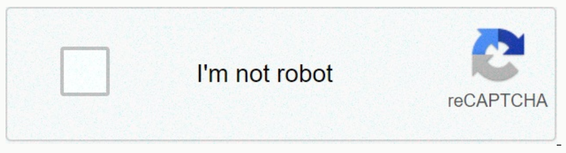


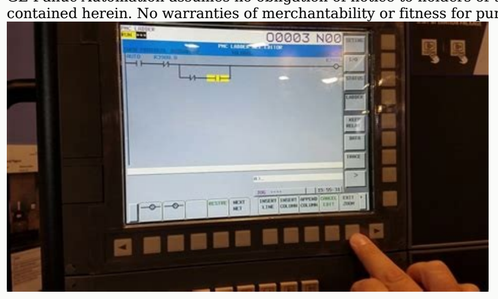
Continue

Pmc ladder language programming manual

Fanuc pmc ladder language programming manual vol 1. Fanuc pmc ladder language programming manual vol 2. Fanuc pmc ladder language programming manual.pdf. Fanuc pmc ladder language programming manual.

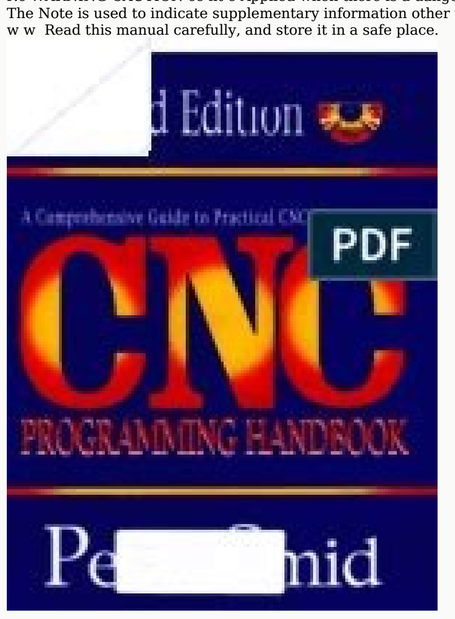
GFZ-61863E B-61863E GE Fanuc CNC PMC Ladder Language Programming Manual Presented By: CNC Center For Product Needs Please Visit: OR Email: OR Call: 1-800-963-3513 GE Fanuc CNC Manuals www.cnccenter.com GE Fanuc Automation Computer Numerical Control Products PMC Model PA1/PA3/SA1/SA2/SA3/SA5 SB/SB2/SB3/SB4/SB5/SB6/SC/SC3/SC4/NB/NB2 Ladder Language Programming Manual GFZ-61863E/10 December 1997 GFL-001 Warnings, Cautions, and Notes as Used in this Publication Warning Warning notices are used in this publication to emphasize that hazardous voltages, currents, temperatures, or other conditions that could cause personal injury exist in this equipment or may be associated with its use. In situations where inattention could cause either personal injury or damage to equipment, a Warning notice is used. Caution Caution notices are used where equipment might be damaged if care is not taken. Note Notes merely call attention to information that is especially significant to understanding and operating the equipment. 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. PowerMotion is a trademark of GE Fanuc Automation North America, Inc. ©Copyright 1997 GE Fanuc Automation North America, Inc. All Rights Reserved.



DEFINITION OF WARNING, CAUTION, AND NOTE B–61863E/10 DEFINITION OF WARNING, CAUTION, AND NOTE om 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.

r.c WARNING CAUTION ce nt e Applied when there is a danger of the user being injured or when there is a damage of both the user being injured and the equipment being damaged if the approved procedure is not observed. .c NOTE nc Applied when there is a danger of the equipment being damaged, if the approved procedure is not observed. w The Note is used to indicate supplementary information other than Warning and Caution.
w w Read this manual carefully, and store it in a safe place.



s–1 Table of Contents B–61863E/10 DEFINITION OF WARNING, CAUTION, AND NOTE .
. . .
. . . . .

. . . . . s–1 PREFACE . . . . . .
.
.
. . . . . .
.
.
. . . . .
. . . . . .
. . . . . .
. . .
. . . . . p–1 I. PMC SEQUENCE PROGRAM 1. SEQUENCE PROGRAM CREATING PROCEDURE . . . . . .
. . . . . 3 SPECIFICATION OF PMCS . . . . . .
.
. . . . . .
. . . . 5 1.2 SUMMARY OF SPECIFICATION OF LADDER PROGRAM . .
. . . .
. . . . . .
. . . . . 15 1.3 WHAT IS A SEQUENCE PROGRAM? . . . . . .
. . . . . .
. . . . . .
. . .
. . . . . 19 1.4 CREATION OF INTERFACE SPECIFICATIONS (STEPS 1 TO 3) . . . . . .
. . . . . 21 1.5 CREATION OF LADDER DIAGRAM (STEP 4) . . . . . .
. . . . . .
. . . . . 21 1.6 CODING (STEP 5) . . . . . .
. . . . . .
. . .
. . . . . .
. . . . . 22 1.7 SEQUENCE PROGRAM ENTRY (STEPS 6, 7) . . . . . .
. . .
. . . . . .

Keep relay D Data table Subprogram Label Fixed timer .c I/O D I/O link w w w D I/O card (R) (A) (T) (C) (K) (D) (P) (L) (I) (O) (I) (O) Sequence program 12 kinds 49 kinds 1100 byte 25 byte 0.1 to 64KB – 14 kinds 66 kinds 1118 byte 25 byte 80 byte 80 byte 80 byte 80 byte 20 byte 20 byte 1860 byte 1860 byte – 512 programs – 9999 labels Timer No.
100 Timer No. 100 devices specified devices specified 1024 points max. 1024 points max. 96 points max. 72 points max. (Note 1) 1024 points max. 1024 points max. 96 points max. 72 points max. (Note 1) Flash ROM 128KB Flash ROM 128KB NOTE 1 Output points of I/O card in 4082 series are following ; PMC–RA1 : 64points, PMC–RA3 : 64points 2
As values indicated with an asterisk (*) in the table, former versions of the programming manual and catalogs have listed the mean processing time of basic commands, but this manual lists the execution time for one step. The actual ladder program execution performance (speed) of each PMC has not been changed. 3 Application PMC for FANUC
Series 21–B loader control function is PMC–RA1. 11 1. SEQUENCE PROGRAM CREATING PROCEDURE PMC SEQUENCE PROGRAM B–61863E/10 Table 1.1 PMC specifications (8) Series 16i/18i/160i/180i Model Number of ladder levels First–level execution period Basic instruction processing time Program capacity D Ladder (step) PMC–RB6 Ladder
Ladder step sequence 2 2 8 ms 8 ms 0.085 (μ sec/step) 0.085 (μ sec/step) 1KB to 128KB 1KB to 128KB 0.1KB to 64KB 0.1KB to 64KB ce nt e D Symbol Comment D Message Instruction (Basic instruction) (Functionalinstruction) nc Internal relay Message request Nonvolatile memory D Variable timer D Counter D Keep replay D Data table Subprogram
Label Fixed timer 14 67 14 67 (R) (A) 1618 bytes 25 bytes (200 points) 3200 bytes 125 bytes (1000 points) (T) (C) (K) (D) (P) (L) 80 bytes (40 each) 80 bytes (20 each) 20 bytes 3000 bytes 512 each 9999 each 100 each (Timer number specification) 300 bytes (150 each) 200 bytes (50 each) 50 bytes 8000 bytes 2000 each 9999 each 100 each (Timer
number specification) 1024 points maximum 1024 points maximum – – 1024 points maximum 1024 points maximum – – Flash ROM 128KB 256KB Flash ROM 128KB 256KB 384KB w .c I/O D I/O link (Input) (Note 2) (Output) D Built–in I/O card (Input) (Output) Sequence program storage media w w About 3,000 About 5,000 About 8,000 About 12,000
About 16,000 About 24,000 About 32,000 r.c About 3,000 About 5,000 About 8,000 About 12,000 About 16,000 About 24,000 om Programming method PMC–RB5 NOTE 1 The PMC–RA1 can be used with the loader control function of the FANUC Series 16i/18i/21i/160i/180i/210i. 2 For I/O of the FANUC Series 16i/18i/21i/160i/180i/210i, only the I/O
link is used. 12 1. SEQUENCE PROGRAM CREATING PROCEDURE PMC SEQUENCE PROGRAM B–61863E/10 Table 1.1 PMC specifications (9) Series 21i/210i Programming method Number of ladder levels First–level execution period Basic instruction processing time Program capacity D Ladder (step) PMC–RA1 (Note 1) PMC–RA5 Ladder Ladder 2
2 8 ms 8 ms 5.0 (μ sec/ step) 0.085 (μ sec/ step) 1KB to 128KB 1KB to 128KB 0.1KB to 64KB 0.1KB to 64KB ce nt e D Symbol Comment D Message Instruction (Basic instruction) (Functionalinstruction) 1118 bytes 25 bytes (200 points) (T) (C) (K) (D) (P) (L) 80 bytes (40 each) 80 bytes (20 each) 20 bytes 1860 bytes – – 100 each (Timer number
specification) 80 bytes (40 each) 80 bytes (20 each) 20 bytes 1860 bytes 512 each 9999 each 100 each (Timer number specification) 1024 points maximum 1024 points maximum – – 1024 points maximum 1024 points maximum – – Flash ROM 128KB Flash ROM 128KB .c w 14 kinds 66 kinds 1100 bytes 25 bytes (200 points) I/O D I/O link (Input) (Note
2) (Output) D Built–in I/O card (Input) (Output) w 12 kinds 49 kinds (R) (A) nc Internal relay Message request Nonvolatile memory D Variable timer D Counter D Keep replay D Data table Subprogram Label Fixed timer w About 3,000 About 5,000 About 8,000 About 12,000 About 16,000 r.c About 3,000 About 5,000 om Model Sequence program
storage media NOTE 1 The PMC–RA1 can be used with the loader control function of the FANUC Series 16i/18i/21i/160i/180i/210i. 2 For I/O of the FANUC Series 16i/18i/21i/160i/180i/210i, only the I/O link is used. 13 1. SEQUENCE PROGRAM CREATING PROCEDURE PMC SEQUENCE PROGRAM B–61863E/10 Table 1.1 PMC specifications (10)
Series 15–MODEL B Model PMC–NB (4048 Series) PMC–NB2 Ladder C–language Ladder C–language Step sequence 3 3 8 ms 8 ms Programming method language Number of ladder level 0.1 Execution Time Program capacity S Ladder(step) S Symbol/Comment (Note) S Message S Language only (Basic) (μs/step) (μs/step) Approx. 8,000
Approx.16,000 Approx.16,000 Approx.24,000 Approx.24,000 1 to 128KB 1 to 128KB ce nt e Instruction (Function) 0.1 Approx.

8,000 r.c Basic instruction om Level–1 Cycle Time 0.1 to 64KB 0.1 to 64KB max. 896KB max. 896KB 14 kinds 14 kinds 69 kinds 69 kinds Internal relay (R) 1618 byte 3200 byte Message request (A) 25 byte 125 byte S Var.Timer (T) 80 byte 300 byte S Counter (C) 80 byte 200 byte (K) 20 byte 50 byte S Data table (D) 3000 byte 8000 byte Subprogram
(P) 512 programs 2000 programs (L) 9999 labels 9999 labels Max 100 timers specified by timer No. Max 100 timers specified by timer No. Non–volatile S Keep relay nc Label Fixed timer Input/output .c S I/O link max 1024 points. max 1024 points. (O) max 1024 points. max 1024 points.
(I) – – (O) – – Flash ROM Flash ROM Sequence program storage media w w w S I/O card (I) 64 KB 64 KB 128 KB 128 KB 256 KB 256 KB 512 KB 512 KB 1 MB 1 MB NOTE Please refer to (4) for PMC–NB(4047 Series). The above–mentioned table is a value for PMC–NB/NB2 (4048 Series).
14 PMC SEQUENCE PROGRAM B–61863E/10 1. SEQUENCE PROGRAM CREATING PROCEDURE 1.2 SUMMARY OF SPECIFICATION OF LADDER PROGRAM Table 1.2 Summary of specification of ladder program (1) Model Compatible Incompatible (Note 2) Interfaces between the PMC and machine (X and Y) Compatible Incompatible (Note 2)
Compatible Incompatible Ladder rogram program compatibility ROM format (object) System Divided system Incompatible (Note 1) Compatible r.c Source format (mnemonic) Basic commands Provided Not provided Compatible Not provided (Note 4) Provided COM (SUB9) Not provided (Note 5) Provided JMP (SUB10) Coil count specification Coil
count specification nc JMPE (SUB30) specification Provided Not provided (Note 5) Provided Provided NOTE 1 The same ROM cannot be shared by different models. The ROM must be rewritten using the offline programmer. 2 It is possible that convert the signal address by the operation of “SIGNAL ADDRESS CONVERSION” (APPENDIX G). 3 The
setting item of system parameter IGNORE DEVIDE CODE is not provided. 4 Use the DISPB (SUB41) command instead. 5 The range of the COM (SUB9) and JMP (SUB10) commands cannot be specified with the number of coils. Specify the range with the COME (SUB29) and JMPE (SUB30) commands. If specify the number of coils, no error messages
will be displayed while editing, but “ALARM093” will be displayed when send the data to RAM. .c w Provided DISP (SUB49) COME (SUB29) specification w Incompatible (Note 2) Not provided (Note 3) ce nt e Undivided system w PMC–P Interfaces between the PMC and CNC (F and G) Others (R, A, C, K, D, T) Function commands PMC–PA3 om PMC
address PMC–PA1 15 1. SEQUENCE PROGRAM CREATING PROCEDURE PMC SEQUENCE PROGRAM B–61863E/10 Table 1.2 Summary of specification of ladder program (2) PMC– RA1 Model PMC– RB Interfaces between the PMC and CNC (F and G) Compatible Interfaces between the PMC and machine (X and Y) Compatible Subprogram, label (P
and L) Not provided Provided ROM format (object) System Divided system Source format (mnemonic) Provided Not provided Provided Compatible (Note 3) Not provided (Note 4) Provided Not provided (Note 4) Provided Not provided Provided Unusable r.c Sub program Usable Basic commands Unusable Usable Unusable Usable
Compatible Not provided END3 (SUB48) Not provided (Note 5) Coil count specification Not provided (Note 6) Provided Not provided (Note 6) Not Provided provided (Note 6) Not provided (Note 6) Not Provided provided (Note 6) Provided COME (SUB29) specification JMP (SUB10) Coil count specification Not provided (Note 6) Provided Provided
JMPE (SUB30) specification Not provided FNC9X (SUB9X) MMCWR (SUB98), MMCWW (SUB99) MMC3R (SUB88), MMC3W (SUB89) (Note 7) Provided (Note 7) Not provided nc MOVB (SUB43), MOVW (SUB44) , MOVN (SUB45) Provided Provided ce nt e DISP (SUB49) COM (SUB9) Provided Provided Provided Not provided Provided Not provided
Provided Not provided Provided Not provided Provided Not provided Provided AND (SUB60), OR (SUB61) NOT (SUB62), EOR (SUB59) Not provided Provided Not provided Provided Not provided Provided Commands for subprogram END (SUB64) , CALL (SUB65), CALLU (SUB66) , SP (SUB71), SPE (SUB72) Not provided Provided Not provided
Provided Not provided Provided Extended jump command JMPB (SUB68), JMPC (SUB73) LBL (SUB69) Not provided Not provided Not provided Provided .c DIFU (SUB57), DIFD (SUB58) w w w Function command (for structured programming) rogramming) PMC– RC3/ RC4 Incompatible (Note 2) Undivided system Function d
commands PMC– RC Compatible (Note 1) Others (R, A, C, K, D, T) Ladder rogram program compatibility Structuring PMC– RB2 PMC– RB3/ RB4/ RB5/ RB6 om PMC address PMC– RA2 PMC– RA3/ RA5 16 PMC SEQUENCE PROGRAM B–61863E/10 1. SEQUENCE PROGRAM CREATING PROCEDURE w w w .c nc ce nt e r.c om NOTE 1 The internal
relay and the data table in nonvolatile memory for the PMC-RB3, RC, RC3 are extended, compared with those for other models.

2 The same ROM cannot be shared by different models. The ROM must be rewritten using the offline programmer. However, the ROM for the PMC–RA2 can be used for the PMC–RA3 and the ROM for the PMC–RB2 can be used for the PMC–RB3. 3 The program can be converted by reinputting it after it is output in a source format. 4 The setting item
of system parameter IGNORE DEVIDE CODE is not provided. 5 Use the DISPB (SUB41) command instead. 6 The range of the COM (SUB9) and JMP (SUB10) commands cannot be specified with the number of coils. Specify the range with the COME (SUB29) and JMPE (SUB30) commands. 7 For the FS18A (PMC–RA1/RA2/RA3), only the MMC–III can
be used. For the FS18B, the MMC–III and MMC–IV can be used. For the FS21B (PMC–RA1/RA3), the MMC–IV can be used. For the FS16i/18i/21i, the MMC–IV can be used. For the FS16C/18C, the MMC–IV can be used. 17 1. SEQUENCE PROGRAM CREATING PROCEDURE PMC SEQUENCE PROGRAM B–61863E/10 Table 1.2 Summary of
specification of ladder program (3) Model PMC– NA (4046) Series PMC address Interfaces between the PMC and CNC (F and G) Incompatible Interfaces between the PMC and machine (X and Y) Compatible Not provided Others (R, A, C, K, D, T) Ladder rogram program compatibility ROM format (object) System Divided system Undivided system
Subprogram Step sequence Compatible (Note 1) Compatible (Note 3) Provided Not provided Provided Usable Unusable Unusable ce nt e Basic commands Function commands DISP (SUB49) COM (SUB9) Coil count spesification JMP (SUB10) Coil count specification Not provided Provided Not provided Provided Provided Not provided
Provided JMPE (SUB30) specification Provided Provided Not provided Not provided Provided S Command for subprogram END (SUB64) , CALL (SUB65) , CALLU Not provided (SUB66) , SP (SUB71) , SPE (SUB72) Provided S Extended jump command Not JMPB (SUB68) , JMPC (SUB73) , LBL (SUB69) provided Provided nc .c w w w Compatible
Provided COME (SUB29) specification LIBRY (SUB60), LEND (SUB61) Usable Provided END3 (SUB48) FNC9X (SUB9X) Function command (for structured programming) Provided Incompatible (Note 2) r.c Source format (mnemonic) PMC– NB2 (4048) om Subprogram, label (P and L) Structuring PMC– NB (4047) (4048) MMCWR (SUB98), MMCWW
(SUB99) MMC3R (SUB88), MMC3W (SUB89) MOVB (SUB43), MOVW (SUB44) MOVN (SUB45) DIFU (SUB57), DIFD (SUB58) AND (SUB60), OR (SUB61) NOT (SUB62), EOR (SUB59) NOTE 1 Management of internal relay address and that of datatable are different between the PMC–NB/NB2 and the PMC–NA. 2 The same ROM cannot be shared by
different models. The ROM must be rewritten using the offline programmer. 3 The data can be converted by outputting in the source format and then inputting again. Moreover, a part of functional instruction is not compatible between PMC–NB/NB2 and PMC–NA. 18 PMC SEQUENCE PROGRAM B–61863E/10 1.3 1. SEQUENCE PROGRAM
CREATING PROCEDURE This is paragraph outlines functions of a sequence program before explaining the programming work. WHAT IS A SEQUENCE PROGRAM? A sequence program is a program for sequence control of machine tools and other systems. A program is defined as a processing procedure to enable CPU to execute arithmetic
processing. om This program is converted into a format (machine language instructions) to enable CPU to execute decoding and arithmetic processing, and stored into the RAM or ROM memory. The CPU reads out instructions of the program stored into the memory at high speed every instruction, and executes the program by arithmetic operation.
r.c The programming of a sequence program begins with the production of a ladder diagram which serves as a processing procedure for arithmetic processing by CPU. This ladder program is produced using PMC instructions. ce nt e After producing the ladder diagram, the processing sequence of this ladder diagram is converted into machine
language instructions, and stored into the memory (program input).

Conversion into the machine language instructions and storage into the memory are done by the PMC programmer. The PMC programmer is a function to produce a program. The sequence program being stored into the memory is sequentially read out into the PMC’s CPU every instruction at high speed and executed. nc Fig. 1.3 shows this relation.
The CPU executes instructions at high speed and outputs arithmetic results to the address Y0.0 output circuit.

w w w .c The CPU reads out input circuit signals of address X0.0 by RD X0.0 instruction, and sets them into an operation register. Then, the CPU executes AND operation with internal relay states at address R10.0 according to the AND R10.1 instruction, and sets these results into the operation register. 19 1.
SEQUENCE PROGRAM CREATING PROCEDURE PMC SEQUENCE PROGRAM B–61863E/10 PMC (Programmable Machine Controller) CPU Sequence program memory X0.0 R10.1 C D W R20.3 Y0.0 RD X0.0 Sequence program input AND R10.1 OR X6.1 om B AND.NOT R20.3 X6.1 WRT Y0.0 Controlled system, such as machine tools and other
systems Input circuit ce nt e X0.0 X6.1 r.c A Output circuit Y0.0 Internal relay (RAM) nc R10.1 R20.3 w w w .c 1.3 Execution of sequence program by PMC 20 PMC SEQUENCE PROGRAM 1.4 CREATION OF INTERFACE SPECIFICATIONS (STEPS 1 TO 3) 1.5 CREATION OF LADDER DIAGRAM (STEP 4) 1. SEQUENCE PROGRAM CREATING
PROCEDURE After deciding the control object specifications and calculating the number of input/output signal points, create the interface specifications. Use the input/output signal interface tables in the CONNECTING MANUAL for the creation of the interface specifications. Enter the signal names (within six characters) in the input/output signal
interface table according to the type of the connected signals. For the input/output signals, see CONNECTING MANUAL. Express the control operations decided by step 2 by use of the ladder diagram (relay circuit diagram). For the functions of the timer, counter, etc. which cannot be expressed with the relay symbols (i.e. the functional instructions),
express them with the symbols assigned to the functional instructions. om B–61863E/10 r.c In the offline programmer and built-in editing function, the sequence program can be entered in the ladder diagram format from the keys of the CRT/MDI panel or from the keys of the keyboard of the SYSTEM P series. ce nt e Also, the entered sequence
program can be output to the printer in the ladder diagram format using the SYSTEM P series. Therefore, entry can be performed while the ladder diagram is created on the CRT screen at the time of sequence program entry. Thus no ladder diagram may be prepared in advance. However, in order to shorten the time occupied by the equipment for
the creation of the sequence program or to efficiently create the sequence program, it is recommended to prepare the ladder diagram in advance. nc The ladder diagram is used as a maintenance diagram by the personnel in charge of maintenance in FANUC, the machine tool builder and end user in the world. Therefore, the ladder diagram must be
easy to understand. w w w .c Signal names (max. six characters) can be entered to the input/output signals, comments (max. 30 characters) can be entered to the relay coil, and comments (max. 30 characters) can be entered to the input/output signals of the address tables at the time of entry of the sequence program. Be sure to enter understandable
signal names and comments as much as possible. 21 1. SEQUENCE PROGRAM CREATING PROCEDURE PMC SEQUENCE PROGRAM 1.6 B–61863E/10 In the coding, the contents of control expressed in the ladder diagram are converted into PMC instructions. In the case of using the offline programmer or ladder diagram editting, since sequence
program entry can be performed in the simple ladder diagram format, it is normally unnecessary to perform coding.

CODING (STEP 5) Coding is necessary only when the sequence program is punched on a paper tape and entered from the paper tape. MF MF TF F 7.2 F 7.3 F7.2 TF MFIN SFIN TFIN F7.3 R211.7 R211.5 R211.6 Step number Instruction RD 851 OR 852 OR 853 RD.NOT.STK 854 OR 856 AND.STK 858 859 Remark F7.0 MF F7.2 SF F7.3 TF
RD.NOT.STK OR RD.NOT.STK OR 862 AND.STK w 861 WRT 1.6 w w 863 F7.0 MF R211.7 MFIN F7.2 SF R211.5 SFIN F7.3 TF R211.6 TFIN G4.3 FIN AND.STK .c 860 G4.3 Address number, bit number nc 850 857 Miscellaneous function finish signal r.c F 7.0 FIN ce nt e F7.0 SF SF om Examples of the ladder diagram and the coding are shown in Fig.
1.6. 22 PMC SEQUENCE PROGRAM B–61863E/10 1. SEQUENCE PROGRAM CREATING PROCEDURE 1.7 The sequence program can be entered in five ways as follows: SEQUENCE PROGRAM ENTRY (STEPS 6, 7) (1) Entry with CRT/MDI keys The sequence program is entered in the ladder diagram format by pressing the keys of the CRT/MDI. (2)
Entry with keys of SYSTEM P series keyboard The sequence program is entered in the mnemonic symbol by pressing the keys of SYSTEM P series keyboard. om (3) Entry from PPR of SYSTEM P series The sequence program punched on a paper tape is read out of the PPR and stored in the memory of the SYSTEM P series.
r.c (4) Entry form floppy disk of SYSTEM P series This method is used when a completed sequence program is slightly changed. The sequence program written in the floppy disk is stored in the memory of SYSTEM P series. 1.8 Check the sequence program and write it into the ROM after check is over. The sequence program can be checked in two
ways. (1) Check by simulator Instead of the machine, connect a simulator (consisting of lamps and switches). Instead of using input signals from the machine, enter signals by turning on and off the switches according to the machine movement. Check the output signals on the basis of the activation of the lamps. nc SEQUENCE PROGRAM CHECK
AND WRITE INTO ROM (STEPS 8 TO 11) ce nt e (5) Entry form ROM Writer This method is used when a completed sequence program is slightly changed. The sequence program written in the ROM is stored from the PMC Writer or FA Writer into P-G or Debugging RAM. w w w .c (2) Check by system operation Perform checks by connecting the
machine. Since it sometimes happens that unexpected operations may be executed depending on a sequence program, arrange for safety before starting operations. (3) Writing into ROM When check of the sequence program is over, write the sequence program into the ROM.
The ROMs to be used are as follows. Then, the ROM into the CNC unit, and deliver it as a regular product to an end user.

Writing of the sequence program into the ROM, maintenance and control thereof shall be performed by the machine tool builder. For this purpose, FANUC provides the PMC Writer or FA Writer as the ROM writer and the ROM or the ROM module that is the PC board on which a ROM chip is mounted. Be sure to use these devices for entering a
sequence program in ROMs. 23 1. SEQUENCE PROGRAM CREATING PROCEDURE PMC SEQUENCE PROGRAM B–61863E/10 1.9 (1) Storage and control of sequence program After debugging, the sequence program should be stored and controlled by the machine tool builder. It can be stored in the following ways: (a) Storing in ROM The sequence
program can be stored in the ROM. For control, enter the drawing number, edition number, etc. of the machine tool builder into the label provided in the ROM, and attach it to the ROM for control. The same control is necessary for the ROM for product. om STORAGE AND CONTROL OF SEQUENCE PROGRAM (STEPS 12 TO 14) r.c (b) Storing in
floppy disk The sequence program can be stored in the floppy disk with offline programmer. Many programs can be stored in one floppy disk. (c) Storing in paper tape The sequence program can be stored in the form of a paper tape. ce nt e (d) Storing in FANUC floppy disk cassette The sequence program can be stored in floppy disk cassette. w w w
.c nc (2) Compiling and control of maintenance drawing The sequence program can be output to the printer in the ladder diagram format using the offline programmer or built-in editing function. Be sure to attach the ladder diagram to the machine as a maintenance drawing together with the machine tool magnetic circuit diagrams, etc. 24 PMC
SEQUENCE PROGRAM B–61863E/10 2 2. SEQUENCE PROGRAM SEQUENCE PROGRAM w w w .c nc ce nt e r.c om Since PMC sequence control handled by software and operates on principles different from a general relay circuit, the sequence control method must be fully understood in order to design the PMC sequence. 25 2. SEQUENCE
PROGRAM 2.1 PMC SEQUENCE PROGRAM B–61863E/10 In a general relay sequence circuit, each relay operates at approximately the same time. In the figure below for example, when relay A operates, the relay D and E operate at approximately the same time. (When both contacts B and C are off.) In PMC sequence control, each relay of the circuit
operates sequentially. When relay A operates, relay D operates, then relay E (see Fig. 2.1 (a)). Thus each relay operates in sequence which can be written as a ladder diagram.
(programmed sequence) EXECUTION PROCEDURE OF SEQUENCE PROGRAM A B A C om D r.c E 2.1 (a) Circuit examples ce nt e Although the PMC sequential operation is performed at high speed, the speed will change with the order to be executed. Fig. 2.1 (b) (A) and (B) illustrate operations varying from the relay circuit to PMC program. (P.B) A
C B .c nc A (A) (P.B) A C w A C B (B) w w C 2.1 (b) Circuit examples (1) Relay circuit Operations are the same in both Fig. 2.1 (b) (A) and (B). Turning on A (P.B) causes current to flow to coils B and C, which turns on B and C. When C turns on, B turns off. (2) PMC program In Fig. 2.1 (b) (A), as in the relay circuit, turning on A (P.B) turns on B and C,
and after one cycle of the PMC sequence, turns off B.

But in Fig. 2.1 (b) (B), turning on A (P.B) turns on C, but does not turn on B. 26 PMC SEQUENCE PROGRAM B–61863E/10 2.2 2. SEQUENCE PROGRAM The sequence program is executed from the beginning of coding to the end of coding of the ladder diagram in the sequence written. When the sequence program ends, the program starts over from
the beginning. This is called repetitive operation. REPETITIVE OPERATION w w w .c nc ce nt e r.c om The execution time from the beginning to the end of the ladder diagram is called the sequence processing time, which varies according to the control scale (the number of steps) and the size of the 1st level sequence. The shorter the process time is,
the better the signal response becomes.

27 2. SEQUENCE PROGRAM 2.3 B–61863E/10 A sequence program consists of three parts: 1st level sequence, 2nd level sequence and 3rd level sequence. The 3rd level sequence part is added to the models usable the 3rd level sequence. (see Fig. 2.3 (a)). The 1st level sequence part operates every 8 ms (high-speed sequential operation). If the 1st
level sequence part is long, the total operating time, including the 2nd level sequence part, is extended. Therefore the 1st level sequence part must be programmed to be processed in as short time as possible. The 2nd level sequence part operates every 8×n ms. Here n is a dividing number for the 2nd level sequence part. The 2nd level sequence part
is divided automatically when the sequence program is transferred to the RAM for debugging in the CNC unit or it is written on ROM after the program is created. The time for one cycle of the sequence program is then displayed on the offline programmer screen. The 3rd level sequence part operates during idle time of PMC. r.c om PRIORITY OF
EXECUTION (1ST LEVEL, 2ND LEVEL AND 3RD LEVEL) PMC SEQUENCE PROGRAM Sequence program SUB 1 ce nt e 1st level sequence part Specifies the end of the 1st level sequence part. Division 1 Division 2 2nd level sequence part SUB 2 nc 3rd level sequence part SUB 48 Division n Specifies the end of the 2nd level sequence part.
(Only the models usable the 3rd level sequence) Specifies the end of the 3rd level sequence part. 2.3 (a) Construction of sequence program w w w .c (1) Division of the 2nd level sequence part The 2nd level sequence part must be divided in order to execute the 1st level sequence part. For example a sequence program is executed in the following
sequence when the dividing number is n. (See Fig. 2.3 (b), 2.3 (c)) After the last 2nd level sequence part (division n) is executed, the sequence program is executed again from the beginning. Thus, when the dividing number is n, the cycle of execution is 8mms (8ms×n).
The 1st level sequence operates every 8 msec, and the 2nd level sequence every 8×n msec. If the steps of the 1st level sequence is increased, the steps of the 2nd level sequence operating within 8 msec becomes less, thereby increasing the dividing number and making the processing time longer. Therefore, it is desirable to program so as to reduce
the 1st level sequence to a minimum. In the, PMC-RA1, -RA2, -RB and -RB2, 1.25 ms of 8 ms is assigned to execution of the 1st and 2nd level sequences. The remaining time is assigned to NC processing.
In the PMC-RC, 5 ms of 8 ms is assigned to execution of the 1st and 2nd level sequences. The standard setting value is 5 ms when system parameter LADDER EXEC = 100%. The remaining time is assigned to execution of the 3rd level sequence and the program. 28 2. SEQUENCE PROGRAM PMC SEQUENCE PROGRAM B–61863E/10 8ms 1st level
8ms 8ms 1.25ms 1.25ms 1.25ms Division 1 Division 2 Division n Division 1 2nd level NC processing 8ms 1st level Division 1 8ms 5ms 5ms Division 2 Division n Division 1 r.c 2nd level 8ms 5ms om 2.3 (b) Sequence in which the Sequence Program Is Executed (PMC-RA1, -RA2, -RB and -RB2) 3rd level, program execution, and displaying the PMC screen
2.3 (c) Sequence in which the Sequence Program Is Executed (PMC-RC) w w w .c nc ce nt e (2) 1st level sequence part Only short-width pulse signals are processed. These signals include emergency stop, overtravel of each axis, reference point return deceleration, external deceleration, skip, measuring position arrival and feed hold signals. (3) 3rd
level sequence The purpose of the 3rd level sequence is to execute such programs as display processing or control status monitor having no direct relation to the machine control (operator message, alarm display, etc.), to lighten the load of the 2nd level program having a direct relation to the machine control by transferring former programs to the
3rd level, and to shorten the PMC execution time (cycle time). For PMC-RC, when 3rd level program is not used, command SUB 48 (END3) following SUB 2 instruction. (4) Divided system and undivided system There is a model can use the divided system and undivided system among the PMCs. In the divided system, a ladder program is divided
before being executed if all ladder program run regardless of the sequence state (see Fig. 2.3 (d)).

For an actual ladder program, not all ladder program run. The PMC cannot therefore be used effectively. The PMC can execute the ladder program in the system for terminating one cycle of the program using the time to execute the actual ladder program (undivide system) as well as in the divided system. The time required for the one cycle can be
reduced by the effective use of jump instructions in the ladder program. Since the sequence using many functional instructions requires a lot of processing time, the undivided system should be specified so that the PMC is used more effectively (see Fig.
2.3 (e)).

To operate the PMC in the undivided system, set system parameter IGNORE DIVIDE CODE to YES. The PMC model usable only the undivided system, does not have setting system parameter IGNORE DIVIDE CODE. It is always 29 2. SEQUENCE PROGRAM PMC SEQUENCE PROGRAM B–61863E/10 operated under the undivided system. A B
Functional instruction Functional instruction om C Functional instruction The ladder program is divided if all functional instructions are executed regardless of execution of an actual ladder program. A r.c 2.3 (d) Divisions in the divided system Functional instruction Functional instruction ce nt e B C Functional instruction Execution of an actual
ladder program when A = 0, B = 1, and C = 0. 2.3 (e) Execution of a ladder program w w .c nc (a) Example of effective use of the undivided system Example 1) Many M codes are usually used.
Since more than one M code is not used in the same block, the decoded M code is divided into several parts. Machine instructions are used as these decoded parts. The M code is divided into M codes having two digits such as M21, M22, M24, M28, and so on. Example 2) To reduce the number of ROM types using the same ladder program for multiple
machines, a PMC parameter must be specified so that any of the following ladder program run. w Ladder A (Ladder common to all machines) (Selected by a PMC parameter) Ladder B1 Ladder B2 Ladder B3 (Ladders dedicated to each machine) 2.3 (f) (5) Construction of sequence program in the case of using Sub-program. 30 PMC SEQUENCE
PROGRAM B–61863E/10 2. SEQUENCE PROGRAM 1st level sequence part END1 (SUB 1) 3rd level sequence part (Only about the PMC model usable the 3rd level sequence) END3 (SUB 48) SP r.c END2 (SUB 2) om 2nd level sequence part ce nt e D Sub program must be written between 2nd D level program and 3rd level program. SPE SP nc Sub
program w .c SPE SPE END 2.3 (g) w w End of sequence program SP 31 D The end of sequence program is expressed D by END command. 2. SEQUENCE PROGRAM PMC SEQUENCE PROGRAM B–61863E/10 2.4 Ę : Can be used : Cannot be used SEQUENCE PROGRAM STRUCTURING PA1 PA3 RA1 RA2 RA3 RA5 Ę Ę Ę RB RB2 RB3 RB4 RB5 RB6 Ę
Ę Ę Ę RC RC3 RC4 NB NB2 Ę Ę Ę Ę With the conventional PMC, a ladder program is described sequentially. By employing a ladder language that allows structured programming, the following benefits are derived: D A program error can be found easily. om D A program can be understood and developed easily.
D When an operation error occurs, the cause can be found easily. Three major structured programming capabilities are supported. r.c (1) Subprogramming A subprogram can consist of a ladder sequence as the processing unit. ce nt e Job A FUNC Job B D D D D f D D D D f D D D nc (2) Nesting Ladder subprograms created in (1) above are combined
to structure a ladder sequence. Job A Job B Sub Program1 w 32 Sub Program2 Job A1 Job A11 D D D Job A12 Job An w w .c Main Program 2. SEQUENCE PROGRAM PMC SEQUENCE PROGRAM B–61863E/10 (3) Conditional branch The main program loops and checks whether conditions are satisfied. If a condition is satisfied, the corresponding
subprogram is executed. If the condition is not satisfied, the subprogram is skipped. Main Program Sub Program1 PROCESS11 STATE1 PROCESS1 STATE2 PROCESS2 PROCESS11 D D D om PROCESS12 w w w .c nc ce nt e For details, see Chapter 9. r.c PROCESS13 33 2.
SEQUENCE PROGRAM 2.5 PROCESSING I/O SIGNALS PMC SEQUENCE PROGRAM B–61863E/10 Input signals (M function, T function, etc.) from the CNC and those (cycle start, feed hold, etc.) from the machine tool are sent to the PMC. Signals for the CNC (cycle start, feed hold, etc.) and those for the machine tool (tunret rotation, spindle stop,
etc.) are output from the PMC. Fig.

2.4 shows the relationship between these signals and the PMC. CNC om Input signals are entered in the input memory of PMC and output signals are issued from PMC. As shown in Fig. 2.5, the input signals are synchronized only in the 2nd level sequence part. PMC Input memory of CNC r.c Sequence program Input signals from CNC ce nt e
Transmitted at the start of 2nd level 1st level sequence part 2nd level synchronous input signal memory Input signals from CNC Transmitted every 8 ms nc Input signals from machine tool 2nd level sequence part Output memory of CNC to w MT signals .c Output CNC to w Output signals machine tool Output signals to the machine Input signal
memory Input signals from machine tool w Output signalmemory Input signals from machine Transmitted every 2 ms 2.5 PMC I/O signals 34 3rd level sequence part (Only PMC-RC) PMC SEQUENCE PROGRAM B–61863E/10 2.5.1 2. SEQUENCE PROGRAM (1) Input memory of CNC The input signals from CNC are loaded in memory of CNC and are
transferred to the PMC at intervals of 8 ms. Since the 1st level and the 3rd level sequence part directly refer to these signals and process operations, these signals do not synchronize with input signals from the CNC. See item 2.5.3. Input Signal Processing om (2) Input signals from machine tool (DI/DO card) Input signals from the machine tool are
transferred to the input signal memory from the input circuit (DI/DO card). 1st level and 3rd level sequence part directly processes by reading signals loaded in the input signal memory.
ce nt e r.c (3) Input signal memory The input signal memory stores signals transferred from the machine tool at intervals of 2 ms period. The PMC 1st level sequence part and 3rd level sequence part are used to read and process signals stored in this memory.
In this case, state of signals set in the input signal memory synchronizes with that of 1st level sequence part but not with that of 3rd level sequence part. See item 2.5.3. w w .c nc (4) 2nd level synchronous input signal memory The 2nd level synchronous input signal memory stores signals processed by the 2nd level sequence section. State of the
signals set in this memory synchronizes with that of the 2nd level sequence part. Input signal memory and input signals from the CNC are transferred to the 2nd level synchronous input signal memory only at the beginning of execution of the 2nd level sequence section. Therefore, the status of the 2nd level synchronous input signal memory does not
change from the beginning to end of the execution of the 2nd level sequence part.

Programmer function makes the processing so that the 1st level sequence section and 3rd level sequence section use the input signal memory and input signals from the CNC side and the 2nd level sequence section uses the 2nd level synchronous input signal memory.
w 2.5.2 Output Signal Processing (1) CNC output memory The output signals are transferred from the PMC to the CNC output memory at intervals of 8 ms. (2) Output signals to machine tool (DI/DO card) Output signals to the machine tool are transferred from the PMC output signal memory to the machine tool. (3) Output signal memory The output
signal memory is set by the PMC sequence program. Signals stored in this memory are transferred to the machine side at a 2 ms period. 35 2.

SEQUENCE PROGRAM PMC SEQUENCE PROGRAM B–61863E/10 2.5.3 Signals input from the CNC are transferred to the PMC at intervals of 8 ms. Signals output to the CNC are transferred from the PMC at intervals of 8 ms. PMC I/O signals are generally transferred at intervals of 8 ms. In this case, note that state of the input signals from the CNC
does not synchronize with that of the 1st level sequence program and the 2nd level sequence program. By this reason, if an input signal from the CNC may change while execution of the 1st level sequence program, for example, some trouble may occur like example in Fig. 2.5.3 (a). To avoid such trouble, write the state of signal TF in an internal relay
at the start of the 1st level sequence, then the 1st level sequence program shall refer to the internal relay as signal TF. See Fig. 2.5.3 (b). ce nt e r.c I/O Signals to CNC om NOTE The status of the CNC input memory, input signals from machine, CNC output memory and output signals to machine can be checked by using the PC self-diagnosis function.
The self-diagnosis number specified is the address number used by the sequence program. W1 TF W2 END 1 If after TF=0 is load, signal state changes to TF=1, state of W1=1 and W2=1 may momentary occur w .c nc TF w w 2.5.3 (a) TF TFM TFM W1 TFM W2 END 1 Make signal TF synchronized one, and state of W1=1 and W2=1 may not occur.
2.5.3 (b) 36 2. SEQUENCE PROGRAM PMC SEQUENCE PROGRAM B–61863E/10 2.5.4 The status of the same input signal may be different in the 1st level and 2nd level sequences. That is, at 1st level, processing is performed using input signal memory and at 2nd level, processing is performed using the 2nd level synchronous input signal memory.
Therefore, it is possible for a 2nd level input signal to delay by a cycle of 2nd level sequence execution at the worst, compared with a 1st level input signal.
This must be kept in mind when writing the sequence program.

A.M ON (short time width pulse signal) B OFF Signal statesO C OF om Difference of Status of Signals Between 1st Level and 2nd Level Differences drawn in Fig. 2.5.4 (a) and Fig. 2.5.4 (b) when the 1st level sequence has been executed are as follows: r.c (a) Fig. 2.5.4 (a) W2 may not be 1 even when W1=1. (Because the A.M signal may be different at
the 1st and 2nd levels.) B A.M B nc A.M ce nt e (b) Fig. 2.5.4 (b) If W1=1, W2=1. When performing the sequence shown in Fig.
2.5.4 (a), proceed as follows: At 1st level, perform a high-speed sequence when the A.M signal changes (operating). At 2nd level, perform sequence processing when the A.M signal does not change (stopped). W1 w .c 1st Level A.M w w END 1 END 1 C 2nd Level W1 W1 C W2 W2 2.5.4 (a) 2.5.4 (b) 37 2. SEQUENCE PROGRAM 2.6 PMC SEQUENCE
PROGRAM B–61863E/10 Interlocking is externally important in sequence control safety. Interlocking with the sequence program is necessary. However, interlocking with the end of the electric circuit in the machine tool magnetics cabinet must not be forgotten. Even though logically interlocked with the sequence program (software), the interlock
will not work when trouble occurs in the hardware used to execute the sequence program.

Therefore, always provide an interlock inside the machine tool magnetics cabinet panel to ensure operator safety and to protect the machine from damage. w w w .c nc ce nt e r.c om INTERLOCKING 38 2. SEQUENCE PROGRAM PMC SEQUENCE PROGRAM B–61863E/10 2.7 SEQUENCE PROGRAM PROCESSING TIME The exact sequence
processing time is displayed on the CRT screen when the sequence programs have been completed.

The time is 2nd level sequence division number n x 8 ms. This section explains how to estimate processing times that are important in sequence control when the ladder diagram, the basis of sequence program control, is almost complete.
r.c om (1) Processing time calculation units Sequence processing time estimation is based on the basic instructions (AND, OR, etc.). The execution time for a functional instruction is given in the execution constant column of the Functional Instruction Table. Converted to a basic instruction; that is the number of basic instructions that a functional
instruction is equivalent to. Processing time is determined for the above using the equation in item below. (2) Processing time estimation equation The number of division (n) in the 2nd level sequence is determined and the processing time is calculated using the following equations: ce nt e Sequence processing time = n (number of division) n= (LT)
msec (ET)msec – (HT)msec 8 msec +1 (n is an integer, fractions are omitted) nc (a) (HT) is the execution time for the 1st level sequence section. (HT)={(number of steps in basic instruction)+(sum of functional instruction execution time constants) 10} (IT) μsec Execution time constant for END.1 (206) must be included in HT.
w w w .c (b) (LT) is the execution time for the 2nd level sequence section. (LT)={(number of steps in basic instruction)+(sum of functional instruction execution time constants) 10} (IT) μsec END.2 execution time (127) must be included. (c) (ET) is the execution time assigned to the 1st and 2nd level parts out of 8 ms. For PMC-RB (ET) = 1.25 ms =
1250μs For PMC-RC (standard setting when LADDER EXEC = 100%) (ET) = 5 ms = 5000μs (d) IT) is the execution constant for calculating the processing time. The value is as follows: (IT) = 0.15μs 39 2. SEQUENCE PROGRAM PMC SEQUENCE PROGRAM B–61863E/10 (3) Processing time calculation example om (a) 1st level sequence Basic
instruction: 100 steps Functional instruction: CTR: 2 times, COMPB: 2 times CTR execution time constant: 26 COMPB execution time constant: 24 END.1 execution time constant: 206 HT={100+(26 2+24 2+206) 10} 0.15 =474 μsec ce nt e r.c (b) 2nd level sequence Basic instruction: 6,000 steps Functional instruction: TMR: 35 times, DECB: 25
times, ROTB: 2 times TMR execution time constant: 23 DECB execution time constant: 20 ROTB execution time constant: 33 END.2 execution time constant: 32 LT={6,000+(23 35+20 25+33 2+32) 10} 0.15=3004.5msec (c) Determination of the number of divisions (n) n= 3004.5 μsec 1250μsec – 474 μsec +1 = 4.87 w w w .c nc (d) Processing time
calculation Sequence processing time=4 (number of division) msec 40 8 msec=32 2. SEQUENCE PROGRAM PMC SEQUENCE PROGRAM B–61863E/10 2.8 SEQUENCE PROGRAM MEMORY CAPACITY In the PMC-RB, one 1M-bit EPROM is used for storing the sequence program. In the PMC-RA1, –RA2, –RB and –RB2, a 128KB, 256KB, 512KB, or
1MB ROM module is used for this purpose. Table 2.8 (a) shows the maximum memory capacity available for the sequence program. The number of bytes in parentheses indicates the size of the area dedicated to the programs other than the sequence program. Table 2.8 (a) Maximum Memory Capacity for a Sequence Program Ladder Symbol and
comment Message Total 1M-bit EPROM 64KB 64KB each 64KB 126KB Note) PMC-RC 128KB ROM module 96KB 64KB each 64KB 126KB Note) PMC-RC3 256KB ROM module 96KB 64KB each 64KB 254KB Note) PMC-NB 512KB ROM module 96KB 64KB each 1MB ROM module 96KB 64KB each r.c RA2, 64KB 288KB (222KB) 64KB 288KB (734KB) ce
nt e PMC-RA1, RB, RB2 om ROM nc NOTE All ladder, symbol, comment, and message data items cannot be created using each maximum memory capacity.

Reduce the memory capacity for any of the data items and create them so that they add up to the total capacity of each ROM. w w w .c Generate a sequence program within a range of bytes shown in Table 2.8 (a).
Calculate the number of bytes of sequence program instructions and data based on Table 2.8 (b). When the program is initialized, symbol and comment areas are allocated 32KB (extendable and reducible in 1KB units) on memory. When the program is initialized, a message area is also allocated 2.1KB (extendable and reducible in 1KB units).
Therefore, program the basic instructions and functional instructions listed in Table 2.8 (b) in the remaining capacity which is the difference of the number of bytes shown in Table 2.8 (a) and the number of bytes for symbols, comments, and messages. Table 2.8 (b) Sizes of sequence program instructions and data Basic instruction Functional
instruction Functional instruction parameters 4 bytes 4 bytes 4 bytes Message data 1 byte/character (alphanumeric characters) 2 bytes/kana characters 41 Symbol Comment 10 bytes 1 byte/character 3. ADDRESS 3 PMC SEQUENCE PROGRAM B–61863E/10 ADDRESS r.c om An address shows a signal location. Addresses include input/output signals
with respect to the machine, the input/output signals with respect to the CNC, the internal relays, the counters, the keep relays (PMC parameters), and data table.

Each address consists of an address number (for every 8 signals) and a bit number (0 to 7). Enter the symbol table showing the relationship between the signal names and the addresses into the programmer by using the keys of the CRT/MDI or the keys of the keyboard of the offline programmer as in the case of the sequence program. For
programming, see Chapter III and IV. ce nt e (1) Addresses related to PMC Four types of addresses as shown in Fig. 3 are necessary for creation of the PMC sequence program. Internal relay .c nc Note) PMC Machine (MT) signal Nonvolatile memory (1) Counter (2) Keep relay (3) Data table (4) Variable Timer w w w CNC signal 3 Addresses related to
PMC (a) The input/output signals with respect to the PMC, which are indicated by the solid lines, are transferred via the receiver and the driver of the I/O board. (b) The input/output signals with respect to the PMC, which are indicated by the broken lines, are transferred only in the memory such as the RAM. All of these signals can be displayed on
the CRT/MDI panel. (2) Address regulations The address comprises the address number and the bit number in the format as shown below. 42 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 X 127. 7 Bit number (0 to 7) Address number (within four numerics after alphabet) om An alphabet must be specified at the beginning of the address
number to indicate the type of the signal as shown in Table 3. When specifying the address in the byte unit by the functional instruction, specify X127. In this case, “.” and the bit number are not necessary. Table 3 Alphabetic characters in address numbers (1) Model Signal description Power Mate Ć D PMC PA1 PMCĆ PA3 Power MateĆ F Power
MateĆ H PMCĆ PA3 PMCĆ PA3 r.c Character Input signal from the machine to the PMC (MT to PMC) X0 to X127 (I/O Link Master) X1000 to X1003 (Built–in I/O Card) X1020 to X1051 (I/O Link Slave) X1000 to X1005 X1020 to X1027 (Slave) Y Output signal from the PMC to the machine (PMC to MT) (Caution 3) Y0 to Y127 (I/O Link Master) Y1000 to
Y1002 (Built–in I/ O Card) Y1020 to Y1051 (I/O Link Slave) Y1000 to Y1003 Y1020 to Y1027 (Slave) F Input signal from the NC to the PMC (NC to PMC) F0 to F255 F1000 to F1255 (Dual path control) F0 to F255 F0 to F255 G Output signal from the PMC to the NC (PMC to NC) G0 to G255 G1000 to G1255 (Dual path control) G0 to G255 G0 to G255 R
Internal relay (Caution 1) R0 to R999 R9000 to R9099 nc ce nt e X R0 to R999 R9000 to R9117 R0 to R999 R9000 to R9117 X0 to X127 (I/O Link Master) X1000 to X1003 (Built–in I/O Card) X1020 to X1051 (I/O Link Slave) Y0 to Y127 (I/O Link Master) Y1000 to Y1002 (Built–in I/O Card) Y1020 to Y1051 (I/O Link Slave) R0 to R999 R9000 to R9117
Message request signal A0 to A24 A0 to A24 A0 to A24 Counter C0 to C79 C0 to C79 C0 to C79 K Keep relay (Caution 2) K0 to K19 K0 to K19 K0 to K19 T0 to T79 T0 to T79 T0 to T79 D0 to D1859 D0 to D1859 D0 to D1859 L w .c A C Label Number – L1 to L9999 L1 to L9999 L1 to L9999 P Subprogram Number – P1 to P512 P1 to P512 P1 to P512 T
Data table w w D Variable timer CAUTION 1 R9000 to R9117 are areas reserved for the PMC system program; these areas cannot be used for output by a sequence program. 2 K17 to K19 are areas reserved for the PMC system program; these areas cannot be used for output by a sequence program. 3 I/O Link Master function is not available in the
Power Mate–MODEL F. You cannot use the address X0–127 and Y0–127. 43 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Table 3 Alphabetic characters in address numbers (2) Model Character FS20A Signal description PMCĆRA1 PMCĆRA3 FS18A PMCĆRA1 PMCĆRA2 Input signal from the machine to the PMC (MT to PMC) X0 to X127
X1000 to X1013 (Caution 1) X0 to X127 X1000 to X1019 Y Output signal from the PMC to the machine (PMC to MT) Y0 to Y127 Y1000 to Y1013 (Caution 1) Y0 to Y127 Y1000 to Y1014 F Input signal from the NC to the PMC (NC to PMC) F0 to F255 F1000 to F125 F0 to F255 F1000 to F1255 G Output signal from the PMC to the NC (PMC to NC) G0 to
G255 G1000 to G1255 R Internal relay (Caution 2) R0 to R999 R9000 to R9099 A Message request signal A0 to A24 C Counter C0 to C79 K Keep relay (Caution 3) K0 to K19 D Data table T Variable timer L Label number P Subprogram number om X PMCĆRA3 G0 to G255 G1000 to G1255 D0 to D1859 R0 to R999 R9000 to R9117 A0 to A24 C0 to C79
K0 to K19 D0 to D1859 ce nt e T0 to T79 R0 to R999 R9000 to R9099 r.c R0 to R999 R9000 to R9117 T0 to T79 – L1 to L9999 – L1 to L9999 – P1 to P512 – P1 to P512 w w w .c nc CAUTION 1 X1000 to X1007 and Y1000 to Y1007 are configured as a matrix.
2 R9000 to R9117 are areas reserved for the PMC system program; these areas cannot be used for output by a sequence program. 3 K17 to K19 are areas reserved for the PMC system program; these areas cannot be used for output by a sequence program. 44 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Table 3 Alphabetic characters in
address numbers (3) Character Signal description Model PMCĆRB PMCĆRB2 PMCĆRB3 PMCĆRC PMCĆRC3 PMCĆNB Input signal from the machine to the PMC (MT to PMC) X0 to X127 X1000 to X1039 X0 to X127 Y Output signal from the PMC to the machine (PMC to MT) Y0 to Y127 Y1000 to Y1029 Y0 to Y127 F Input signal from the NC to the
PMC (NC to PMC) F0 to F255 F1000 to F1255 F0 to F319 G Output signal from the PMC to the NC (PMC to NC) G0 to G255 G1000 to G1255 R Internal relay (Caution 1) A Message request signal C Counter K Keep relay (Caution 2) D Data table T Variable timer L Label number P Subprogram number R0 to R999 R9000 to R9099 R0 to R999 R9000 to
R9117 R0 to R1499 R9000 to R9117 om X G0 to G511 R0 to R1499 R9000 to R9099 R0 to R1499 R9000 to R9117 r.c A0 to A24 R0 to R1499 R9000 to R9117 C0 to C79 K0 to K19 D0 to D1859 D0 to D2999 ce nt e T0 to T79 – – L1 to L9999 – L1 to L9999 – – P1 to P512 – P1 to P512 w w w .c nc CAUTION 1 R9000 to R9117 are areas reserved for the
PMC system program; these areas cannot be used for output by a sequence program. 2 K17 to K19 are areas reserved for the PMC system program; these areas cannot be used for output by a sequence program. 45 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Table 3 Alphabetic characters in address numbers (4) Model Character Signal
description Series 16-MODEL B/Series 18-MODEL B PMCĆRB3 PMCĆRC3 PMCĆRB4 Series 18-MODEL B PMCĆRC4 PMCĆRA1 Input signal from the machine to the PMC (MT to PMC) X0 to X127 X1000 to X1019 X1020 to X1039 X0 to X127 X1000 to X1019 X1020 to X1039 Y Output signal from the PMC to the machine (PMC to MT) Y0 to Y127
Y1000 to Y1014 Y1020 to Y1034 Y0 to Y127 Y1000 to Y1014 Y1020 to Y1034 F Input signal from the F0 to F255 NC to the PMC (NC to F1000 to F1255 PMC) F0 to F255 F1000 to F1255 F0 to F511 F1000 to F1511 F2000 to F2511 F0 to F511 F1000 to F1511 F2000 to F2511 F0 to F255 F1000 to F1255 G Output signal from G0 to G255 the PMC to the
NC G1000 to G1255 (PMC to NC) G0 to G255 G1000 to G1255 G0 to G511 G1000 to G1511 G2000 to G2511 G0 to G511 G1000 to G1511 G2000 to G2511 G0 to G255 G1000 to G1255 R Internal relay R0 to R1499 R9000 to R9117 R0 to R2999 R9000 to R9199 R0 to R2999 R9000 to R9199 R0 to R999 R9000 to R9099 A Message signal A0 to A24 A0 to
A124 A0 to A124 A0 to A24 C Counter C0 to C79 C0 to C79 C0 to C199 C0 to C199 C0 to C79 K Keep relay K0 to K19 K0 to K19 K0 to K39 K900 to K909 K0 to K39 K900 to K909 K0 to K19 are request A0 to A24 r.c R0 to R1499 R9000 to R9117 om X Data table T0 to T79 T0 to T79 T0 to T299 T0 to T299 T0 to T79 Variable timer D0 to D2999 D0 to
D2999 D0 to D7999 D0 to D7999 D0 to D1859 L Label number L1 to L9999 L1 to L9999 L1 to L9999 L1 to L9999 – P Subprogram number P1 to P512 P1 to P512 P1 to P2000 P1 to P2000 – w w w .c nc T D 46 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Table 3 Alphabetic characters in address numbers (5) Model Character Series 16-
MODEL C/Series 18-MODEL C Signal description PMCĆRB5 PMCĆRC3 PMCĆRB6 PMCĆRC4 Input signal from the machine to the PMC (MT to PMC) X0 to X127 X1000 to X1019 X1020 to X1039 Y Output signal from the PMC to the machine (PMC to MT) Y0 to Y127 Y1000 to Y1014 Y1020 to Y1034 F Input signal from the NC to the PMC (NC to F0 to
F255 PMC) F1000 to F1255 F0 to F255 F1000 to F1255 F0 to F511 F1000 to F1511 F2000 to F2511 F0 to F511 F1000 to F1511 F2000 to F2511 G Output signal from the PMC to the NC G0 to G255 (PMC to NC) G1000 to G1255 G0 to G255 G1000 to G1255 G0 to G511 G1000 to G1511 G2000 to G2511 G0 to G511 G1000 to G1511 G2000 to G2511 R
Internal relay R0 to R1499 R9000 to R9117 R0 to R1499 R9000 to R9117 R0 to R2999 R9000 to R9199 R0 to R2999 R9000 to R9199 r.c om X Message request signal A0 to A24 A0 to A24 A0 to A124 A0 to A124 Counter C0 to C79 C0 to C79 C0 to C199 C0 to C199 Keep relay K0 to K19 K0 to K19 K0 to K39 K900 to K909 K0 to K39 K900 to K909 T
Data table T0 to T79 T0 to T79 T0 to T299 T0 to T299 D Variable timer D0 to D2999 D0 to D2999 D0 to D7999 D0 to D7999 L Label number L1 to L9999 L1 to L9999 L1 to L9999 L1 to L9999 P Subprogram number P1 to P512 P1 to P512 P1 to P2000 P1 to P2000 w w w .c nc ce nt e A C 47 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10
Table 3 Alphabetic characters in address numbers (6) Model Character Signal description Series 21/210-MODEL B PMCĆRA1 PMCĆRA3 Input signal from the machine to X0 to X127 the PMC (MT to PMC) X1000 to X1011 Y Output signal from the PMC to the Y0 to Y127 machine (PMC to MT) Y1000 to Y1008 (Note) F Input signal from the NC to the
F0 to F255 PMC (NC to PMC) F1000 to F1255 G Output signal from the PMC to the G0 to G255 NC (PMC to NC) G1000 to G1255 R Internal relay A Message request signal C Counter K Keep relay D Data table T Variable timer L Label number – L1 to L9999 P Subprogram number – P1 to P512 om X R0 to R1999 R9000 to R9099 R0 to R1499 R9000 to
R9117 ce nt e r.c A0 to A24 C0 to C79 K0 to K19 – D0 to D1859 T0 to T79 w w .c nc NOTE The Y addresses for the 4082 series are Y0 to Y127 and Y1000 to Y1007. 48 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Table 3 Alphabetic characters in address numbers (7) Model Sym Symbol Type of signal FANUC Series 16/18/160/180 PMC–
RB5 PMC–RB6 Input signal from the machine to PMC (MT ³ PMC) X0 to X127 (Note) X0 to X127 (Note) Y Output signal from the PMC to machine (PMC ³ MT) Y0 to Y127 (Note) Y0 to Y127 (Note) F Input signal from the NC to PMC (NC ³ PMC) F0 to F255 F1000 to F1255 G Output signal from the PMC to NC (PMC ³ NC) G0 to G255 G1000 to G1255 R
Internal relay R0 to R1499 R9000 to R9117 A Message request signal A0 to A24 C Counter C0 to C79 K Keep relay D Variable timer T Data table L Label number P Subprogram number om X F0 to F511 F1000 to F1511 F2000 to F2511 ce nt e r.c G0 to G511 G1000 to G1511 G2000 to G2511 K0 to K19 R0 to R2999 R9000 to R9199 A0 to A124 C0 to
C199 K0 to K39 K900 to K909 T0 to T79 T0 to T299 D0 to D2999 D0 to D7999 L1 to L9999 L1 to L9999 P1 to P512 P1 to P2000 w w w .c nc NOTE With the 16/18–C, the addresses (X1000 and up, Y1000 and up) for the built–in I/O card are reserved. However, these areas cannot be used for I/O.
Never use X1000 and up, or Y1000 and up. 49 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Table 3 Alphabetic characters in address numbers (8) Model Sym Symbol Type of signal FANUC Series 21/210 PMC–RA5 X Input signal from the machine to PMC ³ X0 to X127 (Note) Y Output signal from the PMC to machine ³ Y0 to Y127 (Note) F
Input signal from the NC to PMC ³ F0 to F255 F1000 to F1255 G Output signal from the PMC to NC ³ G0 to G255 G1000 to G1255 R Internal relay A Message request signal C Counter K Keep relay D Variable timer T Data table L Label number P Subprogram number R0 to R999 R9000 to R9117 r.c R0 to R999 R9000 to R9099 om PMC–RA1 A0 to
A24 C0 to C79 ce nt e K0 to K19 T0 to T79 D0 to D1859 – L1 to L9999 – P1 to P512 w w w .c nc NOTE With the 16/18–C, the addresses (X1000 and up, Y1000 and up) for the built–in I/O card are reserved. However, these areas cannot be used for I/O. Never use X1000 and up, or Y1000 and up. 50 3. ADDRESS PMC SEQUENCE PROGRAM B–
61863E/10 Table 3 Alphabetic characters in address numbers (9) Model Character Series 15-MODEL B Signal description PMC-NB (4048) PMC-NB2 Input signal from the machine to the PMC (MT to PMC) X0 to X127 Y Output signal from the PMC to the machine (PMC to MT) Y0 to Y127 F Input signal from the NC to the PMC (NC to PMC) F0 to F319
G Output signal from the PMC to the NC (PMC to NC) R Internal relay om X G0 to G511 R0 to R1499 R9000 to R9117 r.c R0 to R1499 R9000 to R9099 Message request signal A0 to A24 A0 to A124 Counter (Non–volatile memory) C0 to C79 C0 to C199 K Keep relay (Non–volatile memory) K0 to K19 K0 to K39 K900 to K909 ce nt e A C D Data table
(Non–volatile memory) T Variable timer (Non–volatile memory) L Label number P Subprogram number D0 to D2999 D0 to D7999 T0 to T79 T0 to T299 P1 to P512 L1 to L9999 P1 to P2000 w w w .c nc CAUTION 1 R9000 to R9199 are areas reserved for the PMC system program; these areas cannot be used for output by a sequence program. 2 K17 to
K19 or K900 to K909 are areas reserved for the PMC system program; these areas cannot be used for output by a sequence program. 3 Please refer to (3) PMC–NB(Series 4047). 51 3. ADDRESS PMC SEQUENCE PROGRAM 3.1 ADDRESSES BETWEEN PMC AND CNC (PMCNC) B–61863E/10 Addresses of the interfaces are outlined below.
CONNECTING MANUAL of Series 16. For details, see (1) Basic machine interface (a) PMCCNC related signals The addresses for Series 15 are from F0 to F511, for the others are from F0 to F255. For details of the signals, see CONNECTING MANUAL of CNC. w w w .c nc ce nt e r.c om (b) PMCCNC related signals The addresses for Series 15 are
from G0 to G511, for the others are from G0 to G255. For details of the signals, see CONNECTING MANUAL of CNC. 52 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 3.2 ADDRESSES BETWEEN PMC AND MACHINE TOOL (PMCMT) 3.2.1 Addresses Between PMC and Machine Tool for PMC-RB/RC (a) PMC ← MT Addresses are from X0 to
X127. (b) PMC → MT Addresses are from X0 to X127. om (1) When the FANUC I/O UNIT-MODEL A is used r.c Up to 1024 input and 1024 output points can be assigned to any address within the above range in byte units. (2) When the built-in I/O card is used (except Series 15) ce nt e (a) PMC ← MT Addresses are from X1000 to X1019. (b) PMC → MT
Addresses are from X1000 to X1014. The addresses in the above range are always specified.

They cannot therefore be changed when the I/O points are assigned to them. (3) NC signals whose addresses are fixed and that are input from the machine tool nc Be sure to assign the following signals to be input from the machine tool to the specified addresses because the NC unit refers to the following fixed addresses during processing. w w w .c
NOTE If both I/O Link and built-in I/O card are provided, the address of the I/O card is valid. (Except Series 15) 53 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Table 3.2.1 (a) Input Signals Whose Addresses Are Fixed (Series 16/Series 18) M t system When the I/O Link is used When the built-in I/O card is used Signal indicating that X-axis
measurement position is reached XAE X4.0 X1004.0 Signal indicating that Z-axis measurement position is reached ZAE X4.1 X1004.1 Function B for directly entering the measurement value of tool compensation in the positive X direction +MIT1 X4.2 X1004.2 Function B for directly entering the measurement value of tool compensation in the negative
X direction –MIT1 X4.3 X1004.3 Function B for directly entering the measurement value of tool compensation in the positive Z direction +MIT2 X4.4 X1004.4 Function B for directly entering the measurement value of tool compensation in the negative Z direction –MIT2 X4.5 X1004.5 Signal indicating that X-axis measurement position is reached XAE
X4.0 X1004.0 Signal indicating that Y-axis measurement position is reached YAE X4.1 X1004.1 Signal indicating that Z-axis measurement position is reached ZAE X4.2 X1004.2 Common Skip signal om Signal T system t Symbol r.c Address SKIP X4.7 X1004.7 *ESP X8.4 X1008.4 Deceleration signal for 1st axis reference position return *DEC1 X9.0
X1009.0 Deceleration signal for 2nd axis reference position return *DEC2 X9.1 X1009.1 Deceleration signal for 3rd axis reference position return *DEC3 X9.2 X1009.2 Deceleration signal for 4th axis reference position return *DEC4 X9.3 X1009.3 Deceleration signal for 5th axis reference position return *DEC5 X9.4 X1009.4 Deceleration signal for 6th
axis reference position return *DEC6 X9.5 X1009.5 Deceleration signal for 7th axis reference position return *DEC7 X9.6 X1009.6 Deceleration signal for 8th axis reference position return *DEC8 X9.7 X1009.7 ce nt e Emergency stop signal .c nc If the NC is a TT system, the signals for tool post 2 listed in Table 3.2.1 (b) are always assigned to the
following addresses. In addition, the system does not have the signals for tool post 1, DEC5 to DEC8 (X9.4 to X9.7). Table 3.2.1 (b) Input Signals Whose Addresses Are Fixed (TT) (Series 16/Series 18) When the I/O Link is used When the built-in I/O card is used XAE X13.0 X1013.0 ZAE X13.1 X1013.1 +MIT1 X13.2 X1013.2 Function B for directly
entering the measurement value of tool compensation in the negative X direction –MIT1 X13.3 X1013.3 Function B for directly entering the measurement value of tool compensation in the positive Z direction +MIT2 X13.4 X1013.4 Function B for directly entering the measurement value of tool compensation in the negative Z direction –MIT2 X13.5
X1013.5 w Symbol Signal indicating that X-axis measurement position is reached w TT system t Address Signal indicating that Z-axis measurement position is reached w Function B for directly entering the measurement value of tool compensation in the positive X direction Skip signal SKIP X13.7 X1013.7 Deceleration signal for 1st axis
reference position return *DEC1 X7.0 X1007.0 Deceleration signal for 2nd axis reference position return *DEC2 X7.1 X1007.1 Deceleration signal for 3rd axis reference position return *DEC3 X7.2 X1007.2 Deceleration signal for 4th axis reference position return *DEC4 X7.3 X1007.3 54 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Table
3.2.1 (c) Input Signals Whose Addresses Are Fixed (Series 15) Emergency g y stop address nit Type of I/O unit Skip p signal g address Measurement position reached signal AE1 (XAE) AE2 (ZAE) Connection unit X6.4 X11.6 X8.3 X8.4 I/O unit X6.4 X11.6 X8.3 X8.4 ce nt e r.c Assignment of I/O Module Addresses The sequence program addresses of
each module should be decided by the machine tool builder. These decided addresses are set to the programmer memory by using programmer. The address information being set to the programmer is written together with a sequence program into ROM when a sequence program is written into ROM.
No I/O address is changeable in the written stage of the address information into ROM. These addresses are determined by the connecting position (group number and base number) of the I/O base unit, each module position (slot number) mounted inside the I/O base unit and each module name. Fig. 3.2.2 (a) and Fig. 3.2.2 (b) indicate the
configuration of the I/O base unit in PMC-RB/RC. For the specifications and details of connections of the I/O interface module, I/O module, CPU module, and other modules, see CONNECTING MANUAL of Series 16. om 3.2.2 PMC w w w .c nc I/O control unit 55 I/O Unit I/O Unit AIF01A AIF01B Base#0 Base#1 3.2.2 (a) Group #0 3. ADDRESS PMC
SEQUENCE PROGRAM B–61863E/10 PMC I/O Unit I/O Unit AIF01A AIF01B Base#0 Base#1 I/O Unit I/O Unit AIF01A Base#0 Group #1 AIF01B Base#1 Group #2 r.c Power Mate Group #0 om I/O control unit Base#0 Connection unit Group #3 ce nt e Base#0 Possible to connect max. 16 groups Max. 2 Base/Group Max. 10 Slot/Base 3.2.2 (b) w w w
.c nc (1) Group No. Up to two I/O units can be connected using the additional I/O interface module AIF01B, based on I/O interface AIF01A. Up to two I/O units extended from AIF01A are called a group. When only one interface module is not enough to accommodate the required I/O modules, or when multiple I/O units are separately located remote

from the machine, connect the first AIF01A and the second AIF01A with the cable. Up to 16 groups of I/O units can be connected. (2) Base No. In one group, there are 2 max. I/O base units. The I/O unit with the I/O interface module IF01A is assigned to base No. 0 and another is assigned to base No.1. (3) Slot No. A maximum of 5 or 10 I/O modules can be mounted on the I/O base unit ABU05A, ABU10A, respectively. The module mounting position on the I/O base unit is expressed with slot No. In each base unit, the mounting position of the I/O interface module is assigned to slot No. 0, and slot No. 1, 2, 3 ... are assigned in order from the left. In the case of I/O base unit (BU10B) for 10 slots, slot No. 1, 2 follow slot No. 8. The last slot No. 1, 2 are assigned for the next base address. Each module can be mounted on an arbitrary slot.
It is possible to mount modules by skipping some slots. 56 3.
ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 (4) Module name For module names, see Table 3.2.2 (a) to (c). An actual module name begins with A. When specifying a module name, omit the first letter A from the module name. Example) When specifying module AID16D, enter ID16D.
Table 3.2.2 (a) Input Modules 3 Rated voltage Rated current Polarity Response time Non-insulation DC input ID32A (AID32A) 24VDC 7.5mA Both 20 ms max. ID32B (AID32B) 24VDC 7.5mA Both 2 ms max.
ID16C (AID32C) 24VDC 7.5mA NEG 16 ID16D (AID16D) 24VDC 7.5mA POS 20 ms max. ID32E (AID32E) 24VDC 7.5mA Both 20 ms max. ... Number of points 16 AC input IA16G (AIA16G) 100 to 120VAC Insulation DC input Non-insulation DC input Polarity Number of input points 14.5mA (AC120V) Terminal Indication by LED ...
[illegible]
AD04A (AAD04A) Input: 8 bytes 4 Operator's panel connection unit OC01I Input: 12 bytes OC01O Output: 8 bytes /8 Input: 8 bytes /4 Output: 4 bytes OC02I Input: 16 bytes OC02O Output: 16 bytes OC03I Input: 32 bytes OC03O Output: 32 bytes /V Input: Output: V bytes V bytes OC02I Input: 16 bytes OC02O Output: 16 bytes OC03I Input: 32 bytes OC03O Output: 32 bytes #V Input: Output: V bytes V bytes Specify 1 to 10 in ## Input: 4 bytes Specify an area for reading the power-on/off state of each I/O unit model B. /V Input: Output: V bytes V bytes Specify 1 to 8 in V. 8 I/O unit model B Special modules not listed in Tables 3.1.2(a) and 3.1.2(b) r.c Ordering drawing No. A16B-2200-0661 (Sink type) ty e) A16B-2201-0731 (Source type) ce nt e w w 9 I/O link connection unit nc 7 Machine operator's panel interface unit .c 6 Operator's panel connection unit w 5 Ordering drawing No. A16B-2200-0660 (Sink type) ty e) A16B-2201-0730 (Source type) 58 Specify 1 to 8 in V. V.
Module name (actual module name) 10 Connector panel I/O module (Note 3) Specifications CM03I (/3) Input 3 bytes Basic unit only CM06I (/6) Input 6 bytes Expansion unit 1 is used.
CM09I Expansion unit 2 is used. CM12I (OC01I) Input 12 bytes Expansion unit 3 is used. CM13I Input 13 bytes The first MPG unit is used. CM14I Input 14 bytes The second MPG unit is used. CM15I Input 15 bytes The third MPG unit is used. CM16I (OC02I) Input 16 bytes DO alarm detection is used.
CM02O (/2) Output 2 bytes Basic unit only CM04O (/4) Output 4 bytes Expansion unit 1 is used. CM06O (/6) Output 6 bytes Expansion unit 2 is used. CM08O (/8) Output 8 bytes Expansion unit 3 is used.
CM12I (OC01I) Input 12 bytes CM13I Input 13 bytes The first MPG unit is used. ce nt e Operator's panel I/O module (Note 3) Occupied address om Name r.c 11 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 CM14I Input 14 bytes The second MPG unit is used. CM15I Input 15 bytes The third MPG unit is used. CM16I (OC02I) Input 16 bytes DO alarm detection is used. CM08O (/8) Output 8 bytes w w w .c nc NOTE 1 For the method of I/O link connection unit assignment, see Subsection 3.2.3. 2 For the method of I/O link model B assignment, see Subsection 3.2.4. 3 If the version of the programming system (FAPTLADDER, FAPTLADDER–II) is too old to match a module above, use the compatible module indicated in parentheses. When a compatible module having the same number of points is not available, use a compatible module having a greater number of points. 59 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 GROUP BASE ce nt e ADDRESS r.c om (5) How to set address to each module The character and the mount position of each module is now decided with the group number, base number, slot number, and module name, so the address of each module can now be decided, corresponding these data and the input/output addresses. After display the I/O unit address screen as shown below on the programmer's CRT, set necessary data on the screen, Then the module address is now assigned. The occupying DI/DO points (bytes) of each module are stored in the programmer, so just assign the address of the head byte of each module, and the addresses of the other bytes in the module are automatically assigned by the programmer. For instance, when the module ID32A is assigned address X5 as in Fig. 3.2.2 (d), the necessary 4 bytes are automatically secured. For details on operation, see Chapters III, IV, "Programmer". The input/output addresses of each module can be freely decided in this method at the machine tool builder, so the address can be decided when making the ladder diagram, as long as it does not duplicate with the addresses of each module. SLOT NAME X000 X001 X002 X003 X004 0 0 5 ID32A X006 0 0 5 ID32A X007 0 0 5 ID32A X008 0 0 5 ID32A nc Automaticall set X005 3.2.2 (d) I/O unit address screen NOTE When assigning Connection unit 1, Connection unit 2 or Connection unit for operator's panel, set base number to 0 and slot number to 1. w w w .c X009 60 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 (6) Notes when setting addresses ADDRESS GROUP om (a) The head bytes of the analog input module (AD04A) and analog output module (DA02A) must be assigned to even number addresses of input address (XVVV), and output address (YVVV) each. When reading the A/D-converted digital value from the input address (XVVV) or when writing the D/A-converting value to the output address (YVVV), readout and write-in must always be done in word (16 bits) units. BASE SLOT 1 ID16C 1 ID16C X000 0 0 X001 0 0 X002 0 0 X003 0 X004 1 X005 1 X006 1 X007 1 NAME ID16D 2 ID16D 0 1 IA16G 0 1 IA16G 0 2 IA16G 0 2 IA16G ce nt e r.c 2 0 X008 2 0 1 ID16D X009 2 a 0 1 ID16D nc 3.2.2 (e) Concept: 3.2.3 In conventional data transfer, when data is to be transferred between CNC A and CNC B, the I/O units indicated by (a) (figure below) must be connected with each other. (In this case, data can be transferred using any I/O unit.) CNC A I/ O Unit Model A CNC B (a) I/ O Unit Model A I/ O Unit Model A I/ O Unit Model A I/ O Unit Model A The I/O link connection unit replaces these I/O units, thus eliminating the need to connect them with, for example, cables.
61 3. ADDRESS PMC SEQUENCE PROGRAM I/ O Unit Model A I/ O Unit Model A CNC B I/ O Unit Model A ²This portion is replaced by an I/O link connection unit. (a) I/ O Unit Model A I/ O Unit Model A om CNC A B–61863E/10 I/ O Unit Model A r.c Consequently, when the I/O link connection unit is used, the connections become as shown below. CNC A I/ O Unit Model A CNC B ce nt e I/ O Unit Model A I/O Link connection unit nc I/ O Unit Model A Method of assignment: 1 to 8 Input unit name at the time of assignment Output unit name at the time of assignment / V (V represents a number from 1 to 8.) / V (V represents a number from 1 to 8.) OC02I OC02O OC03I OC03O w 16 32 The assignment data depends on what type of I/O unit is to be replaced with an I/O link connection unit. .c Occupied address I/ O Unit Model A When a connection unit that occupies 16-byte addresses is attached to the input side in GROUP = 1, enter "1.0.1.OC02I." w w Setting: 62 3.
ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 D Related hardware publications: FANUC I/O Unit-MODEL B Connection and Maintenance Manual (B–62163E) 3.2.4 I/O Unit MODEL B Assignment om I/O Unit MODEL Bs can be used together with a Power Mate operator panel interface unit, connection unit, and I/O Unit MODEL As. In this case, the I/O Unit MODEL Bs occupy one group; that is, no other type of unit can be present in that group. An example of connection is shown below. [GROUP] =0 r.c I/ O LINK MASTER Power Mate [BASE] =0, [SLOT] =1 ce nt e [GROUP] =1 Operator's panel interface unit [BASE] =0, [SLOT] =1 [GROUP] =2 →SLOT No. →SLOT No. I/O Unit- A I/O Unit- A nc [GROUP] =3 [BASE] =0 [BASE] =1 w w w .c I/O unit model B interface unit I/O unit model B DI/DO unit (Unit No.=1) [BASE] =0, [SLOT] =1 I/O unit model B DI/DO unit (Unit No.=5) [BASE] =0, [SLOT] =5 I/O unit model B DI/DO unit (Unit No.=30) [BASE] =0, [SLOT] =30 63 I/O unit model B DI/DO unit (Unit No.=20) [BASE] =0, [SLOT] =20 I/O unit model B DI/DO unit (Unit No.=10) [BASE] =0, [SLOT] =10 I/O unit model B DI/DO unit (Unit No.=9) [BASE] =0, [SLOT] =9 3. ADDRESS PMC SEQUENCE PROGRAM Specify a group number in [GROUP]. Always specify 0 in [BASE]. Specify the unit number of an I/O unit model B in [SLOT]. But when you assign the power–on/off information, specify 0 in [SLOT]. The data specified by [SLOT] and [NAME] is as follows: [SLOT] = 0, 1, ...30: Unit number (1 to 30) of an I/O unit model B DI/DO unit [NAME]: Addresses occupied by an I/O unit model B Input/output size of ([base unit] + [extended unit]) Assigned name 1 byte #1 2 bytes #2 3 bytes #3 4 bytes #4 6 bytes #6 8 bytes #8 10 bytes Power-on/off information Occupied address om Input/output: 1 byte Input/output: 2 bytes Input/output: 3 bytes Input/output: 4 bytes Input/output: 6 bytes Input/output: 8 bytes r.c Method of assignment: B–61863E/10 #10 Input/output: 10 bytes ## Input: 4 bytes When an I/O unit model B assigned unit number 10 and occupying an area of 3 bytes is attached to the input with GROUP = 1, enter "1.0.10.#3." 3.2.5 When a Power Mate–MODEL D/H is used as I/O Link slave, it need to be assigned on the I/O Link master side. On the I/O link slave side, fixed addresses are used, so that no address needs to be assigned. (See Table 3 for the addresses used.) An example of connection is shown below.
Power Mate–model D/H Assignment ce nt e Setting: nc I/ O LINK MASTER Operator's panel interface unit [BASE] =0, [SLOT] =1 [GROUP] =1 w w .c [GROUP] =0 w Method of assignment: Power Mate [BASE] =0, [SLOT] =1 Specify a group number in [GROUP]. Always specify 0 in [BASE]. Always specify 1 in [SLOT]. The data specified by [NAME] is as follows: I/O points (input/output) Input unit name at the time of assignment Output unit name at the time of assignment 32/32 FS04A FS04A 64/64 FS08A FS08A 128/128 OC02I OC02O 256/256 OC03I OC03O 64 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Setting: w w w .c nc ce nt e r.c om When a Power Mate–D of 256/256 points is connected with group 1, input the undermentioned assignment data. ·Input side : "1.0.1.OC03I" ·Output side : "1.0.1.OC03O" 65 3. ADDRESS PMC SEQUENCE PROGRAM PA1 PA3 Number of bytes 1100 1118 Model RA1 RA2 RA3/RA5 Number of bytes 1100 1118 1118 Model RB RB2 RB3/RB5 RB4/RB6 Number of bytes 1100 1118 1618 3200 Model RC RC3 RC4 Number of bytes 1600 1618 NB NB2 1618 3200 om Model ce nt e INTERNAL RELAY ADDRESSES (R) In each model, the following signals (bytes) can be used as internal relays. This area is cleared to zero when the power is turned on. r.c 3.3 B–61863E/10 Model w w w .c nc Number of bytes 66 3200 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Address number ± 7 6 5 4 3 2 1 0 R0 PMC-PA1 PMC-PA3 PMC-RA1 PMC-RA2 PMC-RA3 PMC-RA5 PMC-RB PMC-RB2 R1 · · · · · · · R1499 r.c · · · ce nt e R2999 PMC-PA1 PMC-RA1 PMC-RB PMC-RC R9000 · · · · R9099 · R9117 · · .c · nc · · PMC-RB4 PMC-RB6 PMC-RC4 PMC-NB2 om R999 PMC-RB3 PMC-RB5 PMC-RC PMC-RC3 PMC-NB w w w R9199 67 PMC-PA3 PMC-RA2 PMC-RA3 PMC-RA5 PMC-RB2 PMC-RB3 PMC-RB5 PMC-RC3 PMC-NB PMC-RB4 PMC-RB6 PMC-RC4 PMC-NB2 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 3.3.1 Area Managed by the System Program (1) R9000 (Operation output register for the ADDB, SUBB, MULB, DIVB, and COMPB functional instructions) 7 6 5 4 3 2 1 0 R9000 om The result is 0. The result is a negative value. The result overflows. 7 6 5 r.c (2) R9000 (Error output for the EXIN, WINDR, WINDW, MMCWR, MMCWW, MMC3R, and MMC3W functional instructions) 4 3 1 0 The instruction ended in error. ce nt e R9000 2 (3) R9002 to R9005 (Operation output registers for the DIVB functional instruction) The data remaining after the DIVB functional instruction is executed is output. w w w .c nc (4) R9010 to R9027 (Interface area for the FNC9x functional instruction) (PMC-RC only) The area is provided as an interface between the FNC9x functional instruction to be executed and a desired function. 7 6 5 4 3 2 1 0 R9010 97 96 95 94 93 92 91 90 Contents of the command to be executed R9011 97 96 95 94 93 92 91 90 Data output when the processing is completed R9012 Addresses of the control data for SUB90 R9013 R9014 Addresses of the control data for SUB91 R9015 R9026 Addresses of the control data for SUB97 R9027 68 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 (5) R9100 to R9117 (Interface area for the FNC9x functional instruction) (PMC-RC3/RC4/NB/NB2 only) The area is provided as an interface between the FNC9x functional instruction to be executed and a desired function. 6 5 4 3 2 1 0 R9100 97 96 95 94 93 92 91 90 Contents of the command to be executed R9101 97 96 95 94 93 92 91 90 Data output when the processing is completed om 7 R9102 Addresses of the control data for SUB90 R9103 r.c R9104 Addresses of the control data for SUB91 R9105 ce nt e R9116 Addresses of the control data for SUB97 R9117 (6) R9091 (System timer) 4 signals can be used as system timer.
The specifications of every signal are as following.
5 4 3 2 w w 1 0 always OFF always ON Cyclic signal of 200 ms (104 ms ON, 96 ms OFF) w .c R9091 6 nc 7 Cyclic signal of 1 second. (504 ms ON, 496 ms OFF) CAUTION In the beginning, every signal is OFF. The signals of R9091.0 and R9091.1 are always set at the beginning of 1st level in every cycle. Every pulse signal (ON-OFF) includes 8 ms errors.
69 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 R9091. 5 104ms 96ms 200ms 504ms 496ms w w w .c nc ce nt e r.c 1 second om R9091. 6 70 3. ADDRESS PMC SEQUENCE PROGRAM ADDRESSES FOR MESSAGE SELECTION DISPLAYED ON CRT (A) This area is used as message display request. In each model, the following number of messages can be used. Where "Number of Messages" = "Number of Bytes" 8 This area is cleared to zero when the power is turned on. For information about using the message, see the subsection "5.43". Model PA1 PA3 25 25 Number of messages 200 200 Model RA1 RA2 RA3/ RA5 25 25 25 Number of messages 200 200 200 Model RB RB2 RB3/ RB5 RB4/ RB6 25 25 25 125 Number of messages 200 200 200 1000 Model RC RC3 RC4 25 25 125 200 200 1000 NB NB2 Number of bytes ce nt e Number of bytes om 3.4 r.c B–61863E/10 Number of bytes Number of bytes nc Number of messages Model .c Number of bytes w w w Number of messages 71 25 200 125 1000 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Address number 6 5 4 3 2 1 PMC-PA1 PMC-PA3 PMC-RA1 PMC-RA2 PMC-RA3 PMC-RA5 PMC-RB PMC-RB2 PMC-RB3 PMC-RB5 PMC-RC PMC-RC3 PMC-NB A0 PMC-RB4 PMC-RB6 PMC-RC4 PMC-NB2 r.c A1 ce nt e A24 A124 0 om 7 w w w .c nc 3.4 Address of message display reguest 72 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 3.5 PA1 PA3 Number of bytes 80 80 Number of counters 20 20 Model RA1 RA2 RA3/ RA5 80 80 Number of bytes 80 Number of counters 20 r.c Model 20 20 Model RB RB2 RB3/ RB5 RB4/ RB6 Number of bytes 80 80 80 200 Number of counters 20 20 20 50 Model RC RC3 RC4 Number of bytes 80 80 200 Number of counters 20 20 50 Model NB NB2 Number of bytes 80 200 20 50 .c nc ce nt e ADDRESS OF COUNTER (C) om This area is used as counters. In each model, the following number of counters can be used. Where "Number of Counters" = "Number of Bytes" /4 Since this area is nonvolatile, the contents of the memory do not disappear even when the power is turned off.
w w w Number of counters 73 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Address number 7 C0 6 5 4 3 2 1 0 PMC-PA1 PMC-RB4 PMC-PA3 PMC-RB6 PMC-RA1 PMC-RC4 Counter PMC-RA2 PMC-NB2 No. 1 PMC-RA3 Preset value C1 C2 om current value PMC-RA5 PMC-RB C3 PMC-RB2 PMC-RB3 PMC-RB5 r.c PMC-RC PMC-RC3 C76 PMC-NB Preset value C77 ce nt e Counter No. 20 C78 current value C79 C197 Counter No. 50 current value .c C198 Preset value nc C196 3.5 Address of Counter w w w C199 74 3. ADDRESS PMC SEQUENCE PROGRAM 3.6 ADDRESS OF KEEP RELAY AND NONVOLATILE MEMORY CONTROL (K) The area is used as keep relays and PMC parameters. In each model, the following number of bytes can be used. Since this area is nonvolatile, the contents of the memory do not disappear even when the power is turned off. Model Number of bytes Nonvolatile memory control address PMC control software parameter PA1 PA3 20 20 K16 K16 K17 to K19 r.c K19 K17 to om B–61863E/10 Model RA2 RA3/ RA5 20 20 20 ce nt e Number of bytes RA1 Nonvolatile memory control address K16 K16 K16 PMC control software parameter K17 to K17 to K17 to K19 K19 K19 RB RB2 RB3/ RB5 RB4/ RB6 20 20 20 50 Nonvolatile memory control address K16 K16 K16 K16 PMC control software parameter K17 to K17 to K17 to K900 to Model .c nc Number of bytes w w w Model Number of bytes Nonvolatile memory control address PMC control software parameter K19 K19 K19 RC RC3 RC4 20 20 50 K16 K16 K16 K17 to K900 to K17 to K19 Model K19 NB NB2 20 50 Nonvolatile memory control address K16 K16 PMC control software parameter K17 to K900 to Number of bytes K19 75 K909 K909 K909 3. ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Address number 6 5 4 3 2 1 K0 PMC-PA1 PMC-PA3 PMC-RA1 PMC-RA2 PMC-RA3 PMC-RA5 PMC-RB PMC-RB2 PMC-RB3 PMC-RB5 PMC-RC PMC-RC3 PMC-NB PMC-RB4 PMC-RB6 PMC-RC4 PMC-NB2 nc ce nt e K19 K900 PMC-RB4 PMC-RB6 PMC-RC4 PMC-NB2 r.c K1 K39 0 om 7 .c K909 w w w 3.6 Address of Keep Relay and Nonvolative Memory Control For the information about using "Nonvolatile memory control", see the section "6.1". PMC control software parameter area is used by PMC control software. For more information about PMC control software parameter, see the section "II 4.3".
76 3.
ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 3.7 Model PA1 PA3 Number of bytes 1860 1860 Model RA1 RA2 RA3/ RA5 Number of bytes 1860 1860 1860 RB RB2 RB3/ RB5 1860 1860 3000 RC RC3 RC4 3000 3000 8000 Model Number of bytes Model Model Number of bytes Address number 6 5 4 3 2 1 D0 .c D1 NB NB2 3000 8000 0 nc 7 8000 ce nt e Number of bytes RB4/ RB6 r.c ADDRESS OF DATA TABLE (D) om Data table is the area of nonvolatile memory. In each model, the following number of bytes can be used. PMC-PA1 PMC-RB3 PMC-RB4 PMC-PA3 PMC-RB5 PMC-RB6 PMC-RA1 PMC-RC PMC-RC4 PMC-RA2 PMC-RC3 PMC-NB2 PMC-RA3 PMC-NB w PMC-RA5 PMC-RB PMC-RB2 w w D1859 D2999 D7999 3.7 Address of Data Table 77 3. ADDRESS PMC SEQUENCE PROGRAM Model PA1 PA3 Number of bytes 80 80 Number of timers 40 40 RA1 RA2 RA3/ RA5 Number of bytes 80 80 80 Number of timers 40 Model RB 40 40 RB2 RB3/ RB5 RB4 RB6 ce nt e Model om TIMER ADDRESSES (T) This area is used by TMR instruction as variable timers. In each model, the following number of timers can be used. Where "Number of timers" = "Number of Bytes" / 2 Since this area is nonvolatile, the contents of the memory do not disappear even when the power is turned off. r.c 3.8 B–61863E/10 80 80 80 300 Number of timers 40 40 40 150 Model RC RC3 RC4 Number of bytes 80 80 300 Number of timers 40 40 150 Model NB NB2 Number of bytes 80 300 40 150 .c nc Number of bytes w w w Number of timers 78 3.
ADDRESS PMC SEQUENCE PROGRAM B–61863E/10 Address number 6 5 4 3 2 1 0 T0 Timer PMC-PA1 PMC-RB4 No. 1 PMC-PA3 PMC-RB6 PMC-RA1 PMC-RC4 PMC-RA2 PMC-NB2 T1 T2 Timer PMC-RA3 No. 2 PMC-RA5 T3 om 7 PMC-RB PMC-RB2 PMC-RB3 PMC-RB5 r.c PMC-RC PMC-RC3 Timer T78 No. 40 ce nt e T79 PMC-NB Timer T298 No. 150 nc T299 w w w .c 3.8 Timer Address 79 3. ADDRESS PMC SEQUENCE PROGRAM Model Number of labels Model Number of labels Model PA3 – 9999 RA1 RA2 RA3/ RA5 – – 9999 RB RB2 – – RB3/ RB5 9999 RB4/ RB6 9999 ce nt e Number of labels PA1 om LABEL ADDRESSES (JMPB, JMPC, LBL) (L) Label addresses are used to specify jump destination labels (positions in a sequence program) in the JMPB and JMPC instructions. The same label number can appear in different LBL instructions in the same sequence program as long as it is unique in the program unit (main program, subprogram). In each model, the following number of label can be used.
r.c 3.9 B–61863E/10 Model Number of labels RC RC3 RC4 – 9999 9999 NB/ NB2 Number of labels 9999 w w w .c nc Model 80 3. ADDRESS PMC SEQUENCE PROGRAM SUBPROGRAM NUMBERS (CALL, CALLU, SP) (P) Subprogram numbers are used to specify jump destination subprogram labels in the CALL and CALLU instructions. Subprogram number must be unique in the entire sequence program. In each model, the following number of subprograms can be used. Model Number of subprograms Model Number of subprograms Model PA3 – 512 RA1 RA2 – – RA3/ RA5 512 RB RB2 RB3/ RB5 RB3/ RB6 – – 512 2000 ce nt e Number of subprograms PA1 om 3.10 r.c B–61863E/10 Model RC RC3 RC4 – 512 2000 Model NB NB2 Number of subprograms 512 2000 w w w .c nc Number of subprograms 81 4. PMC BASIC INSTRUCTIONS 4 PMC SEQUENCE PROGRAM B–61863E/10 PMC BASIC INSTRUCTIONS om Designing a sequence program begins with writing a ladder diagram. The ladder diagram is written using relay contact symbols and functional instruction code. (These will be described later.) Logic written in the ladder diagram is entered as a sequence program in the Programmer. ce nt e r.c There are two sequence program entry methods. One is the entry method with the mnemonic language (PMC instructions such as RD, AND and OR). The other is the relay symbol method (using relay symbols such as , and) in which the sequence program is entered by using the relay contact symbols and the functional instruction symbols of the ladder diagram. When the relay symbol method is used, the ladder diagram format can be used and programming can be performed without understanding the PMC instructions (basic instructions such as RD, AND and OR). Actually, however, the sequence program entered by the relay symbol method is also internally converted into the instruction corresponding to the PMC instruction. When the sequence program is punched on a paper tape and then entered to the programmer, programming must be performed with the PMC instructions. nc Also, the meanings of the functional instructions described later must be understood fully. See Subsection 4.1 and Section 5. On how to enter the sequence program into the programmer by using the PMC instructions and relay symbols, see Chapter III or IV. .c The following should be noted first before reading the explanation on PMC instructions.
w w w This manual describes the entry method using mnemonic language. (1) Signal address Relay coils and contacts written in a ladder diagram are each given an address, represented with an address number and a bit number. (See Fig. 4 (a)) Zero suppression is possible for the head zero. For details of address, see Section 3. Signal name Relay name A B RO X8.1 C R12.6 R9.0 Y20.4 Bit number Address number 4 (a) Address of signal 82 4. PMC BASIC INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (2) Type There are two types of PMC instructions, basic and functional. (a) Basic instruction Basic instructions are most often used when designing sequence programs. They perform one-bit operations, such as AND, or OR. There are 12 types. om (b) Functional instruction Functional instructions ease programming of machine movements that are difficult to program with basic instructions. Refer to Chapter V about the type of functional instruction. r.c (3) Storage of logical operation results A register is provided for storing the intermediate results of a logical operation during operation of a sequence program. This register consists of 9 bits. (See Fig. 4 (b)) . ce nt e Stack register (which temporarily stores the inter- The result of an operation currently being executed mediate result of an operation) enters here. ST8 ST7 ST6 ST5 ST4 ST3 ST2 ST1 ST0 4 (b) w w w .c nc Execution of an instruction (RD.STK or the like) to temporarily store the intermediate results of an operation as in the above figure, shifts left and stacks the status stored so far; conversely, execution (AND.STK or the like) to retrieve a stacked signal shifts it right. The signal stacked last is retrieved first. Refer to explanations of each instruction for concrete applications and operations. 83 4. PMC BASIC INSTRUCTIONS PMC SEQUENCE PROGRAM 4.1 B–61863E/10 The type of instructions and contents of processing are listed in the Table 4.1.(a). DETAILS OF BASIC INSTRUCTIONS Information format 1: This is used when writing instructions on a coding sheet, punching out them on a paper tape or displayed on the CRT/MDI or offline programmer. om Information format 2: This is used when inputting instructions through programmer. This format is to simplify an input operation. RN, for instance, means RD.NOT and represents an input operation using both keys, "R" and "N". Details of each basic instruction will be given here. r.c Table 4.1 (a) Basic instruction and processing Instruction Format 1 (coding) Contents of processing Format 2 (keys ofFAPT LADDER) RD R 2 RD.NOT RN 3 WRT W 4 WRT.NOT 5 AND 6 AND.NOT 7 OR 8 OR.NOT 9 RD.STK 10 RD.NOT.STK 11 AND.STK 12 WN A AN Inverts the logical status of a specified signal, reads and sets it in ST0. Outputs the results of logical operations (status of ST0) to a specified address. Inverts the results of logical operations (status of ST0) and outputs it to a specified address. Induces a logical product. Inverts the status of a specified signal and induces a logical product. Induces a logical sum. nc O Reads the status of a specified signal and sets it in ST0. ce nt e 1 Inverts the status of a specified signal and induces a logical sum. RS Shifts the stack register left one bit, read and sets the status of a specified signal in ST0. RNS Shifts the stack register left one bit reads the inveried logical status of a specified signal, and sets it in ST0. .c ON Sets the logical product of ST0 and ST1, and shifts the stack register right one bit. OR.STK OS Sets the logical sum of ST0 and ST1, and shifts the stack register right by one bit. 13 SET SET Calculates the logical OR of the contents of ST0 and the status of the signal at the specified address and outputs the result to the specified address. 14 RST RST Calculates the logical AND of the inverted contents of ST0 and the specified address and outputs the result to the address. w w w AS 84 PMC SEQUENCE PROGRAM B–61863E/10 4. PMC BASIC INSTRUCTIONS Basic instructions available on each models are as shown in the "Table 4.1(b)". Table 4.1(b) Basic instruction Model Instruction PMCĆ PA1 PMCĆ PA3 PMCĆ PMCĆ RA1/ RB/ RA2 RB2 PMCĆ RC PMCĆ PMCĆ PMCĆ PMCĆ RA3/ RB3/ RC3/ NB/ RA5 RB4/ RC4 NB2 RB5/ RB6 RD f f f f f f f f f 2 RD.NOT f f f f f f f f f 3 WRT f f f f f f f f f 4 WRT.NOT f f f f f f f f f 5 AND f f f f f f f f f 6 AND.NOT f f f f f f f f f 7 OR f f f f f f f f f 8 OR.NOT f f f f f f f f f 9 RD.STK f f 10 RD.NOT.STK f f 11 AND.STK f f 12 OR.STK f f 13 SET f f 14 RST f f ce nt e r.c 1 om No. f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f nc : Cannot be used f : Can be used w w w .c NOTE SET/RST are not available on PMC-RA3 for Series 20. 85 4. PMC BASIC INSTRUCTIONS PMC SEQUENCE PROGRAM 4.1.1 B–61863E/10 (1) Format RD (Address) Bit number om Address number (2) Reads the status (1 or 0) of a signal at a specified address and sets it in ST0. r.c (3) Is used when beginning coding with contact A ( ). See the ladder diagram of Fig. 4.1.1 and entries in the coding sheet of Table 4.1.1 for an example of using the RD instruction. (4) The signal read by the RD instruction may be any signal entered as the logical condition for one coil (output). B C ce nt e A X2.0 G X5.1 E R5.4 W2 R200.1 Y5.3 4.1.1 Ladder diagram Table 4.1.1 Coding for Fig. 4.1.1 w w Instruction R200.0 Y5.2 F Coding sheet Step Number R2.1 D .c nc X10.1 W1 Address No. Status of operating result Bit No. Remarks ST2 ST1 ST0 RD X10 . 1 A A 2 AND X2 . 0 B A.B 3 AND . NOT 4 WRT 5 w 1 R2 . 1 C A.B.C R200 . 0 W1 output A.B.C RD X5 . 1 D D 6 OR . NOT Y5 . 2 E D+E 7 OR Y5 . 3 F D+E+F 8 AND R5 . 4 G (D+E+F)@G 9 WRT R200 . 1 W2 output (D+E+F)@G 10 86 4. PMC BASIC INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 4.1.2 (1) Format RD . NOT (Address) Bit number om Address number (2) Inverts the status of a signal at a specified address and set it in ST0. (3) Is used when beginning coding with contact B ( ). See the ladder diagram of Fig. 4.1.2 and entries in the coding sheet of Table 4.1.2 for an example of using the RD.NOT instruction. B C ce nt e A r.c (4) The signal read by the RD.NOT instruction may be any contact B entered as the logical condition of one coil. R1.1 F2.2 D R210.1 G G5.1 E R10.5 W2 R210.2 X4.2 F Y10.7 4.1.2 Ladder diagram .c nc F3.3 W1 Table 4.1.2 Coding for Fig. 4.1.2 Step Number Instruction Address No. Bit No. Status of operating result Remarks ST1 ST0 R1 . 1 2 AND . NOT F2 . 2 B A.B 3 AND . NOT F3 . 3 C @C 4 WRT R210 . 1 W1 output @C 5 RD. NOT G5 . 1 D D 6 OR . NOT X4 . 2 E D+E 7 OR Y10 . 7 F D+E+F 8 AND R10 . 5 G (D+E+F)@G 9 WRT R210 . 2 W2 output (D+E+F)@G w A ST2 RD. NOT w 1 w Coding sheet 87 A 4.
PMC BASIC INSTRUCTIONS PMC SEQUENCE PROGRAM 4.1.3 B–61863E/10 (1) Format WRT (Address) Bit number om Address number (2) Outputs the results of logical operations, that is, the status of ST0 to a specified address.
r.c (3) The results of one logical operation can also be output to two or more addresses. How to use the WRT instruction in this case is shown in Fig. 4.1.3 and Table 4.1.3. C ce nt e A R220.1 B G2.2 X4.2 W1 Y11.1 W2 Y14.6 4.1.3 Ladder diagram nc Table 4.1.3 Coding for Fig.
4.1.3 Coding sheet Instruction RD 2 OR 3 AND 4 Remarks ST2 ST1 ST0 1 A A X4 . 2 B A+B G2 . 2 C (A+B)@C WRT Y11 . 1 W1 output (A+B)@C WRT Y14 . 6 W2 output (A+B)@C w w 5 Bit No. R220 . w 1 Address No. .c Step Number Status of operating result 88 4. PMC BASIC INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 4.1.4 (1) Format WRT. NOT (Address) Bit number Address number A C om (2) Inverts the results of logical operations, that is, the status of ST0 and outputs it to a specified address. Fig. 4.1.4 and Table 4.1.4 show an example on using the WRT.NOT instruction. r.c W1 R220.1 B G2.2 Y11.1 W2 ce nt e X4.2 Y14.6 4.1.4 Ladder diagram Table 4.1.4 Coding for Fig. 4.1.4 Coding sheet Instruction RD R220 . 2 OR 3 AND 4 WRT 5 WRT. NOT Bit No. 1 Remarks ST2 ST1 ST0 A A X4 . 2 B A+B G2 . 2 C (A+B) · C Y11 .
1 W1 output (D+E) · F Y14 . 6 W2 output (A+B) · C w w w .c 1 Address No. nc Step Number Status of operating result 89 4. PMC BASIC INSTRUCTIONS 4.1.5 PMC SEQUENCE PROGRAM B–61863E/10 (1) Format AND (Address) Bit number Address number 4.1.6 om (2) Induces a logical product. (3) See Fig. 4.1.1 and Table 4.1.1 for an example of using the AND instruction. (1) Format r.c AND. NOT (Address) ce nt e Bit number Address number (2) Inverts the status of a signal at a specified address and induces a logical product. (3) See Fig. 4.1.1 and Table 4.1.1 for an example of using the AND.NOT instruction. 4.1.7 (1) Format (Address) Bit number w w .c nc OR 4.1.8 Address number (2) Induces a logical sum. (3) See Fig. 4.1.1 and Table 4.1.1 for an example of using the OR instruction. (1) Format w OR. NOT (Address) Bit number Address number (2) Inverts the status of a signal at a specified address and induces a logical sum. (3) See Fig. 4.1.1 and Table 4.1.1 for an example of using the OR.NOT instruction.
90 4. PMC BASIC INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 4.1.9 (1) Format RD. STK (Address) Bit number Address number r.c om (2) Stacks the intermediate results of a logical operations. After shifting the stack register left one bit, sets a signal at a specified address to ST0.
(3) Is used when the signal to be specified is contact A ( ). (4) See Fig. 4.1.9 and Table 4.1.9 for an example of using the RD.STK instruction. A B W1 X1.1 C ce nt e Y1.2 D X1.3 E Y1.4 F R2.1 R3.5 Y15.0 nc 4.1.9 Ladder diagram Table 4.1.9 Coding for Fig. 4.1.9 Coding sheet Instruction RD 2 AND 3 4 Remarks ST2 ST1 ST0 1 A A Y1 . 2 B RD. STK X1 . 3 C C AND Y1 .
4 D   OR. STK w 5 Bit No. X1 . w 1 Address No. .c Step Number Status of operating result RD. STK R2 . 1 E E 7 AND R3 .
5 F   8 OR.STK 9 WRT w 6 Y15 . 0 W1 output 10 91 4. PMC BASIC INSTRUCTIONS PMC SEQUENCE PROGRAM 4.1.10 B–61863E/10 (1) Format RD. NOT.
STK (Address) Bit number Address number om (2) Stacks the intermediate results of a logical operations.
Shifts the stack register left one bit, inverts the status of a signal at a specified address and sets it in ST0. (3) Is used when the signal to be specified is contact B ( ).
A r.c (4) See Fig. 4.1.10 and Table 4.1.10 for an example of using the RD.NOT.STK instruction. B E F ce nt e W1 X1.0 C X1.1 D Y1.2 G Y1.3 H R1.4 R1.5 X1.6 Y1.7 Y15.7 4.1.10 Ladder diagram Table 4.1.10 Coding for Fig. 4.1.10 Instruction RD 2 AND. NOT 3 RD.NOT.STK 4 AND. NOT 5 OR. STK Bit No. X1 . 0 Remarks ST2 ST1 A ST0 A X1 . 1 B R1 . 4 C C R1 . 5 D   E E w 6 Address No. .c 1 Status of operating result nc Coding sheet Step Number   RD. STK Y1 . 2 AND Y1 . 3 F   RD.STK X1 . 6 G   G 9 AND. NOT Y1 . 7 H   10 OR. STK 7 w w 8 11 AND. STK 12 WRT   ()@() Y15 . 7 W1 output ()@() 13 14 92 PMC SEQUENCE PROGRAM B–61863E/10 4.1.11 4. PMC BASIC INSTRUCTIONS (1) Format AND.
STK (Address) Bit number Address number om (2) Induces a logical product from the operation results in ST0 and ST1, sets the result in ST1, and shifts the stack register right one bit. 4.1.12 r.c (3) See Fig. 4.1.10 and Table 4.1.10 for an example of using the AND.STK instruction. (1) Format OR. STK ce nt e (Address) Bit number Address number (2) Induces a logical sum from the operation results in ST0 and in ST1, sets the result in ST1, and shifts the stack register right one bit. nc (3) See Fig. 4.1.9 and Table 4.1.9 or Fig. 4.1.10 and Table 4.1.10 for examples of using the OR.STK instruction. w w w .c NOTE In Table 4.1.9 putting OR.STK at step 5 between steps 7 and 8 brings about the same result. But it is recommended to code as shown in Table 4.1.9, because coding OR.STK or AND.STK in succession is prone to cause an error. 93 4. PMC BASIC INSTRUCTIONS PMC SEQUENCE PROGRAM 4.1.13 B–61863E/10 (1) Format SET (Address) Bit No. Address No. om (2) Logical sum of the logical operation result ST0 with the content of the specified address is outputted to the same address. (3) Refer to the figure below for an example of using the SET instruction. A r.c C (S) Y0.0 ce nt e R0.0 B X0.0 4.1.13 Ladder diagram nc Table 4.1.13 Coding for Fig. 4.1.13 Coding sheet Instruction RD 2 OR 3 SET R0 . Bit No. Remarks 0 ST2 ST1 ST0 A A C X0 . 0 B A+B C Y0 . 0 Y0.0 output – (A+B) +C w w w 1 Address No. .c Step Number Status of operating result 94 – PMC SEQUENCE PROGRAM B–61863E/10 4. PMC BASIC INSTRUCTIONS (4) Remarks (a) Restriction of using Do not use SET/RST like the following example 1, use them alone like the following example 2. R0.0 Y0.0 R0.0 Y0.0 X0.0 (S) X0.0 Y0.1 f R0.0 D D D D D D om (S) Y0.1 r.c f X0.0 END2 ce nt e D D END2 Example 1 Example 2 w w w .c nc D The relation between COM and COME. The operation of SET/RST in the section of COM/COME is as follows. COM condition ON (ACT=1) : It operates usually.
COM condition OFF (ACT=0) : SET does not operate. 95 4. PMC BASIC INSTRUCTIONS PMC SEQUENCE PROGRAM 4.1.14 B–61863E/10 (1) Format RST (Address) Bit No. Address No. om (2) Logical product of inverted logical operation result ST0 with the content of the specified address is outputted to the same address. r.c (3) Refer to the figure below for an example of using the RST instruction. A C (R) Y0.0 ce nt e R0.0 B X0.0 nc 4.1.14 Ladder diagram Table 4.1.14 Coding for Fig.
4.1.14 Coding sheet Instruction RD 2 OR Remarks SET ST2 ST1 ST0 A A C X0 . 0 B A+B C 0 Y0.0 output – (A+B) +C Y0 . w w 3 Bit No. R0 . w 1 Address No. Status of operating result .c Step Number 96 PMC SEQUENCE PROGRAM B–61863E/10 4. PMC BASIC INSTRUCTIONS (4) Remarks (a) Restriction of using Do not use SET/RST like the following example 1, use them alone like the following example 2. R0.0 Y0.0 R0.0 Y0.0 (S) X0.0 X0.0 Y0.1 f . R0.0 . om (S) Y0.1 f r.c . X0.0 . . . END2 ce nt e . END2 Example 1 Example 2 D The relation between COM and COME. w w w .c nc The operation of SET/RST in the section of COM/COME is as follows. COM condition ON (ACT=1) : It operates usually. COM condition OFF (ACT=0) : RST does not operate. 97 5. FUNCTIONAL INSTRUCTIONS 5 PMC SEQUENCE PROGRAM B–61863E/10 FUNCTIONAL INSTRUCTIONS om In preparing a sequence program, some functions such as the function for controlling rotation via the shorter path, are difficult to program with basic instructions, which perform only one-bit logical operations. Therefore, functional instructions are available to facilitate programming. See Table 5 (a).
r.c Table 5 (a) Types and processing of functional instructions (1) Instruction Format 1 (Ladder) Format 2 (paper tape punch program) Processing Format 3 (program input) Model PMCPA1 PMCPA3 End of a first-level ladder program f f End of a second-level ladder program f f SUB1 S1 END2 SUB2 S2 END3 SUB48 S48 TMR TMR S3 or TMR Timer processing f f TMRB SUB24 S24 Fixed timer processing f f TMRC SUB54 S54 Timer processing f f DEC DEC S4 or DEC Decoding f f DECB SUB25 CTR SUB5 CTRC SUB55 ROT SUB6 ROTB COD ce nt e END1 Binary decoding f f S5 Counter processing f f S55 Counter processing f f S6 Rotation control f f SUB26 S26 Binary rotation control f f SUB7 S7 Code conversion f f SUB27 S27 Binary code conversion f f SUB8 w S8 Data transfer after logical AND f f MOVOR SUB28 S28 Data transfer after logical OR f f MOVB SUB43 S43 Transfer of 1 byte f f MOVW SUB44 S44 Transfer of 2 bytes f f MOVN SUB45 S45 Transfer of an arbitrary number of bytes f f COM SUB9 S9 Common line control CODB MOVE w .c nc S25 w End of a third-level ladder program : Cannot be used f : Can be used 98 f f PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS Table 5 (a) Types and processing of functional instructions (2) Instruction Format 1 (Ladder) Format 2 (paper tape punch program) Model Processing Format 3 (program input) PMCPA1 PMCPA3 SUB29 S29 End of common line control f f JMP SUB10 S10 Jump f f JMPE SUB30 S30 End of a jump f f JMPB SUB68 S68 Label jump 1 JMPC SUB73 S73 Label jump 2 LBL SUB69 S69 Label PARI SUB11 S11 Parity check DCNV SUB14 S14 Data conversion DCNVB SUB31 S31 COMP SUB15 S15 COMPB SUB32 S32 COIN SUB16 S16 SFT SUB33 S33 DSCH SUB17 S17 DSCHB SUB34 S34 XMOV SUB18 S18 XMOVB SUB35 S35 ADD SUB19 ADDB SUB36 SUB SUB20 SUBB MUL f f r.c f f Extended data conversion f f Comparison f f Binary comparison f f Coincidence check f f Shift register f f Data search f f Binary data search f f Indexed data transfer f f Binary indexed data transfer f f S19 Addition f f S36 Binary addition f f S20 Subtraction f f SUB37 S37 Binary subtraction f f SUB21 S21 Multiplication f f SUB38 S38 Binary multiplication f f SUB22 S22 Division f f .c nc ce nt e f w DIV f f w MULB om COME SUB39 S39 Binary division f f NUME SUB23 S23 Constant definition f f f f w DIVB NUMEB SUB40 S40 Binary constant definition DISP SUB49 S49 Message display DISPB SUB41 S41 Extended message display f f EXIN SUB42 S42 External data input f f WINDR SUB51 S51 Window data read f f WINDW SUB52 S52 Window data write f f : Cannot be used f : Can be used 99 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Table 5 (a) Types and processing of functional instructions (3) Instruction Format 1 (Ladder) Format 2 (paper tape punch program) Model Processing Format 3 (program input) PMCPA1 PMCPA3 SUB50 S50 Position signal output f f PSGN2 SUB63 S63 Position signal output 2 f f DIFU SUB57 S57 Rising edge detection f DIFD SUB58 S58 Falling edge detection f EOR SUB59 S59 Exclusive OR AND SUB60 S60 Logical AND OR SUB61 S61 Logical OR NOT SUB62 S62 Logical NOT END SUB64 S64 End of a subprogram f CALL SUB65 S65 Conditional subprogram call f CALLU SUB66 S66 SP SUB71 S71 SPE SUB72 S72 AXCTL SUB53 S53 f f ce nt e r.c f f Unconditional subprogram call f Subprogram f End of a subprogram f PMC axes control f : Can be used w w w .c nc : Cannot be used om PSGNL 100 f f PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS Table 5 (a) Types and processing of functional instructions (4) Model SUB number Processing PMCĆ RA1 PMCĆ RA2 PMCĆ RA3 PMCĆ RB PMCĆ RB2 PMCĆ RB3 PMCĆ RC PMCĆ RC3 PMCĆ NB/ NB2 END1 1 End of a first-level ladder program f f f f f f f f f END2 2 End of a second-level ladder program f f f f f f f f f END3 48 End of a third-level ladder program f f f f f TMR 3 Timer processing f f f f TMRB 24 Fixed timer processing f f f f TMRC 54 Timer processing f f f f f DEC 4 Decoding f f f f f DECB 25 Binary decoding f f f f CTR 5 Counter processing f f f f om InstrucInstr c tion CTRC 55 Counter processing f f f ROT 6 Rotation control f f ROTB 26 Binary rotation control f f COD 7 Code conversion f f CODB 27 Binary code conversion MOVE 8 MOVOR f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f ce nt e r.c f f f f f f f f f f f f f f Data transfer after Logical AND f f f f f f f f f 28 Data transfer after logical OR f f f f f f f f MOVB 43 Transfer of 1 byte MOVW 44 Transfer of 2 bytes MOVN 45 Transfer of an arbitrary number of bytes COM 9 Common line control COME 29 JMP f f f f f f f f f f f f f f f f f f f f End of common line control f f f f f f f f f 10 Jump f f f f f f f f f JMPE 30 End of a jump f f f f f f f f JMPB 68 JMPC 73 LBL 69 .c nc f f f f f f Label jump 2 f f f f Label f f f f w Label jump 1 11 Parity check f f f f f f f f f 14 Data conversion f f f f f f f f f DCNVB 31 Binary data conversion f f f f f f f f f COMP 15 Comparison f f f f f f f f f PARI w w DCNV COMPB 32 Binary comparison f f f f f f f f f COIN 16 Coincidence check f f f f f f f f f SFT 33 Shift register f f f f f f f f f DSCH 17 Data search f f f f f f f f f DSCHB 34 Binary data search f f f f f f f f f : Cannot be used f : Can be used 101 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Table 5 (a) Types and processing of functional instructions (5) Model InstrucInstr c tion SUB number Processing PMCĆ RA1 PMCĆ RA2 PMCĆ RA3 PMCĆ RB PMCĆ RB2 PMCĆ RB3 PMCĆ RC PMCĆ RC3 PMCĆ NB/ NB2 18 Indexed data transfer f f f f f f f f f XMOVB 35 Binary indexed data transfer f f f f f f f f f ADD 19 Addition f f f f f f f f f ADDB 36 Binary addition f f f f f f f f f SUB 20 Subtraction f f f f f f f f f SUBB 37 Binary subtraction f f f f f f f f f MUL 21 Multiplication f f f f f f f f f MULB 38 Binary multiplication f f f f f f f f f DIV 22 Division f f f f f f f f f DIVB 39 Binary division f f f NUME 23 Constant definition f f f NUMEB 40 Binary constant definition f f f DISP 49 Message display DISPB 41 Extended message display f f f EXIN 42 External data input f f f SPCNT 46 Spindle control WINDR 51 NC window data read WINDW 52 NC window data write FNC9X 9X Arbitrary functional instruction (X = 0 to 7) MMC3R 88 MMC3 window data read f f f f f f MMC3W 89 MMC3 window data write f f f f f MMCWR 98 MMC window data read f f f f MMCWW 99 MMC window data write f f f f DIFU 57 f f f f f f f f f f f f f f f f f f f f ce nt e r.c f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f nc .c f Rising edge detection f f f f 58 Falling edge detection f f f f 59 Exclusive OR f f f f w EOR f w DIFD om XMOV f NOT 62 Logical NOT f f f f END 64 End of a subprogram f f f f CALL 65 Conditional subprogram call f f f f CALLU 66 Unconditional subprogram call f f f f SP 71 Subprogram f f f f SPE 72 End of a subprogram f f f f AXCTL 53 PMC axes control 60 Logical AND f f f OR 61 Logical OR f f f f w AND : Cannot be used f : Can be used 102 f f f f f f f f 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Table 5 (a) Types and processing of functional instructions (6) Model Name SUB number Processing Series 16/18ĆMODEL B Series 16/18ĆMODEL B/C Series 18MODEL B PMCĆRB3 PMCĆRB4 PMCĆRC4 PMCĆRA1 END1 1 First level program end f f f f f END2 2 Second level program end f f f f f END3 48 Third level program end f f f f f TMR 3 Timer processing f f f f TMRB 24 Fixed timer processing f f f f TMRC 54 Timer processing f f f f DEC 4 Decoding f f f f DECB 25 Binary decoding f f f CTR 5 Counter processing f f f CTRC 55 Counter processing f f ROT 6 Rotation control f ROTB 26 Binary rotation control f COD 7 Code conversion f CODB 27 Binary code conversion MOVE 8 ANDed data transfer MOVOR 28 ORed data transfer MOVB 43 Byte data transfer MOVW 44 Word data transfer MOVN 45 Block data transfer COM 9 Common line control COME 29 JMP JMPE om f f f f f f f f f f f f f f r.c f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f Common line control end f f f f f 10 Jump f f f f f 30 Jump end f f f f f JMPB 68 Label jump 1 f f f f JMPC 73 Label jump 2 LBL 69 PARI 11 DCNV 14 nc ce nt e f f f f f Parity check f f f f f Data conversion f f f f f 31 Extended data conversion f f f f f 15 Comparison f f f f f 32 Binary comparison f f f f COIN 16 Coincidence check f f f f f SFT 33 DCNVB w COMP w .c f Label f Shift register f f f f f DSCH 17 Data search f f f f f DSCHB 34 Binary data search f f f f f XMOV 18 Indexed data transfer f f f f f XMOVB 35 Binary indexed data transfer f f f f f ADD 19 Addition f f f f f w COMPB : Cannot be used f : Can be used 103 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Table 5 (a) Types and processing of functional instructions (7) Model Name SUB number Series 16/18ĆMODEL B Processing Series 16/18ĆMODEL B/C Series 18MODEL B PMCĆRB4 PMCĆRC3 PMCĆRC4 PMCĆRA1 36 Binary Addition f f f f f SUB 20 Subtraction f f f f f SUBB 37 Binary subtraction f f f f f MUL 21 Multiplication f f f f f MULB 38 Binary multiplication f f f f DIV 22 Division f f f f DIVB 39 Binary division f f f f NUME 23 Definition of constant f f f NUMEB 40 Definition of binary constant f f f DISP 49 Message display n DISPB 41 Extended message display f f EXIN 42 External data input f AXCTL 53 PMC axis control f WINDR 51 Window data read WINDW 52 Window data write FNC9X 9X Arbitrary functional ins. MMC3R 88 MMC3 window data read f MMC3W 89 MMC3 window data write f MMCWR 98 MMC2 window data read MMCWW 99 MMC2 window data write DIFU 57 Rising edge detection DIFD 58 Falling edge detection EOR 59 Exclusive OR AND 60 OR 61 NOT 62 END 64 CALL 65 SP f f f f f f f f r.c f n f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f f Logical production f f f f Logical Add nc ce nt e n f f f f f f End of subprograms f f f f Conditional subprogram call .c f Logical Negation f f f f 66 Unconditional subprogram call f f f f 71 Subprogram f f f f 72 End of a subprogram f f f f w SPE f n w CALLU om PMCĆRB3 ADDB f : Can be used n : Can be used (with some restrictions) w : Cannot be used NOTE On the PMC-RB3/RB4/RC3/RC4, DISP is provided only for the compatibility with Series 16/18 MODEL A. On the Series 16/18 MODEL B, it is recommended to use DISPB instead of DISP because some extended functions such as high speed display and display of double sized character are available only with DISPB. On the Series 16/18 MODEL B, if both DISP and DISPB are used in the same sequence program, double sized character can not be displayed by DISPB. 104 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS Table 5 (a) Types and processing of functional instructions (8) Model Name SUB number Series 16-MODEL C/Series 18-MODEL C Processing PMCĆRB5 PMCĆRB6 First level program end f f END2 2 Second level program end f f END3 48 Third level program end TMR 3 Timer processing f f TMRB 24 Fixed timer processing f f TMRC 54 Timer processing f DEC 4 Decoding f DECB 25 Binary decoding f CTR 5 Counter processing f CTRC 55 Counter processing f ROT 6 Rotation control ROTB 26 Binary rotation control COD 7 Code conversion CODB 27 Binary code conversion MOVE 8 ANDed data transfer MOVOR 28 ORed data transfer MOVB 43 Byte data transfer MOVW 44 Word data transfer MOVN 45 Block data transfer COM 9 Common line control COME 29 JMP 10 JMPE 30 JMPB JMPC LBL r.c f f f f f f f f f f f f f f f f f f f f Common line control end f f Jump f f Jump end f f 68 Label jump 1 f f 73 Label jump 2 f f 69 Label f f Parity check f f Data conversion f f DCNVB 31 nc .c 11 14 ce nt e f f DCNV f f Comparison f f 32 Binary comparison f f w Extended data conversion 15 COIN 16 Coincidence check f f SFT 33 Shift register f f DSCH 17 Data search f f DSCHB 34 Binary data search f f XMOV 18 Indexed data transfer f f XMOVB 35 Binary indexed data transfer f f ADD 19 Addition f f w w COMPB f f PARI COMP om 1 END1 : Cannot be used f : Can be used 105 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Table 5 (a) Types and processing of functional instructions (9) Model Name SUB number Series 16-MODEL C/Series 18-MODEL C Processing PMCĆRB5 PMCĆRB6 36 Binary Addition f f SUB 20 Subtraction f f SUBB 37 Binary subtraction f f MUL 21 Multiplication f f MULB 38 Binary multiplication f f DIV 22 Division f DIVB 39 Binary division f NUME 23 Definition of constant f NUMEB 40 Definition of binary constant f DISP 49 Message display DISPB 41 Extended message display EXIN 42 External data input AXCTL 53 PMC axis control WINDR 51 Window data read WINDW 52 Window data write FNC9X 9X Arbitrary functional ins. MMC3R 88 MMC3W MMCWR n r.c (Note) om ADDB f f f f n f f f f f f f f f MMC3 window data read f f 89 MMC3 window data write f f 98 MMC2 window data read f f MMCWW 99 MMC2 window data write f f DIFU 57 Rising edge detection f f DIFD 58 Falling edge detection f f EOR 59 Exclusive OR f f AND 60 Logical production f f OR 61 Logical Add f f NOT 62 Logical Negation f f END 64 End of subprograms f f CALL 65 Conditional subprogram call f f CALLU 66 .c nc ce nt e f f Unconditional subprogram call f f SP 71 Subprogram f f SPE 72 End of a subprogram f f w SP w SPE : Cannot be used f : Can be used n : Can be used (with some restrictions) w NOTE On the PMC-RB5/RB6, DISP is provided only for the compatibility with Series 16 MODEL A/B. On the Series 16/18 MODEL C, it is recommended to use DISPB instead of DISP because some extended functions such as high speed display and display of double sized character are available only with DISPB. On the Series 16/18 MODEL C, if both DISP and DISPB are used in the same sequence program, double sized character can not be displayed by DISPB. 106 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS Table 5 (a) Types and processing of functional instructions (10) Model Name SUB number Series 21-MODEL B Processing PMCĆRA1 PMCĆRA3 First level program end f f END2 2 Second level program end f f END3 48 Third level program end TMR 3 Timer processing f f TMRB 24 Fixed timer processing f f TMRC 54 Timer processing f DEC 4 Decoding f DECB 25 Binary decoding f CTR 5 Counter processing f CTRC 55 Counter processing f ROT 6 Rotation control ROTB 26 Binary rotation control COD 7 Code conversion CODB 27 Binary code conversion MOVE 8 ANDed data transfer MOVOR 28 ORed data transfer MOVB 43 Byte data transfer MOVW 44 Word data transfer MOVN 45 Block data transfer COM 9 Common line control COME 29 JMP 10 JMPE 30 Jump end JMPB 68 Label jump 1 f JMPC 73 Label jump 2 f LBL 69 Label f 11 DCNV 14 DCNVB 31 r.c f f f f f f f f f f f f f ce nt e f f f f f f f Jump f f f f nc Common line control end Parity check f f Data conversion f f Extended data conversion f f 15 Comparison f f 32 Binary comparison f f COIN 16 Coincidence check f f SFT 33 Shift register f f DSCH 17 Data search f f DSCHB 34 Binary data search f f XMOV 18 Indexed data transfer f f XMOVB 35 Binary indexed data transfer f f ADD 19 Addition f f w w COMPB f f COMP f f .c PARI om 1 END1 : Cannot be used f : Can be used 107 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Table 5 (a) Types and processing of functional instructions (11) Model Name SUB number Series 21-MODEL B Processing PMCĆRA1 PMCĆRA3 36 Binary Addition f f SUB 20 Subtraction f f SUBB 37 Binary subtraction f f MUL 21 Multiplication f f MULB 38 Binary multiplication f f DIV 22 Division f DIVB 39 Binary division f NUME 23 Definition of constant f NUMEB 40 Definition of binary constant f DISP 49 Message display DISPB 41 Extended message display EXIN 42 External data input AXCTL 53 PMC axis control WINDR 51 Window data read WINDW 52 Window data write FNC9X 9X Arbitrary functional ins. MMC3R 88 MMC3W MMCWR r.c om ADDB f f f f f f f f f f f f f MMC3 window data read f f 89 MMC3 window data write f f 98 MMC2 window data read f f MMCWW 99 MMC2 window data write f f DIFU 57 Rising edge detection DIFD 58 Falling edge detection EOR 59 Exclusive OR f AND 60 Logical production f OR 61 Logical Add f NOT 62 Logical Negation f END 64 End of subprograms f CALL 65 Conditional subprogram call f CALLU 66 nc .c f Unconditional subprogram call f Subprogram f 72 End of a subprogram f w SPE f 71 w SP ce nt e f f : Can be used w : Cannot be used 108 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS Table 5 (a) Types and Processing of Functional Instructions (6) Part 1 Model Name SUB number Series 16 MODEL Series 18 MODEL Processing PMC-RB5 PMC-RB6 1 First level program end f f END2 2 Second level program end f f END3 48 Third level program end TMR 3 Timer processing f f TMRB 24 Fixed timer processing f TMRC 54 Timer processing f DEC 4 Decoding f DECB 25 Binary decoding f CTR 5 Counter processing f CTRC 55 Counter processing ROT 6 Rotation control ROTB 26 Binary rotation control COD 7 Code conversion CODB 27 Binary code conversion MOVE 8 ANDed data transfer MOVOR 28 ORed data transfer MOVB 43 Transfer of one byte MOVW 44 Transfer of two bytes MOVN 45 COM COME JMP r.c om END1 f f f f f f f f f f f f ce nt e f f f f f f f f f f f f f Transfer of arbitrary bytes f f 9 Common line control f f 29 Common line control end f f 10 Jump f f JMPE 30 Jump end f f JMPB 68 Label jump 1 f f JMPC 73 Label jump 2 f f LBL 69 Label specification f f w .c nc PARI Parity check f f 14 Data conversion f f DCNVB 31 Binary data conversion f f COMP 15 Comparison f f COMPB 32 Binary comparison f f COIN 16 Coincidence check f f SFT 33 Shift register f f DSCH 17 Data search f f DSCHB 34 Binary data search f f XMOV 18 Indexed data transfer f f XMOVB 35 Binary indexed data transfer f f ADD 19 BCD addition f f w w DCNV 11 : Cannot be used f : Can be used 109 5.
FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Table 5 (a) Types and Processing of Functional Instructions (6) Part 2 Model Name SUB number Series 16 MODEL Series 18 MODEL Processing PMC-RB5 PMC-RB6 36 Binary addition f f SUB 20 BCD subtraction f f SUBB 37 Binary subtraction f f MUL 21 BCD multiplication f f MULB 38 Binary multiplication f DIV 22 BCD division f DIVB 39 Binary division f NUME 23 Definition of constant f NUMEB 40 Definition of binary constant f DISP 49 Message display DISPB 41 Extended message display EXIN 42 External data input AXCTL 53 PMC axis control WINDR 51 Window data read WINDW 52 Window data write FNC9X 9X Arbitrary functional instruction MMC3R 88 MMC3 window data read MMC3W 89 MMC3 window data write MMCWR 98 MMCWW DIFU r.c om ADDB f f f f n n f f f f f f f f f f MMC2 window data read f f 99 MMC2 window data write f f 57 Rising edge detection f f DIFD 58 Falling edge detection f f EOR 59 Exclusive OR f f AND 60 Logical product f f OR 61 Logical add f NOT 62 Logical negation f END 64 End of subprograms f f 65 Conditional subprogram call f f 66 Unconditional subprogram call f f 71 Subprogram f f 72 End of a subprogram f f w .c nc ce nt e (Note) CALL CALLU w SP SPE w : Cannot be used f : Can be used n : Can be used (with some restrictions) NOTE With PMC–RB5/RB6 of the Series 16i/18i MODEL A, the DISP instruction can be used only to ensure compatibility with the Series 16 MODEL A/B.
With the Series 16i/18i MODEL A, FANUC recommends the use of the DISPB instruction that provides extended functions such as high–speed display and kanji character display.
With the Series 16i/18i MODEL A, if both the DISP instruction and DISPB instruction are used in the same sequence program, the kanji display function of the DISPB instruction cannot be used. 110 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS Table 5 (a) Types and Processing of Functional Instructions (7) Part 1 Model Name SUB number Series 21 MODEL Processing PMC-RA1 PMC-RA 1 First level program end f f END2 2 Second level program end f f END3 48 Third level program end TMR 3 Timer processing f f TMRB 24 Fixed timer processing f f TMRC 54 Timer processing f f DEC 4 Decoding f DECB 25 Binary decoding f CTR 5 Counter processing f CTRC 55 Counter processing ROT 6 Rotation control ROTB 26 Binary rotation control COD 7 Code conversion CODB 27 Binary code conversion MOVE 8 ANDed data transfer MOVOR 28 ORed data transfer MOVB 43 Transfer of one byte MOVW 44 Transfer of two bytes MOVN 45 Transfer of arbitrary bytes COM 9 Common line control f f COME 29 Common line control end f f JMP 10 Jump f f JMPE 30 Jump end f f JMPB 68 Label jump 1 f f JMPC 73 Label jump 2 f f LBL 69 Label specification f r.c f f f f f f f f ce nt e nc f f f f f f f f f f f 11 Parity check f f 14 Data conversion f f DCNVB 31 Binary data conversion f f COMP 15 Comparison f f COMPB 32 Binary comparison f f COIN 16 Coincidence check f f SFT 33 Shift register f f DSCH 17 Data search f f DSCHB 34 Binary data search f f XMOV 18 Indexed data transfer f f XMOVB 35 Binary indexed data transfer f f ADD 19 BCD addition f f w w DCNV f f .c w PARI om END1 : Cannot be used f : Can be used 111 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Table 5 (a) Types and Processing of Functional Instructions (7) Part 2 Model Name SUB number Series 21 MODEL Processing PMC-RA1 PMC-RA 36 Binary addition f f SUB 20 BCD subtraction f f SUBB 37 Binary subtraction f f MUL 21 BCD multiplication f f MULB 38 Binary multiplication f DIV 22 BCD division f DIVB 39 Binary division f NUME 23 Definition of constant f NUMEB 40 Definition of binary constant f DISP 49 Message display DISPB 41 Extended message display EXIN 42 External data input AXCTL 53 PMC axis control WINDR 51 Window data read WINDW 52 Window data write FNC9X 9X Arbitrary functional instruction MMC3R 88 MMC3 window data read MMC3W 89 MMC3 window data write MMCWR 98 MMCWW r.c om ADDB f f f f f f f ce nt e f f f f f f MMC2 window data read f f 99 MMC2 window data write f f DIFU 57 Rising edge detection f f DIFD 58 Falling edge detection f f EOR 59 Exclusive OR f AND 60 OR 61 NOT 62 .c f Logical add f Logical negation f 64 End of subprograms f 65 Conditional subprogram call f w CALL Logical product w END nc f 66 Unconditional subprogram call f SP 71 Subprogram f End of a subprogram f w CALLU SPE 72 : Cannot be used f : Can be used 112 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 The execution time constant is a ratio of the execution time of a functional instruction to the execution time of 10 basic instruction steps (1.5 μs). Execution time constants are used when a ladder program is executed in the separate mode. Table 5 (b) Execution Time Constants of Functional Instructions (1) Instruction SUB Number Model Processing PMC-RC 171 1033 1 End of a first-level ladder program END2 2 End of a second-level ladder program 26 45 END3 48 End of a third-level ladder program 0 TMR 3 Timer processing 19 33 TMRB 24 Fixed timer processing 19 34 TMRC 54 Timer processing 17 29 DEC 4 Decoding 21 28 DECB 25 Binary decoding 16 23 CTR 5 Counter processing 21 35 CTRC 55 Counter processing 18 26 ROT 6 Rotation control 37 53 ROTB 26 Binary rotation control 27 39 COD 7 Code conversion 20 29 CODB 27 Binary code conversion 19 29 MOVE 8 Data transfer after Logical AND 19 27 MOVOR 28 Data transfer after logical OR 13 19 nc ce nt e r.c om END1 COM 9 Common line control 11 14 COME 29 End of common line control 0.1 0.1 JMP 10 Jump 12 16 JMPE 30 End of a jump 9 11 PARI 11 Parity check 13 19 DCNV 14 Data conversion 25 37 DCNVB 31 Binary data conversion 132 233 COMP 15 Comparison 22 36 COMPB 32 Binary comparison 20 31 COIN 16 Coincidence check 21 36 SFT 33 Shift register 15 22 DSCH 17 Data search 237 287 DSCHB 34 Binary data search 351 596 XMOV 18 Indexed data transfer 26 38 XMOVB 35 Binary indexed data transfer 27 37 .c w w w PMC-RB 113 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Table 5 (b) Execution Time Constants of Functional Instructions (2) Instruction SUB Number Model Processing PMC-RB PMC-RC 19 Addition 22 33 ADDB 36 Binary addition 25 39 SUB 20 Subtraction 21 32 SUBB 37 Binary subtraction 25 39 MUL 21 Multiplication 42 63 MULB 38 Binary multiplication DIV 22 Division DIVB 39 Binary division NUME 23 Constant definition NUMEB 40 Binary constant definition 13 20 DISP 49 Message display 51 93 DISPB 41 Extended message display 177 297 EXIN 42 External data input 29 49 WINDR 51 NC window data read 101 293 WINDW 52 NC window data write 101 293 FNC9X 9X Arbitrary functional instruction (X=0 to 7) 21 MMC3R 88 MMC3 window data read 342 375 MMC3W 89 MMC3 window data write 385 421 MMCWR 98 MMC window data read 100 293 MMCWW 99 MMC window data write 100 293 om ADD 45 44 66 33 53 18 25 r.c ce nt e nc .c w w w 114 28 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Execution time constant: This constant represents how many times the execution time of a functional instruction corresponds to the execution time of 10 basic instructions (about 1.5μs) .
The execution time of a basic instruction is about 0.15 μs. The general format and restrictions common to each functional instruction are given below, details on each instructions will follow later.
Refer to this paragraph without fail, since it covers the provisions on using a functional instruction and other important items. Control conditions A B L1 D Parameter (Note) (3) I n s t r u c t i o n ce nt e L0 C r.c om (1) Format Since the functional instructions cannot be represented with relay symbols, the format shown in Fig. 5 (a) must be used. The format includes control conditions, an instruction, parameters, W1, R9000 to R9005 (Functional instruction operation result register). R 2.4 RST R 3.1 (1) R 5.7 ACT (2) Parameter (3) (4) W1 (1) R 10.1 (0) (E1) R 7.1 nc (2) 7 6 5 4 3 2 1 0 R9001 R9002 R9003 R9004 R9005 w w w .c R9000 5 (a) Function instruction format 115 5.

FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Table 5 (c) Coding of function instruction Coding sheet Step Number Address No. Instruction Status of operating result Bit No. Remarks ST3 ST2 ST1 ST0 RD R1 . 0 A A 2 AND R1 . 1 B 3 RD. STK R2 . 4 C C 4 AND. NOT R3 . 1 D   5 RD. STK R5 . 7 RST   RST 6 RD. STK R7 . 1 RST ACT 7 SUB RST ACT 8 (PRM) (Note 2) RST ACT RST ACT RST ACT RST ACT ACT Instruction ffff Parameter 1 ff (PRM) ffff Parameter 2 (PRM) ffff Parameter 3 11 (PRM) ffff Parameter 4 12 WRT R10 . 1 W1 output r.c 9 10 om 1   RST W1 ce nt e NOTE 1 Numbers in parentheses under control conditions indicate the position of the stored register. 2 (PRM) of steps 8 to 11 under Instruction means that P must be input when a parameter is input from the programmer, and PRM is not required to be input when a parameter is input from a paper tape. nc (2) Control condition The number and meaning of control conditions vary with each functional instruction. The control conditions are entered in the stick register as shown in Table 5 (b). The sequence is fixed and cannot be changed or omitted. w w w .c NOTE For the functional instructions, with a RST as a control condition, the RST has the highest priority. Accordingly when RST=1, the RST processing is done even when ACT=0. (3) Instruction The types of instructions are shown in Table 5 (a). The Programmer has exclusive keys for functional instructions TMR and DEC. They are input by T and D keys, respectively. The other functional instructions are given by “S” key and a following number. When instructions are input by relay symbols, software keys are used to input them. Refer to chapter III or IV for details. (4) Parameter Unlike basic instructions, functional instructions can handle numeric values. Thus the reference data or addresses containing data are entered under Parameter. The number and meaning vary with each functional instruction. The P key is used to enter parameters in the Programmer. 116 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (5) W1 The operation results of a functional instruction, when represented with one bit of 1 or 0, is output to W1 whose address can be determined freely by the programmer. Its meaning varies with each functional instruction. Note that some functional instructions have no W1. om (6) Data to be processed Data handled by functional instructions are of binary coded decimal (BCD) code and binary code. In the conventional PMCs, the numeric data is processed mainly based on the BCD code. However, in the PMC-RB/RC, it is recommended to handle all pieces of numeric data with the binary code. The reasons for this are: nc ce nt e r.c (a) In the Series 16, the numeric data (M, S, T, B code) between the CNC and the PMC should be of the binary code. (b) Numeric data on which the CPU performs processing must be in binary format. When numeric data is always processed in binary format, therefore, neither BCD–to–binary nor binary–to–BCD conversion is necessary, thus enabling faster PMC processing. (c) When the data is of the binary code, the range of the numeric data processable becomes wide. Also, negative numeric data can be processed easily, and the arithmetic operation functions are strengthened. The binary numeric data is handled, as a rule, on the basis of 1 byte (–128 to+127), 2 bytes (–32768 to +32767), and 4 bytes (–99999999 to +99999999). (d) When various numeric data items are entered or displayed using the keys on the CRT/MDI panel, all the numeric data items in binary are conveniently specified or displayed in decimal. Therefore, no problem arises, though the data stored in the internal memory is of the binary code. Pay attention to this only when referring to the memory by the sequence program. See (7).

In the functional instructions, binary data is mainly handled. w w w .c (7) Example of numeric data (a) BCD code data The basic data handled with the BCD code is of 1 byte (0 to 99) or 2 bytes (0 to 9999). The BCD 4-digit data is entered into two bytes of continuous addresses as shown below. Example: When BCD data 1234 is stored to addresses R250 and R251. R250 7 6 5 4 3 2 1 0 0 0 1 1 0 1 0 0 3 R251 4 7 6 5 4 3 2 1 0 0 0 0 1 0 0 1 0 1 2 Specify smaller address R250 by a functional instruction. (Note) The low order digits are entered to the smaller address. 117 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (b) Binary code data The basic data handled with the binary code is of 1 byte (–128 to +127), 2 bytes (–32,768 to +32,767) and 4 bytes (–99,999,999 to +99,999,999). The data is stored at addresses R200, R201, R202 and R203 as shown below. 1 byte data (–128 to +127) 7 6 5 4 3 2 1 0 " 26 25 24 23 22 21 20 7 6 5 4 3 0 0 0 0 0 1 1 1 1 1 0 : Positive 2 byte data (–32,768 to + 32,767) 7 6 5 4 3 2 1 0 R200 27 26 25 24 23 22 21 20 R201 " 214 213 212 211 210 29 28 4 byte data (–99999999 to +99999999) 1 6 5 4 3 2 R200 27 26 25 24 23 R201 215 214 213 212 211 R202 223 222 221 R203 " 230 229 0 1 (+1) 1 1 1 (–1) 1 1 1 1 1 1 1 0 22 21 20 210 29 28 0 0 0 0 0 1 1 (+127) 0 1 (–127) nc 7 0 0 ce nt e 0 1 r.c 1 : Negative 2 om R200 Example: 1 byte data 219 218 217 216 228 227 226 225 224 .c 220 w w w By a functional instruction, specify smaller address R200. A negative value is set by the two’s complement code. (8) Addresses of numerical data handled in the function instructions When numerical data handled in the function instructions are 2 bytes or 4 bytes, addresses of numerical data specified by parameters of function instructions are better to take even numbers. The use of even addresses slightly reduces the execution time of functional instructions. These parameters of the functional instructions mainly handling binary data are marked with an asterisk as follows. 118 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 * When 2-byte or 4-byte data is handled, assigning even addresses to addresses marked with * reduces the time required to execute functional instructions. A RST ADDB + B = Error output C ffff * ffff * ffff * ffff Specifying a format Address for an augend Address for an addend Address for outputting the sum (SUB36) om W1 ACT r.c 5 (b) In even addresses, the number after R is even with internal relays, and the number after D is even in data tables. w w w .c nc ce nt e (9) Functional instruction calculation result register (R9000 to R9005) (See Fig. 5 (c)) The result of calculation of the functional instruction is set in the register. This register is used commonly to the functional instructions. Therefore, refer to the information in the register immediately after the functional instruction is executed. Otherwise, the previous information disappears when the next functional instruction is executed. The calculation information in the register cannot be transferred between different levels of the sequence program. For example, it is impossible to read the set information by referring to registers R9000’s by the 2nd level program. When the subtraction instruction (SUBB) is executed by the 1st level program. The calculation information set in the register is guaranteed up to the point just before the functional instruction for setting the next calculation information is executed between the same level of programs. The calculation information set in this register differs according to the functional instruction.

It can be read out by the sequence program, but cannot be written. 7 6 5 4 3 2 1 0 R9000 R9001 R9002 R9003 R9004 R9005 5 (c) This register is a 6 byte register (R9000 to R9005), and the data of 1 bit unit or 1 byte unit can be referred to. When reading the data of bit 1 of R9000, specify RD R9000.1. 119 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.1 END1 (1ST LEVEL SEQUENCE PROGRAM END) 5.1.1 5.1.2 Fig. 5.1.2 shows the format of END.1 and Table 5.1.2 shows the coding. om Function Must be specifies once in a sequence program, either at the end of the 1st level sequence, or at the beginning of the 2nd level sequence when there is no 1st level sequence. r.c Format ce nt e END1 (SUB 1) Fig.5.1.2 Format of END.1 Table 5.1.2 Coding of END.1 Coding sheet Step Number SUB w w w .c nc 0082 Instruction 120 Address Number Bit Number 1 Remarks End of 1st level 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.2 END2 (2ND LEVEL SEQUENCE PROGRAM END) 5.2.1 Specify at the end of the 2nd level sequence. om Function 5.2.2 Fig.5.2.2 shows the expression format and Table 5.2.2 shows the coding format. r.c Format ce nt e END2 (SUB 2) Fig.5.2.2 Format of END.2 Table 5.2.2 Coding of END.2 Coding sheet Step Number SUB w w w .c nc 1362 Instruction 121 Address Number Bit Number 2 Remarks 2nd level sequence program end 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.3 END3 (END OF 3RD LEVEL SEQUENCE) (PMC-RC/RC3/RC4/ NB/NB2 ONLY) 5.3.1 5.3.2 Fig.5.3.2 shows description format and Table 5.3.2 shows coding format. r.c om Function Specify this command at the end of the 3rd level sequence program, i.e. it indicates the end of the sequence program. If there is no 3rd level sequence program, specify this command immediately after END.2 command. Format ce nt e END3 (SUB 48) Fig.5.3.2 END.3 description format Table 5.3.2 END.3 coding format Coding sheet Instruction nc Step Number SUB w w w .c 2122 122 Address Number Bit Number 48 Remarks End of 3rd level program 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.4 TMR (TIMER) 5.4.1 This is an on-delay timer. Function 5.4.2 Fig.5.4.4 (a) shows description format and Table 5.4.4 shows coding format. om Format 5.4.3 5.4.4 Timer Relay (TMff) ACT When the time preset is reached with ACT=1 as shown in Fig.5.4.4 (b), the timer relay turns on. The address of the timer relay is determined by designer. ce nt e Control Condition r.c ACT=0: Turns off the timer relay (TMff). ACT=1: Initiates the timer. ff TMR fff.f Timer relay TMff fff.f Timer number Instruction nc Control condition .c Fig.5.4.4 (a) Format of TMR w w w Step Number Table 5.4.4 Coding of TMR Instruction Address Number 1 RD ffff.

2 TMR ff 3 WRT fff. Bit Number Remarks f ACT f TMff ACT TMff T T indicates the time set in this timer command. Fig.5.4.4 (b) Operation of the timer 123 5. FUNCTIONAL INSTRUCTIONS 5.4.5 B–61863E/10 The timer can be set via the CRT/MDI unit of the CNC (See Chapter II). The setting time is every 48 ms for timer number 1 to 8 and every 8 ms for timer number 9 to 40. A time less than 48 ms is discarded for timer number 1 to 8. The time set by timers 9 to 40 is every 8 ms. Any remainder is discarded. For example, if 38 ms is set, the remainder 6 (38=8 4+6) is discarded, and only 32 ms is actually set. PA1 PA3 RA1 RA2 RA3 RA5 RB 48 ms timer number 1 to 8 1 to 8 1 to 8 1 to 8 1 to 8 1 to 8 1 to 8 8 ms timer number 9 to 40 9 to 40 9 to 40 9 to 40 9 to 40 9 to 40 9 to 40 RB5 RB6 RC RC3 RC4 NB NB2 48 ms timer number 1 to 8 1 to 8 1 to 8 1 to 8 1 to 8 1 to 8 1 to 8 8 ms timer number 9 to 40 9 to 150 9 to 40 Model 5.4.6 Timer Accuracy 48 ms timer 9 to 150 Setting time 1 to 8 1 to 8 9 to 40 9 to 40 9 to 150 9 to 40 9 to 150 Error –48 to 0 ms 8 ms to 262.1 s –8 to 0 ms Variation in time is caused only by operation time of the Timer Instruction. For example, when a timer instruction is used in the 2nd level sequence part, the variation does not include the delay time (Max. 2nd level sequence one cycle time) until the sequence actuates after the set time is reached. w w w RB4 1 to 8 48 ms to 1572.8 s .c 8 ms timer 9 to 40 nc Type of timer ce nt e Type of timer RB3 r.c Type of timer RB2 om Setting Timers Model PMC SEQUENCE PROGRAM 124 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.5 TMRB (FIXED TIMER) 5.5.1 This timer is used as a fixed on-delay timer. The variable timer in section 5.4 sets time of the timer into the nonvolatile memory, and can be reset via the CRT/MDI when necessary. Time present in this fixed timer is written to the ROM together with the sequence program, so the timer time once set cannot be changed unless the whole ROM is exchanged. om Function 5.5.2 r.c The format is expressed as follows (Fig.5.5.2). Format Timer relay TMRB fff f····· f (SUB24) Timer number Preset time ce nt e ACT TMB fff nc 5.5.2 Format of TMRB 5.5.3 5.5.4 As shown in Fig.5.5.4, timer relay is set ON after certain time preset in the parameter of this instruction pasts after ACT=1. The designer will decide the address of the internal relay in the timer relay. ACT TMB T w w w Timer Relay (TMBfff) .c Control Conditions ACT=0: Turns off timer relay (TMBfff).

ACT=1: Start timer. T indicates the time set in this timer command. 5.5.4 Timer operation 125 5. FUNCTIONAL INSTRUCTIONS 5.5.5 PMC SEQUENCE PROGRAM B–61863E/10 (a) Timer number Sets timer number (1 to 100) of the fixed timers. Parameter w w w .c nc ce nt e Precision of the Timer Time varies –8 to 0 ms from the setting time. The varing time in this timer is caused only the error occurred when the timer instruction performs operation process. Error caused by sequence program processing time (time of 1 cycle of the second level), etc. are not included. r.c 5.5.6 om (b) Preset time (8 to 262,136 ms) Processing is done every 8 ms in this fixed timer. The preset time is therefore integral times of 8 ms and the odds are omitted. For example, when set 38 ms, 38=8 4+6, the odd 6 is omitted, and the preset time becomes 32 ms. The range of the preset time is 8 to 262,136 ms. 126 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.6 TMRC (TIMER) 5.6.1 Function 5.6.2 om This is the on-delay timer. A timer setting time is set at an arbitrary address. The selection of an address determines whether the timer is a variable timer or fixed timer. No limit is imposed on the number of timers provided areas can be allocated. Fig.5.6.2 and Table 5.6.2 show the expression format and the coding format, respectively. f ffff ffff Timer accuracy Time set time address Time resister address TMRC ACT ce nt e (SUB54) TM ff r.c Format Fig.5.6.2 TMRC expression format Table 5.6.2 TMRC coding format Step Number Address Number 1 RD ffff. 2 SUB 54 TMRC command 3 (PRM) f Timer accuracy 4 (PRM) fff Timer set time address (PRM) ffff 6 WRT ffff. nc 5 Bit Number Remarks f Timer register address f TMff ACT=0 : Turns off the timer relay (TMff). ACT=1 : Starts the timer. .c 5.6.3 w Control Condition 5.6.4 Instruction w w Timer Accuracy Timer precision Setting value Setting time Error 8 ms 0 1 to 262,136 –8 to 0 ms 48 ms 1 1 to 1,572,816 –48 to +0 ms 1 second (Note) 2 1 to 32,767 0 to +1 s 10 s (Note) 3 1 to 327,670 0 to +10 s 1 m (Note) 4 1 to 32,767 0 to +1 m NOTE This function is usable only with the following models: FS16C/18C PMC–RB5/RB6 FS16i/18i PMC–RB5/RB6 FS21i PMC–RA5 127 5. FUNCTIONAL INSTRUCTIONS 5.6.5 PMC SEQUENCE PROGRAM B–61863E/10 Sets the first address of the timer set time field. The continuous 2-byte memory space is required for the timer set time field. Field D is normally used as this field. Timer Set Time Address Timer set time + 0 TIME Time : Timer set time (1 to 32,767) om Timer set time + 1 ce nt e r.c The timer set time is converted into the binary value in 8 ms (48 ms) units. The timer set time is shown as follows: 8 ms 8 to 262,136 ms 48 ms 48 to 1,572,816 ms 1s 1 to 32,767s 10 s 1 to 327,670s 1m 1 to 32,767m 5.6.6 Set the start address of a timer register area. A timer register area must be allocated to a continuous four–byte memory area starting from the set address. Normally, the R area is used as a timer register area.

This area should be used by the PMC system, and therefore should not be used by the sequence program. Timer Register Address nc Timer register + 0 Timer register + 1 Timer register 5.6.7 w .c Timer register + 2 w w Timer Relay (TMff) Timer register + 3 As shown in Fig. 5.6.7, after ACT is set to 1, the timer relay is turned on once the time specified in this command has elapsed.

ACT TM ff T T indicates the time set in this timer command. 5.6.7 Timer operation 128 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.7 DEC (DECODE) 5.7.1 Outputs 1 when the two-digit BCD code signal is equal to a specified number, and 0 when not. Is used mainly to decode M or T function. Function 5.7.2 Format om Fig.5.7.2 and Table 5.7.2 show the expression format and Table 5.7.2 show the coding format. Decoding result output ACT ffff ffff fff. f fff. f Decode instruction ce nt e Control condition W1 r.c DEC Instruction ff ff Number of digits instruction Number of digits instruction Address of decode signal Fig.5.7.2 Format of DEC Table 5.7.2 Coding of DEC Instruction w 5.7.3 .c nc Step Number w w Control Condition 5.7.4 Address Number Bit Number fff. f 1 RD 2 DEC ffff 3 (PRM) ffff 4 WRT fff. f Remarks ACT W1, Decoding result output ACT=0 : Turns the decoding result output off (W1). ACT=1 : Performs decoding. When the specified number is equal to the code signal, W1=1; when not, W1=0. Specify the address containing two-digit BCD code signals.

Code Signal Address 129 5. FUNCTIONAL INSTRUCTIONS 5.7.5 PMC SEQUENCE PROGRAM B–61863E/10 There are two paths, the number and the number of digits. Decode specification Decode Specification f f f f Number of digits specification Number specification om (i) Number: Specify the decode number. Must always be decoded in two digits. 5.7.6 ce nt e r.c (ii) Number of digits: 01 : The high-order digit of two decimal digits is set to 0 and only the low-order digit is decoded. 10 : The low-order digit is set to 0 and only the high-order digit is decoded.

11 : Two decimal digits are decoded. W1 is 1 when the status of the code signal at a specified address is equal to a specified number, 0 when not.

The address of W1 is determined by designer. W1 (Decoding Result Output) DEN nc MF DEC F10 M30 X 3011 F1.3 R228.1 .c F7.0 M30 w w w Fig.5.7.6 Ladder diagram using the DEC instruction Table 5.7.6 Coding for Fig.5.7.6 Coding sheet Step Number Instruction Address Number 1 RD F 7.0 2 AND F 1 .3 3 DEC F 10 4 (PRM) 3011 5 WRT 130 Bit Number R228 . 1 Remarks M30X 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.8 DECB (BINARY DECODING) 5.8.1 Function (Fig.5.8.2 (a)) 5.8.2 Format ffff DECB Decode result output ffff (SUB 25) 1, 2 or 4-byte binary code data 7 6 5 4 3 2 1 ce nt e Code data r.c Fig.5.8.2 (b) show the expression format. om DECB decodes one, two, or four-byte binary code data. When one of the specified eight consecutive numbers matches the code data, a logical high value (value 1) is set in the output data bit which corresponds to the specified number. When these numbers do not match, a logical low value (value 0) is set. Use this instruction for decoding data of the M or T function. 0 Decode designating +0 number Decode designating +1 number Decode designating +7 number Decode designating numbers nc Eight numbers, each of which is added by 0, 1, 2, . . . , and 7 to the specified number are decoded. When number 62 is specified, for example, eight numbers of 62 to 69 are decoded. If code data is 62, 0 bit of output data is turned on; if 69, 7th bit is turned on. w .c 5.8.2 (a) Function of DECB w w ACT DECB f * ffff (SUB 25) Format specifi-c ation Code data address ffff ffff ffff Decode designation Decode result output address 5.8.2 (b) Expression format of DECB 131 5. FUNCTIONAL INSTRUCTIONS 5.8.3 Control Conditions 5.8.4 PMC SEQUENCE PROGRAM B–61863E/10 (a) Command (ACT) ACT=0: Resets (all the eight) output data bits. ACT=1: Decodes data. Result of processing is set in the output data address. (a) Format specification Specifies code data format: 1 : Code data is in binary format, occupying 1 byte 2 : Code data is in binary format, occupying 2 bytes 4 : Code data is in binary format, occupying 4 bytes om Parameters r.c (b) Code data address Specifies an address which stores code data. (c) Number specification decode designation Specifies the first of the eight successive numbers to be decoded. w w w .c nc ce nt e (d) Decode result address Specifies an address where the decoded result shall be output. A one-byte area is required in the memory for the output. 132 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.9 CTR (COUNTER) 5.9.1 CTR is used as a counter. Counters are used for various purposes for NC Machine tools. Numerical data such as preset values and count values can be used with either BCD format or binary format by a system parameter. Function om NOTE When a incollect BCD data was set to a BCD type counter, the morement of CTR cannot be sured. r.c This counter has the following functions to meet various applications. (a) Preset counter Outputs a signal when the preset count is reached. The number can be preset from the CRT/MDI panel, or set in the sequence program. ce nt e (b) Ring counter Upon reaching the preset count, returns to the initial value by issuing another count signal. (c) Up/down counter The count can be either up or down. 1 8 7 2 6 3 5 4 Presetting : 8 Initial value : 1 Such a counter permits the position of a rotor to be memorized. w w w .c nc (d) Selection of initial value Selects the initial value as either 0 or 1. A combination of the preceding functions results in the ring counter below. 133 5. FUNCTIONAL INSTRUCTIONS 5.9.2 PMC SEQUENCE PROGRAM B–61863E/10 Fig.5.9.2 show the expression format and Table 5.9.2 show the coding format. Format CN0 fff. f UPDOWN ffff CTR (SUB 5) fff. f RST W1 fff. f r.c fff. f ACT om Count up output fff. f Counter number ce nt e Instruction (SUB 5) Control condition 5.9.2 Format of CRT instruction Table 5.9.2 Coding for Fig.5.9.2 Coding sheet 1 Instruction RD RD. STK 3 RD. STK 4 RD. STK 5 SUB 6 (PRM) 7 WRT Remarks fff . f fff . f UPDOWN fff . f RST fff . f ACT CTR instruction 5 ff fff . w w 5.9.3 Control Conditions w Bit No. .c 2 Address No. f ST3 ST2 ST1 ST0 CN0 nc Step Number Memory status of control condition CN0 CN0 UPDOWN CN0 UPDOWN RST CN0 UPDOWN RST ACT CN0 UPDOWN RST ACT Counter number CN0 UPDOWN RST ACT W1 output number CN0 UPDOWN RST W1 (a) Specify the initial value. (CN0) CN0=0: Begins the value of the counter with 0. 0, 1, 2, 3 ····· n.

CN0=1: Begins the value of the counter with 1 (0 is not used). 1, 2, 3 ····· n. (b) Specify up or down counter. UPDOWN=0: Up counter. The counter begins with 0 when CN0=0; 1 when 1. UPDOWN=1: Down counter. The counter begins with the preset value. 134 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS (c) Reset (RST) RST=0: Releases reset. RST=1: Enables reset. W1 becomes 0. The integrated value is reset to the initial value. NOTE Set RST to 1, only when reset is required. om (d) Count signal (ACT) “1” Count 5.9.4 Model RA1 RA2 RA3 RA5 RB RB2 RB3 RB4 1 to 20 1 to 20 1 to 20 1 to 20 1 to 20 1 to 20 1 to 20 1 to 20 1 to 20 1 to 50 RC RC3 RC4 NB NB2 1 to 50 1 to 20 1 to 20 1 to 50 1 to 20 1 to 50 .c 1 to 20 RB6 w 5.9.5 Count is made by catching the rise of ACT.

PA3 RB5 Counter number Counter does not operate. W1 does not change. 1: PA1 nc Counter number 0: Counters of 2 bytes (2 bytes for each of the preset values and cumulative values) are available. The usable numbers are listed below. Counter Number Model Count ce nt e ACT r.c “0” w w Countup Output (W1) When the count is up to a preset value, W1=1. The address of W1 can be determined arbitrarily. When the counter reaches the set value, W1 is set to 1. When the counter reaches 0 or 1, W1 is set to 1. 135 5. FUNCTIONAL INSTRUCTIONS 5.9.6 PMC SEQUENCE PROGRAM B–61863E/10 [Example 1] As a preset counter (See Fig.5.9.6 (a)) The number of workpieces to be machined is counted. When the number reaches the preset count, a signal is output.

L1 is a circuit to make logic 1. D Since the count ranges from 0 to 9999, contact B of L1 is used for making CN0=0. D Since it is to be up counter, contract B of L1 is used make UPDOWN=0. D The reset signal of the counter uses input signal CRST.M from the machine tool. D The count signal is M30X, which was decoded from the CNC output M code. M30X contains contact B of CUP to prevent counting past the preset value, as long as reset is not enabled after countup. om D r.c Examples of Using the Counter ce nt e L1 L1 R200.1 L1 L1 R200.1 L1 R200.1 L1 R200.1 (3) (CN0) R200.1 L1 (1) nc (2) (UPDOWN) R200.1 CRST.M .c (RST) CTR (SUB 5) 0001 CUP (1) Y6.1 X36.0 CUP M30X (0) w (ACT) R200.3 5.9.6 (a) Ladder diagram for the counter, example 1 w w Y6.1 136 Count up output 5.

FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 [Example 2] Use of the counter to store the position of a rotor. (See Fig.5.9.6 (b)) L1 “1” R200.1 L1 R200.1 L1 om R200.1 (3) (CN0) R200.1 REV (2) (UPDOWN) 0002 r.c CTR R200.1 L1 (SUB 5) (1) (RST) R200.1 POS (ACT) X36.0 ce nt e R200.0 (0) 5.9.6 (b) Ladder diagram for the counter, example 2 3 4 5 7 12 8 11 w w w 6 1 .c nc 2 10 9 Fixed position for indexing 5.9.6 (c) Indexing for a rotor Fig.5.9.6 (b) shows a ladder diagram for a counter to store the position of a rotor of Fig.5.9.6 (c). (1) Control conditions (a) Count start number When a 12-angle rotor shown in Fig.5.9.6 (c) is used, the count starting number is 1. Contact A of L1 is used for making CN0=1. (b) Specify up and down The signal REV changes according to the then direction of rotation. It becomes 0 for forward rotation and 1 for reverse rotation. Thus, the counter is an up counter for forward rotation and a down counter for reverse rotation. 137 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (c) Reset In this example, since W1 is not used, RST=0, and contact B of L1 is used. (d) Count signal The count signal POS turns on and off 12 times each time the rotor rotates once. om (2) Counter number and W1 In this example, the second counter is used.

The result of W1 is not used, but its address must be determined. (3) Operation r.c (a) Setting the preset value Since the rotor to be controlled is 12-angle as shown in Fig.5.9.6 (c), 12 must be preset in the counter. It is set from the CRT/MDI panel. ce nt e (b) Setting the current value When the power is turned on, the position of the rotor must be equated with the count on the counter. The count is set via the CRT/MDI panel. Once a current value is set, then correct current positions will be loaded to the counter every time. w w w .c nc (c) The POS signal turns on and off each time the rotor rotates. The number of times of the POS signal turns on and off is counted by the counter, as below. 1, 2, 3, . . . 11, 12, 1, 2, . . . for forward rotation 1, 12, 11, . . . 3, 2, 1, 12 . . . for reverse rotation 138 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.10 CTRC (COUNTER) 5.10.1 The numeral data of this counter are all binary. This counter has the following functions and can be used according to the application: Functions om (a) Preset counter Preset the count value and if the count reaches this preset value, outputs to show that. (b) Ring counter This is the ring counter which is reset to the initial value when the count signal is input after the count reaches the preset value. r.c (c) Up/down counter This is the reversible counter to be used as both the up counter and down counter. 5.10.2 ce nt e (d) Selection of the initial value Either 0 or 1 can be selected as the initial value. Fig.5.10.2 and Table 5.10.2 show the expression format and the coding format, respectively. Format CN0 ffff ffff Counter preset value address Counter register address CTRC nc UPDOWN SUB 55 W1 .c RST w w w ACT Fig.5.10.2 CTRC expression format Table 5.10.2 CRTC coding format Step Number Instruction Address Number 1 RD ffff. f CN0 2 RD.STK ffff.

f UPDOWN 3 RD.STK ffff. f RST 4 RD.STK ffff. f ACT 5 SUB 55 CRTC command 6 (PRM) ffff Counter preset address 7 (PRM) ffff Counter register address 8 WRT ffff.

139 Bit Number f Remarks W1 5. FUNCTIONAL INSTRUCTIONS 5.10.3 B–61863E/10 (a) Specifying the initial value (CN0) CN0=0 : The count value starts with “0”. 0, 1, 2, 3, . . . n CN0=1 : The count value starts with “1”. 1, 2, 3, . . . n (b) Specifying up or down count (UPDOWN) UPDOWN=0: Up counter. The initial value is “0” when CN0=0 or “1” when CN0=1.

UPDOWN=1: Down counter. The initial value is the preset value. (c) Reset (RST) RST=0 : Reset cancelled. RST=1 : Reset. W1 is reset to “0”. The accumulated value is reset to the initial value. (d) Count signal (ACT) ACT=0 : The counter does not operate. W1 does not change. ACT=1 : The counter operates at the rise of this signal. 5.10.4 Counter Preset Value Address ce nt e r.c om Control Conditions PMC SEQUENCE PROGRAM The first address of the counter preset value field is set. The continuous 2-byte memory space from the first address is required for this field. Field D is normally used. Counter preset value+0 CTR (0 to 32,767) nc Counter preset value+1 CTR: Preset value 5.10.5 .c The counter preset value is binary. Therefore, it ranges from 0 to 32767. NOTE When field R is specified as the counter register address, the counter starts with count value “0” after powered on.

w w w Counter Register Address The first address of the counter register field is set.

The continuous 4-byte memory space from the first address is required for this field. Field D is normally used. 140 PMC SEQUENCE PROGRAM B–61863E/10 5.10.6 Count-up Output (W1) 5.

FUNCTIONAL INSTRUCTIONS If the count value reaches the preset value, W1 is set to “1”. The W1 address can be determined freely. Counter register +0 CTR Count value Counter register +2 WORK WORK : Unusable w w w .c nc ce nt e r.c Counter register +3 om Counter register +1 141 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.11 ROT (ROTATION CONTROL) 5.11.1 Controls rotors, such as the tool post, ATC, rotary table, etc., and is used for the following functions. Function (a) Selection of the rotation direction via the shorter path om (b) Calculation of the number of steps between the current position and the goal position r.c (c) Calculation of the position one position before the goal or of the number of steps up to one position before the goal 5.11.2 Format RN0 (5) ffff. f DIR ffff. f POS (3) ROT (SUB 6) ffff (2) ffff (3) (4) ffff ffff Rotating direction output W1 (2) ffff. f .c ffff. f INC (4) (1) nc ffff. f BYT ce nt e Fig.5.11.2 shows the expression format and Table 5.11.2 shows the coding format. w ffff. f ACT (1) (0) w ffff. f w Control condition Calculating result output address Goal position address Current position address Instruction Rotor indexing address 5.11.2 ROT instruction format 142 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS Table 5.11.2 Coding for Fig.5.11.2 Coding sheet Step Number Instruction Address No. Bit No. Status of operating result Remarks ST5 ST4 ST3 ST2 ST1 ST0 1 RD ffff . f RN0 2 RD. STK ffff . f BYT 3 RD. STK ffff . f DIR 4 RD. STK ffff . f POS 5 RD. STK ffff . f INC 6 RD. STK ffff .

f ACT RN0 BYT DIR POS INC ACT 7 SUB ROT RN0 BYT DIR POS INC ACT 8 (PRM) ffff Rotor indexing number RN0 BYT DIR POS INC ACT RN0 RN0 BYT BYT DIR RN0 BYT DIR POS BYT DIR POS INC om 6 RN0 RN0 (PRM) ffff Current position RN0 BYT DIR POS INC ACT (PRM) ffff Goal position address RN0 BYT DIR POS INC ACT 11 (PRM) ffff 12 WRT fff . Calculating result output address f 13 15 5.11.3 Control Conditions RN0 BYT DIR POS INC ACT RN0 BYT DIR POS INC W1 ce nt e 14 r.c 9 10 (a) Specify the starting number of the rotor. RN0=0 : Begins the number of the position of the rotor with 0. RN0=1 : Begins the number of the position of the rotor with 1. nc (b) Specify the number of digits of the process data (position data). BYT=0: BCD two digits BYT=1: BCD four digits w w w .c (c) Select the rotation direction via the shorter path or not. DIR=0 : No direction is selected.

The direction of rotation is only forward.

DIR=1 : Selected. See (8) for details on the rotation direction. (d) Specify the operating conditions. POS=0 : Calculates the goal position. POS=1 : Calculates the position one position before the goal position. (e) Specify the position or the number of steps.

INC=0 : Calculates the number of the position. If the position one position before the goal position is to be calculated, specify INC=0 and POS=1 INC=1 : Calculates the number of steps. If the difference between the current position and the goal position is to be calculated, specify INC=1 and POS=0. (f) Execution command ACT=0: The ROT instruction is not executed. W1 does not change. ACT=1: Executed. Normally, set ACT=0. If the operation results are required, set ACT=1. 143 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM 5.11.4 B–61863E/10 Specify the rotor indexing number. Rotor Indexing Number 5.11.5 Specify the address storing the current position. Current Position Address 5.11.6 om Specify the address storing the goal position (or command value), for example the address storing the CNC output T code. Goal Position Address 5.11.7 Calculate the number of steps for the rotor to rotate, the number of steps up to the position one position before, or the position before the goal. When the calculating result is to be used, always check that ACT=1. r.c Operation Result Output Address 5.11.8 ce nt e The direction of rotation for control of rotation via the shorter path is output to W1. When W1=0, the direction is forward (FOR) when 1, reverse (REV). The definition of FOR and REV is shown in Fig.5.11.8. If the number given to the rotor is ascending, the rotation is FOR; if descending, REV. The address of W1 can be determined arbitrarily. When, however, the result of W1 is to be used, always check that ACT=1. Rotating Direction Output (W1) An example of a 12-position rotor (a) 11 2 2 .c REV w 9 7 FOR 4 4 5 10 9 5 6 6 Indexing fixed position w 12 11 REV w 8 1 3 3 FOR 10 1 nc 12 (b) 7 8 Indexing fixed position 5.11.8 Rotation direction 144 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.12 ROTB (BINARY ROTATION CONTROL) 5.12.1 This instruction is used to control rotating elements including the tool post, ATC (Automatic Tool Changer), rotary table, etc. In the ROT command (5.11) a parameter indicating the number of rotating element indexing positions is a fixed data in programming. For ROTB, however, you can specify an address for the number of rotating element index positions, allowing change even after programming. The data handled are all in the binary format. Otherwise, ROTB is coded in the same way as ROT. r.c om Function 5.12.2 Fig.5.12.2 shows the expression format of ROTB RN0 DIR POS INC (SUB 26) Format specifi-c ation w w .c ACT f 5.12.3 w * * * * ffff ffff ffff ffff Current position address Target position address Arith-me tic result output address nc ROTB ce nt e Format Control Conditions Rotating element indexed position address W1 5.12.2 Expression format of ROTB The control conditions do not differ basically from those for ROT command described in section 5.11. However, BYT has been eliminated from ROTB (it forms part of the ROTB parameters). For the reset, see ROT. 145 5. FUNCTIONAL INSTRUCTIONS 5.12.4 PMC SEQUENCE PROGRAM B–61863E/10 (a) Format " # #   ! &#" " # !"# # # !#! # "& # $! &#"   &#   &#"   &#" $! # $! % "#" ! # !## #" $!!# !"" # ! # !& !# !! #& ! $! # !& " " & # # om Parameters r.c (b) ## # % "# !"" " # !"" # # $! !#!& # "#" # % 5.12.5 ce nt e (c) #! !#!" ! # $#" $" # #! !#!" " # See Section 5.11. Output for Rotational Direction (W1) 5.12.6 w w .c nc Example of Using the ROTB Instruction Fig. 5.12 (b) illustrates a ladder diagram for a 12-position rotor to be controlled for rotation via the shorter path and for deceleration at the position one position before the goal. D The goal position is specified with CNC 32B of binary code (address F26 to F29). D The current position is entered with the binary code signal (address X41) from the machine tool. D The result of calculating the position one position before the goal is output to address R230 (work area). D Operation starts with the output TF (address F7.3) from the CNC. D The coincidence check instruction (COIN) is used to detect the deceleration and stop positions. 146 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS A A R0228.0 Logic 1 R0228.0 A R0228.0 (4) ROTB (SUB 26) 4 D0000 X0041 F0026 R0228.0 A Reference data format R0228.0 Rotor indexing number Current position address Goal position address R0228.0 TF F0007.3 CW-M CCW-M (0) Y0005.6 Y0005.6 TF (0) F0007.3 R0228.2 TCOMPB TF F0007.3 DEC-M R0230 Reference data X0041 Compari -son data address TDEC R0228.2 DEC-M (0) w F0007.3 TF COMPB (SUB 32) 1004 Reference data format Deceleration command Y0005.5 F0026 X0041 Refer-e Compari nce -son data data address TCO MPB R9000.0 F0007.3 CR-CCW TCOMPB TF Goal position (stop position) detection w R0228.3 CW-M F0007.3 R0228.1 R0228.3 TF CR-CCW TCOMPB Y0005.6 CCW-M F0007.3 Deceleration position detection Y0005.5 w R0228.3 TF F0007.3 1004 Reference data format .c R9000.0 TDEC COMPB (SUB 32) nc TF Calculation result output address ce nt e (1) R0228.0 Shorter path or not r.c (2) A CRCCW R0228.1 (3) A R0230 om A R0228.1 R0228.3 Y0005.6 5.12.6 Example of a Ladder Diagram for the ROTB Instruction 147 Forward r otation command Reverse rotation command 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.13 COD (CODE CONVERSION) 5.13.1 Converts BCD codes into an arbitrary two- or four-digits BCD numbers. For code conversion shown in Fig.5.13.1 the conversion input data address, conversion table, and convert data output address must be provided. Set a table address, in which the data to be retrieved from the conversion table is contained, to conversion table input data address in a two-digits BCD number.

The conversion table is entered in sequence with the numbers to be retrieved in the two- or four-digits number. The contents of the conversion table of the number entered in the conversion input data address is output to the convert data output address. As shown in Fig.5.13.1, when 3 is entered in the conversion input data address, the contents 137 located at 3 in the conversion table is output to the convert data output address. ce nt e r.c om Function Table internal address Conversion input data address ffff 0 3 1 ÅÅÅÅÅÅÅ ÅÅÅÅÅÅÅ Specifies table internal number (BCD two-digits).

2 3 4 ÅÅÅÅÅÅÅ ÅÅÅÅÅÅÅ nc Convert data output address ffff Conversion table .c Data of the specified table internal address is output to this address. n w w w 5.13.1 Code conversion diagram 148 137 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.13.2 Fig.5.13.2 shows the format for the COD instruction and Table 5.13.2 shows the coding format. Format BYT ffff. f RST Error output COD ffff ffff om ffff (SUB 7) W1 ffff. f ACT ffff. f r.c Converted data output address Conversion input data address Size of table data Control condition Instruction ce nt e Conversion data table Convert data 1 f f f f 2 f f f f 3 f f f f 4 f f f f nc Table address 5.13.2 COD instruction .c Table 5.13.2 Coding for Fig.5.13.2 Coding sheet 1 2 Addres s No. Bit No. RD fff . f BYT RD. STK fff .

f RST RD. STK fff . f ACT w 3 Instruction w Step Number SUB 5 (PRM) w 4 7 Memory status of control condition Remarks BYT ffff Size of table data (1) ffff Conversion input data address (2) 7 (PRM) ffff Convert data output address (3) 8 (PRM) ffff Convert data at table address 0 (4) 9 (PRM) ffff Convert data at table address 1 (5) 11 WRT ST1 ST0 BYT RST RST ACT COD instruction (PRM) : ST2 BYT 6 10 ST3 : fff . : f Error output W1 149 5. FUNCTIONAL INSTRUCTIONS 5.13.3 Control Conditions PMC SEQUENCE PROGRAM B–61863E/10 (a) Specify the data size. BYT=0 : Specifies that the conversion table data is to be BCD two digits. BYT=1 : Specifies that the conversion table data is to be BCD four digits. om (b) Error output reset RST=0 : Disable reset RST=1 : Sets error output W1 to 0 (resets). r.c (c) Execution command ACT=0 : The COD instruction is not executed. W1 does not change. ACT=1 : Executed. 5.13.4 Size of Table Data 5.13.5 Conversion Input Data Address The conversion table address includes a table address in which converted data is loaded. Data in the conversion table can be retrieved by specifying a conversion table address. One byte (BCD 2-digit) is required for this conversion input data address. The convert data output address is the address where the data stored in the table is to be output. The convert data BCD two digits in size, requires only a 1-byte memory at the convert data output address. Convert data BCD four digits in size, requires a 2-byte memory at the convert data output address. nc 5.13.6 ce nt e A conversion table data address from 0 to 99 can be specified. Specify n+1 as the size of table when n is the last table internal number. w 5.13.7 .c Convert Data Output Address w w Error Output (W1) 5.13.8 Conversion Data Table If an error occurs in the conversion input address during execution of the COD instruction, W1=1 to indicate an error. For example, W1=1 results if a number exceeding the table size specified in the sequence program is specified as the conversion input address. When W1=1, it is desirable to effect an appropriate interlock, such as having the error lamp on the machine tool operator’s panel light or stopping axis feed. The size of the conversion data table is from 00 to 99. The conversion data can be either BCD two digits or four digits, which is specified depends on the control conditions 150 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.14 CODB (BINARY CODE CONVERSION) 5.14.1 Table address Conversion data address ffff Conversion table r.c Function 0 2 ÅÅÅÅÅÅÅ ÅÅÅÅÅÅÅ ÅÅÅÅÅÅÅ ÅÅÅÅÅÅÅ 1 Specify table address here.(binary format1 byte) (Note 1) This table data is binary format 2-byte data. ce nt e 2 Conversion data output address ffff om This instruction converts data in binary format to an optional binary format 1-byte, 2-byte, or 4-byte data. Conversion input data address, conversion table, and conversion data output address are necessary for data conversion; as shown in Fig.5.14.1. Compared to the 5.13 ”COD Function Instruction”, this CODB function instruction handles numerical data 1-, 2- and 4-byte length binary format data, and the conversion table can be extended to maximum 256. ÅÅÅÅÅÅÅ ÅÅÅÅÅÅÅ 1250 3 Data stored in the specified table address is output to this address (Note 2) Conversion table is written in the ROM together together with the program, because it is defined in the sequence program. n nc n : max. 255 5.14.2 Fig.5.14.2 shows the expression format of CODB. w Format .c 5.14.1 Code conversion diagram * w w RST ACT CODB f fff ffff ffff Number of conver-s ion table data Conversi on input data address Conversi on data output address Error output W1 (SUB 27) Format designation 5.14.2 Expression format of CODB 151 5. FUNCTIONAL INSTRUCTIONS 5.14.3 B–61863E/10 (a) Reset (RST) RST=0 : Do not reset. RST=1 : Reset error output W1 (W1=0). (b) Activate command (ACT) ACT=0: Do not execute CODB instruction ACT=1: Execute CODB instruction.

om Control Conditions PMC SEQUENCE PROGRAM 5.14.4 (a) Format designation Designates binary numerical size in the conversion table. 1 : Numerical data is binary 1-byte data. 2 : Numerical data is binary 2-byte data. 4 : Numerical data is binary 4-byte data. (b) Number of conversion table data Designates size of conversion table. 256 (0 to 255) data can be made. (c) Conversion input data address Data in the conversion data table can be taken out by specifying the table number. The address specifying the table number is called conversion input data address, and 1-byte memory is required from the specified address.

(d) Conversion data output address Address to output data stored in the specified table number is called conversion data output address. Memory of the byte length specified in the format designation is necessary from the specified address. nc ce nt e r.c Parameters 5.14.5 w .c Conversion Data Table Size of the conversion data table is maximum 256 (from 0 to 255). This conversion data table is programmed between the parameter conversion data output address of this instruction and the error output (W1). 5.14.6 w w Error Output (W1) If there are any abnormality when executing the CODB instruction, W1=1 and error will be output. 152 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.15 MOVE (LOGICAL PRODUCT TRANSFER) 5.15.1 7 6 5 4 3 2 Input data 0 0 0 0 0 0 Logical multi-plication data f f f 1 0 0 0 r.c Function om ANDs logical multiplication data and input data, and outputs the results to a specified address. Can also be used to remove unnecessary bits from an eight-bit signal in a specific address, etc. (Logical multiplication data) (Input data) to a specified address The input data is one byte (eight bits). f f f f f Low-order four-bit logical multiplication data ce nt e High-order four-bit logical multiplication data 5.15.2 Fig.5.15.2 shows the expression format and Table 5.15.2 shows the coding format. Format MOVE (1) (2) (3) (4) ffff ffff ffff ffff nc ACT (SUB 8) ffff. f .c Output address Input data address w w w Control condition Step Number Instruction Low-order 4-bit logical multiplication data High-order 4-bit logical multiplication data 5.15.2 Move instruction format Table 5.15.2 Coding for Fig.5.15.2 Coding sheet Address No. fff . Bit No. f Remarks ST3 1 RD 2 SUB 3 (PRM) ffff High-order 4-bit logical multiplication data (1) 4 (PRM) ffff Low-order 4-bit logical multiplication data (2) 8 ACT ST2 ST1 ST0 ACT MOVE instruction 5 (PRM) ffff Input data address (3) 6 (PRM) ffff Output data address (4) 153 5. FUNCTIONAL INSTRUCTIONS 5.15.3 PMC SEQUENCE PROGRAM B–61863E/10 ACT=0: Move instruction not executed. ACT=1: Executed. Execution Command 5.15.4 Example of Using the MOVE Instruction 7 6 5 4 3 2 1 om If a code signal and another signal co-exist at address X35 for an input signal from the machine tool, to compare the code signal and a code signal at another address, the rest of signals in address X35 becomes an obstacle. Thus, the MOVE instruction can be used to output only the code signal at address X35 address R210. 0 r.c Address X35 Code signal Logical multiplication data ce nt e Another signa 0 0 0 1 1 1 1 1 Low-order four-bit logical multiplication data High-order four-bit logical multiplication data A 0 0 0 nc Address R210 MOVE .c (SUB 8) (1) 0001 Code signal (2) 1111 (3) (4) X035 R210 5.15.4 MOVE instruction ladder diagram w w w R228.1 154 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.16 MOVOR (DATA TRANSFER AFTER LOGICAL SUM) 5.16.1 Function Input data om This instruction ORs the input data and the logical sum data and transfers the result to the destination. Logical sum data OR ce nt e r.c Output data 5.16.2 Fig.5.16.2 shows the expression format of MOVOR.

Format MOVOR ffff ffff ffff (SUB 28) Input data address Logical sum data address Output address nc ACT w .c 5.16.2 Expression format of MOVOR 5.16.3 w w Control Conditions 5.16.4 Parameters (a) Command (ACT) ACT=0: Do not execute MOVOR. ACT=1: Execute MOVOR. (a) Input data address Specifies the address for the input data. (b) Logical sum data address Specifies the address of the logical sum data with which to OR the transferred data.

(c) Output address This is the address to contain the logical sum obtained. It is also possible to obtain the logical sum (OR) of the input and the logical sum data and output the result in the logical sum data address. For this, you must set the logical sum data address for the output address. 155 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.17 COM (COMMON LINE CONTROL) 5.17.1 COM (Common Line Control) Ę : Can be used : Cannot be used PA1 PA3 RA1 RA2 RA3 RA5 RB RB2 RB3 RB4 RB5 RC RC3 RC4 NB 5.17.1.1 f The specified number of coils or the coils in a region up to the common line control end instruction (COME) are turned off. (See Fig.5.17.1.1) Relay number specification is set when a numeric other than zero is specified in a parameter for the number of turned off coils. Specification of the region up to the common line control end instruction is set when zero is specified for the number of turned off coils. When the common line control end instruction is programmed in the relay number specification, error is indicated when programming is completed. ACT ce nt e r.c Function COM SUB 9 ffff Number of turned-off coils nc 0 to 9999 w w .c 0 : Region specification Other than 0 : Numeric specification w NB2 om f RB6 COME Effective only when the number of turned-off coils is set to zero. SUB 29 5.17.1.1 Function of COM 156 PMC SEQUENCE PROGRAM B–61863E/10 5.17.1.2 5.

FUNCTIONAL INSTRUCTIONS Fig.5.17.1.2 (a) shows the expression format of COM Format ffff SUB 9 Number of turned-off coils om COM ACT (a) A B W1 ACT=1 11.0 C (b) 2 ce nt e COM r.c 5.17.1.2 (a) Expression format of COM 11.1 W2 ACT=0 F (a) G .c B nc D (b) W3 When COM ACT=1, execution begins with the step after COM. When COM ACT=0, coil W1 and coil W2 are turned OFF unconditionally, and execution begins with the next step after W2. w w 5.17.1.2 (b) Ladder diagram for the COM instruction 5.17.2 w Control Conditions ACT=0 : The specified number of coils or the coils within the region specified are unconditionally turned off (set to 0). ACT=1 : No processing is performed. Processing is performed from the step next to the COM instruction. 157 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM 5.17.3 B–61863E/10 (a) Number of turned-off coils Specify 0 to 9999. 0 : Region specification Other than 0: Coil number specification Parameter ACT A B W1 C om ACT W2 D F G r.c E W3 ce nt e 5.17.3 (a) Relay circuit example nc NOTE 1 A functional instruction in a range specified by COM executes processing, regardless of COM ACT. However, if COM ACT=0, the coil of the execution result becomes 0. 2 Another COM instruction cannot be specified in the range specified by the COM instruction. 3 If COM ACT=0, the coil written in by a WRT. NOT instruction in a range specified by COM becomes 1 unconditionally. 4 The number of coils cannot be specified in PMC-RA1, PMC-RA2, or PMC-RB2.

Assume the number of coils to be 0 and specify the region with the common line control end (COME) command. ACT 3 .c COM A w B High-orde r 4-bit MOVE logical multiplicat ion data w w ACT1 W1 Regardless of the ACT condition of the COM instruction, if ACT1=1, the input data of MOVE function is transferred to the output address W2 When ACT=0 in the COM instruction, W2=0 unconditionally. Low-order Input Output 4-bit data address logical address multiplicat ion data C D COIN Reference Comparis data on data (address) (address) ACT2 E W3 5.17.3 (b) 158 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.17.4 COM (Common Line Control) 5.17.5 Ę : Can be used : Cannot be used PA1 PA3 RA1 RA2 RA3 RA5 f f f f f f RB RB2 RB3 RB4 RB5 RB6 f f f f f RC RC3 RC4 NB NB2 f f f f Function ACT COM ce nt e SUB 9 0 r.c om The COM instruction controls the coils in a range up to a common line control end instruction (COME). (See Fig.5.17.5) Specify 0 as the number of coils, and specify a range to be controlled using the common line end instruction.

When the common line end instruction is not specified, the message COM FUNCTION MISSING is displayed. f Valid range of the COM instruction f COME nc SUB 29 .c w 5.17.6 5.17.5 Function of COM ACT COM w w Format Fig.5.17.6 shows the expression format of the functional instruction COM. 0 SUB 9 5.17.6 Expression Format of COM 159 5. FUNCTIONAL INSTRUCTIONS 5.17.7 PMC SEQUENCE PROGRAM B–61863E/10 Control Conditions ACT = 0 : The coils in the specified range are unconditionally turned off (set to 0). ACT = 1 : The same operation as when COM is not used is performed. 5.17.8 (a) Specify 0. (Range specification only) Parameters ACT SUB 9 OUT1 ce nt e ON 0 r.c COM om NOTE 1 COM instruction operation Suppose the following Ladder diagram including a COM instruction exists: OFF OUT2 Then, for the coil ”OUTx,” this Ladder diagram has the same effect as the following Ladder diagram: ACT OUT1 OFF ACT OUT2 nc ON w w w .c So, the functional instructions in the range specified with a COM instruction are processed, regardless of the setting of ACT of the COM instruction. Note, however, that the coil for the execution of a functional instruction is unconditionally set to 0 when COM ACT = 0. 2 In the range specified with a COM instruction, no additional COM instruction can be specified. 3 As explained in the figures in Note 1, the coil for WRT.NOT in the range specified with a COM instruction is unconditionally set to 1 when COM ACT = 0. 160 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.18 COME (COMMON LINE CONTROL END) 5.18.1 5.18.2 Fig.5.18.2 shows the expression format of COME om Function This instruction indicates the division in the region specification of the common line control instruction (COM). This instruction cannot be used alone. It must he used together with the COM instruction. r.c Format COME ce nt e SUB 29 w w w .c nc 5.18.2 Expression format of COME 161 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.19 JMP (JUMP) 5.19.1 Ę : Can be used : Cannot be used JMP (Jump) PA1 PA3 RA1 RA2 RA3 RA5 RB RB2 RB3 RB4 RB5 RC RC3 RC4 NB NB2 f om f RB6 5.19.2 This instruction jumps the specified number of coils or the logic instructions (including the functional instructions) contained within the region up to the jump end instruction (JMPE). Coil number specification is set when a numeral other than zero is specified in the parameter for the number of coils. Specification of the region up to the jump end instruction is set when zero is set for the number of coils. Nesting of jump instructions is not allowed. ACT ce nt e r.c Function JMP ffff SUB 10 Number of jumped coils 0 to 9999 JMPE Effective only when the number of jumped coils is set to zero. SUB 30 5.19.2 w w w .c nc 0 : Region specification Other tha 0 : Coil number specification 5.19.3 Format ACT 162 JMP ffff SUB 10 Number of jumped coils PMC SEQUENCE PROGRAM B–61863E/10 5.19.4 5. FUNCTIONAL INSTRUCTIONS ACT=0 : Nojump.Processing begins with the step after the JMP instruction. ACT=1 : The logic instructions contained within the specified number of coils or the specified region are jumped. Processing is performed from the next step. Control Conditions 5.19.5 Step Number Instruction Table 5.19.5 JMP instruction coding Address Bit Remarks Number Number r.c Parameter om (a) Number of jumped coils Specify 0 to 9999. 0 : Region specification jump Other than 0 : Coil number specification jump When the jump end instruction is programmed in the coil number specification, error is indicated when programming is completed.

1 RD fff. f ACT 2 3 SUB (PRM) 10 ffff JMP instruction Number of coils to be jumped ce nt e NOTE The number of coils can be specified only for the PMC–RB/RC. Assume the number of coils to be 0 and specify the region with the jump end (JMPE) command. 5.19.6 Fig.5.19.6 shows a ladder diagram for the JMP instruction. When ACT=0, the next step to the JMP instruction is executed. When ACT=1, logical operations are skipped according to the specified number of coils. Note that, when ACT=1, even if signal A changes from 1 to 0 or vice versa as shown in Fig.5.19.6, W1 remains in a status before ACT=1. Similarly, W2 remains unchanged, even if signals B, C, and D change. Using the JMP instruction does not reduce the execution time of the sequence. .c nc Operation JMP 2 (SUB 10) A W1 20.1 ACT=0 10.1 B w w w ACT W2 C D E F W3 ACT=1 Fig.5.19.6 Ladder diagram for the JMP instruction 163 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.19.7 Ę : Can be used : Cannot be used JMP (Jump) 5.19.7.1 PA1 PA3 RA1 RA2 RA3 RA5 f f f f f f RB RB2 RB3 RB4 RB5 RB6 f f f f f f RC RC3 RC4 NB NB2 f f f f Function r.c om The JMP instruction causes a departure from the normal sequence to executing instructions. When a JMP instruction is specified, processing jumps to a jump and instruction (JMPE) without executing the logical instructions (including functional instructions) in the range delimited by a jump end instruction (JMPE). (See Fig.5.19.7.1) Specify 0 as the number of coils, and specify a range to be skipped using the jump end instruction. When the jump end instruction is not specified, the message JUMP FUNCTION MISSING is displayed. ce nt e ACT JMP 0 SUB 10 f Valid range of the JMP instruction nc f JMPE .c SUB 30 w w 5.19.7.1 Function of JMP 5.19.7.2 w Format Fig.5.19.7.2 shows the expression format of the functional instruction JMP. ACT JMP 0 SUB 10 5.19.7.2 Expression Format of JMP 164 PMC SEQUENCE PROGRAM B–61863E/10 5.19.7.3 5. FUNCTIONAL INSTRUCTIONS Control Conditions ACT=1 : The logical instructions (including functional instructions) in the specified range are skipped; program execution proceeds to the next step.

ACT=0 : The same operation as when JMP is not used is performed. 5.19.7.4 (a) Specify 0. (Range specification only) Parameters w w w .c nc ce nt e r.c NOTE JMP instruction operation When ACT = 1, processing jumps to a jump end instruction (JMPE); the logical instructions (including functional instructions) in the specified jump range are not executed. When the Ladder program is executed in the nonseparate mode, this instruction can reduce the Ladder execution period (scan time). 165 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.20 JMPE (JUMP END) 5.20.1 Function 5.20.2 Format JMPE w w w .c nc ce nt e r.c SUB 30 om This instruction indicates the division in the region specification of the jump instruction (JMP). It cannot be used alone. It must be used together with the JMP instruction. 166 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.21 PARI (PARITY CHECK) 5.21.1 5.21.2 Fig.5.21.2 shows the expression format and Table 5.21.2 shows the coding format. Format O.E r.c om Function Checks the parity of code signals, and outputs an error if an abnormality is detected. Secifies either an even- or odd-parity check.

Only one-byte (eight bits) of data can be checked. (2) ffff. f (1) ce nt e RST Error output PARI (SUB 11) ffff. f ACT (0) ffff. f ffff W1 ffff. f Check data address Control condition Instruction Instruction nc 5.21.2 PARI instruction format Table 5.21.2 PARI instruction coding Coding sheet Instruction RD 2 RD. STK 3 RD. STK SUB w 4 5 Bit No. (PRM) ACT fff. f ACT fff. f ACT 11 Memory status of control condition ST3 ST2 ST1 ST0 O.E O.E O.E RST RST ACT PARI instruction ffff Check data address ffff. f Error output W1 w 6 Remarks fff. f w 1 Address No. .c Step Number 167 5. FUNCTIONAL INSTRUCTIONS 5.21.3 PMC SEQUENCE PROGRAM (a) Control Conditions B–61863E/10 Specify even or odd. O.E=0 : Even-parity check O.E=1 : Odd-parity check om (b) Reset RST=0 : Disables reset. RST=1 : Sets error output W1 to 0.

That is, when a parity error occurs, setting RST to 1 results in resetting. (c) Execution command ACT=0 : Parity checks are not performed. W1 does not alter. ACT=1 : Executes the PARI instruction, performing a parity check. 5.21.4 r.c If the results of executing the PARI instruction is abnormal, W1=1 and an error is posted. The W1 address can be determined arbitrarily. Error Output (W1) 5.21.5 ce nt e Fig.5.21.5 shows odd-parity checking of a code signal entered at address X036. Example of Using the PARI Instruction 7 Address X036 6 5 4 3 2 1 0 0 6-bit code signal nc Odd-parity bit A .c R228.0 R228.0 A R228.0 w A w w R228.0 ERST.M A (2) PARI (1) (SUB 11) X036 ERR X32.7 (0) TF F7.3 5.21.5 Ladder diagram for the PARI instruction NOTE For bits 0 to 7, bits other than those for the parity check must be 0. 168 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.22 DCNV (DATA CONVERSION) 5.22.1 Converts binary-code into BCD-code and vice versa. Function 5.22.2 om Fig.5.22.2 shows the expression format and Table 5.22.2 shows the coding format. Format (3) (1) fff. CNV (2) (SUB 14) f RST ffff ffff Error output ce nt e fff. (1) DCNV f r.c BYT (1) W1 fff. f ACT fff. (0) fff. f f Conversion result output address Instruction Input data address nc Control condition 5.22.2 DCNV instruction format Table 5.22.2 DCNV instruction coding 1 2 f fff . f BYT RD. STK fff .

f CNV RD. STK fff . f RST fff . f ACT RD w 3 Instruc- Address Bit No. tion No. w Step Number .c Coding sheet Remarks RD. STK SUB 6 (PRM) ffff (1) Input data address 7 (PRM) ffff (2) Conversion result output address 8 WRT w ST2 BYT 5 fff . ST3 ST1 ST0 BYT 4 14 Memory status of control condition BYT CNV BYT CNV CNV RST RST ACT DCNV instruction f W1 error output W1 169 5. FUNCTIONAL INSTRUCTIONS 5.22.3 PMC SEQUENCE PROGRAM (a) Control Conditions B–61863E/10 Specify data size. BYT=0 : Process data in length of one byte (8 bits) BYT=1 : Process data in length of two byte (16 bits) r.c om (b) Specify the type of conversion CNV=0 : Converts binary-code into BCD-code. CNV=1 : Converts BCD-code into binary-code. (c) Reset RST=0 : Disables reset. RST=1 : Resets error output W1.

That is, setting RST to 1 when W1, makes W1=0. (d) Execution command ACT=0 : Data is not converted. W1 does not alter. ACT=1 : Data is converted. 5.22.4 W1=0 : Normal W1=1 : Conversion error W1=1 if the input data which should be BCD data, is binary data, or if the data size (byte length) specified in advance is exceeded when converting binary data into BCD data. w w w .c nc ce nt e Error Output (W1) 170 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.23 DCNVB (EXTENDED DATA CONVERSION) 5.23.1 This instruction converts 1, 2, and 4-byte binary code into BCD code or vice versa. To execute this instruction, you must preserve the necessary number of bytes in the memory for the conversion result output data. om Function 5.23.2 Fig.5.23.2 shows the expression format of DCNVB SIN CNV DCNVB RST (SUB 31) f Format specifi-ca tion ffff Input data address ffff W1 Conver-si on result output address nc ACT * ce nt e * r.c Format w 5.23.3 .c 5.23.2 Expression format of DCNVB w w Control Conditions (a) Sign of the data to be converted (SIN) This parameter is significant only when you are converting BCD data into binary coded data. It gives the sign of the BCD data. Note that though it is insignificant when you are converting binary into BCD data, you cannot omit it. SIN=0 : Data (BCD code) to be input is positive. SIN=1 : Data (BCD code) to be input is negative. (b) Type of conversion (CNV) CNV=0 : Convert binary data into BCD data CNV=1 : Convert BCD data into binary data. (c) Reset (RST) RST=0 : Release reset RST=1 : Reset error output W1. In other words, set W1=0. (d) Execution command (ACT) ACT=0 : Data is not converted.

The value of W1 remains unchanged. ACT=1 : Data is converted. 171 5. FUNCTIONAL INSTRUCTIONS 5.23.4 PMC SEQUENCE PROGRAM B–61863E/10 (a) Format specification Specify data length (1,2, or 4 bytes). Use the first digit of the parameter to specify byte length. 1 : one byte 2 : two bytes 4 : four bytes (b) Input data address Specify the address containing the input data address. (c) Address for the conversion result output Specify the address to output the data converted to BCD or binary format.

om Parameters 5.23.5 r.c W1=0 : Correct conversion W1=1 : Abnormally (The data to be converted is specified as BCD data but is found to be binary data, or the specified number of bytes cannot contain (and hence an overflow occurs) the BCD data into which a binary data is converted.) ce nt e Error Output (W1) 5.23.6 This register is set with data on operation. If register bit 1 is on, they signify the following. For the positive/negative signs when binary data is converted into BCD data, see R9000. Operation Output Register (R9000) nc 7 6 5 4 3 2 1 0 .c R9000 Negative w w w Overflow (data exceeds the number of bytes specified) 172 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE

PROGRAM B–61863E/10 5.24 COMP (COMPARISON) 5.24.1 Compares input and comparison values. Function 5.24.2 Format COMP BYT f ffff ffff (SUB 15) fff.f r.c om Fig.5.24.2 shows the expression format and Table 5.24.2 shows the coding format. Comparison result output W1 ACT ce nt e ffff.f fff.f Instruction Control condition Input value Comparison value Input data format (constant or address) nc 5.24.2 COMP instruction format Table 5.24.2 COMP instruction coding Coding sheet Instruction RD 2 RD. STK 4 Bit No. Remarks fff . f BYT fff . f ACT w 1 3 Address No. .c Step Number COMP instruction (PRM) f Input data format 6 (PRM) ffff 7 WRT w w ffff fff . ST2 ST1 BYT 15 (PRM) ST3 ST0 BYT SUB 5 Memory status of control condition ACT Input data Comparison data address f W1: Comparison result output 173 W1 5. FUNCTIONAL INSTRUCTIONS 5.24.3 PMC SEQUENCE PROGRAM (a) Control Conditions B–61863E/10 Specify the data size. BYT=0 : Process data (input value and comparison value) is BCD two digits long. BYT=1 : Process data (input value and comparison value) is four digits long.

om (b) Execution command ACT=0 : The COMP instruction is not executed. W1 does not alter. ACT=1 : The COMP instruction is executed and the result is output to W1. 5.24.4 5.24.5 The input data can be specified as either a constant or the address storing it. The selection is made by a parameter of format specification. ce nt e r.c Input Data Format 0 : Specifies input data with a constant. 1 : Specifies input data with an address Not specify input data directly, but specify an address storing input data. Input Data 5.24.6 Specifies the address storing the comparison data. nc Comparison Data Address 5.24.7 w w w .c Comparison Result Output W1=0 :Reference data > Comparison data W1=1 :Reference data x Comparison data 174 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.25 COMPB (COMPARISON BETWEEN BINARY DATA) 5.25.1 5.25.2 Fig.5.25.2 shows the expression format of COMPB. om Function This instruction compares 1, 2, and 4-byte binary data with one another. Results of comparison are set in the operation output register (R9000). Sufficient number of bytes are necessary in the memory to hold the input data and comparison data.

r.c Format * ce nt e * ACT COMPB ffff ffff ffff SUB 32 Format specification Input data (address) Address of data to be compared nc 5.25.2 Expression format of COMPB 5.25.3 (a) w 5.25.4 .c Control Conditions Command (ACT) ACT=0 : Do not execute COMPB. ACT=1 : Execute COMPB. Format specification Specify data length (1,2, or 4 bytes) and format for the input data ('constants data' or 'address data'). w Parameters (a) 0 0 w Specification of data length Specification of format 1 : 1 byte length data 2 : 2 byte length data 4 : 4 byte length data 0 : Constants 1 : Address (b) Input data (address) Format for the input data is determined by the specification in a). (c) Address of data to be compared Indicates the address in which the comparison data is stored. 175 5. FUNCTIONAL INSTRUCTIONS 5.25.5 PMC SEQUENCE PROGRAM B–61863E/10 The data involved in the operation are set in this register.

This register is set with data on operation. If register bit 1 is on, they indicate the following: Operation Output Register (R9000) 7 6 5 4 3 2 1 0 om R9000 Zero (input data=data compared) w w w .c nc ce nt e Overflow r.c Negative (input data 176 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.26 COIN (COINCIDENCE CHECK) 5.26.1 Checks whether the input value and comparison value coincide. This instruction is available with BCD data. om Function 5.26.2 Format BYT (1) COIN (1) (2) (3) r.c Fig.5.26.2 shows the expression format and Table 5.26.2 shows the coding format. Result fff.f ffff ffff Output ce nt e ACT f (SUB 16) W1 (0) ffff.f fff.f Instruction Control conditions Input value Comparison value address Input value format (constant or address) nc 5.26.2 COIN instruction format Table 5.26.2 COIN instruction coding 2 3 RD Address No. RD. STK SUB (PRM) Bit No. Remarks fff . f BYT fff . f ACT 16 Reference value 6 (PRM) ffff Comparison value address w ST1 ST0 ACT Reference value format ffff fff . ST2 BYT (PRM) WRT ST3 BYT 5 7 Memory status of control condition COIN instruction f w 4 Instruction w 1 .c Coding sheet Step Number f W1: Checking result output 177 W1 5. FUNCTIONAL INSTRUCTIONS 5.26.3 PMC SEQUENCE PROGRAM (a) Control Conditions B–61863E/10 Specify the data size.

BYT=0 : Process data (input value, and comparison values). Each BCD is two digits long.

BYT=1 : Each BCD four digits long. 5.26.4 r.c Input Data Format.

0 : Specifies input data as a constant. 1 : Specifies input data as an address. om (b) Execution command ACT=0 : The COIN instruction is not executed. W1 does not change. ACT=1 : The COIN instruction is executed and the results is output to W1. 5.26.5 Input Data 5.26.6 ce nt e The input data can be specified as either a constant or an address storing it. The selection is made by a parameter of format designation. Specifies the address storing the comparison data. Comparison Data Address 5.26.7 w w w .c nc Comparison Result Output W1=0 : Input data 0 Comparison data W1=1 : Input data = Comparison data 178 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.27 SFT (SHIFT REGISTER) 5.27.1 Function om This instruction shifts 2-byte (16-bit) data by a bit to the left or right. Note that W1=1 when data "1" is shifted from the left extremity (bit 15) in left shift or from the right extremity (bit 0) in right shift. 5.27.2 Format r.c DIR SFT * ffff ce nt e CONT Address of shift data RST W1 ACT nc (SUB 33) 5.27.3 (a) w w w .c Control Conditions Shift direction specification (DIR) DIR=0 : Left shift DIR=1 : Right shift (b) Condition specification (CONT) CONT=0: On "1" bit shifts by one bit in the specified direction. The condition of an adjacent bit (eighter right or left adjacent bit according to the specification of shift direction DIR) is set to the original bit position of the on "1" bit. Also, "0" is set to bit 0 after shifting in the left direction or set to hit 15 after shifting in the right direction.

In case of leftward shift; 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 Left shift Bit shifts leftward every bit Shift out at bit 15 Zero is set to bit 0. CONT=1: Shift is the same as above, but 1s are set to shifted bits. 179 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM 15 14 13 12 11 10 Left shift 0 0 0 0 0 0 B–61863E/10 9 8 7 6 5 4 3 2 1 0 1 1 1 1 0 0 1 1 0 0 Each bit shifts leftward. Status 1 remains unchanged 5.27.4 (a) Shift data addresses Sets shift data addresses.

These designated addresses require a continuous 2-byte memory for shift data. Bit numbers are represented by bit 0 to 15 as shown below. When addresses are designated for programming, an address number is attached every 8 bits, and the designable bit numbers are 0 to 7. ce nt e Parameters r.c om (c) Reset (RST) The shifted out data (W1=1) is reset (W1=0). RST=0 : W1 is not reset. RST=1 : W1 is reset (W1=0). (d) Actuation signal (ACT) Shift processing is done when ACT=1. For shifting one bit only, execute an instruction when ACT=1, and then, set ACT to 0 (ACT=0). 7 6 5 4 3 2 1 0 15 14 13 12 11 10 9 8 nc Designated address W1=0 : "1" was not shifted out because of the shift operation. W1=1 : "1" was shifted out because of the shift operation. w w W1 w 5.27.5 .c Designated address +1 180 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.28 DSCH (DATA SEARCH) 5.28.1 Function om DSCH is only valid for data tables (see section 6.3) which can be used by the PMC. DSCH searches the data table for a specified data, outputs an address storing it counting from the beginning of the data table. If the data cannot be found, an output is made accordingly. Data table Table internal number 0 1 Search data 2 Search data result output ce nt e 2 100 r.c 100 n 5.28.1 w Format Fig.5.28.2 shows the expression format and Table 5.28.2 shows the coding format. .c 5.28.2 nc NOTE Parameter of this functional instruction and the data table heading address specified here are table internal number 0. The table internal number specified here, however, is different from that mentioned in 6.3. BYT (2) DSCH (1) (2) (3) (4) ffff ffff ffff ffff w ffff.f (1) BYT (SUB 17) ffff.f w ffff.f (0) ACT W1 ffff.f Instruction Control condition Search result output address Search data address Data table heading address Number of data of the data table (Table capacity) 5.28.2 DSCH instruction format 181 Search data presence/absence output address 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Table 5.28.2 DSCH instruction coding Coding sheet Instruction Address No. Bit No. Remarks 1 RD fff . f BYT 2 RD. STK fff . f RST 3 RD. STK fff . f ACT 4 SUB 5 (PRM) ffff Number of data of the data table 6 (PRM) ffff Data table heading address 7 (PRM) ffff Search data address 8 (PRM) ffff Search result output address 9 WRT fff . ST2 ST1 ST0 BYT RST RST ACT BYT BYT DSCH instruction f 5.28.3 Search data presence/absence output adress (a) Specify data size. BYT=0 : Data stored in the data table, BCD two digits long. BYT=1 : Data stored in the data table, BCD four digits long. ce nt e Control Conditions W1 r.c 17 ST3 om Step Number Memory status of control condition nc (b) Reset RST=0 : Release reset RST=1 : Enables a reset, that is, sets W1 to 0. (c) Execution command ACT=0 : The DSCH instruction is not executed.

W1 does not change. ACT=1 : The DSCH is executed, and the table internal number storing the desired data is output., If the data cannot be found, W1=1. 5.28.4 w 5.28.5 .c Number Of Data of the Data Table Specifies the size of the data table.

If the beginning of the data table is 0 and the end is n, n+1 is set as the number of data of the data table. w Data Table Head Address w 5.28.6 Addresses that can be used in a data table are fixed. When preparing a data table, the addresses to be used must be determined beforehand, specify the head address of a data table here. Indicates the address of the data to be searched. Search Data Address 5.28.7 Search Result Output Address If the data being searched for is found, the internal number of the table storing the data is output to this field. This address field is called a search result output address field. The search result output address field requires memory whose size is the number of bytes conforming to the size of the data specified by BYT. 182 PMC SEQUENCE PROGRAM B–61863E/10 5.28.8 W1=0 : The data to be searched exists. W1=1 : The data to be searched does not exist. w w w .c nc ce nt e r.c om Search Data Presence/Absence Output 5. FUNCTIONAL INSTRUCTIONS 183 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.29 DSCHB (BINARY DATA SEARCH) 5.29.1 Function Data table Table number 100 Search result output 1 2 3 100 ce nt e 2 r.c 0 Search data om Alike the DSCH instruction of Section 5.28, this function instruction instructs data search in the data table. There are two differences; the numerical data handled in this instruction are all in binary format; and number of data (table capacity) in the data table can be specified by specifying the address, thus allowing change in table capacity even after writing the sequence program in the ROM. n nc 5.29.1 5.29.2 DSCHB w RST .c Format f * ffff * ffff * ffff * ffff (SUB 34) w w ACT Storage Data table Search Output address of head data address of number of address address search Format data in result designa- data table tion 5.29.2 184 Search result W1 PMC SEQUENCE PROGRAM B–61863E/10 5.29.3 (a) Control Conditions 5. FUNCTIONAL INSTRUCTIONS R eset (RST) RST=0 : Release reset RST=1 : Reset.

W1="0". 5.29.4 (a) Format designation Specifies data length.

Specify byte length in the first digit of the parameter. 1 : 1-byte long data 2 : 2-byte long data 4 : 4-byte long data r.c Parameter om (b) Activation command ACT=0 : Do not execute DSCHB instruction. W1 does not change. ACT=1 : Execute DSCHB instruction. If the search data is found, table number where the data is stored will be output. If the search data is not found, W1 becomes 1. w 5.29.5 .c nc ce nt e (b) Storage address of number of data in data table Specifies address in which number of data in the data table is set. This address requires memory of number of byte according to the format designation. Number of data in the table is n+1 (headnumber in the table is 0 and the last number is n). (c) Data table head address Sets head address of data table. (d) Search data address Address in which search data is set.

(e) Search result output address After searching, if search data is found, the table number where the data is stored will be output. The searched table number is output in this search result output address. This address requires memory of number of byte according to the format designation. w w Search Result (W1) W1=0 :Search data found. W1=1 :Search data not found. 185 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.30 XMOV (INDEXED DATA TRANSFER) 5.30.1 Reads or rewrites the contents of the data table. Like the DSCH instruction, XMOV is only valid for data tables which can be used by the PMC. Function r.c om NOTE The data table heading address specified here is table internal number 0.

The table internal number specified here, however, is different from that mentioned in 6.3. Data table Table internal number 0 2 3 ce nt e ÅÅÅÅÅÅ ÅÅÅÅÅÅ Input or output data ÅÅÅÅÅÅÅ ÅÅÅÅÅÅÅ 1 1 2 Table internal storing input or output data 1 2 2 n Read out data from the data table. Write data in the data table. nc 5.30.1 Reading and writing of data 5.30.2 .c Fig.5.30.2 shows the expression format and Table 5.30.2 shows the coding format. w Format BYT (3) w ffff.f RW (1) (2) (3) (4) ffff ffff ffff ffff XMOV (2) w ffff.f RST (1) Error output (SUB 18) W1 ffff.f ACT (0) ffff.f ffff.f Address storing tabel internal number Instruction Address storing input/output data Control condition Data table heading address Number of data of the data table (Table capacity) 5.30.2 XMOV instruction format 186 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS Table 5.30.2 XMOV instruction coding Coding sheet Instruction Address No. Bit No. Remarks 1 RD fff . f BYT 2 RD. STK fff . f RW 3 RD. STK fff .

f RST 4 RD.

STK fff . f ACT 5 SUB 6 (PRM) ffff Number of data of the data table 7 (PRM) ffff Data table heading address 8 (PRM) ffff Address storing input/output data 9 (PRM) ffff 10 WRT 18 5.30.3 ST1 ST0 BYT BYT BYT RW BYT RW RST RW RST ACT XMOV instruction Address storing table internal number f Error output W1 Specify the number of digits of data. BYT=0 : Data stored in the data table, BCD in two digits long. BYT=1 : Data stored in the data table, BCD in four digits long. ce nt e (a) Control Conditions ST2 r.c fff . ST3 om Step Number Memory status of control conditions 5.30.4 .c nc (b) Specify read or write RW=0 : Data is read from the data table. RW=1 : Data is write in the data table.

(c) Reset RST=0 : Release reset. RST=1 : Enables reset, that is, sets W1 to 0.

(d) Execution command ACT=0 : The XMOV instruction is not executed. W1 does not change.

ACT=1 : The XMOV instruction is executed. w Number of Data Of the Data Table w 5.30.5 Specifies the size of the data table.

If the beginning of the data table is 0 and the end is n, n+1 is set as the number of data of the data table. Address that can be used in a data table are fixed. When preparing a data table, the addresses to be used must be determined beforehand, and the head address placed in that data table . 5.30.6 The input/output data storage address is the address storing the specified data, and is external to the data table. The contents of the data table is read or rewritten.

w Data Table Head Address Address Storing Input/Output Data 187 5. FUNCTIONAL INSTRUCTIONS 5.30.7 Address Storing the Table Internal Number 5.30.8 PMC SEQUENCE PROGRAM B–61863E/10 The table internal number storage address is the address storing the table internal number of the data to be read or rewritten. This address requires memory specified by the formaat designation (BYT). W1=0 :There is no error. W1=1 :There is an error. An error occurs if a table internal number exceeding the previously programmed number of the data table is specified. w w w .c nc ce nt e r.c om Error Output 188 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.31 XMOVB (BINARY INDEX MODIFIER DATA TRANSFER) 5.31.1 Function om Alike the XMOV instruction of Section 5.30, this function instruction instructs reading and rewriting of data in the data. There are two differences; the numerical data handled in this instruction are all in binary format; and number of data (table capacity) in the data table can be specified by specifying the address, thus allowing change in table capacity even after writing the sequence program in the ROM. Data table Table internal number ÅÅÅÅÅÅ ÅÅÅÅÅÅ ÅÅÅÅÅÅ ÅÅÅÅÅÅ 1 2 3 1 ce nt e Input or output data r.c 0 2 Table number where input/output data is stored 1 2 2 n Data read from data table Data write to data table nc 5.31.1 Data read and data write .c 5.31.2 w Format RW XMOVB f w RST w ACT (SUB 35) Format designation * ffff * ffff * ffff * ffff Storage address of number of data in data table Data table head address I/O data storage address Table number storage address 189 W1 5. FUNCTIONAL INSTRUCTIONS 5.31.3 PMC SEQUENCE PROGRAM (a) Control Conditions B–61863E/10 Read, write designation (RW) RW=0 : Read data from data table. RW=1 : Write data to data table. 5.31.4 Format designation Specifies data length. Specify byte length in the first digit of the parameter. 1 : 1-byte long data 2 : 2-byte long data 4 : 4-byte long data r.c (a) om (b) Reset (RST) RST=0 : Reset release. RST=1 : Reset. W=1.

(c) Activation command (ACT) ACT=0: Do not execute MOV instruction. There is no change in W1. ACT=1: Execute MOV instruction. ce nt e Parameters w w .c nc (b) Storage address of number of data in data table Specifies address in which number of data in the data table is set. This address requires 1 or 2-byte memory according to the format designation. Number of data in the table is n+1 (head number in the table is 0 and the last number is n).

(c) Data table head address Sets head address in the data table. (d) Input/Output data storage address Reads and rewrites data in the data table. The data read or rewritten is stored in this address. (e) Table number storage address When reading or rewriting data in the data table, table number of where to read or rewrite must be specified. The specified table number is stored in this table number storage address. This address requires 1 or 2-byte memory according to the format designation. 5.31.5 w Error Output (W1) W1=0 :No error W1=1 :Error found. Error will be output if number of data more than pre-programmed in the data table is specified. 190 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.32 ADD (ADDITION) 5.32.1 Adds BCD two-or four-digit data.

Function 5.32.2 Format om Fig.5.32.2 shows the expression format and Table 5.32.2 shows the coding format. A + B = C BYT (2) (1) f ffff.f ACT (0) (2) (3) (4) ffff ffff ffff (SUB 19) Instruction Control conditions Error output W1 fff.f ce nt e ffff.f r.c ADD ffff.f RST (1) Sum output address Addend Summand address Format of addend (Constant or address) nc 5.32.2 ADD instruction format Table 5.32.2 DSCH instruction coding Coding sheet Memory status of control conditions f BYT fff . f RST f ACT RD 2 RD. STK 4 fff .

RD. STK SUB (PRM) 19 Remarks BYT f Summand address 7 (PRM) ffff Addend (address) ffff Sum output address w ffff 9 WRT fff . ST1 ST0 BYT RST RST ACT Addend format (PRM) (PRM) ST2 BYT 6 8 ST3 ADD instruction w 5 Bit No. fff . 1 3 Address No. .c Instruction w Step Number f Error output W1 191 5. FUNCTIONAL INSTRUCTIONS 5.32.3 PMC SEQUENCE PROGRAM (a) Control Conditions B–61863E/10 Specify the number of digits of data. BYT=0 : Data is BCD two digits long. BYT=1 : Data is BCD four digits long. Data Format of Addend 5.32.5 0 : Specifies addend with a constant. 1 : Specifies addend with an address. r.c 5.32.4 om (b) Reset RST=0 : Release reset. RST=1 : Resets error output W1, that is, sets W1 to 0. (c) Execution command ACT=0 : The ADD instruction is not executed. ACT=1 : The ADD instruction is executed. Summand Address 5.32.6 ce nt e Set the address storing the summand. Addressing of the addend depends on 4). Addend (Address) 5.32.7 Set the address to which the sum is to be output. 5.32.8 If the sum exceeds the data size specified in 3), a), W1=1 is set to indicate an error. w w w .c Error Output nc Sum Output Address 192 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.33 ADDB (BINARY ADDITION) 5.33.1 This instruction performs binary addition between 1-, 2-, and 4-byte data. In the operation result register (R9000), operating data is set besides the numerical data representing the operation results. The required number of bytes is necessary to store each augend, the added, and the operation output data. om Function 5.33.2 r.c Format A + B = C Error output RST ACT 5.33.3 ffff * * ffff ffff * ffff ce nt e ADDB Format (SUB 36) specification Augend address W1 Addend Result address or (sum) constant address Reset (RST) RST=0 : Release reset RST=1 : Resets error output W1. In other words, makes W1=0. (b) Command (ACT) ACT=0 : Do not execute ADDB. W1 does not change now. ACT=1 : Execute ADDB. nc (a) w .c Control Conditions 5.33.4 Format specification Specifies data length (1,2, and 4 bytes) and the format for the addend (constant or address). w w Parameters (a) 0 0 Data length specification 1 : 1 byte length data 2 : 2 bytes length data 4 : 4 bytes length data Format specification 0 : Constant data 1 : Address data 193 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (b) Augend address Address containing the augend. (c) Addend data (address) Specification in (a) determines the format of the addend. (d) Result output address Specifies the address to contain the result of operation. 5.33.5 5.33.6 This register is set with data on operation.

If register bit is on, they signify the following operation data: Operation Output Register (R9000) 6 5 4 3 2 1 ce nt e 7 r.c om Error Output (W1) W1=0 :Operation correct W1=1 :Operation incorrect W1 goes on (W1=1) if the result of addition exceeds the specified data length. 0 w w w .c nc R9000 194 Zero Negative Overflow PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.34 SUB (SUBTRACTION) 5.34.1 Subtracts BCD two-or four-digit data. Function 5.34.2 Format om Fig.5.34.2 shows the expression format and Table 5.34.2 shows the coding format. (2) (1) ffff.f (0) ACT f ffff ffff r.c SUB ffff.f (1) RST ffff (SUB 20) Error output W1 ffff.f ffff.f Instruction ce nt e BYT Difference output address Subtrahend (address or constant) Control condition Subtrahend (data address) Minuend data format nc 5.34.2 SUB Instruction format Table 5.34.2 SUB instruction format Coding sheet Memory status of control conditions Instruction Remarks f BYT RD. STK fff . f RST RD. STK fff . f ACT 2 w RD SUB 20 ST2 ST1 ST0 BYT BYT f 5 (PRM) 6 (PRM) ffff Minuend address 7 (PRM) ffff Subtrahend (address) 8 (PRM) ffff Difference output address 9 WRT w ST3 BYT RST RST ACT SUB instruction w 4 Bit No. fff . 1 3 Address No. .c Step Number Data format of subtrahend fff . f Error output 195 W1 5. FUNCTIONAL INSTRUCTIONS 5.34.3 PMC SEQUENCE PROGRAM B–61863E/10 5.34.4 RST=0 : Release reset. RST=1 : Resets error output W1, that is, sets W1 to 0. Reset 5.34.5 5.34.6 5.34.7 ce nt e 0 : Specifies subtrahend with a constant.

1 : Specifies subtrahend with an address. Data Format of Subtrahend Set the address storing the minuend.

Minuend Address 5.34.8 Addressing of the subtrahend depends on (6).

Subtrahend (Address) 5.34.9 .c nc Sets the address to which the difference is output. Difference Output Address 5.34.10 ACT=0 : The SUB instruction is not executed. W1 does not change. ACT=1 : The SUB instruction is executed. r.c Execution Command om Control Conditions (a) Specification of the number of digits of data. BYT=0 : Data BCD two digits long BYT=1 : Data BCD four digits long W1 is set 1 to indicate an error if the difference is negative. w w w Error Output 196 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.35 SUBB (BINARY SUBTRACTION) 5.35.1 This instruction subtracts one data from another, both data being in the binary format of 1, 2 or 4 bytes.

In the operation result register (R9000), operation data is set besides the numerical data representing the operation. A required number of bytes is necessary to store the subtrahend, minuend, and the result (difference). om Function 5.35.2 A SUBB ACT (SUB 37) 5.35.3 ffff – B * ffff = * ffff C Error output * ffff ce nt e RST r.c Format Format specification Menuend address W1 Subtrahend Differenc addressor e output constant address Reset (RST) RST=0 : Release reset RST=1 : Resets error output W1. (Set W1 to 0.) (b) Command (ACT) ACT=0: Do not execute SUBB. W1 does not change now. ACT=1: Execute SUBB. nc (a) w .c Control Conditions 5.35.4 Format specification Specifies data length (1, 2, and 4 bytes) and the format for the subtrahend (constant or address). 0 w w Parameters (a) 0 Data length specification 1 : 1 byte length data 2 : 2 bytes length data 4 : 4 bytes length data Format specification 0 : Constant data 1 : Address data (b) Minuend address Address containing the minuend. 197 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (c) Minuend data (address) Specification in (a) determines the format of the minuend. (d) Result output address Specifies the address to contain the result of operation.

5.35.5 5.35.6 This register is set with data on operation. If register bit is on, they signify the following operation data: Operation Output Register (R9000) 7 5 4 3 2 1 ce nt e R9000 6 r.c om Error Output (W1) W1=0 :Operation correct W1=1 :Operation incorrect W1 goes on (W1=1) if the result of subtraction exceeds the specified data length.

w w w .c nc Overflow 198 0 Zero Negative 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.36 MUL (MULTIPLICATION) 5.36.1 Function 5.36.2 om Multiplies BCD two-or four-digit data. The product must also be BCD two-or four-digit data. Fig.5.36.2 shows the expression format and Table 5.36.2 shows the coding format. Format A × B = C (2) MUL (1) fff.f (1) RST (2) (3) (4) ffff ffff ffff f (SUB 21) fff.f (0) ACT r.c BYT Error output W1 ce nt e ffff.f fff.f Instruction Product output address Multiplier (address or constant) Control conditions Multiplicand address nc Data format of multiplier (constant or address) 5.36.2 MUL instruction format .c Table 5.36.2 MUL instruction coding Coding sheet 1 2 Bit No. RD fff . f BYT RD. STK fff . f RST RD. STK fff . f ACT w 3 Instruc- Address tion No. w Step Number SUB 5 (PRM) w 4 21 Memory status of control conditions Remarks ST3 ST2 ST1 ST0 BYT BYT BYT RST RST ACT MUL instruction f Data format of multiplier 6 (PRM) ffff Multiplicand address 7 (PRM) ffff Multiplier (address) 8 (PRM) ffff 9 WRT fff . Product output address f Error output W1 199 5. FUNCTIONAL INSTRUCTIONS 5.36.3 PMC SEQUENCE PROGRAM (a) Control Conditions B–61863E/10 Specify the number of digits of data. BYT=0 : Data is BCD two digits long. BYT=1 : Data is BCD four digits long. 5.36.4 r.c 0 : Specifies multiplier with a constant. 1 : Specifies multiplier with an address. om (b) Reset RST=0 : Releases reset. RST=1 : Resets error output W1, that is, sets W1 to 0. (c) Execution command ACT=0 : The MUL instruction is not executed. W1 does not change. ACT=1 : The MUL instruction is executed. Data Format of Multiplier 5.36.5 Multiplicand Address 5.36.6 ce nt e Sets the address storing the multiplicand. Addressing of the multiplier depends on 4). Multiplier (Address) 5.36.7 Product Output Address W1=1 is set to indicate an error if the product exceeds the size specified in 3), a). .c 5.36.8 nc Set the address to which the product is output. w w w Error Output 200 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.37 MULB (BINARY MULTIPLICATION) 5.37.1 This instruction multiplies 1-, 2-, and 4-byte binary data items. In the operation result register (R9000), operation data is set besides the numerical data representing the operation. A required number of bytes is necessary to store multiplicand, multiplier, and the result (product). om Function 5.37.2 A MULB ACT * ffff B * ffff = C Error output * ffff Format specification W1 Multi-plic Multiplier Product and address or output address constant address nc (SUB 38) ffff × ce nt e RST r.c Format 5.37.3 (a) w .c Control Conditions Reset (RST) RST=0 : Release reset RST=1 : Resets error output W1. In other words, makes W1=0.

(b) Command (ACT) ACT=0: Do not execute MULB. W1 does not change now.

ACT=1: Execute MULB. 5.37.4 Format specification Specifies data length (1, 2, and 4 bytes) and the format for the multiplier (constant or address). w w Parameters (a) 0 0 Data length specification 1 : 1 byte length data 2 : 2 bytes length data 4 : 4 bytes length data Format specification 0 : Constant data 1 : Address data 201 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (b) Multiplicand address Address containing the multiplicand. (c) Multiplier data (address or constant) Specification in (a) determines the format of the multiplier. (d) Result output address Specifies the address to contain the result of operation. 5.37.5 5.37.6 This register is set with data on operation. If register bit is on, they signify the following operation data: Operation Output Register (R9000) 6 5 4 3 2 1 ce nt e 7 r.c om Error Output (W1) W1=0 :Operation correct W1=1 :Operation incorrect W1 goes on (W1=1) if the result of multiplication exceeds the specified data length. 0 R9000 w w w .c nc Overflow 202 Zero Negative 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.38 DIV (DIVISION) 5.38.1 Divides BCD two-or four-digit data. Remainders are discarded. Function 5.38.2 Format om Fig.5.38.2 shows the expression format and Table 5.38.2 shows the coding format. (2) DIV (1) ffff.f (1) RST f (2) (3) (4) ffff ffff ffff (SUB 22) ffff.f (0) ACT r.c BYT Error output W1 ce nt e fff.f ffff.f Instruction Quotient output address Divisor (address or constant) Control conditions Dividend address Data format of divider nc 5.38.2 DIV instruction format Table 5.38.2 DIV instruction coding Step Number 2 3 Bit No. fff . f BYT RD. STK fff . f RST RD. STK fff . f ACT RD SUB w 4 Instruc- Address tion No. w 1 .c Coding sheet (PRM) 6 (PRM) w 5 22 Remarks Memory status of control conditions ST3 ST2 ST1 ST0 BYT BYT BYT RST RST ACT DIV instruction f Data format of divider ffff Dividend address 7 (PRM) ffff Divider (address) 8 (PRM) ffff Quatient output address 9 WRT fff . f Error output W1 203 5. FUNCTIONAL INSTRUCTIONS 5.38.3 PMC SEQUENCE PROGRAM (a) Control Conditions B–61863E/10 Specify the number of digits of data. BYT=0 : Data is BCD two digits long. BYT=1 : Data is BCD four digits long. Divisor Data Format Designation 5.38.5 0 : Specifies divisor data by constant. 1 : Specifies divisor data by address. r.c 5.38.4 om (b) Reset RST=0 : Releases reset. RST=1 : Resets error output W1, that is, sets W1 to 0. (c) Execution command ACT=0 : The DIV instruction is not executed. W1 does not change. ACT=1 : The DIV instruction is executed. Sets the address storing the dividend.

ce nt e Dividend Address 5.38.6 Addressing of the divisor depends on 4). Divisor (Address) 5.38.7 Quotient Output Address 5.38.8 nc Sets the address to which the quotient is output. W1=1 is set to indicate an error if the divider is 0. w w w .c Error Output 204 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.39 DIVB (BINARY DIVISION) 5.39.1 This instruction divides binary data items 1, 2, and 4 byte in length. In the operation result register (R9000), operation data is set and remainder is set to R9002 and following addresses. A required number of bytes is necessary to store the dividend, divisor, and the result (quotient). om Function Format A / B = RST C ffff * ffff SUB 39 Format specification Divisor Dividend (address) Quotient address or address constant Error data * ffff W1 nc ce nt e DIVB ACT * ffff r.c 5.39.2 5.39.3 (a) Control Conditions Reset (RST) RST=0 : Release reset RST=1 : Resets error output W1. In other words, makes W1=0. w .c (b) Command (ACT) ACT=0: Do not execute DIVB. W1 does not change now. ACT=1: Execute DIVB. w 5.39.4 Format specification Specifies data length (1, 2, and 4 bytes) and the format for the divisor (constant or address). w Parameters (a) 0 0 Data length specification Format specification 0 : Constant data 1 : Address data 205 1 : 1 byte length data 2 : 2 bytes length data 4 : 4 bytes length data 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (b) Dividend address Address containing the dividend (c) Divisor data (address) Specification in (a) determines the format of the divisor.

(d) Result output address Specified the address to contain the result of operation. 5.39.5 5.39.6 This register is set with data on operation. If register bit is on, they signify the following operation data: 7 5 4 3 2 ce nt e R9000 6 r.c Operation Output Register (R9000) om Error Putput (W1) W1=0 :Operation correct W1=1 :Operation incorrect W1 goes on (W1=1) if the divisor is 0. w w w .c Remainder Output Address 0 Zero Negative Depending on its length, the remainder is stored in one or more of registers R9002 to R9005.

nc 5.39.7 1 206 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.40 NUME (DEFINITION OF CONSTANT) 5.40.1 Defines constants, when required.

In this case, constants are defined with this instructions.

Function 5.40.2 BYT (1) fff.f ACT (0) (1) NUME ffff Control condition ffff (SUB 23) ce nt e fff.f (2) r.c Format om Fig.5.40.2 shows the expression format and Table 5.45.40.2 shows the coding format. Instruction Constant Constant outputaddress 5.40.2 NUME instruction format Table 5.40.2 NUME instruction coding Coding sheet Step Number Instruc- Address tion No. Bit No. Remarks 1 RD fff . 2 RD. STK fff . 3 SUB 4 (PRM) ffff Constant 5 (PRM) ffff Constant output address ST2 ST1 ST0 BYT ACT BYT nc f ST3 ACT BYT NUME instruction .c 23 w 5.40.3 f Memory status of control conditions w w Control Conditions 5.40.4 Constant 5.40.5 (a) Specify the number of digits of a constant. BYT=0 : Constant is BCD two digits long. BYT=1 : Constant is BCD four digits long. (b) Execution command ACT=0 : The NUME instruction is not executed. ACT=1 : The NUME instruction is executed. Sets the constant as the number of digits specified in Item (a) in Subsec. 5.40.3. Sets the address to which the constant defined in Subsec. 5.40.4 is output. Constant Output Address 207 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.41 NUMEB (DEFINITION OF BINARY CONSTANTS) 5.41.1 This instruction defines 1, 2, or 4-bytes long binary constant. Data entered in decimal during programming is converted into binary data during program execution. The binary data is stored in the specified memory address(es). om Function 5.41.2 r.c Format * NUMEB ACT (SUB 40) 5.41.3 (a) (a) w w w Parameters Constant Constant output address Command (ACT) ACT= 0: Do not execute NUMEB. ACT= 1: Execute NUMEB. Format specification Specifies data length (1, 2, or 4 bytes). Use the first parameter digit to specify byte length: 1 : 1 byte 2 : 2 bytes 4 : 4 bytes .c 5.41.4 Format specification ffff nc Control Conditions f@@@f ce nt e f (b) Constant Defined constants in decimal format. The constant must not executed the number of bytes specified (format specification). (c) Constant output address Specifies the address of the area for output of the binary data. A continuous memory area of the specified number of bytes is required from the specified address.

208 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.42 DISP(MESSAGE DISPLAY) (PMC–RB/RB2/RB3/ RB4/RB5/RB6/RC/ RC3/RC4 ONLY) 5.42.1 om DISP is used to display messages on the CRT screen, CNC of which enters alarm status. Message data to be displayed is specified after the parameters of the functional instruction. One DISP functional instruction can define up to 16 types of message. Display is performed by setting the control condition ACT to 1. In order to display and then clear a message, set the display-request bit corresponding to the message data number to 1 and 0, respectively. Up to one alarm message (message data putting the CNC in alarm status) can be displayed on one screen.

When one message is cleared, a message is displayed. Similarly, each time one of the message is displayed. One operator message (message data not putting the CNC in alarm status) can be displayed on a screen. When an operator message is cleared in a state when four operator messages are displayed, the subsequent operator message is displayed. ce nt e r.c Function 5.42.2 Fig.5.42.2 shows the instruction format and Table 5.42.2 shows the coding format. w w w .c nc Format 209 5.

FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM (1) (2) B–61863E/10 Process end (3) DISP (SUB 49) ffff ffff ffff W1 fff. f om ACT Message control address Control condition r.c Number of data of one message data Instruction Total sum of data of message data ce nt e Message data Message number 1 2 Message data 1 Message characters m nc Message number 1 2 .c Message data 2 m w w w Message characters Message number 1 2 Message data n Message characters m 1 x n x 16 5.42.2 DISP instruction format 210 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS Table 5.42.2 DISP instruction coding Coding sheet Address No. RD SUB Bit No. Remarks ST2 ACT ACT 49 DISP ACT ffff Total sum of data of message data (PRM) ffff Number of data of one message item (PRM) ffff Message control address (PRM) ffff Message number (PRM) ffff 2 (PRM) ffff 3 : : : : : Message characters (PRM) ffff m (PRM) ffff Message number (PRM) ffff 2 (PRM) ffff 3 : : : : ffff : : : (PRM) ffff (PRM) ffff : : : 2 3 : : : : Message characters ffff m ffff Process end (W1) w w w WRT : Message number .c (PRM) Message characters nc ffff : m : (PRM) r.c : : ST0 fff . f (PRM) (PRM) ST1 om Instruction ce nt e Step Number Memory status of control conditions 211 W1 5.

FUNCTIONAL INSTRUCTIONS 5.42.3 PMC SEQUENCE PROGRAM B–61863E/10 Control Condition ACT=0: Nothing is processed. W1 does not change. ACT=1: The specified message data is displayed or cleared. ACT must remain 1 until processing end is reported by W1. 5.42.4 (a) Parameters Total sum of message data of data: m n om (b) Number of data of one massage data: m Note) (c) Message control address: Specifies the address of the RAM of internal relay area (see (7) for details). w w w .c nc ce nt e r.c NOTE The number of data used by each message data item, m, must be the same. Since 00 is ignored, it can be set for unnecessary data. For example, for particular messages with a different number of displayed characters, set 00 so that the number of data, m, are the same. 212 PMC SEQUENCE PROGRAM B–61863E/10 5.42.5 (a) Message number: The specified number produces an appropriate event as follows. 1000 to 1999 (alarm message): The CNC is put in alarm status and the number and following data are displayed. The maximum number of the displayed characters is up to 32, except for the message number. When an alarm status occurs, the operation being executed stops. To release the alarm status, set the display-request bit (see Fig.5.42.7) to 0. 2000 to 2099 (operator message): The CNC is not put in alarm status and the number and following data are displayed. The maximum number of the displayed characters is 255, except for the message number. 2100 to 2999 (operator message): The CNC is not put in alarm status and the number is not displayed. Only the following data (up to 255 characters) is displayed. 5000–5999 (alarm messages on path 2): Path 2 is placed in the alarm state. A displayed message number is a specified number from which 4000 is subtracted. The number of displayed characters excluding this number is 32 or less. If the alarm state arises during axis movement, a gradual stop occurs. The alarm state can be released by setting the display request bit to 0. 7000–7999 (alarm messages on path 3): Path 3 is placed in the alarm state. The displayed message number is a specified number from which 6000 is subtracted. The number of displayed characters excluding this number is 32 or less. If the alarm state arises during axis movement, a gradual stop occurs. The alarm state can be released by setting the display request bit to 0. nc ce nt e r.c om Message Data 5. FUNCTIONAL INSTRUCTIONS w w w .c NOTE If all characters in the operator message are kana characters, up to 254 kana characters are displayed. (b) Message character An alphanumeric character is specified with a two-digit decimal (two characters per step). Table 5.42.6 shows the correspondence between characters and specified numbers. The above message data is always specified because it is written on ROM. The message data cannot therefore be changed as desired. However, arbitrary numeric data of up to four BCD digits can be displayed according to the specified variable data. The spindle tool number which changes whenever ACT tools are changed and the number of the tool at the tool-change position can be displayed, for example. For specifying variable data, see (10) below. 213 5. FUNCTIONAL INSTRUCTIONS 5.42.6 PMC SEQUENCE PROGRAM B–61863E/10 W1=0 :Processing ends.

Normally, W1=0. If W1=0 after W1=1, processing ends. W1=1 :In process. W1=1 when ACT=1. Error Output (W1) Table 5.42.6 Correspondince between characters and specified numbers Specified number 64 160 33 65 161 34 66 162 35 67 163 36 68 164 37 69 165 38 70 166 39 71 167 40 72 168 41 73 42 74 43 75 44 76 45 77 46 78 47 79 48 80 49 81 50 82 51 52 53 55 57 58 59 193 194 195 196 197 198 199 200 201 170 202 171 203 172 204 173 205 174 206 175 207 176 208 177 209 178 210 83 179 211 84 180 212 85 181 213 86 182 214 87 183 215 88 184 216 89 185 217 90 186 218 91 187 219 92 188 220 93 189 221 62 94 190 222 95 191 223 w 63 Corresponding character 192 61 w 60 w 56 Specified number 169 .c 54 Corresponding character ce nt e (space) Corresponding character nc 32 Specified number om Corresponding character r.c Specified number *1) minus *2) Under bar *3) Long bar 214 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.42.7 The parameters and message data used by this functional instruction are as follows. Parameters and Message Data SUB49 Total sum of data of message data Number of data of one message item Parameter Message data 1 Message data 2 6 5 4 3 2 1 0 0 0 0 0 0 0 1 0 R200 0 R201 0 0 0 0 0 0 0 0 R202 0 0 0 0 0 0 1 0 R203 0 0 0 0 0 0 0 0 ce nt e 3 n 7 r.c RAM address Display request Display state Two bytes of R200 and R201, the address specified in the message control address and that address plus +1, (display request), are required to specify a message to be displayed. Even if the number of message data items is small, two bytes are always required. 0 is set to unnecessary data. 0 is set automatically when the CNC is powered on. Two bytes of R202 and R203, the specified address plus +2 and the same address plus +3, (displayed state), show the message displayed on the CRT screen. When there is more than one display request, only a prescribed number of message are displayed on the CRT screen. Actually displayed messages are known by the displayed state. The displayed state is set automatically in the two bytes of the displayed state and can be referred to by the sequence program. Those bytes must not be written in. w w w .c Message data Message data 2 is displayed on the CRT screen nc Message data om Message control address (Specify a message to be displayed, using an address of RAM in the internal relay area is taken to here, R200 is taken.) One DISP functional instruction requires the four consecutive bytes following the address specified in the above message control address in order to check the display request and displayed status. When messages are displayed or cleared, message data 1 to n (n x 16) and display-request bits correspond to each other as shown in Fig.5.42.7. To display and clear a message data item, set the corresponding bit to 1 and 0, respectively, and the control condition ACT to 1. If the sequence program checks messages displayed on the screen, message data 1 to n and display-request bits correspond to each other as shown in Fig.5.42.7. Message data for which 1 is set among the 16 displayed status bits, is the message data currently being displayed. 215 5. FUNCTIONAL INSTRUCTIONS Note) Specified address Display request Specified address +1 Specified address +2 Display state 6 Message data 7 Message data 15 Message data 7 Message data 15 5 Message data 6 Message data 14 Message data 6 Message data 14 4 Message data 5 Message data 13 Message data 5 Message data 13 3 Message data 4 Message data 12 Message data 4 Message data 12 B–61863E/10 2 Message data 3 Message data 11 Message data 3 Message data 11 1 Message data 2 Message data 10 Message data 2 Message data 10 0 Message data 1 Message data 9 Message data 1 Message data 9 om Specified address +3 7 Message data 8 Message data 16 Message data 8 Message data 16 PMC SEQUENCE PROGRAM 5.42.7 Correspondence between message data and display request/displayed status 5.42.8 w w w .c nc Remarks on Using the DISP Instruction CNC external data input function The DISP instruction displays mes-sages using external data input function or external message display, which in-volves external work-number search, external tool offset, external work co-ordinate system shift, etc. as well as message display. The DISP instruction cannot display messages when any of these functions is being executed. To check this, EPCA (any address in inter-nal relay area) and EPCB (any address in control relay area) are used as interlock signal.

The sequence program sets EPCA to 1 while the message is displayed, and to 0 upon competion of processing. The sequence program sets EPCB to 1 while any function other than the above is being processed, and to 0 upon completion of processing. When EPCB = 1, messages must not be displayed (DISP ACT must not be 1). Set ACT to 1 after making sure that EPCB = 0. When the function other than message display is executed, execute after making sure that EPCA = 0. DISP instruction and external data input function (external tool offset, external work number search) must be programmed in the same sequence level. ce nt e (a) r.c NOTE "Specified address" means an address specified in the message control address of a DISP instruction parameter. (b) External data input function address During DISP instruction execution (EPCA = 1), the PMC ³ CNC interface of the external data input function must not be used for processing of external tool offset, external work-number search or external work coordinate system shifting. If EPCA = 1, use the JUMP instruction, for example, to skip writing data, so that nothing is written in the interface. (c) ACT and W1 of the DISP instruction (a) Timing of ACT ON If EPCB = 0, ACT may be set to 1 with any timing. For instance, when all display-request bits are off or when the status displayed on the screen and the display requests are the same, that is, when there are no new display requests, even if ACT = 1, the DISP instruction processes nothing and the operation terminates (W1 = 0). Even if another display-request bit is set on and ACT is set to 1 216 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS with a prescribed number of messages (four alarm messages or one operator messages) displayed on the screen, no message is displayed for that request, but W1 = 0 after W1 = 1 and W1 = 1 again during execution of the next cycle. In other words, W1 only changes back and forth between 1 and 0. w w w .c nc ce nt e r.c om (b) Using two or more DISP instructions If EPCB = 0, ACT of each DISP instruction may be turned on simultaneously.

Until the DISP instruction whose ACT was set to 1 earlier, has been completed (W1 = 0), executing of the next DISP instruction is kept waiting. W1 of the DISP instruction kept waiting remains 0 at this time. Consequently, no messages more than those specified number are displayed, as discussed in (a). From (i) and (ii) above, set ACT to 1 whenever EPCB = 0. Do not set ACT to 1 when EPCB = 1. 217 5. FUNCTIONAL INSTRUCTIONS 5.42.9 Examples of Using The DISP Instruction PMC SEQUENCE PROGRAM B–61863E/10 (a) Display three types of messages with the following conditions. SPER = 1 and "SPINDLE ALARM" (Message data 1) ATCER = 1 and "ACT ALARM" (Message data 2) WORK = 1 and "WORK SET UP" (Message data 3) [Message data specified] 7 6 5 4 3 AddressR220 0 0 0 0 0 R221 0 0 0 0 0 R222 0 0 0 0 0 R223 0 0 0 0 0 2 1 0 MSI AL2 AL1 om Display request 0 0 0 0 0 0 Display state SPER AL1 r.c R220.0 D SPINDLE ALARM D ATC ALARM D WORK SET UP ATCER AL2 R220.1 WORK ce nt e MS1 Message data 1 : AL1 Message data 2 : AL2 Message data 3 : MS1 R220.2 EPCB ACT Whenever EPCB=0, ACT=1 R201.2 ACT W1 When ACT=1, promptly R1=1, display being begun. When display is completed, automatically W1=0.

R201.3 .c R201.2 SUB 49 One Total Control Messa sum messag ge of es 10 addres message s s R220 30 nc DISP 1 1010 SPINDLE ALARM 2 1020 ATC ALARM w 3 2100 WORK SET UP w W1 w R201.3 EP CA R295.0 Interlock signal for external data input function 5.42.9 (a) 218 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Table 5.42.9 Instruction RD Address No. Bit No. Remarks ST2 R201.2 49 30 Total sum of message data (PRM) 10 Number of data of one message (PRM) (PRM) R220 Message control address (PRM) 1010 Message No. 8380 SP 7378 IN 6876 DL 6932 E_ 6576 AL 6582 AR 7700 M (Note1) 0000 1020 Message No. 6584 AT 6732 C_ 6576 AL 6582 AR 7700 0000 .c nc 0000 8275 Message data 2 (10 data m=10) M 0000 8779 Message data 1 (10 data m=10) ce nt e 0000 om SUB 2100 ST0 ACT (PRM) 0000 ST1 r.c Step Number 3200 Message No. W0 RK _ Message data 3 10) m=10) (10 data m 0192 w 0222 0221 0196 w 0222 (PRM) 0216 R201.3 RD R201.3 W1 WRT R295.0 W1 w Process end (W1) ACT W1 WRT NOTE 1 00 is ignored data. 2 Display example (The following is displayed on the screen in message data 1). 1010 SPINDLE ALARM 219 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (b) Using three DISP instructions and one external tool offset EPCB ACT W1 1st DISP instruction W2 2nd DISP instruction DISP DISP W3 W1 EP CA W2 nc W3 ACT JMP n .c w w Use as external tool offset interlock. EPCA=1 during execution of each DISP instruction. When EPCA=0, see (8), (iii) W3 Processing of external tool offset W3 Writing into external data input interface w External tool offset end condition 3rd DISP instruction Use a JMP instruction, for example, so that no external data input interface (addresses) may be DISP instruction (EPCA=1). Jump External tool offset start condition om DISP r.c ACT Set ACT to 1 whenever no external tool offset is being precessed (EPCB=0) ce nt e ACT ACT En EPCA EP CB EPCB 5.42.9 (b) 220 Always turn EPCB off on completion of external tool offset. "completion" is when processing has been exactly completed and NC signal REND=0. PMC SEQUENCE PROGRAM 5.42.10 Variable Data Display by Specifying Variable Data 5. FUNCTIONAL INSTRUCTIONS Conform to the following instruction format. Variable data, i.e., any numeric value of up to four BCD digits, can be displayed. SUB49 Instruction format ffff Total number of steps in message data ffff Number of steps in one message data item ffff ffff Message control address Message number om B–61863E/10 ffff 990m VVVV Variable data specification Variable data address Message number ce nt e ffff r.c Message characters Message characters ffff ffff 990m VVVV Message number Variable data specification Variable data address nc Message characters 990m w w w .c VVVV Variable data specification Variable data address NOTE 1 One step is used at variable data specification 990m. 2 The number of steps is the same for each message data item. The number of characters to be displayed varies according to the value specified for m. 3 Multiple variable data items can be used in one message data item. (1) Specifying variable data 990m Specifies the number of digits in the variable data.

(1xmx4) Variable data 221 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (2) Variable data address VVVV: AddressR300 0011 0100 R301 0001 0010 om (3) Variable data Specify variable data consisting of up to four BCD digits (the number of digits specified for m) to be displayed at the address specified by the variable data address using the sequence program. For example, variable data 1234 is specified at variable data address R300 in BCD as shown below: w w w .c nc ce nt e r.c (4) Example To display TOOL NO 123 SUB49 0007 Total number of steps in message data 0007 Number of steps in one message data item R300 Message control address 2100 Message number 8479 TO 7396 OL 3278 N 7932 O 9903 Variable data specification R350 Variable data address 222 AddressR350 0011 0100 R351 0001 0010 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.43 DISPB 5.43.1 This instruction displays messages on the CRT/MDI screen. You can also specify the message number to generate an alarm in the CNC. This instruction supports special functions (numerical data display and kanji character display) in addition to the same basic functions as those of the message display instruction (DISP), described in Section 5.42. However, it performs a special additional function, namely, it displays numerical data. You can program up to 200 messages. You must use the special message addresses in your program (see Sec. 3, 'Address') to simplify use of the messages. The following are the features of this function. om Function ce nt e r.c (a) In the program you define the total number of messages by using DISPB, and set ACT=1.It does not matter if ACT is already set at '1'.

If, however, ACT = 0, DISPB will not process the messages at all. When ACT = 1, messages are displayed according to the contents of the message display request memory (addresses A0 to A24) and the message data table. Relation between the message display request memory address and the message data table appears in Table 5.43.1. Message display request memory (RAM) Address Number of message data table A0.0 (Message datacorresponding to address A0.0) A0.1 (Message datacorresponding to address A0.1) 7 6 5 4 3 2 1 0 .c (Maximum) A24.7 (Message datacorresponding to address A24.7) w (Maximum) A2 nc A0 A1 A2 Message data table (written in ROM) w w 5.43.1 Message display request memory and message data table (i) Message display request memory (RAM) Addresses A0 to A24 constitute a 200-bit area. This is a display request memory for up to 200 messages, each bit corresponding to a message. If you want to display a message on the CRT, set the corresponding display request memory 1. Set 0 to erase the message of CRT. (ii) Message data table This table stores messages corres-ponding to the message display request bits. The table is stored in the EPROM together with the sequence program. Message data table numbers correspond to the message display request memory addresses. The message data table capacity is prepared by the maximum 223 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 capacity of a message, or, 255 characters (255 bytes). Produce a message data within this capacity. A character prepared in CRT/MDI key consists of one byte, and 4 bytes are necessary for a message number (consisting of 4 characters) in the next item. A character not covered by the CRT/MDI keys requires two bytes (a half–width kana character) or four bytes (a kanji character or other full–width character). For details, see Section 5.43.6. om (iii) Message number This message number consisting of 4 digits must always be defined at the start of each message data. The CRT display is as specified below by this message number. Message number Alarm message screen Display contents Alarm message D CNC is turned to alarm state. ce nt e 1000 to 1999 CNC screen r.c D FS16-M/T, FS18-M/T, FS15, FS20, Power Mate-D (single path control), Power Mate-F and Power Mate-H 2000 to 2099 2100 to 2999 Operator message screen Operator message Operator message (without message number) D Only message data, no message number, is displayed. D FS16-TT and FS18-TT Message number 2000 to 2099 2100 to 2999 w w w .c 5000 to 5999 Display contents Alarm message screen (The 1st tool post side) Alarm message D The 1st tool post side of CNC is turned to alarm state. Operator message screen Operator message Alarm message screen (The 2nd tool post side) Alarm message D The 2nd tool post side of CNC is turned to alarm state. D The displayed message number is a value by witch 4000 is subtracted from specified number. nc 1000 to 1999 CNC screen Operator message (without message number) D For 3–path control Message number CNC screen Display contents 1000 to 1999 Alarm screen (on path 1) Alarm message D Path 1 is placed in the alarm state. 2000 to 2099 Operator message screen Operator message 5000 to 5999 Alarm screen (on path 2) Alarm message D Path 2 is placed in the alarm state. D The displayed message number is a specified number from which 4000 is subtracted. 7000 to 7999 Alarm screen (on path 3) Alarm message D Path 3 is placed in the alarm state. D The displayed message number is a specified number from which 6000 is subtracted. 2100 to 2999 224 Operator message (with no message number) PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS D Power Mate-D (dual path control) Message number CNC screen Display contents 1000 to 1999 Alarm message screen (The 1st path side) Alarm message D The 1st path side of CNC is turned to alarm state. 2000 to 2099 Operator message screen (Th 1st path h (The side) Operator message 5000 to 5999 Alarm message screen (The 2nd path side) Alarm message D The 2nd path side of CNC is turned to alarm state. D The displayed message number is a value by witch 4000 is subtracted from specified number.

6000 to 6099 Operator message screen (The 2nd path side) Operator message D The displayed message number is a value by witch 4000 is subtracted from specified number.

6100 to 6999 om Operator message (without message number) r.c 2100 to 2999 Operator message (without message number) ce nt e Moreover, the DPL/ MDI display with Power Mate is as specified below by this message number. D Power Mate-D (single path control), Power Mate-F and Power Mate-H Message number CNC screen Alarm message screen Message number D CNC is turned to alarm state. D Only message number, no message data, is displayed. nc 1000 to 1999 .c 2100 to 2999 D Power Mate-D (dual path control) w w Message number w Display contents CNC screen Display contents 1000 to 1999 Alarm message screen (The 1st path side) Message number D The 1st path side of CNC is turned to alarm state. D Only message number, no message data, is displayed. 2000 to 2099 Operator message screen (Th 1st path h (The side) Operator message D Only message data, no message number, is displayed. 5000 to 5999 Alarm message screen (The 2nd path side) Message number D The 2nd path side of CNC is turned to alarm state. D Only message number, no message data, is displayed. D The displayed message number is a value by witch 4000 is subtracted from specified number. 6000 to 6099 Operator message screen (Th 2nd d path h (The side) Operator message D Only message data, no message number, is displayed. 2100 to 2999 6100 to 6999 225 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 om NOTE 1 The number of message number which you can display at the same time to the alarm screen on DPL/MDI is up to 3. 2 The number of character which you can display to the operator message screen on DPL/MDI is up to 32 characters. The message data since the 33rd character is not displayed.

3 A "~" character (code A0H) is displayed as space character to the screen on DPL/MDI. 4 The DPL/MDI cannot display kanji (double-byte) characters. r.c (b) You need not use numerical codes for message data input. Instead, when programming, directly key in the characters making up the messages (from the CRT/MDI keyboard). For the characters that CRT/MDI does not provide for, you must enter these characters by numerical data with special symbols "@". For details, refer to Subsec. 5.43.6). ce nt e (c) Use external data input command (described later) where you must combine the DISPB instruction with external data input function (for external tool compensation, external workpiece No. search, etc.).

Such use of the DISPB instruction does not affect the interface of external data input function though the common interface is used between DISPB instruction and external data input function. nc (d) If you write the message data items in the ROM after programming, you cannot change them any more (they will become fixed data items). However, you can still change and display only the numerical data forming part of the messages if you specify addresses storing the numerical data as the message data and assign the required numerical data in these addresses through sequence program. Use of this function makes it possible for you to display frequently varying numerical data (such as tool number etc.) during automatic operations. w w .c (e) A message is displayed on the CNC alarm message/operator message screen. When using the DISPB instruction, you must satisfy the following conditions: To use DISPB, the optional External Data Input function or External Message Display is necessary for CNC. w 5.43.2 Format DISPB fff (SUB 41) Number of message ACT 226 PMC SEQUENCE PROGRAM B–61863E/10 5.43.3 5. FUNCTIONAL INSTRUCTIONS ACT=0 : Do not display messages on the CRT. ACT=1 : Display the messages on the CRT. Conditions 5.43.4 (a) Number of messages Specifies the total number of messages (up to 200). Numerical Data Display To change the numerical data contained within the messages, enter in the messages the number of digits making up the data and the memory address to contain the data. To differentiate between the numerical data from the other message data, write it within [ ] in the message. Since the brackets, [ ], are used to contain numerical data, they are not themselves treated as symbols to be included in the messages.

(a) Numerical data format [Ibid, ffff] om 5.43.5 r.c Parameters ce nt e Memory address storing the numerical data. The numerical data must be of binary format. Set the bid data after character i: b: i: d: Number of bytes (1, 2, or 4) Number of digits in the integer part (0 to 8) Number of digits in the fractional part (0 to 8) w w w .c nc NOTE 1 Sum of integer part digits and fractional part digits must be within 8.

2 Blank is displayed for digits exceeding 8 digits. 5.43.6 Defining Characters not found in the CRT/MDI (b) Example The following message includes a 3-digit tool number at the spindle and the offset data (f.ff) for this tool. And these data is contained in a 2-byte memory address: SPINDLE TOOL No. = [I 230,VVVV] OFFSET DATA = [I 212, nnnn] Message characters not covered by the CRT/MDI keys (kanji and half–width kana characters) can be input as follows: (a) Half–width kana characters (i) Data format Numerical code enclosed by @ and @ (ii) Input method Enter the numerical codes corresponding to the characters to be input, by referring to the character code table (Table 5.43). Each character requires two bytes. Characters covered by the CRT/MDI keys can also be input in this way. 227 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (iii) Example To input ATC?

OK when characters A, T, C, O, and K are registered in the CRT/MDI unit, enter the following: (b) Kanji (full–width) characters FS20 FS21A FS21B FS21i FS18A RA1 RA3 RA1 RA3 RA1 RA5 RA1 RA3 f f f FS16A FS16B FS18B FS16C FS18C Mate RA2 RB RB3 f RB2 RC RC3 RB3 RB4 RC3 RC4 f f f FS16i FS18i RB5 RB6 RC3 RC4 RB5 RB6 f f f FS15B NB NB2 r.c PA1 PA3 om Power Ę : Can be used : Cannot be used ce nt e NOTE 1 The PMC–RA1 for the FS18–A can be used when the PMC management software series is 4071. 2 The PMC–RB for the FS16–A can be used when the PMC management software series is 4063. 3 The PMC–RC/RC3 for the FS16–A cannot be used depending on the series and edition of the CNC software. 4 For the FS16–A, set the following CNC parameter: – No. 6300 bit 6 = 0: Kanji characters are used for the DISPB instruction (default). 1: Kanji characters are not used for the DISPB instruction. When kanji characters are used, the DISP instruction cannot be used. 5 On the CNC, the external data input option or external message option must be selected. 4434 3A3A 01 w w w .c nc (i) Data format Numerical code enclosed by @02 and 01@ (ii) Input method Enter the numerical codes corresponding to the characters to be input, by referring to the kanji, hiragana, and special code table in Appendix O.

Each character requires four bytes.

(iii) Example To input ATC? OK when characters A, T, C, O, and K are registered in the CRT/MDI unit, enter the following: 228 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS ce nt e r.c om NOTE 1 To define @, enter @40...@, where 40 is the code corresponding to @.

@40 . . . . . @ Code for @ 2 To renew the message line displayed on the CRT/MDI screen, input as: @ OA @ at the end of the data. 3 When using numerical codes, @ code occupies 1 byte, and space code occupies 2 bytes. (Space code = 20, 2 and 0 occupies 1 byte each). 4 The following control codes are used: 02 : 2–byte code (kanji and hiragana characters) 01 : 1–byte code (alphanumerics and half–width kana characters) Do not specify 02 or 01 between @02 and 01@, as follows. The characters may not be correctly displayed. @02 ... 02 ... 01@ @02 ... 01 ... 01@ Table 5.43.6 Character code table 2 (Space) 4 A B to _ *3) 0 @ P ! 1 A Q 2 # 2 B R 3 # 3 C S 4 $ 4 D T % 5 E U 6 & 6 F V 7 ' 7 G W 8 (8 H X 9) 9 I Y A * : J Z B + ; K [C , < L ¥ D ± *1) = M] E · > N η F / ? O __ *2) w .c 5 w w 5 1 nc 0 3 5.43.7 C D *1) Minus, *2) Under bar, *3) Long bar *4) Dakuten *5) Han-dakuten Refer to Sec. 9.3. Notes when this Functional Instruction is Used in Subroutine 229 5.

FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.43.8 Foreign Language Display FS20/ FS21B FS18A FS16A PA1 PA3 RA1 RA3 RA1, RA2 RA3 RB, RB2,RB3 RC, RC3 FS16B FS18B RB3 RC3 RB4 RC4 f FS16C FS18C RB5 RC3 RB6 RC4 f FS21 RA1 RA5 FS16 FS18 RB5 RB6 FS15B NB NB2 om Power Mate/ FS21A f ce nt e r.c (a) General In the message data areas corresponding to contiguous message display request memory locations, message data can be displayed in any of several languages. The language in which a message is displayed is selected by shifting the message display request bit according to the address bit shift amount set in setting parameter 2. A0.0 Language 1 A0.1 Language 2 A0.2 Language 3 When A0.0 is turned on after setting the message display request bit shift amount to 2, the message display request bit is shifted by 2 bits to display language 3. A0.3 Language 4 A0.4 Language 5 The parameters set on the setting parameter 2 screen are listed below. nc D MESSAGE SHIFT VALUE Message display request bit shift amount w w w .c D MESSAGE SHIFT START ADDRESS Start bit address of the message display request bit area to be shifted (b) Examples Example 1: Message data in any of four languages is set starting at A0.0 in the order of Japanese, English, Italian, German, Japanese and so on. The Italian message data is displayed. Set the parameters as follows: MESSAGE SHIFT VALUE : 2 MESSAGE SHIFT START ADDRESS : A0.0 (MESSAGE SHIFT VALUE = 0:Japanese/1:English/2:Italian/3:German) Manipulate A0.0, A0.4, A1.0, A1.4, and so forth with the ladder. 230 5.

FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Message table A0.0 Japanese 1 A0.1 English 1 A0.2 Italian 1 When A0.0 is turned on, Italian 1 is displayed. (The message data is shifted by 2 bits). A0.3 German 1 A0.4 Japanese 2 A0.5 English 2 A0.6 Italian 2 When A0.4 is turned on, Italian 2 is displayed. (The message data is shifted by 2 bits). : : r.c ; Am.n om A0.7 German 2 .c nc ce nt e Example 2: As common alarm messages, English message data is displayed with A0.0 through A9.7. Operator messages are set starting at A10.0 in the order of Japanese, English, Italian, German and so on, and German message data is displayed. Set the parameters as follows: MESSAGE SHIFT VALUE : 3 MESSAGE SHIFT START ADDRESS : A10.0 (MESSAGE SHIFT VALUE = 0:Japanese/1:English/2:Italian/3:German) Manipulate A10.0, A10.4, A11.0, A11.4, and so forth with the ladder. When any of A0.0 to A9.7 is turned on, the message corresponding to the bit is displayed. w w w Message table A0.0 English A A0.1 English B (ALARM) A0.2 English C (ALARM) (ALARM) A10.0 Japanese 1 (OPE) A10.1 English 1 (OPE) A10.2 Italian 1 (OPE) A10.3 German 1 (OPE) A10.4 Japanese 2 (OPE) A10.5 English 2 (OPE) A10.6 A10.7 Italian 2 German 2 (OPE) (OPE) : : ! : : : 231 When A0.1 is turned on, English B is displayed. When A10.0 is turned on, German 1 is displayed. (The message data is shifted by 3 bits). When A10.4 is turned on, German 2 is displayed. (The message data is shifted by 3 bits).

5.

FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Example 3: r.c om As common alarm messages, English message data is displayed with A0.0 through A9.7. Operator messages are set starting at A10.0 in the order of Japanese, English, Italian, German and so on, with 40 successive messages assigned to each language. For these messages, German message data is displayed. Set the parameters as follows: MESSAGE SHIFT VALUE : 120 (40 x 3) MESSAGE SHIFT START ADDRESS : A10.0 (MESSAGE SHIFT VALUE = 0:Japanese/40:English/80:Italian/120:German) Manipulate A10.0 through A14.7 with the ladder. When any of A0.0 to A9.7 is turned on, the message corresponding to the bit is displayed. Message table English A A0.1 English B (ALARM) (ALARM) A0.2 English C (ALARM) When A0.1 is turned on, English B is displayed. ce nt e A0.0 Japanese 1 (OPE) A10.1 Japanese 2 (OPE) A15.0 English 1 (OPE) A15.1 English 2 (OPE) A20.0 Italian 1 (OPE) w w w .c nc A10.0 A20.1 Italian 2 (OPE) A25.0 German 1 (OPE) A25.1 German 2 (OPE) : : Am.n : When A10.0 is turned on, German 1 is displayed. When A10.1 is turned on, German 2 is displayed. (The message data is shifted by 120 bits). (c) Notes The same message number should be assigned to a message in each language that has the same meaning. Message table A0.0 A0.1 1000 1001 A10.0 1000 Japanese 1 (OPE) A10.1 1001 Japanese 2 (OPE) 232 English A English B (ALARM) (ALARM) 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.44 EXIN (EXTERNAL DATA INPUT) 5.44.1 This instruction is used for external data (external tool compensation, external message function, external program number search, external workpiece coordinates shift, etc.) input. You must use this instruction when combining the message display instruction (DISP, DISPB) with the external data input function.

If you are not used DISP or DISPB, you need not use this instruction either. Instead, use the external data input interface PMCCNC directly in your program. The DISPB instruction uses the interface between the PMC and CNC provided by the external data input function during display. The DISP instruction prevents the interface signal transferred between the PMC and CNC from being changed due to external cutter compensation or others. You can use the EXIN instruction only when the PMCCNC interface is of BMI (Basic Machine Interface) and optional external data input function is provided with CNC. An 4-byte control data as described below is required for external data input function (option). ce nt e r.c om Function 5.44.2 Format ACT EXIN w .c nc (SUB 42) 5.44.3 w w Control Conditions 5.44.4 Parameter ffff W1 Control data address 5.44.2 ACT=0 : Do not process external data input/output. ACT=1 : Process external data input/output. ACT is to be maintained '1' till the end of external data input/output. After external data input, reset ACT (W1 = 1).

(a) Control data (except PMC–NB) The control data requires an area of four consecutive bytes beginning with an address to be specified. In 16-TT and 18-TT, the first-byte area is used for specifying a tool post. In systems other than 16-TT and 18-TT, specify 0 for the area. Specify data to be set for addresses G0 to G2 of the interface from PMC to NC for the remaining three-byte area in sequence. For tool post 2 of 16-TT or 18-TT, specify data to be set for addresses G1000 to G1002 in sequence.

(Be sure to set the strobe signal (ESTB) to ON.) 233 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 CTL+0 HEAD.NO (TT) +1 ED0 to ED7 +2 ED8 to ED15 +3 EA0toEA6,ESTB +4 r.c om [For systems other than 16-TT and 18-TT] CTL+0 : 0 CTL+1 to CTL+3 : Data to be specified for G0 to G2 [For 16-TT and 18-TT] (i) Tool post 1 CTL+0 : 0 CTL+1 to CTL+3: Data to be specified for G0 to G2 ce nt e (ii) Tool post 2 CTL+0 : 2 CTL+1 to CTL+3: Data to be specified for G1000 to G1002 [3–path control] (i) On path 1 CTL+0: Set 0. CTL+1 to 3: Data to be set in G0 to G2 (ii) On path 2 CTL+0: Set 0. CTL+1 to 3: Data to be set in G1000 to G1002 nc (iii) On path 3 CTL+0: Set 0. CTL+1 to 3: Data to be set in G2000 to G2002 w w w .c NOTE Refer to the "Series 16 or 18 Connection Manual" for detailed data to be specified concerning external data input. (b) A consecutive area in eight bytes is necessary as the control data. In 15- M/ T, set command data in this CTL+0 - +7 by the same data form as G32- 39 of BMI interface. In 15- TT, set command data in this CTL+0 - +7 by the same data form as G112- 119 of BMI interface. 234 PMC SEQUENCE PROGRAM B–61863E/10 5.

FUNCTIONAL INSTRUCTIONS CTL+0 EISTB, EOREND etc.

+1 EIA0 to EIA7 +2 EID32 to EID39 +3 EID40 to EID47 +4 +5 EID8 to EID15 +6 EID16 to EID23 +7 r.c EID25 to EID31 om EID0 to EID7 ce nt e A consecutive area in 16 bytes is necessary as the control data. In 15-M/T, set command data in first CTL+0 - +7 by the same data form as G32-39 of BMI interface. The data output from NC is written in CTL+8 - +15 in the same data form as BMI interface F32-39. In 15-TT, set command data in first CTL+0 - +7 by the same data form as G112-119 of BMI interface. The data output from NC is written in CTL+8 - +15 in the same data form as BMI interface F112-119. CTL+0 nc to EISTB, EOREND etc +1 EIA0 to EIA7 EOSTB, EIREND etc +9 EOA0 to EOA7 EID32 to EID39 +10 EID40 to EID47 +11 .c +2 EOD32 to EOD39 +3 EOD40 to EOD47 w +4 EID0 to EID7 +12 EID8 to EID15 +13 EID16 to EID23 +14 EID25 to EID31 +15 EOD0 to EOD7 w +5 EOD8 to EOD15 +6 EOD16 to EOD23 +7 w to CTL+8 EOD25 to EOD31 to to NOTE Refer to the following manuals in detail of BMI interface. "FANUC Series 15-MODEL B Connection Manual (BMI interface)" 235 5. FUNCTIONAL INSTRUCTIONS 5.44.5 End of Transfer (W1) PMC SEQUENCE PROGRAM B–61863E/10 This indicates end of transfer of external data. This transfer end condition shows the end of a series of external data input sequence. This functional instruction executes a series of transfer sequence, and finally sets ESTB = 0 in the PMC ^{3} NC interface. As a result, W1 is set to 1 (W1 = 1) after confirming that EREND = 0. When W1 = 1, transfer of data is over. Reset ACT now. om CAUTION 1 The EXIN command cannot input multiple external data items at the same time. Be sure to issue the next EXIN command (ACT = 1) after external data transfer ends (W1 = 1). 2 Be sure to specify an interlock when the external data input function r.c is used by commands other than the function commands, DISP, DISPB, and EXIN. 5.44.6 ce nt e If any of the following errors occurs during external data input, the bit in the operation output register is set. In this case, external data transfer ends (W1 = 1). Operation Output Register 7 6 5 4 3 2 1 0 nc R9000 (Description of errors) D When the EXIN command (ACT = 1) is started, the strobe signal (ESTB) or EREND signal is already on.

The external data may be input by commands other than the function commands, DISP, DISPB, and EXIN. D An invalid head number was specified for 16-TT or 18-TT. (Data other than 0 to 2 was specified.) D The specification of HEAD.NO is incorrect. (Data other than 0 to 3 is set for 3–path control.) .c w w w 5.44.7 EXIN error Refer to Sec. 9.3. Notes when this Functional Instruction is Used in Subroutine 236 PMC SEQUENCE PROGRAM B–61863E/10 5.

FUNCTIONAL INSTRUCTIONS 5.45 WINDR (READING CNC WINDOW DATA) 5.45.1 This function reads various data items via the window between the PMC and the CNC. The "WINDR" is classified into two types.

One type completes reading a data during one scan time. Another type completes reading a data during a few scan time. The former is called the function of a high–speed response and the latter is called the function of a low–speed response. om Function r.c 5.45.2 Format ACT ce nt e W1 WINDR Control data address (SUB 51) 5.45.3 ACT=0 : The WINDR function is not executed.

ACT=1 : The WINDR function is executed. Using the function of a high–speed response, it is possible to read the data continuously by always keeping ACT on. However, using the function of a low–speed response, as soon as reading a data is completed, reset "ACT" once (ACT=0). w .c Control Condition nc 5.45.2 5.45.4 w w Parameter (a) Control data address The PMC byte address is used to specify the area where control data is stored. 237 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.45.5 CTL+0 Function code +2 Completion code +4 Data length +6 Data number +8 Data attribute +10 Read data * Only the size of the read data is necessary for the data area below to "CTL+10" usually. r.c +n * Set the control data area by sequence program before executing the "WINDR" or "WINDW". om Control Data See Appendix B WINDOW FUNCTION DESCRIPTION. w w w .c nc ce nt e CAUTION 1 In the functional instructions "WINDR" and "WINDW", the control data area may be temporarily rewritten. Therefore, set the control data area by sequence program before the "WINDR" or "WINDW" is executed even when you specify the none volatile memory area like "D" address for the control data area.

Because, when the power supply is turned off during the control data is rewritten, this rewritten data may be memorized in a none volatile memory. Therefore, note that the "WINDR" or "WINDW" might be executed with the wrong control data when the power supply is turned on next if the control data area is not set by sequence program.

2 Set the control data in the same program level as the "WINDR" or "WINDW" is executed. If you set the control data in the different program level, note that the "WINDR" or "WINDW" might not be executed correctly, because the control data is rewritten during the execution of "WINDR" or "WINDW". 3 In the diagnosis screen, it might be seen that the value of control data is changing.

This is not abnormal. Because the display processing and the execution processing of a sequence program are asynchronously executed. Therefore, the value when the control data is rewritten (above–mentioned) is occasionally displayed. Even in this case, the "WINDR" or "WINDW" is executed correctly. 5.45.6 Reading Completion (W1) W1=0 : "W1" is usually reset. The "W1=0" indicates that the "WINDR" is not executed or the "WINDR" being executed now. W1=1 : "W1" is set when the reading a data is completed by the reading command (ACT=1). If the function of a low–speed response is used, as soon as reading a data is completed, reset "ACT" (ACT=0). 238 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.45.7 If an error occurs during execution of the "WINDR" or "WINDW", the bit in the operation output register is set. At the same time, the reading completion is set (W1=1). Details of the error are output to the completion code (CTL+2) in the control data area. See Appendix B WINDOW FUNCTION DESCRIPTION. Operation Output register 7 6 5 4 3 2 1 0 nc R9000 w w w .c nc Notes when this Functional Instruction is Used in Subroutine When you use the function of a low–speed response, there are a few limitation. Refer to "9.3 NOTE FOR SUBROUTINES WHEN YOU USE SUBROUTINES" When you use the function of a high–speed response, there is no limitation. ce nt e 5.45.8 r.c WINDR error 239 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.46 WINDOW (WRITING CNC WINDOW DATA) 5.46.1 Function om This function writes various data items via the window between the PMC and the CNC. The "WINDR" is classified into the function of a low–speed response. 5.46.2 Format ACT WINDW Control data address ce nt e (SUB 52) r.c W1 5.46.2 5.46.3 nc Control Condition (a) Control data address The PMC byte address is used to specify the area where control data is stored.

w .c 5.46.4 Parameter ACT=0 : The WINDW function is not executed. ACT=1 : The WINDW function is executed. As soon as writing a data is completed, reset "ACT" once (ACT=0). 5.46.5 w w Control Data CTL+0 Function code +2 Completion code +4 Data length +6 Data number +8 Data attribute +10 Writing data * Set the control data area by sequence program before executing the "WINDR" or "WINDW". +42 See Appendix B WINDOW FUNCTION DESCRIPTION. 240 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 nc ce nt e r.c om CAUTION 1 In the functional instructions "WINDR" and "WINDW", the control data area may be temporarily rewritten. Therefore, set the control data area by sequence program before the "WINDR" or "WINDW" is executed even when you specify the none volatile memory area like "D" address for the control data area. Because, when the power supply is turned off during the control data is rewritten, this rewritten data may be memorized in a none volatile memory. Therefore, note that the "WINDR" or "WINDW" might be executed with the wrong control data when the power supply is turned on next if the control data area is not set by sequence program. 2 Set the control data in the same program level as the "WINDR" or "WINDW" is executed. If you set the control data in the different program level, note that the "WINDR" or "WINDW" might not be executed correctly, because the control data is rewritten during the execution of "WINDR" or "WINDW". 3 In the diagnosis screen, it might be seen that the value of control data is changing. This is not abnormal. Because the display processing and the execution processing of a sequence program are asynchronously executed. Therefore, the value when the control data is rewritten (above–mentioned) is occasionally displayed. Even in this case, the "WINDR" or "WINDW" is executed correctly. 5.46.6 w w .c Writing Completion (W1) W1=0 : "W1" is usually reset.

The "W1=0" indicates that the "WINDW" is not executed or the "WINDW" being executed now.

W1=1 : "W1" is set when the writing a data is completed by the writing command (ACT=1). As soon as writing a data is completed, reset "ACT" (ACT=0). 5.46.7 w Operation Output Register If an error occurs during execution of the "WINDR" or "WINDW", the bit in the operation output register is set. At the same time, the writing completion is set (W1=1).

Details of the error are output to the completion code (CTL+2) in the control data area. See Appendix B WINDOW FUNCTION DESCRIPTION. 7 6 5 4 3 2 1 0 R9000 WINDW error 241 5. FUNCTIONAL INSTRUCTIONS 5.46.8 B–61863E/10 When you use the function of a low–speed response, there are a few limitation. Refer to "9.3 NOTE FOR SUBROUTINES WHEN YOU USE SUBROUTINES" w w w .c nc ce nt e r.c om Notes when this Functional Instruction is Used in Subroutine PMC SEQUENCE PROGRAM 242 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.47 ARBITRARY FUNCTIONAL INSTRUCTIONS 5.47.1 5.47.1.1 om FNC 90 to 97 (Arbitrary Functional Instructions) (Only for PMC-RC/RC3/NB/NB2) 5.47.1.2 Fig.5.47.1.2 shows the notation format. Table 5.47.1.2 shows the coding format. ce nt e r.c Function These functional instructions (SUB90 to SUB97) are used to execute the arbitrary functional instructions. These instructions consist of the addresses specifying the start condition, process end output, and control condition. Format ACT SUB90 ffffffff .c nc FNC 90 w w w Step number 5.47.1.3 W1 Control data address Fig.5.47.1.2 FUNC 90 Notation Format Table 5.47.1.2 FUNC 90 Coding Format Command Address No. Bit No. ffff. f Remarks 1 RD 2 SUB 3 (PRM) ffff Control data address 4 WRT ffff. f W1 90 ACT FUNC90 command Control Condition (a) Execution command (ACT) This is used as the start condition of an arbitrary functional instruction. 5.47.1.4 (a) Control data address Specifies the first address in the control data area. Parameter 243 5. FUNCTIONAL INSTRUCTIONS 5.47.1.5 PMC SEQUENCE PROGRAM B–61863E/10 Set the control data to be used by an arbitrary functional instruction. If the control data is determined as follows, for example, the person who created the ladder program determines a control address to set the control data using the ladder program. Control data 7····· addres CTL + 0 CTL + 1 CTL + 2 2 1 0 CNO UPDOWN RST om Control Data Counter number r.c CTL + 3 NOTE If this functional instruction is displayed by the PCLAD display function, an arbitrary functional instruction is displayed as SUB9X, FNC99X. nc Process End Output (W1) This is used as the process end output of an arbitrary functional instruction. ce nt e 5.47.1.6 5.47.2 w 5.47.2.1 .c Creating an Arbitrary Function w w Arbitrary Functional Instruction and Interface (a) Execution command (ACT) The contents of the execution command can be referenced by bit 1 at R9010. (b) Control data address The address where the control data is stored can be referenced in the byte address format at R9012 or later. (c) Process end output (W1) The data output when the process terminates can be referenced by bit 1 at R9011. 244 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.47.2.2 Use of the R Field 96 95 94 93 92 91 90 Execution command data (See (3) in 5.46.1.) R9011 97 96 95 94 93 92 91 90 Process end output data (See (5) in 5.46.1.) R9014 Control data address of SUB91 R9026 Control data address of SUB97 Byte address om Control data address of SUB90 r.c R9012 Reference the start condition (ACT) of the arbitrary function by bit 1 at R9010. Reference the address at which the control data is stored in the byte address format by the fields at R9012 and later. Set the end signal (W1) of an arbitrary function in bit 1 at R9011. For example, to execute the arbitrary function using SUB90, reference the start condition by R9010.0. Reference the control data address in the byte address format by R9012. Set the end signal at R9011.0. w w w .c nc Creating an Arbitrary Function 97 ce nt e 5.47.2.3 R9010 245 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.48 MMCWR (READING MMC WINDOW DATA) (OTHER THAN PMC–PA1/PA3) 5.48.1 This command reads up to 32 bytes of data via the window between PMC and MMC. The data can be determined as required between the PMC ladder program and MMC application program. om Function r.c 5.48.2 Format ffff Input data length address MMCWR ce nt e ACT (SUB98) ffff Input data address W1 Fig.5.48.2 MMCWR Command Format Table 5.48.2 MMCWR Coding Format Coding sheet Command w .c nc Step number w 5.48.3 w Control Condition 5.48.4 Parameters Address No. Bit No. fff.

f Remarks 1 RD ACT 2 SUB 3 (PRM) ffff Input data length address 4 (PRM) ffff. Input data address 5 WRT 98 fff. f W1, processing completion ACT=0 : The MMCWR function is not executed. ACT=1 : The MMCWR function is executed. Hold ACT = 1 until processing is completed and specify ACT = 0 immediately after processing is completed (W1 = 1). (a) Input data length address (two bytes) Specifies the length of input data transferred from MMC. When transfer is completed, the length of data actually transferred is stored. The maximum data length is 32 bytes. (b) Input data address Specifies the area containing data transferred from MMC. An area large enough for the specified input data length is required. 246 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.48.5 Processing Completion (W1) 5.48.6 W1=0 :This value is usually set. W1 = 1 indicates that processing is completed. As soon as processing is completed, specify ACT=0. W1=1 :This value is set when data transfer from MMC is completed or if an error occurs. Operation Output Register 7 5 4 3 2 1 0 r.c R9000 6 om If an MMC window transfer error occurs, the bit in the operation output register is set to indicate the error. If an error occurs, the transferred data is not stored in the input data area.

5.48.7 The completion status information is specified in R9002 and R9003. The completion codes and contents, W1, and error bits are as follows: -11 ····· Initialization at MMC is not completed. (W1 = 0, R9000#0 = 0) -10 ····· Processing is in progress (W1 = 0, R9000#0 = 0) 0 ····· Processing is completed.(W1 = 1, R9000#0 = 0) 2 ····· Data length error (W1 = 1, R9000#0 = 1) (0, a negative value, or a value exceeding 33 bytes was specified for the data length.

The length of data actually transferred exceeded the specified value.) 6 ····· MMC is not provided (W1 = 1, R9000#0 = 1) 5.48.8 .c nc Completion Status Information ce nt e MMCWR error Refer to Sec. 9.3. w w w Notes when this Functional Instruction is Used in Subroutine 247 5.

FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.49 MMCWW (WRITING MMC WINDOW DATA) (OTHER THAN PMC–PA1/PA3) 5.49.1 om This command writes data containing up to 32 bytes via the window between PMC and MMC. The data can be determined as required between the PMC ladder program and MMC application program. Function Format MMCWW ffff Output data length address ffff Output data address W1 ce nt e ACT r.c 5.49.2 (SUB99) Fig.5.49.2 MMCWW Command Format nc Table 5.49.2 MMCWW Coding Format Coding sheet Command Address No. Bit No. fff. f Remarks 1 RD 2 SUB 3 (PRM) ffff Output data length address 4 (PRM) ffff. Output data address 5 WRT ACT 99 fff.

f W1, processing completion w w .c Step number ACT=0 : The MMCWW function is not executed. ACT=1 : The MMCWW function is executed. Hold ACT = 1 until processing is completed and specify ACT = 0 immediately after processing is completed. 5.49.4 (a) Output data length address (two bytes) Specifies the length of output data transferred to MMC. The maximum data length is 32 bytes. 5.49.3 w Control Condition Parameters (b) Output data address Specifies the area storing data to be transferred to MMC. An area large enough for the specified output data length is required. 248 5.

FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.49.5 Processing Completion (W1) 5.49.6 W1=0 :This value is usually set.

W1 = 1 indicates that processing is completed. As soon as processing is completed, specify ACT=0. W1=1 :This value is set when data transfer to MMC is completed or if an error occurs. Operation Output Register 7 5 4 3 2 1 0 r.c R9000 6 om If an MMC window transfer error occurs, the bit in the operation output register is set to indicate the error. If an error occurs, the transferred data is not transferred to MMC. 5.49.7 The completion status information is specified in R9002 and R9003. The completion codes and contents, W1, and error bits are as follows: -11 ····· Initialization at MMC is not completed.(W1 = 0, R9000#0 = 0) -10 ····· Processing is in progress.

(W1 = 0, R9000#0 = 0) 0 ····· Processing is completed. (W1 = 1, R9000#0 = 0) 2 ····· Data length error (W1 = 1, R9000#0 = 1) (0, a negative value, or a value exceeding 33 bytes was specified for the data length.) 6 ····· MMC is not provided. (W1 = 1, R9000#0 = 1) nc Completion Status Information ce nt e MMCWW error 5.49.8 Refer to Sec.

9.3. w w w .c Notes when this Functional Instruction is Used in Subroutine 249 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.50 MOVB (TRANSFER OF 1 BYTE) 5.50.1 f : Can be used : Cannot be used PA1 PA3 RA1 RA2 f RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC RC3 RC4 NB NB2 f f f f Function 5.50.2 Format ACT Transfer Transfer source destinatio address n address r.c MOVB om The MOVB instruction transfers 1-byte data from a specified source address to a specified destination address. 5.50.3 (a) Execution specification ACT=0 : No data is transferred. ACT=1 : One-byte data is transferred. w w w .c nc Control Conditions ce nt e SUB 43 250 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.51 MOVW (TRANSFER OF 2 BYTES) 5.51.1 f : Can be used : Cannot be used PA1 PA3 RA1 RA2 f RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC RC3 RC4 NB NB2 f f f f Function 5.51.2 Format ACT Transfer Transfer source destinatio address n address r.c MOVB om The MOVW instruction transfers 2-byte data from a specified source address to a specified destination address. 5.51.3 (a) Execution specification ACT=0 : No data is transferred. ACT=1 : Two-byte data is transferred.

w w w .c nc Control Conditions ce nt e SUB 44 251 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.52 MOVN (TRANSFER OF AN ARBITRARY NUMBER OF BYTES) 5.52.1 f : Can be used : Cannot be used PA1 PA3 RA1 RA2 f RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC RC3 RC4 NB NB2 f f f f Function 5.52.2 Format ACT MOVN Number of 5.52.3 Control Conditions 5.52.4 transferred ce nt e SUB 45 Transfer source address Transfer destinatio n address r.c bytes to be om The MOVN instruction transfers data consisting of an arbitrary number of bytes from a specified source address to a specified destination address. (a) Execution specification ACT=0 : No data is transferred. ACT=1 : A specified number of bytes are transferred. (a) Number of bytes to be transferred Specify the number of bytes to be transferred. An odd number can also be specified. A number from 1 to 200 can be specified. w w w .c nc Parameters 252 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.53 f : Can be used : Cannot be used DIFU (RISING EDGE DETECTION) PA1 PA3 RA1 RA2 RA3 RA5 f f f 5.53.1 RB RB2 RB3 RB4 RB5 RB6 f f f f RC RC3 RC4 NB NB2 f f f f Function om The DIFU instruction sets the output signal to 1 for one scanning cycle on a rising edge of the input signal. 5.53.2 Format ACT DIFU ce nt e r.c SUB 57 OUT f Rising edge number 5.53.3 (a) Control Conditions Input signal On a rising edge (0→1) of the input signal, the output signal is set to 1.

nc (b) Output signal The output signal level remains at 1 for one scanning cycle of the ladder level where this functional instruction is operating. 5.53.4 (a) Rising edge number Model PA1 PA3 RA1 RA2 RA3 RA5 RB RB2 RB3 RB4 RB5 RB6 RC RC3 RC4 NB NB2 Rising edge number _ 1 to 256 _ _ 1 to 256 1 to 256 _ _ 1 to 256 1 to 500 1 to 256 1 to 500 _ 1 to 256 1 to 500 1 to 256 1 to 500 If the same number is used for another DIFU instruction or a DIFD instruction (described later) in one Ladder diagram, operation is not guaranteed. w w .c Parameters 5.53.5 w Operation 1 ACT OUT 253 2 3 4 Execution period 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.54 f : Can be used : Cannot be used DIFD (FALLING EDGE DETECTION) PA1 PA3 RA1 RA2 f 5.54.1 RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC RC3 RC4 NB NB2 f f f f Function om The DIFD instruction set the output signal to 1 for one scanning period on a falling edge of the input signal.

5.54.2 Format ACT DIFD ce nt e r.c SUB 58 OUT f Falling edge number 5.54.3 (a) Input signal On a falling edge(1→0)of the input signal, the output signal is set to 1. Control Conditions nc (b) Output signal The output signal level remains at 1 for one scanning period of the ladder level where this functional instruction is operating. 5.54.4 (a) Falling edge number Model PA1 PA3 RA1 RA2 RA3 RA5 RB RB2 RB3 RB4 RB5 RB6 RC RC3 RC4 NB NB2 Falling edge number _ 1 to 256 _ _ 1 to 256 1 to 256 _ _ 1 to 256 1 to 500 1 to 256 1 to 500 _ 1 to 256 1 to 500 1 to 256 1 to 500 If the same number is used for another DIFD instruction or a DIFU instruction (described above) in one ladder diagram, operation is not guaranteed. w w .c Parameters 5.54.5 w Operation 1 ACT OUT 254 2 3 4 Execution period 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.55 f : Can be used : Cannot be used EOR (EXCLUSIVE OR) PA1 PA3 RA1 RA2 f 5.55.1 RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC3 RC4 NB NB2 f f f f 5.55.2 Format ACT EOR Format 5.55.3 A address B Control Conditions (a) Input signal ACT=0 : The EOR instruction is not executed. ACT=1 : The EOR instruction is executed. 5.55.4 (a) Format specification Specify a data length (1, 2, or 4 bytes), and an input data format (constant or address specification). w w .c nc Parameters C ce nt e SUB 59 Address Constant or Address r.c specification om The EOR instruction exclusive-ORs the contents of address A with a constant (or the contents of address B), and stores the result at address C. Function W RC Format specification 0 : Constant 1 : Address specification Data length specification 1 : 1 byte 2 : 2 bytes 4 : 4 bytes (b) Address A Input data to be exclusive-ORed.

The data that is held starting at this address and has the data length specified in format specification is treated as input data. (c) Constant or address B Input data to be exclusive-ORed with. When address specification is selected in format specification, the data that is held starting at this address and has the data length specified in format specification is treated as input data. (d) Address C Address used to store the result of an exclusive OR operation. The result of an exclusive OR operation is stored starting at this address, and has the data length specified in format specification. 255 5. FUNCTIONAL INSTRUCTIONS 5.55.5 PMC SEQUENCE PROGRAM B–61863E/10 When address A and address B hold the following data: Operation 1 1 1 0 0 0 1 1 Address B 0 1 0 1 0 1 0 1 om Address A The result of the exclusive OR operation is as follows: 1 0 1 1 0 1 1 0 w w w .c nc ce nt e r.c Address C 256 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.56 f : Can be used : Cannot be used LOGICAL AND PA1 PA3 RA1 RA2 f 5.56.1 RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC RC3 RC4 NB NB2 f f f f Function 5.56.2 Format ACT AND Format specification Constant Address or C address B 5.56.3 ce nt e r.c SUB 60 Address A om The AND instruction ANDs the contents of address A with a constant (or the contents of address B), and stores the result at address C. Control Conditions (a) Input signal ACT=0 : The AND instruction is not executed.

ACT=1 : The AND instruction is executed. 5.56.4 (a) w w w .c nc Parameters Format specification Specify a data length (1, 2, or 4 bytes), and an input data format (constant or address specification).

Format specification 0 : Constant 1 : Address specification Data length specification 1 : 1 byte 2 : 2 bytes 4 : 4 bytes (b) Address A Input data to be ANDed. The data that is held starting at this address and has the data length specified in format specification is treated as input data. (c) Constant or address B Input data to be ANDed with. When address specification is selected in format specification, the data that is held starting at this address and has the data length specified in format specification is treated as input data. (d) Address C Address used to store the result of an AND operation. The result of an AND operation is stored starting at this address, and has the data length specified in format specification. 257 5. FUNCTIONAL INSTRUCTIONS 5.56.5 PMC SEQUENCE PROGRAM B–61863E/10 When address A and address B hold the following data: Operation 1 1 1 0 0 0 1 1 Address B 0 1 0 1 0 1 0 1 om Address A The result of the AND operation is as follows: 0 1 0 0 0 0 1 0 w w w .c nc ce nt e r.c Address C 258 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.57 f : Can be used : Cannot be used LOGICAL OR PA1 PA3 RA1 RA2 f 5.57.1 RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC3 RC4 NB NB2 f f f f 5.57.2 Format ACT OR 5.57.3 A Constant or address B Control Conditions (a) Input signal ACT=0 : The OR instruction is not executed. ACT=1 : The OR instruction is executed. 5.57.4 (a) .c w w C Format specification Specify a data length (1, 2, or 4 bytes), and an input data format (constant or address specification). nc Parameters Address ce nt e SUB 61 Address r.c Format specification om The OR instruction ORs the contents of address A with a constant (or the contents of address B), and stores the result at address C. Function w RC Format specification 0: Constant 1 : Address specification Data length specification 1 : 1 byte 2 : 2 bytes 4 : 4 bytes (b) Address A Input data to be ORed. The data that is held starting at this address and has the data length specified in format specification is treated as input data. (c) Constant or address B Input data to be ORed with. When address specification is selected in format specification, the data that is held starting at this address and has the data length specified in format specification is treated as input data.

(d) Address C Address used to store the result of an OR operation. The result of an OR operation is stored starting at this address, and has the data length specified in format specification. 259 5. FUNCTIONAL INSTRUCTIONS 5.57.5 PMC SEQUENCE PROGRAM B–61863E/10 When address A and address B hold the following data: Operation 1 1 1 0 0 0 1 1 Address B 0 1 0 1 0 1 0 1 om Address A The result of the OR operation is as follows: 1 1 1 1 0 1 1 1 w w w .c nc ce nt e r.c Address C 260 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.58 NOT (LOGICAL NOT) f : Can be used : Cannot be used PA1 PA3 RA1 RA2 f 5.58.1 RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC3 RC4 NB NB2 f f f f 5.58.2 Format ACT 5.58.3 Address Address B A ce nt e SUB 62 Format specification r.c NOT om The NOT instruction inverts each bit of the contents of address A, and stores the result at address B. Function Control Conditions (a) Input signal ACT=0 : The NOT instruction is not executed. ACT=1 : The NOT instruction is executed. 5.58.4 (a) Format specification Specify a data length (1, 2, or 4 bytes), and an input data format (constant or address specification). w w .c nc Parameters w RC Data length specification 1 : 1 byte 2 : 2 bytes 4 : 4 bytes (b) Address A Input data to be inverted bit by bit. The data that is held starting at this address and has the data length specified in format specification is treated as input data. (c) Address B Address used to output the result of a NOT operation. The result of a NOT operation is stored starting at this address, and has the data length specified in format specification. 261 5. FUNCTIONAL INSTRUCTIONS 5.58.5 PMC SEQUENCE PROGRAM B–61863E/10 When address A holds the following data: Operation Address A 1 1 1 0 0 0 1 1 0 0 0 1 1 1 0 0 w w w .c nc ce nt e r.c Address B om The result of the NOT operation is as follows: 262 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.59 f : Can be used Δ : Can be used (with some restrictions) : Cannot be used MMC3 R (MMC-III WINDOW DATA READ) PA1 PA3 RA1 RA2 RA3 RA5 RB RB2 RB3 RB4 Δ f f f f f f f RB5 RB6 RC RC3 RC4 NB NB2 f f f f f om NOTE This functional instruction cannot be used with RA1 of the Series 16i/18i/21i–MODEL A. 5.59.1 Function ce nt e r.c The MMC3R instruction reads MMC-III application data via a PMC-MMC window. Which buffer in the MMC-III is to be read can be specified. The contents of read data can be freely determined by a PMC Ladder program and MMC-III application program. CNC PMC Buffer 1 Offset Data length .c Buffer n n = maximum of 10 w 5.59.2 Buffer 1 MMC-III nc PMC data w Format w ACT MMC3R SUB 88 5.59.3 Control Conditions (ACT) Buffer Offset from specification the beginning Data length Input data W1 f of the buffer ACT=0 : The MMC3R instruction is not executed. ACT=1 : Data is read. 263 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM (a) 5.59.4 Parameters B–61863E/10 Address for storing buffer specifications (2 bytes) A buffer from which data is to be read is specified. Up to 10 buffers can be specified. Specify the address where the buffer specification is held. om NOTE For the method of buffer registration, refer to the relevant MMC-III manual. (b) Address for storing an offset from the beginning of a buffer (2 bytes) An offset from the beginning of a read buffer is specified. Specify the address where the offset is held. r.c (c) Data length storage address (2 bytes) The length of data to be read from the MMC-III is specified. Specify the address where the length of data is held. The maximum allowable data length is 256 bytes. 5.59.5 5.59.6 When W1 indicates the termination of read processing, a termination state is set. 7 6 5 4 3 2 1 w w 5.59.7 Completion Status Information 0 R9000 w .c Operation Output Register W1=0 :When ACT = 0, W1 = 0 is set. If W1 = 0 is set when ACT = 1, it indicates that read processing is in progress. W1=1 :Indicates that read processing has terminated.

Whether read processing has terminated normally or abnormally can be checked with the state of R9000 described below. nc Processing Completion (W1) ce nt e (d) Input data storage address Specify the address where data to be read from the MMC-III is stored. A contiguous area not smaller than the length of data specified in c) above is required. MMC3R error MMC3R=0 : Normal termination MMC3R=1 : Abnormal termination When ACT = 1, completion status information is set in the operation register R9002. –11 : MMC initialization not completed (W1=0, R9000#0=0) 0 : Normal termination (W1=1, R9000#0=0) 2 : Data length error (W1=1, R9000#0=1) The specified length of data is 0, negative data is specified, or the maximum allowable data length is exceeded. 6 : The MMC-III is not attached. (W1=1, R9000#0=1) 3 : Buffer specification error (W1=1, R9000#0=1) 264 PMC SEQUENCE PROGRAM B–61863E/10 5.59.8 5.

FUNCTIONAL INSTRUCTIONS Refer to Sec. 9.3. w w w .c nc ce nt e r.c om Notes when this Functional Instruction is Used in Subroutine 265 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.60 f : Can be used Δ : Can be used (with some restrictions) : Cannot be used MMC3W (MMC-III WINDOW DATA WRITE) PA1 PA3 RA1 RA2 RA3 RA5 RB RB2 RB3 RB4 Δ f f f f f f f RB5 RB6 RC RC3 RC4 NB NB2 f f f f f om NOTE This functional instruction cannot be used with RA1 of the Series 16i/18i/21i–MODEL A. 5.60.1 Function r.c The MMC3W instruction writes data to MMC-III application data via a PMC-MMC window. Which buffer in the MMC-III is to be written to can be specified. The contents of write data can be freely determined by a PMC Ladder program and MMC-III application program. CNC PMC data Buffer 1 MMC- III Buffer 1 ce nt e PMC Offset Data length 5.60.2 w Format .c nc Buffer n n = maximum of 10 MMC3R SUB 89 w w ACT 5.60.3 Control Conditions (ACT) Buffer Offset from specification the beginning Data length Input data W1 f beginning of the buffer ACT=0 : The MMC3W instruction is not executed.

ACT=1 : Data is written. 266 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (a) Address for storing buffer specifications (2 bytes) A buffer to which data is to be written is specified. Up to 10 buffers can be specified. Specify the address where the buffer specification is held.

5.60.4 Parameters om NOTE For the method of buffer registration, refer to the relevant MMC-III manual. (b) Address for storing an offset from the beginning of a buffer (2 bytes) An offset from the beginning of a write buffer is specified. Specify the address where the offset is held. r.c (c) Data length storage address (2 bytes) The length of data to be written to the MMC-III is specified. Specify the address where the length of data is held. The maximum allowable data length is 256 bytes. 5.60.5 When W1 indicates the termination of write processing, a termination state is set. .c 5.60.6 W1=0 :When ACT = 0, W1 = 0 is set. If W1 = 0 is set when ACT = 1, it indicates that write processing is in progress. W1=1 :Indicates that write processing has terminated. Whether write processing has terminated normally or abnormally can be checked with the state of R9000 described below. nc Processing Completion (W1) ce nt e (d) Output data storage address Specify the address where data to be written to the MMC-III is stored.

A contiguous area not smaller than the length of data specified in c) above is required. 7 w 6 5 4 3 2 1 0 R9000 w w Operation Output Register MMC3R error MMC3W=0 : Normal termination MMC3W=1 : MMC3R = 1: Abnormal termination 267 5.60.7 Completion Status Information 5.60.8 PMC SEQUENCE PROGRAM B–61863E/10 When ACT = 1, completion status information is set in the operation register R9002. -11 : MMC initialization not completed (W1=0, R9000#0=0) 0 : Normal termination (W1=1, R9000#0=0) 2 : Data length error (W1=1, R9000#0=1) The specified length of data is 0, negative data is specified, or the maximum allowable data length is exceeded.

6 : The MMC-III is not attached. (W1=1, R9000#0=1) 3 : Buffer specification error (W1=1, R9000#0=1) Refer to Sec. 9.3. w w w .c nc ce nt e r.c Notes when this Functional Instruction is Used in Subroutine om 5. FUNCTIONAL INSTRUCTIONS 268 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.61 f : Can be used : Cannot be used SPCNT (SPINDLE CONTROL) PA1 5.61.1 PA3 RA1 RA2 RA3 RA5 RB RB2 RB3 RB4 RB5 RB6 RC RC3 RC4 NB NB2 f f SPCNT performs the following processing using spindle speed data (16-bit binary data) that is input from the NC or some other device to the PMC: Function om (a) Gear selection (Up to four gears from GR1 to GR4 can be used.) (b) Calculating a spindle motor rotation command (13-bit binary data) when automatic gear selection is enabled (c) Calculating a spindle motor rotation command (13-bit binary data) when direct gear selection is enabled r.c (d) Clamping the spindle motor speed to an upper or lower limit ce nt e (e) Calculating a spindle motor rotation command when a spindle override is specified As shown in Fig.5.61.1, a spindle motor rotation command is calculated from the spindle speed data. The maximum value (8191) of the spindle motor command is equivalent to an analog voltage at 10V.

Spindle motor rotation command (13-bit binary data) Maximum motor speed (8191) Upper limit at which the motor speed is clamped .c Lower limit at which the motor speed is clamped GR2 GR3 GR4 nc GR1 w Maximum spindle speed for GR1 Maximum spindle speed for GR2 Spindle speed (rpm) Maximum spindle speed for GR3 Maximum spindle speed for GR4 w w 5.61.1 Spindle Speeds and Corresponding Spindle Motor Rotation Commands The spindle motor rotation command is calculated as 13-bit binary data. If the spindle amplifier is a D/A converter provided in the machine and can only handle 12-bit binary data, for example, the calculated spindle motor rotation command must be halved before being output (shifted right one bit position in a shift register). (i) Spindle control with automatic gear selection This functional instruction uses spindle speed data (16-bit binary data) and the maximum spindle speeds set in parameters GR1 to GR4 of this functional instruction to select a gear, calculate the spindle motor rotation command for that selected gear, and output the result to the control data address. Based on this output information, the sequence program must perform gear switching as necessary and output the rotation command to the spindle motor. 269 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Spindle control 8 Å Ζ SPCNT Spindle motor rotation command om Spindle speed Information of GR1 to GR2 ɗ ȧ ʐ ȧ 8 ce nt e r.c (ii) Spindle control with direct gear selection When direct gear selection is set, this functional instruction does not perform gear selection. A gear to be used is selected by the sequence program. The target gear must be set at the control data address, which is a parameter of this functional instruction, using the sequence program. According to the set gear, the functional instruction calculates and outputs a spindle motor rotation command. In this case, the spindle motor rotation command has a linear relationship with the spindle speed. The line for the selected gear is assumed to extend to its lower limit (indicated by a dotted line).

See Fig.5.61.1. Spindle control SPCNT w .c nc Spindle speed Information of GR1 to GR2 ɗ ȧ ʐ ȧ 8 w Format w Spindle motor rotation command The spindle motor speed can be clamped at the upper and lower limits also with direct gear specification. When the CNC performs constant surface speed control, spindle control with direct gear specification is generally performed. 5.61.2 CIRC * * * SPCNT ffff ffff ffff (SUB46) Spindle speed data Spindle control Control data address parameter address address OVRD ACT 8 Å Ζ 270 PMC SEQUENCE PROGRAM B–61863E/10 5.61.3 5. FUNCTIONAL INSTRUCTIONS Control Conditions 5.61.4 Direct gear specification (CIRC) CIRC=0 : Disables direct gear specification.

(Enables automatic gear selection.) CIRC=1 : Enables direct gear specification. (b) Override specification (OVRD) OVRD=0 : Disables the override function OVRD=1 : Enables the override function. (c) Instruction execution specification (ACT) ACT=0 : The SPCNT instruction is not executed. ACT=1 : The SPCNT instruction is executed.

(a) Spindle speed data address Specifies an even-numbered address at which the spindle speed data (16-bit binary data) is stored.

r.c Parameters om (a) ce nt e (b) Spindle control parameter address Specifies an even-numbered address at which the parameters for spindle control are stored. Binary data is set in contiguous 24-byte memory locations starting at the specified address. Spindle control parameter +0 Lower spindle motor speed limit data Spindle control parameter +4 Upper spindle motor speed limit data Spindle control parameter +8 Maximum spindle speed for gear 1 Spindle control parameter +12 Maximum spindle speed for gear 2 Spindle control parameter +16 Maximum spindle speed for gear 3 Spindle control parameter +20 Maximum spindle speed for gear 4 w w .c nc Spindle control parameter +24 This 24-byte memory area is specified by addressing, and so it can be allocated in any addressable memory location. For this type of data, however, a data table in nonvolatile memory is most suitable. For maintenance convenience, the memory area should be allocated in the first data table (table group 1). (i) Lower spindle motor speed limit data Sets the lower spindle motor speed limit obtained from the following expression: = Minimum speed (rpm) specified for the spindle motor 8191 Maximum speed (rpm) obtainable by the spindle motor A value from 0 to 8191 can be specified as the lower speed limit data. The maximum spindle motor speed is achieved when 10 V is applied to the motor. (ii) Upper spindle motor speed limit data Sets the upper spindle motor speed limit obtained from the following expression: Upper spindle motor speed limit data = Maximum speed (rpm) specified for the spindle motor Maximum speed (rpm) obtainable by the spindle motor 271 8191 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (iii) Maximum spindle speed for GR1 Sets a maximum spindle speed (rpm) for GR1. The maximum spindle speed must be set in this parameter even when GR1 gear is not provided. The maximum spindle speed is the speed of the spindle when the motor operates at its maximum speed. (iv) Maximum spindle speed for GR2 Sets a maximum spindle speed (rpm) for GR2. When GR2 is not provided, this parameter must be set to 0.

om (v) Maximum spindle speed for GR3 Sets a maximum spindle speed (rpm) for GR3.

When GR3 is not provided, this parameter must be set to 0. r.c (vi) Maximum spindle speed for GR4 Sets a maximum spindle speed (rpm) for GR4. When GR4 is not provided, this parameter must be set to 0. Specified address+0 7 6 R08 R07 Specified address+1 Specified address+2 Specified address+3 SOV128 SOV64 ce nt e (c) Control data address Contiguous 4-byte memory locations starting at the even-numbered address specified in the control data address parameter must be specified. 5 4 3 2 1 0 R06 R05 R04 R03 R02 R01 R13 R12 R11 R10 R09 GR4 GR3 GR2 GR1 SOV8 SOV4 SOV2 SOV1 SOV32 SOV16 8 Å Ζ Spindle motor rotation command } Spindle gear } selection } Spindle override nc (i) Spindle gear selection w .c 7 5 4 3 2 1 0 GR4 GR3 GR2 GR1 [For automatic gear selection] This functional instruction finds an appropriate gear using the spindle speed data and the maximum spindle speed for each gear, then outputs the result to GR1 to GR4. [For direct gear selection] The sequence program sets the gear to be used in GR1 to GR4. This functional instruction calculates the spindle motor rotation commands for all speeds from the upper motor speed limit to the lower speed limit (extended portion indicated by dotted line). See Fig. 5.61. w w 6 (ii) Spindle motor rotation command 7 6 5 4 3 2 1 0 R08 R07 R06 R05 R04 R03 R02 R01 R13 R12 R11 R10 R09 The spindle motor rotation command (13-bit binary data) calculated by this functional instruction is set at these control data addresses. This instruction specifies a spindle motor rotation command with a spindle override applied. 272 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS (iii) Spindle override 7 6 5 4 3 2 1 0 SOV128 SOV64 SOV32 SOV16 SOV8 SOV4 SOV2 SOV1 The sequence program must set a spindle override in binary. A spindle override from 0% to 255% can be set in binary. Use of Spindle Control Spindle control is primarily used to control the spindle speed during normal cutting. It can, however, also be used to: om 5.61.5 r.c (a) Rotate the spindle motor at a specific speed when the gear is switched The sequence program can output appropriate 13-bit binary data as a spindle motor rotation command to rotate the spindle motor at a specific speed, without using this functional instruction. Spindle control Spindle speed data w w w .c Orientation spindle speed data ɗ ȧ ʐ ȧ 8 ɗ ʐ 8 nc Spindle speed data sent from the NC, etc. ce nt e (b) Rotate the spindle at a specific speed during spindle orientation This is enabled by specifying appropriate spindle speed data in the functional instruction (SPCNT). During spindle orientation, the spindle is rotated at the specified orientation spindle speed with the currently selected gear (gear selection is not performed). Gear selection is disabled by setting CIRC to 1 (direct gear specification). Information of GR1 to GR2 8 Å Ζ SPCNT Spindle motor rotation command (c) Control the spindle in a tapping cycle In a tapping cycle, spindle rotation is reversed at the bottom of a hole. Using the HIGH gear to reverse the rotation requires a lower analog voltage than using the LOW gear. So, using the HIGH gear reduces the machining time. To widen the usable range of the HIGH gear, set CIRC to 1 to disable automatic gear selection.

(d) Clamp the spindle speed When the BMI interface is used between the NC and PMC, spindle should be controlled by the PMC (sequence program), as described in the BMI manual. Clamping the spindle speed is one of the spindle control operations. The spindle control functional instruction SPCNT (SUB46) can be used to clamp the spindle speed. The clamping method is outlined below. For precise control, conform to the specifications of the machine supplied by the machine tool builder. 273 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM 8 ȧ ȧ ȧ Å ȧ ȧ ȧ Ζ Spindle control GR1 to GR4 Spindle motor rotation command Spindle motor limit value om ɗ ȧ Spindle speed data ʐ ȧ 8 ɗ Spindle speed limit ʐ data 8 B–61863E/10 ce nt e r.c (e) Example Suppose that the parameters are set as follows: Minimum speed specified for the spindle motor = 1000 rpm Maximum speed specified for the spindle motor = 35000 rpm Maximum speed obtainable by the spindle motor = 40000 rpm (Maximum speed when 10 V is applied to the spindle motor) Maximum speed for gear 1 = 25000 rpm Maximum speed for gear 2 = 40000 rpm Maximum speed for gear 3 = 6000 rpm Maximum speed for gear 4 = 100000 rpm Spindle speed data addresses = F10 to F11 (RO0 to RO15) The specified spindle speed signal is used.

(For details, refer to the BMI connection manual.) Spindle control parameter addresses = D10 to D33 Control data addresses = R0 to R3 nc (1) Create a functional instruction. MOVW .c R9091.1 F10 R10 2 0 R12 D10 R0 w SUB44 w w R9091.1 NUMEB SUB40 CIRC SPCNT R10 OVRD SUB46 ACT 274 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 (2) Set the spindle speed data Copy the spindle speed data (RO0 to RO15) to spindle speed data addres specified at the first porameter of SPCNT. RO0 to 7 Copy from F10 RO8 to 15 Copy from F11 0 Clear by 0 0 Clear by 0 om R10 (3) Set the spindle control parameters. The lower spindle motor speed limit data and the upper spindle motor speed limit data are obtained as follows (see i) and ii) of b) in 4)): = Upper spindle motor speed limit data = r.c Lower spindle motor speed limit data 1000 40000 8191 = 7167 (rpm) ce nt e 8191 = 204 (rpm) Then, the spindle control parameters are set as follows: 204 Lower spindle motor speed limit data 7167 Upper spindle motor speed limit data D18 to D21 25000 Maximum spindle speed for gear 1 D22 to D25 40000 Maximum spindle speed for gear 2 D26 to D29 60000 Maximum spindle speed for gear 3 D30 to D33 100000 Maximum spindle speed for gear 4 nc D10 toD13 D14 to D17 .c (4) Calculate the spindle motor rotation command for the spindle speed GR1 GR2 GR3 GR4 w w Maximum motor speed (8191) Lower limit at which the motor speed is clamped (7167) w Lower limit at which the motor speed is clamped (204) Maximum Maximum spindle speed spindle speed for GR1 (25000) for GR2 (40000) Maximum spindle speed for GR3 (60000) Spindle speed (rpm) Maximum spindle speed for GR4 (100000) From the above graph, the following table can be obtained: 275 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Table 5.61.5 Maximum and Minimum Spindle Speeds for each Gear Minimum spindle speed (rpm) Maximum spindle speed (rpm) GR1 625 21877 GR2 21878 35004 GR3 35005 52506 GR4 52507 87499 (When CIRC = 0, OVRD = 0) R0 to R1 Spindle motor rotation command r.c 4505 8 (GR4) Spindle gear selection w w w .c nc ce nt e R2 om Thus, if the spindle speed data is 55000 (rpm), when the spindle override is not applied (OVRD = 0) and the direct gear specification is not set (CIRC = 0), the spindle motor rotation command and the spindle gear to be used are obtained as follows. 276 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.62 END (END OF A LADDER PROGRAM) 5.62.1 f : Can be used : Cannot be used PA1 PA3 RA1 RA2 f RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC RC3 RC4 NB NB2 f f f f Function 5.62.2 Format r.c END om The END functional instruction designates the end of a ladder program. END must be placed at the end of the ladder program. w w w .c nc ce nt e SUB 64 277 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.63 CALL (CONDITIONAL SUBPROGRAM CALL) 5.63.1 f : Can be used : Cannot be used PA1 PA3 RA1 RA2 f RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC RC3 RC4 NB NB2 f f f f Function om The CALL functional instruction calls a subprogram. When a subprogram number is specified in CALL, a jump occurs to the subprogram if a condition is satisfied. 5.63.2 Format r.c ACT CALL Subprogram number 5.63.3 Control Conditions (a) Input signal ACT=0 : The

CALL instruction is not executed. ACT=1 : The CALL instruction is executed. (a) Subprogram number Specifies the subprogram number of a subprogram to be called. The subprogram number must be specified in the P address form. A number from P1 to P512 can be specified. nc 5.63.4 ce nt e SUB 65 Example : To call subprogram 1 ACT CALL w .c Parameters P1 w w SUB 65 NOTE Be careful when using the CALL instruction with the COM, COME, JMP, or JMPE functional instruction. For details, see Chapter 9 in Part I. 278 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.64 CALLU (UNCONDITIONAL SUBPROGRAM CALL) 5.64.1 f : Can be used : Cannot be used PA1 PA3 RA1 RA2 f RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC RC3 RC4 NB NB2 f f f f Function om The CALLU functional instruction calls a subprogram. When a subprogram number is specified, a jump occurs to the subprogram. 5.64.2 Format Subprogram number r.c CALLU ce nt e SUB 66 5.64.3 (a) Parameters Subprogram number Specifies the subprogram number of a subprogram to be called. The subprogram number must be specified in the P address form. A number from P1 to P512 can be specified. w w w .c nc Example : To call subprogram 1 279 CALLU SUB 66 P1 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.65 SP (SUBPROGRAM) f : Can be used : Cannot be used PA1 PA3 RA1 RA2 f 5.65.1 RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC RC3 RC4 NB NB2 f f f f The SP functional instruction is used to create a subprogram. A subprogram number is specified as a subprogram name. SP is used with the SPE functional instruction (mentioned later) to specify the subprogram range. om Function 5.65.2 r.c Format SP Subprogram number ce nt e SUB 71 5.65.3 (a) nc Parameters Subprogram number Specifies the subprogram number of a subprogram to be coded following this instruction. The subprogram number must be specified in the P address form. A number from P1 to P512 can be specified. The specified subprogram number must be unique within the sequence program. .c Example: When the subprogram number is set to 1 SP w w w SUB 71 280 P1 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.66 f : Can be used : Cannot be used SPE (END OF A SUBPROGRAM) PA1 5.66.1 PA3 RA1 RA2 RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC RC3 RC4 NB NB2 f f f f The SPE functional instruction is used to create a subprogram. SPE is used with the SP functional instruction. It specifies the range of a subprogram. When this functional instruction has been executed, control is returned to the functional instruction that called the subprogram. om Function 5.66.2 r.c Format SPE w w w .c nc ce nt e SUB 72 281 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.67 f : Can be used : Cannot be used JMPB (LABEL JUMP) PA1 PA3 RA1 RA2 f 5.67.1 RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f Program unit NB NB2 f f f f om Program unit LBL AA ce nt e LBL JMPB AA nc 5.67.2 w 5.67.3 w Control Conditions (ACT) JMPB BB JMPB AA LBL BB ACT .c Format AA AA JMPB w RC4 r.c Ladder program Parameters RC3 The JMPB functional instruction transfers control to a Ladder immediately after the label set in a Ladder program. The jump instruction can transfer control freely before and after the instruction within the program unit (main program or subprogram) in which the instruction is coded. (See the description of the LBL functional instruction, which is be explained later.) As compared with the conventional JMP functional instruction, JMPB has the following additional functions: D More than one jump instruction can be coded for the same label. D Jump instructions can be nested. Function 5.67.4 RC JMPB SUB 68 Specification of the jump destination label ACT=0 : The next instruction after the JMPB instruction is executed. ACT=1 : Control is transferred to the Ladder immediately after the specified label. (a) Label specification Specifies the label of the jump destination. The label number must be specified in the L address form. A value from L1 to L9999 can be specified. NOTE 1 For the specifications of this instruction, see Chapter 10 in Part I. 2 When this instruction is used to jump back to a previous instruction, care must be taken not to cause an infinite loop. 282 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS 5.68 f : Can be used : Cannot be used JMPC (LABEL JUMP) PA1 PA3 RA1 RA2 f 5.68.1 RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC RC3 RC4 NB NB2 f f f f Function om The JMPC functional instruction returns control from a subprogram to the main program. Be sure to code the destination label in the main program. The specifications of this JMPC functional instruction are the same as those of the JMPC functional instruction, except that JMPC always returns control to the main program. D More than one jump instruction can be coded for the same label. Ladder program Main program AA r.c LBL Main program LBL AA LBL Subprogram ce nt e Subprogram 5.68.2 JMPC AA JMPC AA JMPC AA JMPC BB ACT JMPC SUB 73 Specification of the jump destination label w .c nc Format BB 5.68.3 w Control Conditions (ACT) w 5.68.4 Parameters ACT=0 : The instruction after the JMPC instruction is executed. ACT=1 : Control is transferred to the Ladder after the specified label. (a) Label specification Specifies the label of the jump destination. The label number must be specified in the L address form. A number from L1 to L9999 can be specified. NOTE 1 For the specifications of this instruction, see Chapter 10 in Part I. 2 When this instruction is used to jump back to a previous instruction, care must be taken not to cause an infinite loop.

283 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.69 f : Can be used : Cannot be used LBL (LABEL) PA1 PA3 RA1 RA2 f 5.69.1 RA3 RA5 f f RB RB2 RB3 RB4 RB5 RB6 f f f f RC RC3 RC4 NB NB2 f f f f Function om The LBL functional instruction specifies a label in a Ladder program. It specifies the jump destination for the JMPB and JMPC functional instructions. (See the explanation of the JMPB and JMPC functional instructions.) Ladder program LBL AA ce nt e JMPB AA JMPC 5.69.2 AA LBL JMPB BB JMPC AA LBL BB Label specification SUB 69 w .c nc Format AA r.c LBL w 5.69.3 w Parameters (a) Label specification Specifies the jump destination for the JMPB and JMPC functional instructions. The label number must be specified in the L address form. A label number from L1 to L9999 can be specified. A label number can be used more than once as long as it is used in a different program unit (main program, subprogram). NOTE For the use of this instruction, see Chapter 10 of Part I. 284 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.70 AXCTL (AXIS CONTROL BY PMC) f : Available : Unavailable PA1 PA3 RA1 RA2 RA3 RA5 RB RB2 RB3 RB4 RB5 RB6 RC RC3 RC4 f f f f f f f f f f f f f f f NB NB2 om NOTE 1 Option for Axis control by PMC function is required.

2 This functional instruction can not be used on the CNC that does not have option for Axis control by PMC. 5.70.1 This function simplifies the handshake of DI/DO signal for the axis control by PMC. Function Format r.c 5.70.2 RST ffff ffff Group No. of DI/DO signal Axis control data address ce nt e AXCTL ACT (SUB 53) W1 f Fig.5.70.2 AXCTL Instruction Format Table 5.70.2 AXCTL Instruction Coding Step Number Instruction w w Control Condition Bit Number Remarks 1 RD ffff. f RST 2 RD. STK ffff. f ACT 3 SUB 4 (PRM) ffff Number of DI/DO signal 5 (PRM) ffff Axis control data address 6 WRT ffff. f W1, processing completion nc .c w 5.70.3 Address Number 53 ACT=0 : The AXCTL function is not executed. ACT=1 : The AXCTL function is executed. ACT is to be maintained ‘1’ till the end of AXCTL processing. And reset ACT immediately after the processing is complete (W1 = 1). RST=0 : Release reset. RST=1 : Set the reset signal (ECLRx) to 1. All the buffered commands are invalidated and the command being executed is stopped. Set RST at the same time as the reset of CNC when CNC becomes the state of alarm. NOTE When RST and ACT become 1 at the same time, RST is prior to ACT. 285 5.

FUNCTIONAL INSTRUCTIONS 5.70.4 PMC SEQUENCE PROGRAM (a) Parameters B–61863E/10 Group number of DI/DO signal ce nt e r.c om Specify the DI/DO signal group by the number.

1 : group A(G142 to G149, F130 to F132) 2 : group B(G154 to G161, F133 to F135) 3 : group C(G166 to G173, F136 to F138) ; Cannot be used on Power Mate-D/F 4 : group D(G178 to G185, F139 to F141) ; Cannot be used on Power Mate-D/F 5 : group E (G226 to G233, F228 to F230) ; Can be used only on Power Mate-H 6 : group F (G238 to G245, F231 to F233) ; Can be used only on Power Mate-H Add 1000 to the above number as follows if you use HEAD2 of FS16/18–TT or 2nd path of Power Matw–D. 1001 : group A (G1142 to G1149, F1130 to F1132) 1002 : group B (G1154 to G1161, F1133 to F1135) 1003 : group C (G1166 to G1173, F1136 to F1138) ; Cannot be used on Power Mate 1004 : group D (G1178 to G1185, F1139 to F1141) ; Cannot be used on Power Mate When 3–path control is used with the Series 16i/18i, the following addresses are used for DI/DO signals: 2001 : Group A (G2142 to G2149, F2130 to F2132) 2002 : Group B (G2154 to G2161, F2133 to F2135) 2003 : Group C (G2166 to G2173, F2136 to F2138) 2004 : Group D (G2178 to G2185, F2139 to F2141) w w w .c nc (b) Axis control data address Select the addresses of the locations that contain PMC axis control data. +0 FANUC reserved Specify 0. 1 Control command Specify the command to set EC0x-EC6x. 2 Command data 1 Specify the data to set EIF0x-EIF15x. Command data 2 Specify the data to set EID0x-EID31x. 3 4 5 6 (x=A / B / C / D) 7 286 PMC SEQUENCE PROGRAM B–61863E/10 5. FUNCTIONAL INSTRUCTIONS The following functions are available. Operation Rapid traverse Control 00H Command data 1 Feedrate Command data 2 Total travel amount Need not to set if CNC PRM. 8002#0 = 0. 01H Feedrate (Note 1) Total travel amount Cutting feed (Note 2) (feed per revolution) 02H Feedrate per revolution not used Skip (feed per min.) (Note 2) 03H Feedrate Total travel amount Dwell 04H not used Dwell time Reference pos. return 05H not used not used Continuous feed (Note 3) 06H Feedrate Feed direction (Note 4) 1st ref. pos. return 2nd ref. pos. return 3rd ref. pos. return 4th ref. pos. return (Note 2) 07H 08H 09H 0AH r.c om Cutting feed (feed per min.) Feed rate ce nt e Need not to set if CNC not used PRM. 8002#0 = 0. External pulse synchronization (Note 2) (Note 3) 0BH 0DH 0EH 0FH Pulse weighting Speed command (Note 2) (Note 5) (Note 6) (Note 2) 10H Feedrate Machine coordinate positioning. (Rapid traverse) (Note 2) (Note 6) 20H Feedrate not used (Only M series) not used w w w .c nc Position of machine coordinate. Need not to set if CNC (absolute) PRM. 8002#0 = 0. NOTE 1 When you specify 0 for feedrate, CNC does not work. Please release this state by RST = 1. 2 It is not available in PMC-MODEL PA1/PA3. 3 When you end a continuous feed or external pulse synchronization, set RST to 1. And, continuous feed can’t be used with buffering inhibits signal = 1. You must set the signal to 0. 4 Specify the direction by most significant bit of command data 2. 5 Command control axis must be specified to rotary axis by setting parameter ROTx (No. 1006#0) to 0. 6 Not applicable to the Power Mate. 7 For details such as the range of command data, please refer to the connecting manual for each CNC models. 8 About the miscellaneous function, please operate the DI/DO signal with basic instruction of ladder program.

287 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 Example 1) In case of cutting feed (feed per min.) +0 0H Not used (Specify 0). 1 01H Command code for cutting feed. (feed per min.) 2 Feedrate unit : mm/min. 4 Total travel amount om 3 unit : 0.001mm 5 6 r.c (x=A / B / C / D) 7 ce nt e Example 2) In case of machine coordinate positioning. +0 0H Not used (Specify 0). 1 20H Command code for machine coordinate positioning. 2 0 or Feedrate 3 Position in machine coordinate system nc 4 In case of CNC PRM8002#0= 0 =1 not used.

Feedrate. (Absolute) 5 7 NOTE It is necessary to set the CNC parameters relating to the axis movement. w w w .c 6 5.70.5 End of Command (W1) W1=0 : It is 0 usually. W1=1 indicates that AXCTL instruction is completed.

Specify ACT=0 immediately after processing is completed. (W1=1). W1=1 : It will become 1 when the command of the axis control by PMC is buffered on CNC (when EMBUFx=0) or when axis movement is completed (when EMBUFx=1). 288 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.70.6 When error occurs by processing the axis control by PMC, the bit of the operation output register will be set. At the same time, processing is over. Operation Output Register (R9000) 7 6 5 4 3 2 1 0 R9000 om Group number of DI/DO signal specification error. 5.70.7 (1) The following signals cannot be operated from this function. Please operate by LADDER. D Axis control stop signal ESTPx (G142#5, G154#5, G166#5, G178#5) D Servo-off signal ESOFx (G142#4, G154#4, G166#4, G178#4) D Block stop signal ESBKx (G142#3, G154#3, G166#3, G178#3) D Block stop inhibit signal EMSBKx (G143#7, G155#7, G167#7, G179#7) D Controlled axis selection signal EAX1-EAX8 (G136#0 to #7) D Override signal *FV0E-*FV7E (G151#0 to #7) D Override cancel signal OVCE (G150#5) D Rapid traverse override signal ROV2E, ROV1E (G150#1, #0) D Dry run signal DRNE (G150#7) D Manual rapid traverse RTE (G150#7) selection signal D Skip signal SKIP/ESKIP (X4#7, #6) D Buffering inhibit signal EMBUFx (G142#2, G154#2, G166/#2, G178#2) (x=A/B/C/D) w w w .c nc ce nt e Remarks r.c NOTE 1 W1 becomes 1 regardless of the state of ACT. 2 It is not related to the state of the alarm signal (EIALx). WARNING Movement cannot be sured when controlled axis selection signal (EAXx) is changed in the state of ACT=1. (2) Buffering inhibit signal (EMBUFx) 0 : The commands are buffered on the CNC. Even if one command is being executed, the CNC accepts the next command as long as there is vacancy in the buffer on CNC. W1 will become 1 when the command of the axis control by PMC is buffered on CNC. 1 : Prohibits the buffering on CNC. W1 will become 1 when the movement of the instructed axis control by PMC is completed. 289 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.71 f : Can be used : Cannot be used PSGNL (POSITION SIGNAL OUTPUT) 5.71.1 PA1 PA3 RA1 RA2 RA3 RA5 RB RB2 RB3 RB4 RB5 RB6 RC RC3 RC4 NB NB2 f f × × × × × × × × × × × × × × This function outputs a signal that indicates the are in which the current position in the mechanical coordinate system is located. The area is specified by parameter. om Function 5.71.2 PSGNL ffff Area division ffff Current position (SUB 50) specification data address area output address ce nt e ACT r.c Format 5.71.2 PSGNL instruction format 5.71.4 (a) Area division specification data address Set the top address of area division specification data 29 bytes of continuous memory are necessary in nonvolatile memory area for area division specification data.

w w w Parameters .c Control Condition (a) Execution specification ACT=0 : The PSGNL instruction is not executed. ACT=1 : The PSGNLnstruction is executed. nc 5.71.3 290 PMC SEQUENCE PROGRAM B–61863E/10 Area division specification data +0 +1 5. FUNCTIONAL INSTRUCTIONS (1 : 1st Axis, 2 : 2nd Axis,...) Axis No. or Path No. (1byte) (1 : 1st Path–1st Axis 2 :2nd Path–1st Axis) I (4bytes) II (4bytes) +9 +13 r.c III (4bytes) om +5 Area division specification data ce nt e IV (4bytes) +17 V (4bytes) +21 VI (4bytes) nc +25 VII (4bytes) w w w .c D In case of axis–No. specification Please set axis–No. to select.

(1 byte data of binary format) In case of Power Mate–H, the axis No. ranges from 1 to 6.

(Example) Axis No.=1 : For machine coordinates of the 1st axis Axis No.=2 : For machine coordinates of the 2nd axis OR D In case of path specification (Power Mate–MODEL D dual path control) Please set path–No. of axis to select. (1 byte data of binary format) It is impossible to set path specification for Power Mate–H. (Example) Path spec.=1: For machine coordinates of the 1st axis on the 1st path) Path spec.=2: For machine coordinates of the 1st axis on the 2nd path) Each area division specification data (I, II, III, ..., VII) is 4bytes binary format data. (Scale is 0.001mm or 0.001inch) 291 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 I II (1) (2) III (3) IV (4) V VI (5) VII (6) (7) (8)) – Total stroke area om As shown in the above diagram, check can be performed for the 8 areas (1) to (8) by dividing the total stroke area by 7 division points. r.c NOTE 1 Please set the division points data in ascending order (I < II < .....

ce nt e (b) Current position area output address The address which is output the divided area that the currrent position in the machine coordinates system located. Cureent position area output address 7 6 5 4 3 2 1 0 (1) (2) (3) (4) (5) (6) (7) (8) w w w .c nc Corresponding bit is set to 1 indicates the area in which the current position in the machine coordinates system is located. 292 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 5.72 PSGN2 (POSITION SIGNAL OUTPUT 2) 5.72.1 f : Can be used : Cannot be used PA1 PA3 f f RA1 RA2 RA3 RA5 RB RB2 RB3 RB4 RB5 RB6 RC3 RC4 RC NB NB2 Function 5.72.2 Format ffff (SUB 63) Control data address W1 f r.c PSGN2 ce nt e ACT om Turn W1=1 which th ecurrent position in the machine coordinates system is in the area specified by parameters. 5.72.2 PSGN2 instruction format 5.72.3 (a) Execution specification ACT=0 : The PSGN2 instruction is not executed. ACT=1 : The PSGN2 instruction is executed. Control Condition 5.72.4 (a) Control data address Please set the top address of control data. For the area specification data, 9bytes of continuous memory area in the nonvolatile memory is necessary. nc Parameters w .c Control data+0 Axis No. or Path No. (1byte) (1 : 1st Axis, 2 : 2nd Axis,...) (1 : 1st Path–1st Axis 2 :2nd Path–1st Axis) +1 w Boundary Point 1 (4bytes) Area division specification data w +5 Boundary Point 2 (4bytes) D In case of axis–No. specification Please set axis–No. to select. (1 byte data of binary format) In case of Power Mate–H, the axis No. ranges from 1 to 6. (Example) Axis No.=1 : For machine coordinates of the 1st axis Axis No.=2 : For machine coordinates of the 2nd axis OR 293 5. FUNCTIONAL INSTRUCTIONS PMC SEQUENCE PROGRAM B–61863E/10 D In case of path specification (Power Mate–MODEL D dual path control) Please set path–No. of axis to select.

(1 byte data of binary format) It is impossible to set path specification for Power Mate–H. (Example) Path spec.=1: For machine coordinates of the 1st axis on the 1st path) Path spec.=2: For machine coordinates of the 1st axis on the 2nd path) om Each area divicion specification data (Boundary Point / is 4bytes binary format data. (Scale is 0.001mm or 0.001inch) NOTE Please set the data division specification data in ascending order.

(bounary point 1 x bounary point 2) r.c 1 0 ce nt e W1 Boundary Point < 1 > – F Boundary Point < 2 > F Note) Includes + F : Boundary Point .

Total stroke area W1=0 :The current position in the machine coordinates system is outside of the area specified by parameters. W1=1 :The current position n the machine coordinates system is inside of the area specified by parameters. nc 5.72.5 Current Position Area Output (W1) ACT PSGN2 D0320 W1 f (SUB 63) w w .c w Binary Decimal D0320 00000010 (D0321 D0322 D0323 D0324 01100000 01111001 11111110 11111111 (-100000) 2) D0325 D0326 D0327 D0328 01000000 00001101 00000011 00000000 (200000) In this case, when the current position in the machine coordinates system (second axis) is from –100.000 to 200.000mm ACT=1, the current position area output becomes W1=1.. 294 NONVOLATILE MEMORY (1) Used for the timer Time can be set and displayed from the CRT/MDI panel. The set time can be read or written by a sequence program instruction. (2) Used for the counter This area is used to store the preset and cumulative values of the counter. Values can be set and displayed from the CRT/MDI panel. These values can be read and written by a sequence program instruction. Refer to section 3.5 for details of addresses. The data format is two bytes of BCD or binary, and the higher-order digits are entered at the smaller address. Whether counter is processed by BCD format or binary format is selected by a system parameter. ce nt e TIMER, COUNTER, KEEP RELAY, NONVOLATILE MEMORY CONTROL, DATA TABLE Nonvolatile memory is considered nonvolatile if its contents are not erased when the power is turned off. om 6.1 r.c 6 6. NONVOLATILE MEMORY PMC SEQUENCE PROGRAM B–61863E/10 Example) PMC counter addresses are C0 and C1 and the set value is 1578. nc BCD code (1578) w w w .c C0 C1 7 6 5 4 0 1 1 1 3 1 2 1 0 0 0 0 7 8 7 6 5 4 3 2 1 0 0 0 0 1 0 1 0 1 1 5 Binary code (1578) 7 6 5 C0 C1 4 3 2 1 0 0 1 0 1 0 27 26 25 24 23 22 21 20 7 6 5 4 3 2 1 0 0 0 0 0 0 0 1 1 0 0 0 1 215 214 213 212 211 210 29 28 To change low-order digits of the set value by a sequence program instruction with 1 byte processing, specify C0 as the output address of the functional instruction parameters to enter new data. 295 6. NONVOLATILE MEMORY PMC SEQUENCE PROGRAM B–61863E/10 (3) Keep relay This memory is used as parameters, keep relays, etc. for sequence control. Setting and display are possible from the CRT/MDI panel and sequence program instructions can be used for reading and writing.

Since data set or displayed from the CRT/MDI panel is binary eight bits, each of the eight digits of data is set or displayed as 0 or 1. #7 #6 MWRTF2 MWRTF #5 #4 #3 #2 #1 #0 r.c K16 om (4) Nonvolatile memory control (MWRTF, MWRTF2) (Address K16) This memory is used when the position of a moving part of the machine tool, such as a lathe turret, is stored in code (BCD, etc.) and to maintain it while power is off. ce nt e Setting and display are possible from the CRT/MDI panel, and sequence program instructions can be used for reading and writing. If, for example, power is turned off for some reason during rotation of the turret, the turret stops and a difference between the contents of the memory storing the position and the actual position of the turret occurs.

When power is turned on again, the machine tool will be out of sequence. To prevent this, use the nonvolatile memory control, and a sequence program as follows. (a) Set MWRTF in nonvolatile memory control to 1 before starting the turret. (b) Start the turret.

(c) Set MWRTF to 0 after the turret stops. nc (d) MWRTF remains 1 if power is turned off between a) and c). w w w .c (e) When power is turned on again, automatically MWRTF2 = 1 and an error is reported to the sequence program. Thus, the sequence program processes (a) to (d), check for the error of MWRTF2, and outputs an alarm when MWRTF2 = 1 (error). (f) In response, the operator should set MWRTF and MWRTF2 to 0 from the CRT/MDI panel. (g) Resume operation after the contents of the memory and the turret position are aligned. (5) Data table A sizable amount of numeric data (data table) can be used for sequence control by the PMC. See section 6.3 for details. 296 PMC SEQUENCE PROGRAM B–61863E/10 6.2 6. NONVOLATILE MEMORY All the nonvolatile memory data can be read and written by the sequence program. The memory read and written by the PMC sequence program is actually not a nonvolatile memory, but a nonvolatile memory image (RAM) storing the same data as the nonvolatile memory. When the power supply is turned off, the data in the nonvolatile memory image disappears. Immediately after the power is turned on, the nonvolatile memory data is automatically transferred to the nonvolatile memory image. Before the power is turned off, the data is correctly restored. READING AND WRITING OF NONVOLATILE MEMORY DATA om When the nonvolatile memory image is rewritten by the sequence program, the data is automatically transferred to the CMOS or bubble memory. When the sequence program rewrite nonvolatile memory image of area, the rewritten data is automatically transferred to the nonvolatile memory. r.c Rewriting of nonvolatile memory can also be done by rewriting optional addresses of the nonvolatile memory image in an optional timing. The changed data will be automatically transferred to the nonvolatile memory. w w w .c nc ce nt e Therefore, there is not special processing necessary when the sequence program writes or reads nonvolatile memory. It will only take some time to write in the nonvolatile memory (512 ms). 297 6. NONVOLATILE MEMORY 6.3 PMC SEQUENCE PROGRAM (1) B–61863E/10 Introduction PMC sequence control sometimes requires a sizable amount of numeric data (herein after referred to as data table.

If contents of such data table are free to set or to read, they can be used as various PMC sequence control data, such as tool numbers of tools on the ATC magazine. Each table size can be set optionally in the memory for data table, and 1-, 2-, or 4-byte binary or BCD format data can be used per each table, thus consigning a simple-to-use table. Data in the data table can be set in the nonvolatile memory or displayed via the CRT/MDI panel. Data set in the data table can also be easily read or written by the sequence program using function instructions as data search (DSCHB), or index modification data transfer (XMOVB).

r.c om PMC DATA TABLE ce nt e NOTE For details of the usable range, see the description of PMC sequence program addresses in Part I. (2) Configuration of the PMC data table and notes on programming (b) Data table head address If the data table starts from an odd address, for example, when a data table is created with an odd number of one-byte data, the DSCHB instruction operates slower than when the data table starts from an even address. It is recommended that the starting address of a data table be an even number. w w w .c nc (a) Configuration of the data table PMC data table consists of table control data and data table. Table control data control the size and data format (BCD or binary) of the tables. This table control data must first be set from CRT/MDI before preparing data table. In the sequence program, the table control data cannot be read or written. When the contents of the nonvolatile memory are read or written using the Floppy Cassette, the table control data is read or written together. Figure 6.3 (a) is a general configuration of the data table, and Figure 6.3 (b) is a detailed configuration of the data table. Also refer to 3.7 for data table configuration. 298 PMC SEQUENCE PROGRAM B–61863E/10 6. NONVOLATILE MEMORY Table control data Data table number D0 D1 ce nt e r.c om Basic data table (1860 bytes or 3000 bytes) D1859 or D2999 6.3 (a) General configuration of data table w w w .c nc Data table 299 6. NONVOLATILE MEMORY PMC SEQUENCE PROGRAM B–61863E/10 Number of tables Table parameter Data type Table 1 control data Number of data om Data table head address Table parameter Data type Table 2 control data Number of data r.c Table control data ce nt e Data table head address Table parameter Data type Table n control data Number of data Data table head address Table number 0 Data 1 Data table 1 : n1 nc Address No. D0 D1 D2 Table number Data Data table 2 .c 0 1 2 3 : : n2 w Data table w w Table number Table number 0 1 : np Data Data table n (Note) N1, n2, and np are the last table number of each data table. 6.3 (b) Detailed configuration of data table 300 6. NONVOLATILE MEMORY PMC SEQUENCE PROGRAM B–61863E/10 (3) Table control data The table control data controls a data table If the table control data is not properly set, a data table described in Item (4) cannot be properly created. Referring to the description in Item (3), set the table control data, then create a data table. (a) Number of groups of tables Specify the number of groups of data tables in binary. om (b) Control data for table groups 1 to n Each data table has table control data consisting of the starting address of the table, table parameters, data type, and the number of data items. (ii) Table parameter #6 #5 #4 #3 #2 ce nt e #7 r.c (i) Starting address of the table Specify the starting address of the table from D0 to D1859 or D0 to D2999. d̦ ʐ 8 d̦ MASK ʐ 8 COD #1 #0 MASK COD 0 : A data table is specified in binary.

1 : A data table is specified in BCD. 0 : The contents of the data table are not protected.

1 : The contents of the data table are protected. w w w .c nc (iii) Data type Specify the length of data in the data table. d̦ 0 : One byte ʐ 1 : Two bytes 8 2 : Four bytes (iv) Number of data items Specify the number of data items used in the data table. (4) Data table Data table can be created within the range of the memory (D address) for the data table and separated some groups. This number of groups is decided with the number of tables of table control data. The maximum of the number of table groups. Except series 15b PMC–NB max 100 tables PMC–NB max 50 tables 301 6. NONVOLATILE MEMORY PMC SEQUENCE PROGRAM B–61863E/10 Table number 0 1 2 Table 1 (1-byte data) 3 om n r.c 1 Table 2 (2-byte data) ce nt e p Each data table can be used in 1, 2 or 4 byte data. Table parameter of table control data decides whether to use 1 or 2 byte data. Therefore, 1 table number is taken for a 1-byte data when table data is 1 byte; 2 byte data when table data is 2 bytes. nc (5) Entering data in a data table Specify a location number in the data table from the CRT/MDI panel, then enter the data. A number for each location in the table is defined for each data table group. w w w .c NOTE Reading and Writing of the data table are available from the sequence program. 302 PMC SEQUENCE PROGRAM B–61863E/10 7 7. LADDER DIAGRAM FORMAT LADDER DIAGRAM FORMAT w w w .c nc ce nt e r.c om A designer examines and checks the ladder diagram in the process of design. However, it should be noted that other persons (maintenance servicemen, for example) read the ladder diagram far longer than the designer. Accordingly, the ladder diagram must be written to be easily understood by all persons. For this purpose, applicable symbols, writing method, and other methods are specified as detailed below. 303 7. LADDER DIAGRAM FORMAT 7.1 ADDRESSES, SIGNAL NAMES, COMMENTS, AND LINE NUMBERS 7.1.1 PMC SEQUENCE PROGRAM B–61863E/10 Addresses, signal names, comments, and line numbers must be inserted into a ladder diagram to enable all users to easily read the ladder diagram. Addresses Bit number (0 to 7) om Each address consists of an address number and a bit number, and it is represented as follows. r.c Address number (A numeric of 4 digits or less after an alphabetic character) ce nt e An alphabetic character is prefixed to the start of each address number to represent the kinds of signals as shown in Table 7.1.1. Table 7.1.1 Alphabetic symbols of address numbers Symbol Type of signal X Input signal entered from machine tool to PMC (MT³PMC) Y Output signal sent from PMC to machine tool (PMC³MT) F Input signal entered from CNC to PMC (CNC³PMC) G Output signal sent from PMC to CNC (PMC³CNC) R Internal relay Message display request Counter K Keep relay D Data table T Variable Timer L Label number P subprogram number w w w .c nc A C 304 PMC SEQUENCE PROGRAM B–61863E/10 7.1.2 7. LADDER DIAGRAM FORMAT Suitable symbols shall be attached to I/O signals as signal names according to the following procedure. Signal Names (1) The names of all signals containing CNC signals and machine tool signals are represented within 6 characters. Alphanumeric characters and special symbols described in this manual are all employable. om (2) For CNCPMC signal names, signal names shown in the PMC address table are employable as they are.

ce nt e r.c (3) CNC signals to be entered from the machine tool and CNC signals to be sent to the machine tool are identified from each other by prefixing X or Y to the start of these CNCPMC signal names, respectively. A single block input signal is represented as XSRK by prefixing X, while a start lamp output signal is represented as YSTL by prefixing Y, for example. However, when X or Y is prefixed to the start of an CNCPMC signal name, certain signal names exceed 6 characters. In such a case, omit the last character from such a signal name (*SECLPX*SECL) 7.1.3 7.1.4 A line number should be attached to each line of the ladder diagram. For details, refer to Sec. 7.3. w w w .c Line Numbers nc Comments A comment of within 30 characters can be inserted to a relay coil in a sequence program and each symbol in a symbol table. Since it is difficult to guess the meanings of signals perculier to the machine tool, a detailed comment is necessary.

305 7. LADDER DIAGRAM FORMAT PMC SEQUENCE PROGRAM B–61863E/10 7.2 Symbol SYMBOLS USED IN THE LADDER DIAGRAM Description These are the contacts of relays in the PMC, and A contact are used for other input from the machine side and CNC B contact These are input signals from the CNC. om A contact B contact These are input signals from the machine side A contact (including the built-in manual control panel). r.c B contact These are timer contacts in the PMC A contact ce nt e B contact This is a relay coil whose contact is used only in the PMC. This is a relay coil whose contact is output to .c nc CNC. This is a relay coil whose contact is output to the machine side. w w This is the coil of a timer in the PMC. This is a PMC fucnctional instruction. The actual w form varies depending on the instruction. NOTE If the coil is represented by PMC, and the contact uses 306 or , the relay is within the or . 7. LADDER DIAGRAM FORMAT PMC SEQUENCE PROGRAM B–61863E/10 7.3 (1) LADDER DIAGRAM FORMAT Format The size should be A3 or A4 (JIS standard). (2) Columns are used for wiring. 3A 3B om Line number 3P r.c Spindle control (3) Divide the circuits into several functions.

And program the same function in a single program.

ce nt e Example) Mode control.spindle control, turret control, APC control. (4) Assign a line number to each line as follows: Line symbol (A to Z) Page number (1 to 999) w w w .c nc (5) Write a relay contact with a signal name of the relay coil, line number and address. Signal name Address Line number (6) For complicated timing, timing chart should be on the same page of the ladder diagram. (7) The meaning of the code numbers for the S, T, and M functions should be listed on the ladder diagram.

(8) The 1st level sequence part should be written at the beginning of the ladder diagram. (9) The following data should be written on the first page of the ladder diagram: (i) The sequence program design number Machine tool builder shall assign design numbers of sequence program and ROMs and manage them. (ii) Description of symbol (iii) Setting table of timer, counter, and PMC parameters and meaning of them. (iv) Description of functional instruction. (10) Easy-to-understand name should be assigned. 307 7. LADDER DIAGRAM FORMAT 7.4 PMC SEQUENCE PROGRAM B–61863E/10 A general relay sequence circuit has a finite number of contacts, so several relays use one contact in common so as to reduce the number of contacts used as much as possible. INFINITE NUMBER OF RELAY CONTACTS A B om R1 r.c R2 A R1 B w w w .c nc A ce nt e The PMC is considered to have an infinite number of relay contact and is written as in the figure below. 308 R2 PMC SEQUENCE PROGRAM B–61863E/10 8 8. MISCELLANEOUS ITEM MISCELLANEOUS ITEM w w w .c nc ce nt e r.c om To create a ladder program related to the axis-control function by the PMC, refer to the subsection, ”Axis-control function by the PMC,” in the Connecting Manual. 309 9. SEQUENCE PROGRAM STRUCTURING 9 PMC SEQUENCE PROGRAM B–61863E/10 SEQUENCE PROGRAM STRUCTURING PA1 PA3 RA1 RA2 f RA3 RA5 f f RB RB2 RB3 RB4 f f om f : Can be used : Cannot be used RB5 RB6 f f RC RC3 RC4 NB NB2 f f f f r.c With the conventional PMC, a Ladder program is described sequentially. By employing a Ladder language that allows structured programming, the following benefits are derived: D A program can be understood and developed easily. D A program error can be found easily. w w w .c nc ce nt e D When an operation error occurs, the cause can be found easily. 310 9. SEQUENCE PROGRAM STRUCTURING PMC SEQUENCE PROGRAM B–61863E/10 9.1 EXAMPLES OF STRUCTURED PROGRAMMING Three major structured programming capabilities are supported. Implementation Techniques (1) Subprogramming A subprogram can consist of a Ladder sequence as the processing units.

om 9.1.1 · · · · r.c Job A FUNC ce nt e Job B · · · · f f · · · (2) Nesting Ladder subprograms created in 1 above are combined to structure a Ladder sequence. Main Program Sub Program2 Job A1 Job A11 Job B D D D Job A12 nc Job A .c w w w Main Program Sub Program1 Job An (3) Conditional branch The main program loops and check whether conditions are satisfied. If a condition is satisfied, the corresponding subprogram is executed. If the condition is not satisfied, the subprogram is skipped. Sub Program PROCESS11 STATE1 PROCESS1 STATE2 PROCESS2 PROCESS12 · · · PROCESS13 311 PROCESS11 9. SEQUENCE PROGRAM STRUCTURING 9.1.2 PMC SEQUENCE PROGRAM B–61863E/10 (1) Example Suppose that there are four major jobs. If Y0 is 1, a request to machine a workpiece is assumed, and processing is performed (with a condition). A : 1 Pick up a workpiece from a pallet. (A1) 2 Machine the workpiece. (A2) 3 Return the workpiece to the pallet. (A3) B : 4 Move the pallet. A B om Applications (2) Program structure Sub Program P2 A1 B A2 Sub Program P4 A1 A2 Sub Program P5 Sub Program P6 A3 B ce nt e A Sub Program P3 r.c Sub Program P1 A3 (3) Program description nc Main Program END1 Y0 MAIN 8 À Ʒ Machine a workpiece. END2 w w w .c CALL Sub Program P1 (=MAIN) SP MAIN CALL A CALL B SPE 312 8 À Machine a workpiece. Ʒ 8 À Move a pallet. Ʒ 9. SEQUENCE PROGRAM STRUCTURING PMC SEQUENCE PROGRAM B–61863E/10 Sub Program P2 (=A) A CALLU A1 CALLU A2 CALLU A3 8 À Pick up a workpiece from a pallet. Ʒ 8 À Machine the workpiece. Ʒ 8 À Return the workpiece to the pallet. Ʒ om SP Sub Program P3 (=A1) A1 ce nt e SP r.c SPE Pick up a workpiece from a pallet. 8 À Ʒ Ladder coding 8 À Ʒ Ladder coding 8 À Ʒ Ladder coding SPE nc Sub Program P4 (=A2) w w w .c SP A2 Machine the workpiece. SPE Sub Program P5 (=A3) SP A3 Return the workpiece to the pallet. SPE 313 9. SEQUENCE PROGRAM STRUCTURING PMC SEQUENCE PROGRAM B–61863E/10 Sub Program P6 (=B) SP B Move the pallet. 8 À Ladder coding Ʒ 9.1.3 (1) Main program The main program is the Ladder program consisting of the first- and second-level Ladder programs. One, but only one, main program can be created. A subprogram cannot be called from the first-level Ladder program. Any number of subprograms however, can be called from the second-level Ladder program. The functional instructions JMP and COM must be completed within each main program or subprogram. ce nt e r.c Specifications om SPE (2) Subprogram A subprogram is a program called by the second-level Ladder program. It is a program unit starting with the functional instruction SP and ending with the functional instruction SPE.

Up to 512 subprograms can be created for one PMC. w w w .c nc (3) Nesting A subprogram can call another subprogram. The maximum nesting depth is eight levels. Recursive calls are not allowed.

314 9. SEQUENCE PROGRAM STRUCTURING PMC SEQUENCE PROGRAM B–61863E/10 9.2 SUBPROGRAMMING AND NESTING 9.2.1 Conditional JUMP (or unconditional JUMP) is coded in the main program, and the name of a subprogram to be executed is specified. In the subprogram, the name of the subprogram and a Ladder sequence to be executed are coded. om Function When a subprogram is named Pn (program name), and this name is specified in conditional JUMP, the subprogram is executed by calling it. A symbol and comment can be added to Pn to assign a subroutine name. r.c In the example shown in Fig. 9.2.1, the main program calls three subprograms. These calls are all conditional calls.

Subprogram P1 is named SUBPRO. It calls subprogram PROCS1 unconditionally. Sub Program P1 (=SUBPRO) SP END1 CALL ce nt e Main Program f SUBPRO CALLU P2 CALL P3 nc CALL ADD B END2 0001 0001 F10 R200 .c w PROCS1 SPE 9.2.1 Example of Subprogramming and Nesting w w SUBPRO 315 f f 9. SEQUENCE PROGRAM STRUCTURING PMC SEQUENCE PROGRAM 9.2.2 B–61863E/10 The main program is always active. Subprograms on the other hand, are active only when called by another program. In the following example, subprogram SUBPRO is called by signal A. Execution Method Main Program Sub Program · · · · f SP SUBPRO END1 CALL SUBPRO · · · · · · · · · · · · · · · · · · f END2 r.c SPE Main program a Subprogram b c d a nc Management program ce nt e Program cycle Signal A f om · · · · A .c Flow of execution b : When the end of the subprogram is reached, control is returned to the main program. c : When the end of the main program is reached, the management program performs Ladder program postprocessing. w w w a : Functional instruction CALL calls a subprogram in order to transfer control to the subprogram. 316 9. SEQUENCE PROGRAM STRUCTURING PMC SEQUENCE PROGRAM B–61863E/10 9.2.3 Creating a Program Create subprograms in the same way as the first-, second-, and third-level Ladder programs. Example of creation · · · · f END1 CALL SUBPRO f r.c · · · · om A ce nt e END2 SP SUBPRO · · · · f · · · · · · · · · · · · · · SP P20 · · · · · · · · · · · · · · .c nc SPE SPE w w w END 317 Must create 9. SEQUENCE PROGRAM STRUCTURING PMC SEQUENCE PROGRAM B–61863E/10 Inhibit items (1) Subprograms are nested. SP SUBPRO · · · · f om · · · · · · · · · · · · · · SPE r.c · · · · · · · · · · · · · · SPE ce nt e (2) A subprogram is created within the first-, second-, or third-level Ladder program. · · · · f END1 · · · · · · · · · · · · · · w w w .c nc SP SUBPRO 318 · · · · · · · · · · · · · · END2 PMC SEQUENCE PROGRAM B–61863E/10 9.3 a) DISPB CAUTIONS FOR SUBROUTINES b) EXIN c) 9. SEQUENCE PROGRAM STRUCTURING WINDR (only low-speed response) d) WINDW (only low-speed response) e) MMCWR f) MMCWW om g) MMC3R h) MMC3W r.c When you use the above-mentioned functional instructions, ACT=1 must be held until the transfer completion information(W1) becomes 1. Therefore, be careful of the following when using those instructions in subprograms. ce nt e D Do not stop calling the subprogram at the state which has not been completed yet, that is executed still while using the instructions in the subprogram. (In other words, do not set the ACT of the CALL instruction to 0) ³ If you do it the function of the instructions after that is not guaranteed. D Call the subprogram from other subprograms at the state which has not been completed yet while using the instructions in the subprogram.

³ The movement of the above-mentioned functional instruction after that is not guaranteed so that the last functional instruction may be processing the instruction. w w w .c nc Then, when the subprogram, in which the above-mentioned functional instruction is used, is called from two or more places, it is necessary to control the subprogram exclusively. The case of the WINDR instruction (low-speed response) is given as an example here. 319 9. SEQUENCE PROGRAM STRUCTURING PMC SEQUENCE PROGRAM B–61863E/10 Example) When subprogram is called from two places. (The WINDR instruction is used) Main program Subprogram 1 Subprogram 2 C2 L4 SP C1 S-PRO1 SP B JMPB L1 ON CALL WINDR A S-PRO1 C2 CALL S-PRO2 JMPB L2 B C1 ce nt e L3 LBL LBL C2 DATA2 SET S-PRO1 LBL C2 S-PRO2 L2 nc A A CALLU C2 CALL L1 ON L4 r.c C1 JMPB DATA SPE A A B A DATA1 SET C1 S-PRO2 om JMPB SPE L3 w w w LBL .c C1 Description) Subprogram 1 controls ACT(A)and W1(B)of WINDR (subprogram 2). By ”A” controlled in subprogram 1, the main program decides which relay (C1,C2) to be effective. When the WINDR instruction is completed, the following data will be set and the other CALL instruction is started. It keeps working in this way. 320 10.

JMP INSTRUCTIONS WITH LABEL SPECIFICATION PMC SEQUENCE PROGRAM B–61863E/10 10 JMP INSTRUCTIONS WITH LABEL SPECIFICATION PA1 PA3 RA1 RA2 f (1) SPECIFICATIONS RA5 f f RB RB2 RB3 RB4 f f RB5 RB6 f f RC RC3 RC4 NB NB2 f f f f Relationship between JMPB/JMPC and LBL (Forward and backward jumps to the same label are possible.) JMPB A r.c 10.1 RA3 om f : Can be used : Cannot be used f : Possible ce nt e f LBL A f f JMPC A CAUTION The specifications allow backward jumps. A backward jump, however, may result in an infinite loop or cause the execution time of the first-level Ladder program to exceed 1.5 ms (or 5 ms). Create a program carefully so an infinite loop does not occur. w w w .c nc JMPB A 321 10. JMP INSTRUCTIONS WITH LABEL SPECIFICATION I. PMC SEQUENCE PROGRAM B–61863E/10 (2) Same label (A label can be used more than once as long as it is unique within the main program or each subprogram.) LBL A LBL B END1 LBL C r.c Second level om First level ce nt e END2 SP A SP A LBL C SP B SP B LBL B w w w .c nc SPE SPE NOTE As mentioned in (8) of Section 10.2, the same label must not exist in the first- and second-level Ladder programs. (3) Number of labels First-and second-level Ladder programs : Up to 256 labels Subprogram : Up to 256 labels for each subprogram Label number : L1 to L9999 322 10. JMP INSTRUCTIONS WITH LABEL SPECIFICATION PMC SEQUENCE PROGRAM B–61863E/10 (4) Relationship between JMP/JMPE and JMPB/JMPC (JMPB and JMPC can be used with JMP and JMPE freely.) JMPB A JMPE f : Possible f JMPB D om LBL A f JMPB B ce nt e LBL B f r.c JMP JMPB C f LBL C JMPE nc LBL D JMPB C f : Possible f JMPB A f CALL w w .c (5) Relationship between CALL/CALLU and JMPB/JMPC (JMPB and JMPC can be used with CALL and CALLU freely.) w LBL A JMPB B f CALLU LBL B LBL C 323 10.

JMP INSTRUCTIONS WITH LABEL SPECIFICATION I. PMC SEQUENCE PROGRAM B–61863E/10 (6) Position of JMPC (JMPC coded between COM and COME can cause a jump.) f : Possible LBL A f Second level SP B COM r.c SP A om END2 JMPC A ce nt e COME w w w .c nc SPE 324 PMC SEQUENCE PROGRAM B–61863E/10 10.2 (1) RESTRICTIONS 10. JMP INSTRUCTIONS WITH LABEL SPECIFICATION Jump destination of JMPB (1) (A jump over END1 or END2 is inhibited.) f : Possible : Impossible LBL A JMPB B First level f LBL B END1 Second level r.c LBL C om JMPB C JMPB A ce nt e END2 (2) Jump destination of JMPB (2) (A jump must be performed within a subprogram.) JMPB A f nc LBL A JMPB B SPE SP B SP B LBL B w w .c SP A w SPE 325 f : Possible : Impossible 10. JMP INSTRUCTIONS WITH LABEL SPECIFICATION I. PMC SEQUENCE PROGRAM B–61863E/10 (3) Jump destination of JMPB (3) (A jump over COM or COME is inhibited.) f : Possible : Impossible JMPB A COME LBL A om JMPB D f JMPB B ce nt e LBL B r.c COM JMPB C f LBL C COME w w w .c nc LBL D 326 PMC SEQUENCE PROGRAM B–61863E/10 10. JMP INSTRUCTIONS WITH LABEL SPECIFICATION (4) Jump destination of JMPC (1) (A jump to the first-level Ladder program is inhibited.) f : Possible : Impossible LBL A END1 Second level LBL B om First level r.c f END2 ce nt e SP A SP A JMPC B JMPC A w w w .c nc SPE 327 10. JMP INSTRUCTIONS WITH LABEL SPECIFICATION I. PMC SEQUENCE PROGRAM B–61863E/10 (5) Jump destination of JMPC (2) (A jump to a label between COM and COME is inhibited.) : Impossible COM COME r.c END2 om LBL A Second level SP A ce nt e SP A JMPC A SPE nc (6) Jump destination of JMPC (3) (Control must not be returned to a label that appears earlier than the instruction that has called the subprogram.) : Impossible LBL A CALL A END2 SP A SP A w w w .c Second level JMPC A SPE CAUTION Although Ladder diagrams can be edited, editing a Ladder diagram may cause an infinite loop. So, be careful not to program such processing. 328 10. JMP INSTRUCTIONS WITH LABEL SPECIFICATION PMC SEQUENCE PROGRAM B–61863E/10 (7) LBL for JMPB (1) (There is no LBL in the same subprogram.) LBL A END2 SP A SP A om Second level r.c JMPB A ce nt e SPE (8) LBL for JMPB (2) (The same LBL is found in the first- and second-level Ladder programs.) nc JMPB A LBL A END1 Second level LBL A w w .c First level w END 329 10. JMP INSTRUCTIONS WITH LABEL SPECIFICATION I. PMC SEQUENCE PROGRAM B–61863E/10 (9) LBL for JMPC (There is no LBL in the second-level Ladder program.) LBL A END2 SP A r.c SP A om Second level JMPC B w w w .c nc ce nt e SPE 330 om w w w .c nc ce nt e r.c II. PMC OPERATION (CRT/MDI) 1 1. GENERAL PMC OPERATION (CRT/MDI) B–61863E/10 GENERAL om The following PMC data can be set and displayed by using the CRT/MDI panel.

nc ce nt e r.c 1) PMC I/O signal display and internal relay display (PMCDGN) PMCDGN has following screens. a) Title data display b) Status screen c) Alarm screen d) Trace function e) Memory display f) Signal Wareform display function g) User task execution status display function 2) PMC data setting and display (PMCPRM) The following PMC data are provided. a) Timer b) Counter c) Keep relay d) Data table 3) Display of sequence program ladder diagram (PMCLAD) 4) PMC screen (PMCMDI) for the user Press the function key on the CRT/MDI panel first.

w w w .c NOTE This function key is effective when a user program exists in the PMC-RC. Switch the NC and PMC menus as described below. NC screen to PMC screen Press the SYSTEM function key on the CRT/MDI panel. Selecting the PMC soft key displays the PMC basic menu. PMC screen to NC screen D Pressing the RETURN key (the leftmost key) on the PMC basic menu screen changes the menu to the NC soft key menu. D Selecting a function key on the PMC screen changes the screen to the corresponding NC screen. Figs. 1 l) to 1 a) show the standard CRT/MDI panels. NOTE A key in < > is a function key on the CRT/MDI panel. A key in [ ] is a soft key described below. 333 1.

GENERAL PMC OPERATION (CRT/MDI) B–61863E/10 a) 9ÀÀ small monochrome/color CRT/MDI panel for 16-TA/18-TA (Horizontal type) Function keys Address/numeric keys om SHIFT key Cancel key INPUT key Edit keys HELP key r.c RESET key Page keys Cursor control keys ce nt e Soft keys b) 9ÀÀ monochrome/color CRT/MDI panel for 16-TA/18-TA (Horizontal type) Address/numeric keys nc RESET key HELP key Edit keys .c Cancel key w w INPUT key Soft keys SHIFT key Function keys Page keys Cursor control keys w Power on/off buttom 334 1.

GENERAL PMC OPERATION (CRT/MDI) B–61863E/10 c) 10ÀÀ color LCD/MDI panel for 16-TA/18-TA (Horizontal type) om Function keys Address/numeric keys Cancel key SHIFT key INPUT key r.c HELP key RESET key ce nt e Page keys Edit keys Cursor control keys Soft keys Power on/off buttom d) 10ÀÀ color LCD/MDI panel for 16-TA/18-TA (Vertical type) w .c nc Power on/off buttom w Soft keys RESET key Function keys Edit keys w HELP key Cursor control keys SHIFT key Page keys INPUT key Cancel key Address/numeric keys 335 1. GENERAL PMC OPERATION (CRT/MDI) B–61863E/10 e) 14ÀÀ color CRT/MDI panel for 16-TA/18-TA (Horizontal type) RESET key HELP key om Address/numeric keys SHIFT key r.c Edit keys Power on/off buttom ce nt e INPUT key Cancel key Function keys Cursor control keys Page keys Soft keys w w w .c nc f) 14ÀÀ color CRT/MDI panel for 16-TA/18-TA (Vertical type) Soft keys RESET key Function keys HELP key Edit keys Cursor control keys Power on/off buttom Page keys INPUT key Cancel key Address/numeric keys SHIFT key 336 1. GENERAL PMC OPERATION (CRT/MDI) B–61863E/10 g) 9ÀÀ small monochrome/color CRT/MDI panel for 16-MA/18-MA (Horizontal type) Function keys Address/numeric keys om SHIFT key Cancel keys INPUT key Edit keys r.c HELP key RESET key Page keys Soft keys ce nt e Cursor control keys h) 9ÀÀ monochrome/color CRT/MDI panel for 16-MA/18-MA (Horizontal type) nc RESET key Address/numeric keys HELP key Edit keys .c Cancel key w w INPUT key Soft keys SHIFT key Cursor control keys Function keys Page keys w Power on/off buttom 337 1. GENERAL PMC OPERATION (CRT/MDI) i) B–61863E/10 10ÀÀ color LDC/MDI panel for 16-MA/18-MA (Horizontal type) Address/numeric keys om Function keys Cancel key SHIFT key INPUT key r.c HELP key RESET key ce nt e Edit keys Page keys Cursor control keys Soft keys Power on/off buttom j) 10ÀÀ color LCD/MDI panel for 16-MA/18-MA (Vertical type) w w .c nc Power on/off buttom w Soft keys RESET key Function keys Edit keys HELP key Cursor control keys SHIFT key Address/numeric keys Page keys INPUT key Cancel keys 338 1.

GENERAL PMC OPERATION (CRT/MDI) B–61863E/10 k) 14ÀÀ color CRT/MDI panel for 16-MA/18-MA (Horizontal type) RESET key HELP key om Address/numeric keys SHIFT key r.c Edit keys Power on/off buttom ce nt e INPUT key Cancel key Function keys Page keys Soft keys 14ÀÀ color CRT/MDI panel for 16-MA/18-MA (Vertical type) w w w .c nc l) Cursor control keys Soft keys RESET key Function keys HELP key Edit keys Cursor control keys Power on/off buttom Address/numeric keys Page keys INPUT key Cancel key SHIFT key 339 1.

GENERAL PMC OPERATION (CRT/MDI) FOR MDI UNITS OTHER THAN STANDARD MDI UNITS (FOR FS20 PMC-RA1 AND RA3) Note the followings when you input PMC-address on the original MDI boards made by MTBs without using Standard MDI Unit supplied by FANUC. (1) If the MDI has the keys to input PMC-address (X, Y, F, G, R, A, C, K, D, T), You can operate as same as FANUC Seires 18 (PMC-RA1/RA3). (2) If MDI does not have those keys, input PMC-address as follows. When inputting PMC-address (in PCLAD, STATUS and so on), you can substitute number keys (0 to 9) and a hyphen key (–) for PMC-address capital keys (X, Y, F, G, etc.). PMC-address capital keys are corresponding to the number keys as follows. PMC-address keys G F Y number keys 0– 1– 2– om 1.1 B–61863E/10 X A R T K C D 3– 4– 5– 6– 7– 8– 9– AUTOMATIC OPERATION WHEN THE POWER IS TURNED ON CLEARING THE SEQUENCE PROGRAM When the power for the CNC is turned on for the first time, a RAM PARITY or NMI alarm may occur in the PMC.

This is caused by invalid data in the sequence program storage area in the PMC. The sequence program must be cleared to prevent this. nc 1.3 When a valid sequence program is contained in the PMC, automatic operation can be started immediately after power–on by keep relay setting. This eliminates the need to display the PMC screen and run a sequence program each time the power is turned on. The keep relay setting method depends on the PMC model. See Section 4.3.3. ce nt e 1.2 r.c (Example) If you want to input “X0.0 [SRCH] ”, input “3-0.0 [SRCH] ”. The automatic operation (see 1.2 above) can also be stopped by clearing the sequence program in the PMC. 1. Turn on the power while pressing X and O. 2. Turn on the power, display the PMC screen, and use the programmer function of the PMC (EDIT/CLEAR). NOTE In case of loader control function, turn on the power while pressing X and 5. w w w .c The sequence program can be cleared in either of the following two ways: 340 1. GENERAL PMC OPERATION (CRT/MDI) B–61863E/10 1.4 LOADING THE STANDARD LADDER (FOR Power Mate –D/F PMC–PA1 AND PA3) The PMC-PA1 and PA3 contained in the Power Mate have a sequence program called the “standard ladder” in their ROM to operate the Power Mate without creating a sequence program. Operation) Parameter in the Power Mate #7 #6 #5 #4 #3 #2 #1 8703 #0 FLA 0 : The FANUC standard ladder is not used. 1 : The FANUC standard ladder is used. om #0 (FLA) = (1) Set bit 0 (FLA) of NC parameter 8703 to 1. This generates alarm 000 (power-off request) in the Power Mate. (2) Turn off the power, then turn it on again. r.c If the PMC contains a sequence program (PMC alarm ER22 PROGRAM NOTHING does not occur), turn on the power while clearing the sequence program (pressing X and O). ce nt e (3) The FANUC standard ladder is loaded. NOTE If the sequence program is not cleared in the PMC, the FANUC standard ladder is not loaded. The existing sequence program remains. 1.5 nc LADDER PASSWORD FUNCTION A password can be specified for a ladder program. Specified passwords are stored as sequence program data. A ladder program for which the password has been specified cannot be displayed or edited. Symbols, comments and messages, however, can be displayed and edited whether a password is specified or not. w w w .c (1) Applicable model PMC–RA1/RA5/RB5/RB6 for Series 16i/18i/21i–A PMC–RA1/RB3/RB4/RC3/RC4 for Series 16/18–MODEL B PMC–RB5/RB6 for Series 16/18–MODEL C PMC–RA1/RA3 for Series 21/210–MODEL B PMC–NB/NB2 for Series 15–MODEL B PMC–PA3 for Power Mate–H (2) Types of passwords A password consists of up to eight alphanumeric characters. The following two types of passwords are used. Display permissible : R password (READ) Display and editing permissible : RW password (READ+WRITE) 341 1. GENERAL PMC OPERATION (CRT/MDI) Table 1.5 (a) B–61863E/10 Screens Requiring Password Release and Corresponding Password Types Password Selected screen (soft key) READ ONLEDT READ+WRITE M.SRCH (display) READ M.SRCH (input) READ+WRITE LADDER READ+WRITE CLRLAD READ+WRITE CLRALL READ+WRITE DBGLAD READ ONLEDT READ+WRITE READ+WRITE Screens Requiring Password Release and Corresponding Password Types (DPL/MDI) Selected screen Password READ+WRITE ce nt e LADDER r.c Table 1.5 (b) om PMCLAD w w w .c nc NOTE 1 See the following items for the selected screens listed in Table 1.5 (a). PMCLAD : 5. PMC LADDER DIAGRAM DISPLAY (PMCLAD) in Part II M.SRCH : 3.5 Display the Contents of Memory (M.SRCH) in Part II LADDER : 5.2 Sequence program generation (LADDER) in Part III CLRLAD : 5.6.2 Clear the ladder program (CLRLAD) in Part III CLRALL : 5.6.5 Clear the sequence program (CLRALL) in Part III DBGLAD : 8.4 Ladder Debug Function in Part III ONLEDT : 5.8 On-line Editing in Part II 8.4.2 Soft key menu for ladder debug function in Part III 2 For an explanation of the selection screen of Table 1.5 (b), see the following section: LADDER : III 11.4 Ladder Mnemonic Editing 3 With DPL/MDI of the Power Mate, the use of the following characters only is supported for clearing passwords: Alphabetic characters : D, F, G, K, P, T, X, Y Numeric characters : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 If a character other than those listed above is used for a password, the password cannot be cleared using the DPL/MDI. (3) Setting a password Set a password for a ladder program on the editing/password screen on FAPT LADDER (for personal computers). 342 1.

GENERAL PMC OPERATION (CRT/MDI) B–61863E/10 (4) Releasing password protection A ladder program for which the password has been specified cannot be displayed or edited until the password is input correctly. Once password protection is released, the protection remains being released until the power is turned off then on again. (a) When operation which requires releasing the password protection is performed, the system displays either of the following messages to require the protection to be released, depending on the type of password. om “KEY IN PASSWORD(R)” ··· READ PASSWORD “KEY IN PASSWORD(R/W)” ··· READ+WRITE PASSWORD (b) Enter the password and press the [INPUT] key. *The entered password is not displayed.

(Echo back is not performed.) ce nt e r.c (c) When the password is correctly specified, the protection is released and the corresponding operation becomes available. See Table 1.5(a). If the password is incorrectly specified, the message “FALSE PASSWORD” is displayed. w w w .c nc NOTE The sequence program is cleared by turning on the power with the X and O keys being held down, whether password protection is specified or not. 343 1. GENERAL PMC OPERATION (CRT/MDI) B–61863E/10 (5) Special password Ę : Usable : See Note. : Not usable PA1 PA3 RA1 RA3 FS18A FS16A FS16B FS18B RA1, RA2 RB, RB2, RB3 RA3 RC, RC3 RB3 RC3 FS16C FS18C RB4 RC4 RB5 RC3 FS21 RB6 RC4 f RA1 RA5 FS16 FS18 RB5 RB6 FS15B NB NB2 om Power FS20/F Mate/ S21B FS21A f f ce nt e r.c NOTE Usable editions PMC–RB4 : Series 4066 Edition 08 or later PMC–RC4 : Series 4068 Edition 07 or later Edit card : Series 4073 Edition 06 or later When a password beginning with the character # is set for RW password, the subprogram after P1500 can be edited in spite of the protection by this password. LADDER > LEVEL2 V P0002 V P0009 V P0022 · · V P1501 .c nc LEVEL1 V P0001 V P0008 V P0021 · · · · V P1500 PROGRAM:(STEP SEQUENCE DEMO PROGRAM) P1500 ( ) USER PROGRAM NO.1 LEVEL3 V P0004 V P0014 V P0024 · · V]P1502 V P0005 V P0015 V P0025 · · V P0006 V P0016 V P0026 · · MONIT STOP V P0007 V P0017 V] P0027 w w w example 1) When the cursor is positioned to the subprogram P1500 and [ZOOM] key is pressed, this subprogram P1500 can be edited in spite of the protection by the password. example2) When the cursor is positioned to the subprogram P1 and [ZOOM] key is pressed, if the protection by the password is not released, the message ”KEY IN PASSWORD(R/W)” is displayed and this subprogram can be edited by inputting a correct password. 344 1. GENERAL PMC OPERATION (CRT/MDI) B–61863E/10 1.6 Note the following when PMC of loader control function is operated. PMC OPERATION FOR LOADER CONTOROL FUNCTION D Operate PMC after switching to the screen for the loader control. (The control of the main and the loader changes by pushing the SHIFT key and the HELP key at the same time.) D Connector JD5A of main board is used when communicating with RS232–C. D When ladder data is input and output to the memory card on the PMC I/O screen or an edit card is used,the edit card or the memory card is installed at connector CNMC of the loader board. w w w .c nc ce nt e r.c om D Connector JD1A of loader board is used when using I/O Link function. 345 2. PMC MENU SELECTION PROCEDURE BY SOFTKEY 2 PMC OPERATION (CRT/MDI) B–61863E/10 PMC MENU SELECTION PROCEDURE BY SOFTKEY om Pressing the function key of CRT/MDI and the PMC soft key changes the screen to the PMC basic screen. The soft keys are displayed at the bottom of the screen. 1) PMC basic menu r.c If the control provides a built-in programmer function, a programmer basic menu is selected by depressing the next key. The PMC basic menu and programmer basic menu are alternately selected from each other by depressing the next key. For programmer basic menus and operation, see Chapter III “PMC PROGRAMMER”. ce nt e NOTE 1 In the following description, the relation between soft keys and menu is described based on the 9” CRT/MDI panel. The 10”, 14” CRT/MDI panel is provided with 10 soft keys which are those of the 9” CRT/MDI panel, and thus, it displays many menus as compared with the 9” CRT/MDI panel.

2 The following operations are necessary for using the built–in programmer function: Model Operation PMC–RA1/RA2/RA3/RB/RB2/RB3 (FS16/18–MODEL A), PMC–RA1(FS16–MODEL A loader control) Mount the editing module. nc (A02B–0120C–C160) Mount an editing card. PMC–RC/RC3(FS16/18–MODEL A), PMC–RC3/RC4(FS16/18–MODEL B), PMC–NB/NB2(FS15B) The function is already contained. Common to all the models listed above Set bit 1 of K17 to 1. w w .c PMC–PA1/PA3 (Power Mate–D/H), PMC–RA1/RB3/RB4 (FS16/18–MODEL B), PMC–RB5/RB6 (FS16/18–MODEL C), PMC–RA1/RA5/RB5/RB6 (16i/18i/21i–MODEL A), PMC–RA1/RA3 (FS20, FS21/210–B), PMC–RA1 (FS16–MODEL B/C, 16i/18i/21i–MODEL A, FS21–B loadercontrol function) w The FS18–MODEL A contains the PMC–RA1,RA2,or RA3.

The FS20 contains the PMC–RA1 or RA3. The FS21/210–B contains PMC–RA1 or RA3. The FS21–B(with loader control) contains PMC–RA1. 346 The series number is 4070. The series number is 4080. The series number is 4084. The series number is 4086. PMC OPERATION (CRT/MDI) B–61863E/10 2. PMC MENU SELECTION PROCEDURE BY SOFTKEY PMC DIAGNOSIS FUNCTIONS MONIT RUN SELECT ONE OF FOLLOWING SOFT KEYS Built-in programmer function DYNAMIC LADDER DISPLAY DIAGNOSIS FUNCTION PARAMETER(T/C/K/D) RUN/STOP SEQUENCE PROGRAM EDIT SEQUENCE PROGRAM I/O SEQUENCE PROGRAM SYSTEM PARAMETER PMC MONITOR 2] [ ] r.c [PMCLAD] [PMCDGN] [PMCPRM] [om PMCLAD : PMCDGN : PMCPRM : RUN/STOP: EDIT : I/O : SYSPRM : MONIT : PMC basic menu screen (9”CRT) ce nt e NOTE Without built-in programmer function of PMC-RA1, -RA2, -RA3, -RB, -RB2, -RB3, -RB4, -RB5, or -RB6 there are only RUN/STOP and I/O functions. w w w .c nc 2) Keys on CRT/MDI panel

The following keys are related to PMC operation on CRT/MDI panel. a) key Selects from CNC menu to PMC basic menu. b) key Screen page return key. c) key Screen page advance key. d) key Cursor shift (upward) key.
e) key Cursor shift (downward) key. f) key Cursor shift (leftward) key. Search function with this key is provided in PMCLAD EDIT, LADDER (See chapter II.5 and Chapter III.5.2.5 for details). g) key Cursor shift (rightward) key. Search function with this key is provided in PMCLAD EDIT, LADDER (See chapter II.5 and Chapter III.5.2.5 for details). h) Soft key These keys show operating functions corresponding to individual operations when various PMC operations are done. The soft key functions change (key menus are selected) according to operations. 347 2. PMC MENU SELECTION PROCEDURE BY SOFTKEY PMC OPERATION (CRT/MDI) i) B–61863E/10 Next key This key is used for extending menus of soft keys. By pressing this key, a menu changes, and by pressing it again, the menu is reset as before. j) Return key Various PMC operations are conducted by pressing soft keys related to menus. The menus sequentially change when pressing corresponding soft key. Use this return key to reset a menu to the original one. om 3) Status display Data entry > Status display r.c The alarm status and the name of the sequence program storage that is currently effective are displayed on all the PMC menus. In addition, PMC-RC/RC3/RC4/NB display the states while the debugging function is used.
DBG Soft key display [ ] [ ] [ ] [ ] [ALM [ce nt e [-RAM- D ALM : An alarm occurred in the PMC (For details, see Section 3.3.) D RAM : The currently effective sequence program storage is a RAM module. D ROM : The currently effective sequence program storage is a ROM module. nc D EPROM: Currently effective sequence program storage is EPROM. (EPROM for PMC-RA1, PMC-RA2, PMC-RB, and PMC-RB2) : A break issued by the debugging function of PMC-RC/RC3/RC4/NB in effective. D BRK : The break issued by the debugging function of PMC-RC/RC3/RC4/NB has terminated. w w w .c D DBG 348 PMC OPERATION (CRT/MDI) B–61863E/10 2. PMC MENU SELECTION PROCEDURE BY SOFTKEY 4) Relation between PMC menus and soft keys There are 2 types, A and B, in the series of CNC. A–TYPE I/O Chapter II.5 Chapter II.3 Chapter II.4 PMCLAD PMCDGN PMCPRM RET RET SYSPRM RET TITLE TIMER BOTTOM STATUS COUNTR SRCH ALARM KEEPRL W- SRCH TRACE DATA N- SRCH NEXT NEXT F- SRCH M. SRCH*1 nc ANALYS*2 NEXT ce nt e TOP MONIT om EDIT r.c RUN or STOP USRDGN*3 NOTE The soft keys indicated by *1, *2, *3 are supported only for certain models. See the conditions in the description of each relevant function. w w w .c ADRESS (SYMBOL) 349 2. PMC MENU SELECTION PROCEDURE BY SOFTKEY PMC OPERATION (CRT/MDI) B–61863E/10 B–TYPE RUN or STOP EDIT I/O Chapter II.5 Chapter II.3 Chapter II.4 PMCLAD PMCDGN PMCPRM SEARCH NEXT RET TITLE TIMER ADRESS (SYMBOL) TRIGER STATUS COUNTR ALARM KEEPRL WINDOW TRACE DATA om RET MONIT r.c RET SYSPRM NEXT NEXT DUMP M. SRCH*1 DPARA ANALYS*2 ce nt e SETING USRDGN*3 nc ONLEDT .c IOCHK*4 w w NOTE The soft keys indicated by *1, *2, *3 are supported only for certain models.
See the conditions in the description of each relevant function.
The softkey’ s type for the series of CNC. Power Mate PMC type PA1 PA3 RA1 RA3 RA1 RA2 RA3 RB RB2 RB3 RC RC3 NB NB2 Softkey type A A A A AB A B AB A B AB AB B B w CNC type FS20 FS18 FS16 FS15B Type A or B is selected depending on the Series of PMC control software. Series of PMC control software and type of softkey are related as follows. Type A Type B FS16A 4061 4063 FS18A 4070 4071 350 PMC OPERATION (CRT/MDI) B–61863E/10 2. PMC MENU SELECTION PROCEDURE BY SOFTKEY (Reference) Series of PMC control software is displayed on the [PMCDGN] and [TITLE] screen as shown below. om PMC CONTROL PROGRAM SERIES : 4063 EDITION : 08 w w w .c nc ce nt e r.c Series of PMC control software 351 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) 3 PMC OPERATION (CRT/MDI) B–61863E/10 PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) RET STATUS ALARM TRACE ce nt e TITLE r.c PMCLAD PMCDGN PMCPRM om PMC I/O signals, internal relays, and other PMC diagnosis are displayed on the screen by depressing soft key [PMCDGN]. w w w .c nc M.SRCH 352 ANALYS USRDGN NEXT PMC OPERATION (CRT/MDI) B–61863E/10 3.1 DISPLAYING TITLE DATA 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) Title Data refers to the title of the sequence program created by the machine tool builder. They consist of the following ten items : In addition to the title display : (32 characters) (32 characters) (32 characters) (4 characters) (2 characters) (32 characters) (16 characters) (32 characters) (32 characters) (32 characters) om Machine tool builder name Machine tool name NC and PMC types Sequence program number Version Sequence program drawing number Date when the sequence program was created Sequence program programmer ROM programmer Comment 2) Type of the PMC.
r.c 1) Series and version of the PMC control software. 3) For Editing module or Editing card, the series and version. ce nt e 4) Memory areas used for each sequence data, and execution time of ladder program. 5) Type of PMC control module and PMC sequence program.
6) For the non-dividing system, the present, maximum and minimum values for the execution time of ladder program. nc NOTE When a C board is installed in the Series 16i/18i, the title data for C can be displayed. With the arrow keys [2] and [3], the user can switch the display between the ladder title and C title data. w w w .c To display the previous or next screen on the 9” CRT/MDI, use the or key. PMC TITLE DATA #1 MONIT RUN PMC PROGRAM NO.
: 1234 EDITION NO. : 12 PMC CONTROL PROGRAM SERIES : 4063 EDITION : 08 (SERIES : 4065 EDITION : 08) PMC TYPE CONTROL : RB3 PROGRAM : RB3 MEMORY USED : 007.8 KB LADDER : 007.0 KB SYMBOL : 000.0 KB MESSAGE : 000.8 KB SCAN TIME : 008 MS SCAN MAX : 016 MS MIN : 008 MS [TITLE] [STATUS] [ALARM] [TRACE] [3.1 (a) 353 Title Data 1] 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 PMC TITLE DATA #2 MONIT RUN MACHINE TOOL BUILDER NAME : f · · · · · · · · · · · · · f MACHINE TOOL NAME : f · · · · · · · · · · · · · f om CNC & PMC TYPE NAME : f · · · · · · · · · · · · · f PROGRAM DRAWING NO. : f · · · · · · · · · · · · · f 3.1 (b) Title data 2 MONIT RUN ce nt e PMC TITLE DATA #3] r.c [TITLE] [STATUS] [ALARM] [TRACE] [DATE OF PROGRAMING : f · · · · · · f PROGRAM DESIGNED BY : f · · · · · · · · · · · · · f ROM WRITTEN BY : f · · · · · · · · · · · · · f nc REMARKS : f · · · · · · · · · · · · · f .c [TITLE] [STATUS] [ALARM] [TRACE] [w w w 3.1 (c) 354 Title data 3] 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 TITLE RET ALARM SEARCH M.SRCH TRACE NEXT ANALYS USRDGN I/O CHK Depress [STATUS] soft key. The CRT screen changes as shown in Fig.
3.2, and the soft key menu is changed. ce nt e 1 STATUS om DISPLAY OF SIGNAL STATUS (STATUS) The contents at all addresses (X, Y, F, G, R, A, C, K, D, T, M, N) disignated in programs can be displayed on the CRT screen. This display is all done by “0” and “1” bit patterns, and symbol data is displayed together at address bits where symbol data are difined. r.c 3.2 2 Depress [SEARCH] key after keying in an address to be displayed.
3 A continuous 8 byte data is displayed by a bit pattern from the designated address in the top stage of the CRT screen.
4 Depress [SEARCH] key or page key to display another address.
PMC SIGNAL STATUS MONIT RUN w w w .c nc ADDRESS 7 6 5 4 3 2 1 0 EXDAT1 ED7 ED6 ED5 ED4 ED3 ED2 ED1 ED0 G0000 0 0 0 0 0 0 0 0 EXDAT2 ED15 ED14 ED13 ED12 ED11 ED10 ED9 ED8 G0001 0 0 0 0 0 0 0 0 ESTB EA6 EA5 EA4 EA3 EA2 EA1 EA0 G0002 0 0 0 0 0 0 0 0 ERDRQ EOREND G0003 0 0 0 0 0 0 0 0 MFIN5 MFIN4 MFIN3 MFIN2 MFIN1 G0004 0 0 0 0 0 0 0 0 BFIN AFL TFIN SFIN EFIND MFIN G0005 0 0 0 0 0 0 0 0 DLK OVC *ABSM BRN SRN G0004 0 0 0 0 0 0 0 0 RLSOT EXLM2 *FLWP ST STLK RVS G0007 0 0 0 0 0 0 0 0 [SEARCH] [ ] [ ] [ ] [ ] [ ] 3.2 Status display of PMC I/O signals and internal relays 355 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) 3.3 PMC OPERATION (CRT/MDI) B–61863E/10 If an alarm is issued in the PMC, pressing the PMC soft key displays the alarm message as shown in Fig. 3.3. ALM blinks at the lower right corner of the screen. If a fatal error occurs, a sequence program does not start. ALARM SCREEN (ALARM) PMC ALARM MESSAGE MONIT STOP r.c om ER00 PROGRAM DATA ERROR (ROM) ALM ce nt e [TITLE] [STATUS] [ALARM] [TRACE] [ ] 3.3 Alarm screen w w w .c nc For displayed messages, see the appendix, “Alarm Message List.” 356 PMC OPERATION (CRT/MDI) B–61863E/10 3.4 TRACE FUNCTION (TRACE) 3.4.1 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) This function checks the signal history which cannot be checked in the status display. Using one- or two-byte addressing, the function records a state when the signal changes. In two-byte addressing, discontinuous addresses can be set. Pressing the [TRACE] key on the PMCDGN screen displays the trace screen when signals are being read. When signals are not being read, the parameter setting screen for reading signals is displayed. After displaying either screen, pressing the [TRCPRM] key on the trace screen displays the parameter setting screen and pressing the [T.DISP] key on the parameter setting screen displays the trace screen. TITLE r.c om Operation STATUS TRACE ce nt e RET ALARM T.DISP EXEC NEXT or or TRCPRM STOP 3.4.2 w w w USRDGN Data to be used for reading signals needs to be specified to check the signal history. 1) Parameters .c Parameter Setting Screen ANALYS nc M.SRCH TRACE MODE : Sets a mode used for reading signals 0 : 1-byte data 1 : 2-byte data (discontinuous addresses can be specified) 2 : Word data (with continuous addresses) ADDRESS TYPE : Sets addresses used 0 : PMC address 1 : Physical address ADDRESS : Sets addresses at which a signal is traced MASK DATA : Sets a masked bit or bits (signals can be read with unnecessary bits masked) Range : 00 to FF The above trace parameters are retained if the power is turned off. 357 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) 3.4.3 PMC OPERATION (CRT/MDI) B–61863E/10 EXEC : Starts reading signals NOTE 1 Pressing the [EXEC] key again clears the results of the previous trace. If the trace parameters are not set correctly, the trace is not performed. When signals are being sampled using the function for displaying signal waveforms, the trace is not performed. 2 The result data of the trace is stored latest 256-byte. If the power is turned off, the results of the trace are cleared. 3 Signals R9000 to R9007 cannot be traced. 4 A signal is traced at intervals of 8 ms. If the signal changes within 8 ms, the changed signal state cannot be traced. 5 When the trace address type is specified as a physical address, specify an effective memory address. If an ineffective address is specified to execute the trace, a system error may occur. ce nt e r.c om Starting or Stopping the Trace Function STOP : Stops reading signals. PMC SIGNAL TRACE MONIT RUN [T.DISP] [EXEC] [ ] [ ] [3.4.3 Trace Parameter setting screen w w w .c nc TRACE MODE : 1 (0:1BYTE/1:2BYTE/2:WORD) 1ST TRACE ADDRESS CONDITION ADDRESS TYPE : 1 (0:PMC /1:PHY) ADDRESS : FFE480 MASK DATA : 11 2ND TRACE ADDRESS CONDITION ADDRESS TYPE : 0 (0:PMC /1:PHY) ADDRESS : Y0 MASK DATA : FF 358] 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 3.4.4 Signal history can be checked using data specified on the parameter setting screen. The result of the latest trace is displayed at the cursor position. The cursor moves on the screen as the results of the trace are obtained. If the cursor moves off the screen, the results of the trace can be followed by pressing the page key to display the subsequent screen. Trace Screen PMC SIGNAL TRACE 7 .
. . . .
. . . . . . 2ND 6 . . .
. . . . . .
. .
. . .
. . ADDRESS 5 4 . . . .
. . . .
. . . .
. . = Y0000 (FF) 3 2 1 0 . .
. . . .
. . . .
. . .
. . .
. . . .
. . . .
. . . .
. om (FF) 1 0 . . . .
. . . . r.c 1ST ADDRESS = X0000 7 6 5 4 3 2 . . . .
. . . . . .
. . . .
. .
. .
. .
. . . . . .
. . . .
. . . .
. . .
. . . .
. . . . ce nt e NO. 0000 0001 0002 0003 0004 0005 0006 0007 0008 0009 0010 0011 0012 0013 0014 0015 MONIT RUN [TRCPRM] [STOP] [ ] [ ] [ ] EXEC 3.4.4 Trace Screen Trace operation can be started automatically, immediately after power–on, by setting trace parameters beforehand and by setting the keep relay to start the trace function automatically upon power–on. The keep relay setting method depends on the PMC model.
See Section 4.3.3. nc 3.4.5 w w w .c Automatic Tracing Function at Power on 359 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 3.5 DISPLAYING THE CONTENTS OF MEMORY Power MateD/F Power Mate-H PA1 PA3 PA3 FS20/ FS21A FS21B : Standard Ę : optional : cannot be used : Can be used for the 4084 series. FS18A RA1 RA3 RA1 RA3 RA1 RA2 RA3 Ę Ę Ę FS18B RA1 Ę FS16A FS16A/B FS18B FS16C FS18C FS21i FS16i FS18i RB RB2 RB3 RB4 RB5 RB6 RA1 RA5 RB5 RB6 FS16A FS16A /B/C FS18B/C FS16B/C FS18B/C FS15B RC RC3 RC4 NB NB2 Ę Ę Ę Ę om D The ladder editing module is required for the PMC–RB/RB2/RB3 of the Series 16 –MODEL A and for the PMC–RA1/RA2/RA3 of the Series 18–MODEL A. r.c D This function is provided as a standard function with PMC–RA1/RB3/RB4/RB5/RB6/RC3/RC4 of the Series 16/18–MODEL B/C, and PMC–RA1/RA3 of the Series 21/210–MB. 3.5.1 ce nt e 1) Pressing the [M.SRCH] soft key changes the screen to that shown in Fig. 3.5. The displayed soft keys also change. Operation 2) Enter a physical address in hexadecimal from which the contents of the memory are to be displayed. Then pressing the [SEARCH] key displays 256 bytes of stored data starting from the specified address. Example) Entering 100000, then pressing the [SEARCH] key displays the contents of the memory starting from 100000H. 3) An address can be changed using the or key. nc 4) Pressing either the [BYTE], [WORD], or [D.WORD] soft key displays data of the corresponding type. w w w .c CAUTION If an address at which the memory is not used is specified, a system error occurs. Be sure to specify the correct address. 360 PMC OPERATION (CRT/MDI) B–61863E/10 3.5.2 Function for Storing Data in Memory 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) To store data in memory, set bit 4 of keep relay K17 to 1, move the cursor to a position at which the address of the data to be changed in RAM is displayed, and enter data in units of data type in hexadecimal. Example) Entering 0F41, then pressing the [INPUT] key stores 0F41 at the address specified by the cursor. PMC CONTENTS OF MEMORY om CAUTION Some values cause a system error. MONIT RUN 0000 4142 2020 5152 0000 4344 2020 5354 0000 4546 2020 5556 0000 4748 2020 5758 0000 494A 2020 595A 0000 4B4C 2020 2020 0000 4D4E 2020 2020 0000 4F50 2020 2020 .... ABCDEFGHIJKLMNOP 100040 100050 100060 100070 100080 100090 1000A0 1000B0 0000 0000 0000 0000 4641 0000 0000 0000 0000 0000 0000 4E55 0000 0000 0000 0000 0000 0000 4320 0000 0000 0000 0000 0000 0000 434F 0000 0000 0000 0000 0000 0000 2E2C 0000 0000 0000 0000 0000 0000 5444 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 .... FANUC CO.LTD.... ....
.... 1000C0 1000D0 1000E0 1000F0 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 ....
.... ce nt e r.c 100000 100010 100020 100030 QRSTUVWXYZ > SEARCH] [INPUT w w w .c nc [361] [ ] [3.5.2 Memory Display] [ ] 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 3.6 FUNCTION FOR DISPLAYING SIGNAL WAVEFORMS (ANALYS) PA1 Power Mate–H PA3 PA3 FS20/ FS21A RA1 RA3 FS21B RA1 RA3 Ę FS18A RA1 RA2 FS18B FS16A RB2 FS16A/B FS18B RB3 RB4 FS16C FS18C RB5 FS21i FS16i FS18i FS16A FS16A /B/C FS18B/C FS16B/C FS18B/C RC3 RC4 RA3 RA1 RB RB6 RA1 RA5 RB5 RB6 RC Ę Ę Ę Ę Ę Ę Ę Ę Ę Ę Ę Ę Ę FS15B NB NB2 om Power Mate–D/F : Standard Ę : optional : cannot be used : Can be used for the 4084 series. D The ladder edit module is required with PMC–RB/RB2/RB3 of the Series 16–MODEL A, and PMC–RA2/RA3 of the Series 18–MODEL A. r.c D The ladder edit card is required with PMC–RA1/RB3/RB4 of the Series 16/18–MODEL B, PMC–RA1/RB5/RB6 of the Series 16/18–MODEL C, PMC–RA1/RA3 of the Series 21/210–MODEL B, and PMC–PA3 of the Power Mate–MODEL–H. ce nt e D Work RAM is required with PMC–RC/RC3 of the Series 16–MODEL A. D This function is provided as a standard function with PMC–RC3/RC4 of the Series 16/18–MODEL B/C. D The ladder edit card is required with PMC–RA5/RB5/RB6 of the Series 16i/18i/21i–MODEL A. 3.6.1 1) Maximum number of signals traced at the same time: 16 2) Maximum sampling period: 10 s 3) Sampling interval: 8 ms nc Specifications 3.6.2 w w w .c Operation Pressing the [ANALYS] key on the PMCDGN screen displays the parameter screen for diagnosing signals. Pressing the [SCOPE] soft key on the parameter screen displays the screen showing signal diagnosis. To return to the parameter screen, press the [SGNPRM] soft key. 362 PMC OPERATION (CRT/MDI) B–61863E/10 TITLE STATUS 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) ALARM TRACE NEXT RET M.SRCH ANALYS USRDGN SCOPE DELETE INIT ADRESS or SYMBOL START T.SRCH ADRESS om SGNPRM or SYMBOL EXCHG SCALE r.c SELECT CANCEL RET CANCEL ce nt e TO NEXT EXEC 3.6.3 CANCEL To trace the state of a signal, the trace conditions need to be specified on the parameter screen.
In a 9” screen, a trace address can be specified by pressing the key. (See Fig. 3.6 (b).) nc Parameter Screen w w w .c 1) Setting parameters Move the cursor to a parameter to be specified. Enter a value and press the [INPUT] key. To delete the value of the parameter, move the cursor to the parameter, then press the [DELETE] soft key. a) SAMPLING TIME Specify the maximum trace time in the range of 1 to 10 s. b) TRIGGER ADDRESS Specify a trigger address from which the tracing starts on the PMC address. A symbol name can be used. c) CONDITION Specify the conditions at which the tracing starts. 0 : When the [START] key is pressed 1 : When the [START] key is pressed and the trigger address signal rises 2 : When the [START] key is pressed and the trigger address signal falls NOTE Conditions 1 and 2 are effective when a trigger address is specified. 363 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 d) TRIGGER MODE Sampled data for up to 10 seconds is stored in the trace buffer. A signal is stored in the buffer within 8 ms. This parameter specifies the starting and end points for obtaining data. 0 : AFTER In this mode, signal states are obtained in the period specified in parameter SAMPLING TIME from the time when the trigger conditions are satisfied. om 1 : ABOUT In this mode, signal states are obtained in the period specified in the parameter SAMPLING TIME with the time at the middle when the trigger conditions are satisfied.
2 : BEFORE 3 : ONLY r.c In this mode, signal states are obtained in the period specified in parameter SAMPLING TIME before the trigger conditions are satisfied. ce nt e In this mode, the signal states are obtained only when the trigger conditions are satisfied. NOTE Trigger mode 1 and 2 are effective when condition 1 or 2 is set. e) SIGNAL ADDRESS nc Specify up to 16 addresses at which the tracing is performed with PMC addresses or symbol names. .c 2) Initializing signal diagnosis data Pressing the [INIT] soft key on the parameter screen initializes parameter data and trace data. Pressing the [ADRESS] soft key displays trigger and trace addresses for which symbols are defined and the key changes to the [SYMBOL] key. Pressing the [SYMBOL] key displays the symbols for trigger and trace addresses and the key changes to the [ADRESS] key. w w w 3) Displaying symbols for trigger addresses and trace addresses 364 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 PMC SIGNAL PARAMETER SAMPLING TIME : MONIT RUN 10(1-10SEC) TRIGGER ADDRESS : *ESP CONDITION : 1 (0:START 1:TRIGGER-ON 2:TRIGGER-OFF) om TRIGGER MODE : 1 (0:AFTER 1:ABOUT 2:BEFORE 3:ONLY) >] [ADRESS] [ ] r.c [SCOPE] [DELETE] [INIT ADRESS/SYMBOL Parameter Setting Screen 1 ce nt e 3.6.3 (a) PMC SIGNAL PARAMETER MONIT RUN SIGNAL ADDRESS : : : : : : : : ED0 ED1 ED2 ED3 ED4 ED5 ED6 ED7 nc 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 : : : : : : : : X0000.0 X0000.1 X0000.2 X0000.3 X0000.4 X0000.5 X0000.6 X0000.7 >] [ADRESS] [ ] ADRESS/SYMBOL 3.6.3 (b) w w w .c [SCOPE] [DELETE] [INIT 365 Parameter Setting Screen 2 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) 3.6.4 PMC OPERATION (CRT/MDI) B–61863E/10 After parameters are specified on the parameter screen, select the signal diagnosis screen.
Signal Diagnosis Screen Pressing the [START] soft key starts to trace the specified signal. While the signals are traced, “EXECUTING” is displayed.
When the tracing is finished, the period in which the specified signal was traced is displayed on the screen. om When the optional graphic function is provided, the waveform is displayed by using the graphic function. When the function is not provided, waveform is displayed with “J” indicating the signal is on and “_” indicating the signal off. r.c In the ONLY mode, even when the optional graphic function is provided,“J” and “_” is used to display the waveform as shown in Fig. 3.6.4. 1) Starting or stopping the data sampling ce nt e Pressing the [START] key starts sampling. Pressing the [STOP] key stops sampling and the sampled data is displayed. 2) Displaying traced data by specifying a period Enter a period in ms in which traced data is to be displayed. Pressing the [T.SRCH] key displays the traced data. Example) Entering 800, then pressing the [T.SRCH] key displays the waveform from 512 ms to 1024 ms. 3) Displaying symbols for trigger and trace addresses When symbols are defined for trigger and trace addresses, the symbols and addresses are displayed w w w .c nc 4) Exchanging positions at which traced data is displayed Pressing the [EXCHG] key moves the cursor to the first traced address.
Position the cursor to the trace address to be exchanged, using the or key, then press the [SELECT] key. Next, position the cursor to the trace address with which the selected trace address is to be exchanged, then press the [TO] key. Finally, press the [EXEC] key. The trace data is exchanged. During the above operation, all other soft keys are disabled until the [EXEC] key has been pressed. To cancel the exchange, press the [CANCEL] key. 5) Changing the time division (This function is available when the graphic function is used.) When displaying the signal waveform, the time division can be changed. Setting 8 . . . . . 8 ms/divisions 16 . . . . 16 ms/divisions 32 . . . . 32 ms/divisions Pressing only the [SCALE] key increments the minimum scale from 8 to 32 ms, as follows: 6) Shifting traced data upward or downward Pressing the key shifts traced data upward. Pressing the key shifts traced data downward. 7) Shifting traced data left or right 366 PMC OPERATION (CRT/MDI) B–61863E/10 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) Pressing the “2” key shifts traced data to the left. Pressing the “3” key shifts traced data to the right. PMC SIGNAL ANALYSIS(SCOPE) MONIT RUN SAMPLING TIME : 10 CONDITION : 1 TRIGGER ADDRESS:*ESP TRIGGER MODE: 1 ED0 ED1 om ED2 ED3 ED4 ED3 0(MSEC) r.c -256 [SGNPRM] [START] [T.SRCH] [ADRESS] [START/STOP] ADRESS/SYMBOL 3.6.5 Since parameter and sampling data is stored in nonvolatile memory, data is retained when the power is turned off. Data sampling can be started automatically, immediately after power–on, by setting sampling parameters and setting the keep relay beforehand. The keep relay setting method depends on the PMC model. See Section 4.3.3. w w w .c nc Reading Signals Automatically at Power on ce nt e 3.6.4 Screen Displaying Signal Diagnosis 367 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) 3.7 PMC OPERATION (CRT/MDI) B–61863E/10 Pressing the [USRDGN] key dynamically displays the running states of user tasks (including the third level of a ladder program) in the PMC (Fig. 3.7). DISPLAYING THE RUNNING STATE OF A USER TASK (USRDGN) Ę : Can be used : Cannot be used PA1 PA3 RA1 RA2 RA3 RA5 RB RB2 RB3 RB4 RB5 RB6 RC RC3 RC4 NB NB2 Ę Ę Ę Ę Ę om Work RAM is necessary (A02B-0120-H987 for the PMC-RC and PMC-RC3 and A02B-0162-J151 or A02B-0162-J152 for the PMC-NB). For details, refer to the “PMC-RC/RC3/RC4/NB Programming Manual (C language)” (B-61863E-1). MONIT RUN r.c PMC MONIT USER TASK #1 ce nt e ID NAME LV STATUS WAIT-INF WAIT-ID LAD3 10 READY 10 TASK_O1 @ 10 ACTIVE 11 TASK_O2 # 11 READY 12 TASK_O3 12 WAIT TIM 13 TASK_O4 13 WAIT EVT.O 1 14 TASK_O5 14 WAIT EVT.A 3 15 TASK_O6 15 WAIT PKT 2340 16 TASK_O7 STOP 17 TASK8 17 READY >][ ][ ][ ][ ] nc [3.7 Screen Displaying the Running States of User Tasks NAME TASK1 w ID 11 .c [Displayed items] LV 13 STATUS WAIT WAIT-INF EVT.O WAIT-ID 10 ID for wait information w # Wait information Running state w Task level Operation Task name Task ID 1) Operation Code Description # RS-232C being used @ NC command edit being used 368 PMC OPERATION (CRT/MDI) B–61863E/10 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) 2) Running state Code Description ACTIVE Running READY Ready WAIT Waiting STOP Task stopped The system deleted the task because the task had called library that is not supported. 3) Wait information Code om ERROR Description TIM Waiting for time-out EVT.A Waiting for AND condition of event flags EVT.O Waiting for OR condition of event flags Waiting for semaphore r.c SEM MBX.R Waiting for READ of the mail box MBX.W Waiting for WRITE of the mail box PKT Waiting for a packet to be received Waiting for the PCMDI command to be issued w w w .c nc ce nt e PCMDI 369 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 3.8 Ę : Supprted : Not supported Power Mate– D/F/G Power Mate–H FS21 TA/TB FS20 FS18 FS16–A FS16–B FS18–B FS21i FS16i FS18i FS15–B Ę Ę Ę Ę Ę Ę Ę Ę In case of FS16–B/FS18–B : TITLE STATUS ALARM om DISPLAYING AND SETTING THE CONFIGURATION STATUS OF I/O DEVICES(IOCHK) TRACE NEXT ANALYS IOLNK IOLNK2 USRDGN IOCHK r.c M.SRCH ce nt e RET INPUT DELETE DELALL PRV.CH NXT.CH RET IOLNK2 nc IOLNK NEXT SELECT ONE OF FOLLOWING SOFT KEYS IOLNK : I/O LINK CHECK IOLNK2 : I/O LINK–II SETTING w w w .c PMC I/O CHECK 3.8 I/O Check Menu Screen The I/O check screen has two functions. By pressing the soft key, the following screens are displayed. [IOLNK] : I/O Link connecting check screen.
[IOLNK2] : I/O Link–II parameter setting screen. 370 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 In case of FS15–B : FS15–B has not supported [IOLNK2] screen. By pressing [IOCHK] key, I/O Link connecting check screen is selected directly. TITLE STATUS ALARM TRACE NEXT 3.8.1 IOCHK om USRDGN The I/O Link connecting check screen displays the types and ID codes of the connected I/O devices for each group. When I/O device is not connected, ”NO I/O DEVICE” is displayed. When input to or output from an I/O devices is abnormal, check if the configuration of the connected I/O devices correct is by referring the screen. ce nt e I/O Link Connecting Check Screen ANALYS r.c M.SRCH I/O CHECK ID KIND OF UNIT 00 80 CONNECTION UNIT 01 82 OPERATOR PANEL 02 84 I/O UNIT MODEL A 03 96 CONNECTION UNIT 04 4A POWER MATE w w w .c nc GROUP Fig.3.8.1 (a) Example of the I/O Link Screen Table 3.8.1 I/O Devices and ID Codes Displayed I/O device name ID Actual I/O device CONNECTION UNIT 80 Connection unit OPERATOR PANEL 82 Operator’s panel connection unit I/O-B3 83 Expanded I/O B3 I/O UNIT MODEL A 84 to 87 I/O UNIT MODEL A I/O UNIT MODEL B 9D to 9E I/O UNIT MODEL B POWER MATE CONNECTION UNIT 4A Power Mate 96 I/O Link connection unit I/O MODULE A9 to AA OTHER UNIT 371 Distributed I/O Other than above 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 When the screen is displayed like fig.3.8.1(a) The I/O devices are composed like following fig.3.8.1(2). Connection CNC Group 0 Unit I/O Group 1 om Operator’s Panel Connection Unit Group 2 r.c Unit MODEL A Connection ce nt e Unit Group 3 Power Mate 3.8.1 (b) I/O Link Configuration In case of using the I/O Link–II function, set the following I/O Link–II parameter on this screen. Depending on the kind of I/O Link–II interface board, master/slave screen is displayed automatically. Please refer to FANUC I/O Link–II operating manual (B–62714EN) about details of I/O Link–II and each parameter. nc 3.8.2 Group 4 w w w .c I/O Link–II Parameter Setting Screen (1) Set parameters. Move the cursor to the parameter by using the cursor key. Type the data and press the soft key[INPUT] or MDI key. The set parameter is saved to the I/O Link–II board when the data is input. (2) Change channel. Change the channel by the soft key [PRV.CH],[NXT.CH]. These keys are not displayed when the single channel is used. (3) Delete parameter. Move the cursor to the parameter by using the cursor key. Press the soft key[DELETE]. (4) Delete all parameters. Press the soft key[DELALL]. Press the soft key[EXEC] to delete all parameters. Press the soft key[CANCEL] to cancel the deletion.
(5) Change page. This screen is composed of two pages when the 9 inch CRT is used. Change the page by using (PAGE) key of MDI. 372 PMC OPERATION (CRT/MDI) B–61863E/10 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) (6) Re–start I/O Link–II Press the soft key [START] to re–start I/O Link–II after editing the parameter. When the re–start is completed normally, ”LINK STARTED” is displayed. If the re–start fails, ”START ERROR” is displayed. In this case, check the parameter that is set. Example of parameter setting of master. GENERAL: om PMC I/O LINK–II CH 1 (1/2) = 03 (1–31) SCAN TIME = 0100 (1–9999)MSEC STATUS ADDRESS = R0500 DI/DO SETTING: DI/DO MAP MODE r.c MAX SLAVE NO. = 1 (1,2) DI/DO DATA SIZE = 16 (0–32) = R0100 ce nt e DO ADDRESS DI ADDRESS = R0150 [INPUT] [DELETE] [DELALL] [PRV.CH] [NXT.CH] PMC I/O LINK–II CH 1 (2/2) MESSAGE I/O SETTING: nc MESSAGE SIZE = 032 (0–128) OUTPUT ADDRESS = R0200 INPUT ADDRESS = R0250 REFRESH TIME = 40 MSEC I/O LINK–II = 6546/01 (MASTER) [INPUT] [DELETE] [DELALL] [PRV.CH] [NXT.CH] 3.8.2 (a) w w w .c STATUS: 373 Example of the I/O Link–II Screen.(Master) 3.
PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 Example of parameter setting of slave.
PMC I/O LINK–II CH 1 (1/2) GENERAL: MAX SLAVE NO. = 03 (1–31) STATION NO. = 02 (1–31) STATUS ADDRESS = R0900 DI/DO MAP MODE = 0 (0,2) om DI/DO SETTING: DI/DO DATA SIZE = 16 (0–32) = R0000 DI ADDRESS = R0032 r.c DO ADDRESS [INPUT] [DELETE] [DELALL] [PRV.CH] [NXT.CH] ce nt e PMC I/O LINK–II CH 1 (2/2) MESSAGE I/O SETTING: MESSAGE SIZE = 032 (0–128) OUTPUT ADDRESS = R0256 INPUT ADDRESS = R0296 STATUS: nc I/O LINK–II = 6545/01 (SLAVE) 3.8.2 (b) w w w .c [INPUT] [DELETE] [DELALL] [PRV.CH] [NXT.CH] 374 Example of the I/O Link–II Screen.(Slave) 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 3.9 FORCED INPUT/OUTPUT FUNCTION Ę : Usable : See Note. : Not usable RA1 RA3 FS16A FS16B FS18B RA1, RA2 RB, RB2, RB3 RA3 RC, RC3 RB3 RC3 RB4 RC4 FS16C FS18C RB5 RC3 RB6 RC4 FS21 FS16 FS18 FS15B RA5 RB5 RB6 NB NB2 r.c PA1 PA3 FS18A om Power FS20/ Mate FS21A ce nt e NOTE With the FS16i/18i, the edit card or C board are required. With the FS21i, the edit card is required. With RA1 of the FS16i/18i/21i, only FORCING mode is valid. 3.9.1 This function can forcibly enter a value for the signal of an arbitrary PMC address. With this function, for example, an X value can be forcibly entered to enable sequence program debugging without using a machine, and a Y value can be forcibly entered to enable the signal wiring on the machine to be checked efficiently without using a sequence program. This function is added to the status display function. nc Overview w w w .c (1) Input mode Two input modes are available. The user can choose between the two modes, depending on the application. (a) FORCING mode This mode is applicable to all addresses. When input/output scan is performed by a sequence program, however, a signal modified by the forced input/output function is overwritten, and the result of modification made by the forced input/output function is lost.
Machine I/O address Overwrite X, Y Overwrite Overwrite Ladder 375 G, F, R, T, K, C, D. .
3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 Example 1: In this example, the forced input/output function is applied to R0 in the ladder program below.
X0.0 MOVE 1111 1111 K0 om R0 The initial signal states are as follows: X0.0 = off, K0 = 55H, R0 = 00H b. FFH is forcibly entered to R0. X0.0 = off, K0 = 55H, R0 = FFH c. When X0.0 is turned on, R0 assumes the result of output by the sequence program as follows: X0.0 = on, K0 = 55H, R0 = 55H ce nt e r.c a. w w w .c nc Example 2: In this example, the forced input/output function is applied to X0 in a configuration where the I/O UNIT–MODEL A is connected to X0 over an I/O link. The input value from the I/O UNIT–MODEL A is transferred to X0 at certain intervals. So, even if the value of X0 is forcibly modified, X0 is overwritten in the next cycle. Thus, the value of X0 is restored to the value input from the I/O UNIT–MODEL A. I/O UNIT–MODEL A NC X0 = FFH X0 = FFH The value of X0 is forcibly modified to 00H. X0 = 00H In the next cycle, the value of X0 is restored to that entered from the I/O UNIT–MODEL A. I/O UNIT–MODEL A NC X0 = FFH X0 = FFH Cyclical transfer may also be performed for addresses that are not assigned. So, use the forced input/output function for X in FORCING mode to debug a sequence program when no machine is connected or assigned. Use the OVERRIDE mode to debug a sequence program in the case of I/O connection. (b) OVERRIDE mode That state in which values modified by the forced input/output function cannot be overwritten by a sequence program or 376 PMC OPERATION (CRT/MDI) B–61863E/10 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) machine signal is referred to as OVERRIDE mode. In OVERRIDE mode, OVERRIDE can be set for arbitrary X and Y signals. FORCING is applied to those X and Y addresses where OVERRIDE is not set, and also to addresses other than the X and Y addresses. Machine Overwrite disabled Overwrite Ladder I/O address X, Y (OVERRIDE) om Overwrite disabled X, Y (NOT OVERRIDE) r.c G, F, R, T, K, C, D. . ce nt e Example: In this example, the forced input/output function is applied to X0 in a configuration where the I/O UNIT–MODEL A is connected to X0 with an I/O link. NC X0 = FFH X0 = FFH w w w .c nc I/O UNIT–MODEL A OVERRIDE is set for X0. A value transferred from the I/O UNIT–MODEL A is invalidated. I/O UNIT–MODEL A _ X0 = FFH NC X0 = FFH Value 00H is forcibly entered to X0. I/O UNIT–MODEL A X0 = FFH _ NC X0 = 00H Thus, the forced input/output function for X addresses in OVERRIDE mode can be used to debug a sequence program while a machine is connected. When a Y address is placed in the OVERRIDE state, a value that has been forcibly modified by the forced input/output function is output to the machine. 377 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 ce nt e r.c om CAUTION 1 In OVERRIDE mode, input/output signals are updated at 8–ms intervals in sync with the first ladder level. When an I/O link is used which is usually updated at 2–ms intervals, an input/output signal timing delay occurs. For this reason, note that a sequence that depends on input/output signal timing may perform an unpredictable operation. 2 Note that, when OVERRIDE mode is set, the interval of the second level may be extended slightly.
3 Even if OVERRIDE is set for a Y address, the resultant value of a ladder operation before being modified by the forced input/output function is displayed as the coil on/off value in ladder dynamic display. A value, after being modified by the forced input/output function, is output to the machine. So, note that the on/off indication in ladder dynamic display does not match a value output to the machine. Example: In this example, the forced input/output function is used for Y0.0 in the ladder below in a configuration where the I/O UNIT–MODEL A is connected to Y0 with an I/O link. Before OVERRIDE setting, the on/off indication in dynamic display matches a value output to the I/O UNIT–MODEL A as shown below.
I/O UNIT–MODEL A X0.0 Y0.0 NC (Off) (Off) (Dynamic display) nc Y0.0 = off X0.0 I/O UNIT–MODEL A Y0.0 = on Y0.0 NC (Off) (Off) (Dynamic display) w w w .c If 1 is entered forcibly after setting OVERRIDE for Y0.0, the value modified by the forced input/output function is output to the I/O UNIT–MODEL A as shown below. 3.9.2 Use the procedures below to set the input/output modes. Setting/Operation for Enabling Forced Input/Output (1) Operation for enabling FORCING mode Use the procedure below. (a) Mount an edit card or C board. (b) Turn on the power. (c) Set the PROGRAM ENABLE (bit 1 of K17 or bit 1 of K900) setting parameter to YES. (2) Operation for enabling OVERRIDE mode Use the procedure below. 378 PMC OPERATION (CRT/MDI) B–61863E/10 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) (a) Mount an edit card or C board. (b) Turn on the power. (c) Set the PROGRAM ENABLE (bit 1 of K17 or bit 1 of K900) setting parameter to YES. (d) Set the OVERRIDE ENABLE setting parameter (OVERRIDE) to YES. (e) Turn the power off, then back on. r.c om WARNING Special care must be exercised when modifying a signal with the forced input/output function.
If the forced input/output function is used incorrectly, the operation of the machine may be unpredictable. Never use this function when persons are near the machine. ce nt e CAUTION 1 When shipping a machine, disable this function. 2 This function is disabled by extracting the edit card or setting the PROGRAM ENABLE setting parameter to NO. 3 The setting of OVERRIDE is not maintained when the power is turned off. When the power is turned on again, the setting of OVERRIDE is cleared for all X and Y addresses. Screen Display (1) Soft keys nc 3.9.3 w w w .c TITLE STATUS ALARM TRACE SEARCH FORCE SEARCH ON OVRSET OVRRST EXEC CANCEL STATUS OFF INIT (2) Details of the soft keys (a) [SEARCH] Searches for an address to be displayed. (b) [ON] (Note 1) Forcibly changes the value of a signal to 1. 379 3.
PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 (c) [OFF] (Note 1) Forcibly changes the value of a signal to 0. (d) [FORCE]/[STATUS] (Note 1) Switches between the status display screen and forced input/output screen. (e) [OVRSET] (Note 2) Sets OVERRIDE for a signal.
(f) [OVRRST] (Note 2) Clears an OVERRIDE setting for a signal. om (g) [INIT] (Note 2) Clears OVERRIDE for all the X and Y areas. w w w .c nc ce nt e r.c NOTE 1 The soft key is enabled in FORCING mode. 2 The soft key is enabled in OVERRIDE mode. 380 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 (3) Forced input/output screens FORCING Mode Status Screen PMC SIGNAL STATUS 6 0 0 0 0 0 0 0 1] [4 0 0 0 0 0 1 0 0 3 0 0 0 0 1 0 0 0] [2 0 0 0 1 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0] [FORCE] ce nt e [SEARCH] [5 0 0 0 0 0 0 1 0 om 7 0 0 0 0 0 0 0 0 r.c NO. X0000 X0001 X0002 X0003 X0004 X0005 X0006 X0007 MONIT RUN FORCING Mode Setting Screen PMC SIGNAL FORCING 7 0 0 0 0 0 0 0 0 [SEARCH] [w w w .c nc NO. X0000 X0001 X0002 X0003 X0004 X0005 X0006 X0007 381 6 0 0 0 0 0 0 0 1 5 0 0 0 0 0 0 0 1 0 ON] [4 0 0 0 0 0 1 0 0 OFF 3 0 0 0 0 1 0 0 0] [MONIT RUN 2 0 0 0 1 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0] [STATUS] 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 OVERRIDE Mode Status Screen NO. X0000 X0001 X0002 X0003 X0004 X0005 X0006 X0007 7 0 0 0 0 0 0 0 0 6 0 0 0 0 0 0 0 0 >1 5 0 0 0 0 0 0 >1 0 4 0 0 0 0 0 >1 0 0] [3 0 0 0 0 >1 0 0 0] [MONIT RUN 2 0 0 0 >1 0 0 0 0 1 0 0 >1 0 0 0 0 0 0 0 >1 0 0 0 0 0 0] [FORCE] r.c [SEARCH] [OVERRIDE om PMC SIGNAL STATUS ce nt e OVERRIDE Mode Setting Screen PMC SIGNAL FORCING 7 0 0 0 0 0 0 0 0 nc NO. X0000 X0001 X0002 X0003 X0004 X0005 X0006 X0007 ON 5 0 0 0 0 0 0 >1 0] [OVRSET] [OVRRST] [w w w .c [SEARCH] [6 0 0 0 0 0 0 0 1>1 382 OVERRIDE 4 0 0 0 0 0 >1 0 0 OFF 3 0 0 0 0 0>1 0 0 0] [ ] [MONIT RUN 2 0 0 0 1>1 0 0 0 0 1 0 0 1>1 0 0 0 0 0 0 0 0>1 0 0 0 0 0 0] [STATUS] INIT] [ ] PMC OPERATION (CRT/MDI) B–61863E/10 3.9.4 Modifying the Values of Signals by Forced Input/Output 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) The method described below applies to both FORCING and OVERRIDE modes. (1) Modifying signal values on a bit–by–bit basis Position the cursor to a desired input bit, then enter a desired value by using one of the following three methods: (a) Enter 1, then press the INPUT key or the [ON] soft key. The signal is forcibly turned on. om (b) Enter 0, then press the INPUT key or the [OFF] soft key. The signal is forcibly turned off. (c) Press the INPUT key.
The on/off state of the signal is reversed. 7 0 6 0 MONIT RUN 5 0 4 0 3 0 2 0 1 0 0 0 ce nt e NO. X0000 OVERRIDE r.c PMC SIGNAL FORCING (2) Modifying signal values on a byte–by–byte basis Move the cursor to a desired input byte, then enter a desired value by using one of the following three methods: (a) Enter a binary number of no more than 8 digits, then press the INPUT key. (If an entered number is shorter than 8 digits, the number is entered starting from bit 0.) Example: When 100 is entered, the number is entered at the following bit positions: nc Bit No. 76543210 00000100 (c) Press the [OFF] soft key. All bits of the specified byte are set to 0. PMC SIGNAL FORCING NO. X0000 7 0 w w w .c (b) Press the [ON] soft key. All bits of the specified byte are set to 1. 383 6 0 OVERRIDE 5 0 4 0 3 0 MONIT RUN 2 0 1 0 0 0 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) PMC OPERATION (CRT/MDI) B–61863E/10 3.9.5 Setting/Clearing OVERRIDE PMC SIGNAL FORCING 7 0 6 0 OVERRIDE 5 0 4 0 3 0 MONIT RUN 2 0 1 0 r.c NO. X0000 om (1) Setting OVERRIDE OVERRIDE can be set as described below. By using the [OVRSET] soft key, place the desired bit/byte in the OVERRIDE state. (a) Setting OVERRIDE on a bit–by–bit basis Move the cursor to the desired bit, then press the [OVRSET] soft key. 0 0 Then, the display changes as shown below. PMC SIGNAL FORCING 7 0 6 0 5 0 4 0 MONIT RUN 3 2 0 0> 0 ce nt e NO. X0000 OVERRIDE 1 0 0 0 (b) Setting OVERRIDE on a byte–by–byte basis Position the cursor to the desired byte, then press the [OVRSET] soft key. PMC SIGNAL FORCING 7 0 nc NO. X0000 6 0 OVERRIDE 5 0 4 0 3 0 MONIT RUN 2 0 1 0 0 0 w w w .c Then, the display changes as shown below. PMC SIGNAL FORCING NO. 7 X0000 0>0 6 0>0 5 0>0 OVERRIDE 4 0>0 3 0>0 MONIT RUN 2 0>0 1 0>0 0 0>0 When OVERRIDE is set, the signal resumes the state existing before OVERRIDE setting. When OVERRIDE is set for a pulse signal, for example, the signal state existing when the [OVRSET] soft key is pressed is overridden. Each bit for which OVERRIDE is set has ”>” added in the status display. In OVERRIDE mode, those signal states that are not in the OVERRIDE state are also displayed. For an X signal: (Input signal from the machine) 0 > 1 (input signal to the ladder) For a Y signal (Output signal from the ladder) 0 > 1 (output signal to the machine) 384 PMC OPERATION (CRT/MDI) B–61863E/10 3. PMC I/O SIGNAL DISPLAY AND INTERNAL RELAY DISPLAY (PMCDGN) (2) Clearing OVERRIDE By using the [OVRRST] soft key, clear the OVERRIDE state of the desired bit/byte. This operation returns the state of a signal to the state existing before OVERRIDE setting. (a) Clearing OVERRIDE on a bit–by–bit basis Position the cursor to the desired bit, then press the [OVRRST] soft key. NO. X0000 7 0 6 0 OVERRIDE 5 0 4 0 MONIT RUN om PMC SIGNAL FORCING 3 2 0 1> 0 1 0 0 0 PMC SIGNAL FORCING 7 0 6 0 OVERRIDE 5 0 4 0 3 0 ce nt e NO. X0000 r.c Then, the display changes as shown below. MONIT RUN 2 1 1 0 0 0 (b) Clearing OVERRIDE on a byte–by–byte basis Position the cursor to the desired byte, then press the [OVRRST] soft key. PMC SIGNAL FORCING nc NO. 7 X0000 1>0 6 0>1 OVERRIDE 5 1>0 4 0>0 3 1>0 MONIT RUN 2 1>1 1 1>0 0 0>1 w w w .c Then, the display changes as shown below.
PMC SIGNAL FORCING NO. X0000 7 1 6 0 OVERRIDE 5 1 4 0 3 1 MONIT RUN 2 1 1 1 0 0 (c) Clearing OVERRIDE for all X and Y areas Press the [INIT] soft key. Then, the message ”CLEAR OVERRIDES OK?” is displayed. By using the soft key [EXEC]/[CANCEL], execute or cancel the clearing of OVERRIDE. 385 4. PMC PARAMETERS SETTING AND PMC OPERATION (CRT/MDI) DISPLAY (PMCPRM) PMC PARAMETERS SETTING AND DISPLAY (PMCPRM) w w w .c nc ce nt e r.c om 4 B–61863E/10 386 PMC OPERATION (CRT/MDI) B–61863E/10 4.1 4. PMC PARAMETERS SETTING AND DISPLAY (PMCPRM) Parameters of TIMER, COUNTER, KEEP RELAY and DATA TABLE, which are nonvolatile, are set and displayed with CRT/MDI panel. To use this function, press the soft key [PMCPRM] of PMC basic menu screen. OUTLINE w w w .c nc ce nt e r.c om NOTE The address and contents of the nonvolatile memory are described in 3.5 to 3.8 of I–3. ”ADDRESS” and I–6.”NONVOLATILE MEMORY”.
387 4. PMC PARAMETERS SETTING AND PMC OPERATION (CRT/MDI) DISPLAY (PMCPRM) B–61863E/10 4.2 INPUT PMC PARAMETERS FROM MDI PANEL 1 Place the sequence program in the STOP state. 2 When the sequence program is in the RUN state, perform the setting below. (1) Set NC to ”MDI” mode or ”Emergency Stop” status. TIMER Ę COUNTER Ę KEEP RELAY Ę DATA TABLE Ę 3 KEY4 Ę : Alternative Ę : Alternative r.c PWE om (2) Set ”PWE” of NC setting screen or Program Protect Signal(”KEY4”) to 1. (See the following table.) Press the following soft keys to select the screens. TIMER screen COUNTER screen KEEP RELAY screen DATA TABLE screen ce nt e [TIMER] : [COUNTR] : [KEEPRL] : [DATA] : By using cursor keys, move cursor to the position for setting value. 5 Press the INPUT key after typing the value. 6 Set ”PWE” or ”KEY4” to 0 after setting value. 1 This function is effective on the screen of TIMER, COUNTER, KEEP RELAY, and DATA TABLE.
nc 4.2.1 4 Multiple Data Input Up to 10 data can be inputted at once. 3 The cursor is moved to the final data position of inputted data. .c 2 w w w (1) Input method D ” ; (EOB)” is used for separating data. Press the INPUT key after typing “100; 200; 300”. D ” ; =” is used for inputting the same value as preceding data. Press the INPUT key after typing “100; =; =; 200; =”, and it becomes “100, 100, 100, 200, 200”. D ” ; ; ” is used for skipping an input address. Press the INPUT key after typing “100; ; 100”. The second data is not inputted.
388 PMC OPERATION (CRT/MDI) B–61863E/10 4. PMC PARAMETERS SETTING AND DISPLAY (PMCPRM) 4.3 SETTING AND DISPLAY SCREEN Timer Screen (TIMER) The TIMER times of the functional instruction TMR(SUB 3) are set and displayed on this screen.
om 4.3.1 Page No.(Change pages with the page keys.) The TIMER No.s used by TIMER instruction The addresses refered by sequence program ADDRESS T00 T02 T04 T06 T08 T10 T12 T14 T16 T18 #001 DATA 2016 48 960 1008 0 0 96 0 8 16 MONIT RUN NO. 11 12 13 14 15 16 17 18 19 20 ADDRESS T20 T22 T24 T26 T28 T30 T32 T34 T36 T38 DATA 1000 8 0 32 0 0 2000 0 8 10000 r.c NO. 01 02 03 04 05 06 07 08 09 10 (TIMER) TIMER times(See the following table.) TIMER No.s 1 to 8 Minimum time Maximum time 48 (ms) 1572.8 (s) 8 (ms) 262.136 (s) ce nt e PMC PRM] w w w .c nc [TIMER] [COUNTR] [KEEPRL] [DATA] [389 9 to 40 or 9 to 150 (*1) *1 The usable numbers vary from one model to another. For details, see Section 5.4 in Part I. 4. PMC PARAMETERS SETTING AND PMC OPERATION (CRT/MDI) DISPLAY (PMCPRM) 4.3.2 B–61863E/10 The maximum(PRESET) values and CURRENT values of the functional instruction CTR(SUB 5) are set and displayed on this screen. Counter Screen (COUNTR) The COUNTER No.s used by CTR instruction The addresses refered by sequence program Page No.(Change pages with the page keys.) #001 ADDRESS C00 C04 C08 C12 C16 C20 C24 C28 C32 C36 MONIT RUN PRESET 4 4 4 5 4 545 5 6 6 6 CURRENT 1 2 3 4 5 6 3 2 1 4 [TIMER] [COUNTR] [KEEPRL] [DATA] [Keep Relay (KEEPRL) (KEEP RELAY) DATA 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 w w NO. ADDRESS 01 K00 02 K01 03 K02 04 K03 05 K04 06 K05 07 K06 08 K07 09 K08 10 K09 w] The KEEP RELAYs and the Data for Controlling nonvolatile memory are set and displayed on this screen. .c PMC PRM 0-9999 in BCD(0-32767 in Binary) can be set as the PRESET and CURRENT values. nc 4.3.3 The CURRENT values of COUNTER r.c NO. 01 02 03 04 05 06 07 08 09 10 (COUNTER) ce nt e PMC PRM om The maximum(PRESET) values of COUNTER (The minimum values are specified in CTR instruction.) MONIT RUN The address used by sequence program NO. ADDRESS DATA 11 K10 00000000 12 K11 00000000 13 K12 00000000 14 K13 00000000 15 K14 00000000 16 K15 00000000 17 K16 00000000 18 K17 00000110 19 K18 00000000 20 K19 00000000 J : This area is reserved for special use. (Note) [TIMER] [COUNTR] [KEEPRL] [DATA] [ ] 390 PMC OPERATION (CRT/MDI) B–61863E/10 4. PMC PARAMETERS SETTING AND DISPLAY (PMCPRM) NOTE 1 The Data for Controlling Nonvolatile Memory(K16) Refer to I–6.1(4)”Nonvolatile Memory Control”. 2 The Data for PMC Management Software(K17,18,19) Be careful of using the following KEEP RELAYs, because they are used by PMC Management Software.
Model PA1 PMC control software data 1 K17 PMC control software data 2 K18 Not used K19 PMC control software data 2 K18 K19 RA2 RA3/ RA5 K17 K17 K17 K18 K18 K18 K19 K19 K19 ce nt e Not used RB RB2 RB3/ RB5 RB4/ RB6 PMC control software data 1 K17 K17 K17 K900 PMC control software data 2 K18 K18 K18 K901 Not used K19 K19 K19 K902 to K909 Not used K903 to K909 Model RC RC3 RC4 PMC control software data 1 K17 K17 K900 PMC control software data 2 K18 K18 K901 PMC control software data 3 K19 K19 K902 nc Model NB NB2 PMC control software data 1 K17 K900 PMC control software data 2 K18 K901 PMC control software data 3 K19 K902 .c w K17 RA1 PMC control software data 1 w PA3 r.c Model w om The Data for PMC Management Software Model Not used K903 to K909 PMC control software data 1 (K17 or K900) K17 or K900 #7 DTBLDSP #7 DTBLDSP #6 #5 ANASTAT TRCSTART #4 #3 #2 MEMINP SELCTMDL AUTORUN #1 PRGRAM #0 LADMASK 0 : The PMC parameter data table control screen is displayed. 1 : The PMC parameter data table control screen is not displayed. #6 ANASTAT 0 : In the function for displaying signal waveforms, sampling starts when the [START] soft key is pressed. 1 : In the function for displaying signal waveforms, sampling starts automatically when the power is turned on. 391 4.
PMC PARAMETERS SETTING AND PMC OPERATION (CRT/MDI) DISPLAY (PMCPRM) B–61863E/10 * This bit is effective only for applicable models specified in 3.6, “Function for Displaying Signal Waveforms (ANALYS),” in Part II. #5 TRCSTAT 0 : In the signal trace function, tracing starts when the [EXEC] soft key is pressed. 1 : In the signal trace function, tracing starts automatically when the power is turned on. 0 : Data cannot be entered in the memory content display function. #4 MEMINP om 1 : Data can be entered in the memory content display function. * This bit is effective only for applicable models specified in 3.5, “Display the Contents of Memory (M.SRCH),” in Part II. r.c #3 SELCTMDL 0 : The sequence program stored in ROM (EPROM) is enabled. ce nt e 1 : The sequence program stored in the RAM module or ROM module (only for PMC-RB2/RB3) is enabled. * This bit enables either the EPROM module or ROM/RAM module when both modules are provided.
It is effective for the PMC-RA1, RA2, RA3, RB, RB2, and RB3. (It is not effective for the Series 20 or Series 16/18 MODEL-B.) #2 AUTORUN 0 : In RAM operation, a sequence program is not executed when the power is turned on. nc 1 : In RAM operation, a sequence program is executed automatically when the power is turned on (as in ROM operation). 0 : The sequence program is executed automatically when the power is turned on. 1 : The sequence program is executed when the [RUN] soft key is pressed. 0 : The built-in programmer function is not operated. (The programmer menu is not displayed, either.) #1 PRGRAM #0 LADMASK 1 : The built-in programmer function is operated. (The programmer menu is displayed.) 0 : Ladder dynamic display (PCLAD) is performed. 1 : Ladder dynamic display (PCLAD) is not performed. w w w .c * For the PMC of the Series 16/18 MODEL-B/C, Series 16i/18i/21i–MODEL A, this bit has the following meanings. 392 PMC OPERATION (CRT/MDI) B–61863E/10 4. PMC PARAMETERS SETTING AND DISPLAY (PMCPRM) PMC control software data 2 (K18 or K901) K18 or K901 #7 #6 IGNDINT #7 IGNDINT #5 #4 #3 #2 CHKPRTY CALCPRTY TRNSRAM TRGSTAT #1 #0 DBGSTAT IGNKEY 0 : When the screen is switched to the PCMMDI screen, the CRT is initialized. 1 : When the screen is switched to the PCMMDI screen, the CRT is not initialized. #5 CHKPRTY om * The flag is used to determine whether PMC control software initializes the CRT when the screen is switched to the PCMMDI screen. Design application software so that the CRT is initialized when this flag is on. 0 : The parity check is performed for the system ROM and program ROM/RAM. r.c 1 : The parity check is not performed for the system ROM and program ROM/RAM. #4 CALCPRTY 0 : The built-in programmer function performs RAM parity calculation.
ce nt e 1 : The built-in programmer function does not performs RAM parity calculation. 0 : A ladder program is not automatically sent to the backup RAM after on-line editing is completed. #3 TRNSRAM 1 : A ladder program is automatically sent to the backup RAM after on-line editing is completed. 0 : The trigger stop function does not automatically start when the power is turned on. #2 TRGSTAT w w w .c nc #1 DBGSTAT 1 : The trigger stop function automatically starts when the power is turned on. 0 : In the C language debug function, the break processing does not automatically start when the power is turned on. 1 : In the C language debug function, the break processing automatically starts when the power is turned on. * This flag is effective for the PMC-RC/RC3. #0 IGNKEY 0 : Function keys are enabled when a user program displays the user screen. 1 : Function keys are disabled when a user program displays the user screen. * This flag is effective for the PMC-RC/RC3/RC4/NB/NB2. When this bit is set to 1 of the user screen, the screen cannot be switched to the NC screen using function keys. A program which always sets this bit to 0 or which changes the screen to the NC screen is required. 393 4. PMC PARAMETERS SETTING AND PMC OPERATION (CRT/MDI) DISPLAY (PMCPRM) B–61863E/10 PMC control software data 3 (K19 or K902) #7 K19 or K902 #6 #5 #4 #3 #2

#1 #0 C-REJECT FROM-WRT LCD–MONO #3 LCD–MONO 0 : Ladder–related display is brightness–adjusted when a monochrome LCD is used with the VGA setting. om #1 C-REJECT 1 : Ladder–related display is not brightness–adjusted but is displayed in reverse video when a monochrome LCD is used with the VGA setting. 0 : A C-language program is activated. 1 : A C-language program is forcibly not activated.
r.c * The flag is effective for the PMC-RC/RC3/RC4. #0 FROM-WRT 0 : The program is not automatically written to F–ROM. ce nt e 1 : After a lodder program on C program has been edited, the program is automatically written to F–ROM. CAUTION Be sure to set bits not used in the PMC control software data to 0. #7 nc In case of PMC–PA1/PA3 on Power Mate K17 DTBLDSP .c #7 DTBLDSP w w w #6 ANASTAT #6 #5 ANASTAT TRCSTART #4 MEMINP #3 #2 #1 #0 AUTORUN PRGRAM LADMASK 0 : The PMC parameter data table control screen is displayed. 1 : The PMC parameter data table control screen is not displayed.
0 : Sampling is started with the signal waveform display function by using the execution soft key. 1 : Sampling is started with the signal waveform display function, automatically after the power is turned on. (This setting cannot be used with the Power Mate–D/F.) 0 : Trace operation is started with the signal trace function by using the trace execution soft key. #5 TRCSTAT 1 : Trace operation is started with the signal trace function, automatically after the power is turned on. 0 : Data cannot be entered with the memory contents display function. #4 MEMINP 1 : Data can be entered with the memory contents display function. (This setting cannot be used with the Power Mate–D/F.) 394 PMC OPERATION (CRT/MDI) #2 AUTORUN 4. PMC PARAMETERS SETTING AND DISPLAY (PMCPRM) 0 : The sequence program is executed automatically after the power is turned on. (This setting cannot be used with the Power Mate–D/F.) 1 : The sequence program is executed by using the sequence program execution soft key. 0 : The built–in programmer function is not operated.
(The programmer menu is not displayed either.) #1 PRGRAM #0 LADMASK 1 : The built–in programmer function is operated. (The programmer menu is displayed.) 0 : Ladder dynamic display (PMCLAD) is performed. om B–61863E/10 #7 #6 K18 r.c 1 : Ladder dynamic display (PMCLAD) is not performed. #5 #4 #3 #2 #1 #0 CHKPRTY CALCPRTY TRANSRAM TRGSTAT 0 : System ROM and program ROM/RAM parity checks are performed. ce nt e #5 CHKPRTY 1 : System ROM and program ROM/RAM parity checks are not performed. #4 CALCPRTY 0 : A RAM parity calculation is performed with the built–in programmer function. nc #3 TRNSRAM .c #2 TRGSTAT 1 : Upon the completion of online editing, the ladder program is automatically transferred to RAM for editing. (This setting cannot be used with the Power Mate–D/F.) 0 : When the power is turned on, the trigger stop function is not started automatically. 1 : When the power is turned on, the trigger stop function is started automatically.
(This setting cannot be used with the Power Mate–D/F.) w w w 1 : A RAM parity calculation is not performed with the built–in programmer function. 0 : Upon the completion of online editing, the ladder program is not automatically transferred to RAM for editing. #7 K19 #6 #5 #4 #3 #2 #1 #0 FROM–WRT #0 FROM–WRT 0 : After a ladder is edited, the ladder is not automatically written to F–ROM.
1 : After a ladder is edited, the ladder is automatically written to F–ROM. (This setting cannot be used with the Power Mate–D/F.) 395 4. PMC PARAMETERS SETTING AND PMC OPERATION (CRT/MDI) DISPLAY (PMCPRM) B–61863E/10 CAUTION The unused area of the data for the PMC management software must always be set to 0. 4.3.4 Data Table (DATA) om DATA TABLE consists of two screens, that is, Data Table Controlling Data screen and Data Table screen. (1) Data Table Controlling Data Screen Data Table Controlling Data Screen for controlling Data Table is displayed by pressing the soft key [DATA]. r.c Group No.s The top address of Data Table Table Parameters(Note) Page No. (Change pages with the page keys) PMC DATA TBL CONTROL #001 GROUP TABLE COUNT = Á ADDRESS PARAMETER TYPE D0000 00000000 0 D0020 00000010 0 D0101 00000001 1 D0301 00000000 2 D0501 00000011 0 D0506 8 00000000 0 D0506 Á 00000000 1 D0506 Ź 00000000 2 w w .c [G.DATA] [G.CONT] [NO.SRH] [w MONIT RUN The number of group of Data Table NO. OF DATA 20 81 100 50 5 10 10 10 nc NO. 001 002 003 004 005 006 007 008 ce nt e Data length (0:1byte, 1:2bytes, 2:4bytes)] [INIT] The data numbers of each Data Table * You can set the same address in other groups. You can initialize the Data Table setting data. The initial data is as follows. PMC DATA TBL CONTROL #001 MONIT RUN GROUP TABLE COUNT = 1 NO. 001 ADDRESS PARAMETER D0000 00000000 TYPE NO. 0 OF DATA 1860 * 3000:PMC-RB3/RB5 002 /RC/RC3/NB * 8000:PMC-RB4/RB6/NB2 Press this key after typing the group No., and the cursor is moved to the group. Press this key after typing the number of group, and the Group Table Count is set. You can change the screen to Data Table.
396 PMC OPERATION (CRT/MDI) B–61863E/10 4. PMC PARAMETERS SETTING AND DISPLAY (PMCPRM) NOTE Table Parameter #7 #6 #5 #4 #3 #2 #1 #0 0 : Binary 1 : BCD om 0 : Available to input 1 : Unavailable to input (Protection mode) 0 : Binary of BCD (The bit 0 is valid 1 : Hexadecimal (The bit 0 is invalid.) r.c (2) Data Table Screen ce nt e If the Data Table Controlling Data is set, Data Table Screen is displayed by pressing the soft key [G.DATA]. Group No.s Page No. (Change pages with the page keys) PMC PRM (DATA) nc ADDRESS D0000 D0001 D0002 D0003 D0004 D0005 D0006 D0007 D0008 D0009 MONIT RUN .c NO. 001 002 003 004 005 006 007 008 009 010 001/001 w w w [C.DATA] [G-SRCH] [SEARCH] [ ] [DATA 10 48 5 64 0 48 10 1 1 The address used by sequence program] Press this key after typing the address (ex.D8;D can be omitted), and the cursor is moved to the address in the current group. If you search the Data Table in the other group, press this key after typing the group No., and the cursor is moved to the top of the address in the specified group. You can change the screen to Data Table Controlling Data. 397 4. PMC PARAMETERS SETTING AND PMC OPERATION (CRT/MDI) DISPLAY (PMCPRM) 4.4 B–61863E/10 Part of KEEP RELAY parameters can be set on SETTING Screen. Ę SETTING SCREEN : Can be used : Cannot be used PA1 PA3 RA1 RA2 RA3 RA5 RB RB2 RB3 RB4 RB5 RB6 RC RC3 RC4 NB NB2 Ę Δ Ę Δ Ę Ę Ę Ę Δ Ę Δ Ę Ę Ę r.c om NOTE Δ: Can be used for the specific series of CNC. (Series 16 : B005/11 to, B105/08 to, B305/04 to, B009/03 to, All serieses of model C) (Series 18 : BD03/12 to, BE03/09 to, BG23/03 to, BG03/06 to, BD09/02 to, BE09/14 to, All serieses of model C) PMC–PA3 can be used only with Power Mate–H. D The display items are different according to the type of CNC. D The parameter is set by a soft key or the key with 0 or 1. ce nt e D Once an item has been set, the cursor moves to the next item. [PMC-RA1/RA3/RB/RB3/RB4 on SETTING screen] PMC PRM (SETTING) MONIT RUN PROGRAMMER ENABLE = 0(0:NO 1:YES) (K17. 1) LADDER START (RAM) = 0(0:MANUAL 1:AUTO) (K17. 2) SELECT ROM/RAM = 0(0:ROM 1:RAM) (K17. 3) SIGNAL TRACE START = 0(0:MANUAL 1:AUTO) (K17. 5) DATA TBL CNTL SCREEN = 0(0:YES 1:NO) nc (K17. 7) SIGNAL TRIGGER START = 0(0:MANUAL 1:AUTO) (K18.
2) [NO] [YES] [ ] [ ] [(K18. 3)] * The bracketed addresses show the related KEEP RELAYs. w w w .c TRANS LADDER(ONLEDT) = 0(0:MANUAL 1: AUTO) 398 PMC OPERATION (CRT/MDI) B–61863E/10 4. PMC PARAMETERS SETTING AND DISPLAY (PMCPRM) [PMC–PA3 on SETTING screen] PMC PRM (SETTING) MONIT RUN PROGRAMMER ENABLE = 0(0:NO 1:YES) (K17. 1) LADDER START = 0(0:AUTO 1:MANUAL) (K17. 2) RAM WRITE ENABLE = 0(0:NO 1:YES) (K17. 4) SIGNAL TRACE START = 0(0:MANUAL 1:AUTO) (K17. 5) DATA TBL CNTL SCREEN = 0(0:YES 1:NO) om (K17. 7) SIGNAL TRIGGER START = 0(0:MANUAL 1:AUTO)] [YES] [ ] [ ] [ ] r.c [NO (K18. 2) * The bracketed address show the related KEEP RELAYs ce nt e [PMC-RC/RC3/RC4 on SETTING screen] PMC PRM (SETTING) nc PROGRAMMER ENABLE = LADDER START (RAM) = RAM WRITE ENABLE = SIGNAL TRACE START = SIGNAL ANALYS START = DATA TBL CNTL SCREEN = FUNC KEY INP(CUSTOM) = DEBUG FUNC START = SIGNAL TRIGGER START = TRANS LADDER (ONLEDT)= INITPMC-MDI SCREEN =] [YES] [0(0:NO 1:YES) 0(0:MANUAL 1:AUTO) 0(0:NO 1:YES) 0(0:MANUAL 1:AUTO) 0(0:MANUAL 1:AUTO) 0(0:YES 1:NO) 0(0:AVAL 1:IGNORE) 0(0:MANUAL 1:AUTO) 0(0:MANUAL 1:AUTO) 0(0:MANUAL 1:AUTO) 0(0:YES 1:NO)] [ ] [(K17. 1) (K17. 2) (K17. 4) (K17. 5) (K17. 6) (K17. 7) (K18. 0) (K18. 1) (K18. 2) (K18.
3) (K18. 7)] * The bracketed addresses show the related KEEP RELAYs. w w w .c [NO MONIT RUN 399 4. PMC PARAMETERS SETTING AND PMC OPERATION (CRT/MDI) DISPLAY (PMCPRM) B–61863E/10 [PMC–NB/NB2 on SETTING screen] PMC PRM (SETTING) MONIT RUN] [YES] [ ] [NB NB2 (K17. 1, K900.1) (K17. 2, K900.2) (K17. 4, K900.4) (K17. 5, K900.5) (K17. 6, K900.6) (K17.
7, K900.7) (K18. 0, K901.0) (K18. 1, K901.1) (K18. 2, K901.2) (K18. 3, K901.3) (K18.
7, K901.7) (K19. 0, K902.0) (K19. 1, K902.1) om (0:NO 1:YES) (0:MANUAL 1:AUTO) (0:NO 1:YES) (0:MANUAL 1:AUTO) (0:MANUAL 1:AUTO) (0:YES 1:NO) (0:AVAL 1:IGNORE) (0:MANUAL 1:AUTO) (0:MANUAL 1:AUTO) (0:MANUAL 1:AUTO) (0:YES 1:NO) (0:NO 1:YES) (0:NO 1:YES) (0:GRAPHIC 1:TEXT) (0:AUTO 1:MANUAL) (0:EXPAND 1:STANDARD)] [ ] ce nt e [NO 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 r.c PROGRAMMER ENABLE = AUTOMATIC LADDER START = RAM WRITE ENABLE IN [M.SRC] = SIGNAL TRACE START = SIGNAL ANALYSIS START = DATA TABLE CONTROL SCREEN = NC/PC KEY EFFECTIVE = DEBUG FUNCTION START = SIGNAL TRIGGER START = TRANSFER LADDER (ONLINE–EDIT) = INITIALIZE PMC–MDI SCREEN = WRITE TO F–ROM (EDIT) = REJECT LANGUAGE = SIGNAL ANALYSIS DISPLAY MODE = SPECIFY NC WINDOW FORMAT = NC WINDOW FORMAT (TOOL DATA) = * The bracketed addresses show the related KEEP RELAYs. SIGNAL TRIGGER ENABLE Displayed in case of PMC–NB(4047). Stop function of ladder diagram display by trigger of signal is set. The trigger stop function can be used by selecting "YES" ,and turning off and on the power. nc WRITE TO F–ROM (EDIT) .c Setting to write the LADDER data in F–ROM, when the edit of LADDER ends. When you select "YES" and then get out of the EDIT screen, a message confirming if you write to F–ROM is displayed. w w w REJECT LANGUAGE It is setting of the start of the program of C language. When "YES" is selected, the program of C language is not started.
SIGNAL ANALYSIS DISPLAY MODE The display form in the signal waveform display function is set. The display form can be selected. Select "TEXT" and it is displayed by the character. Select "GRAPHIC" and it is displayed by the line.
SPECIFY NC WINDOW FORMAT The form in functional instruction WINDR and WINDW are set. When "AUTO" is selected, the format is automatically distinguished by the state of bit 4 of NC parameter 7401. When "MANUAL" is selected, the format is selected by "NC WINDOW FORMAT (TOOL DATA)". 400 PMC OPERATION (CRT/MDI) B–61863E/10 4.
PMC PARAMETERS SETTING AND DISPLAY (PMCPRM) NC WINDOW FORMAT (TOOL DATA) The format in functional instruction WINDR and WINDW are set. When "MANUAL" is selected by "SPECIFY NC WINDOW FORMAT", this item is effective. The window instruction of a new format can be used by selecting "EXPAND". (The same meaning as bit 4 of NC parameter 7401 is 1.) An old window instruction can be used by selecting "STANDARD" 4.4.1 om (The same meaning as bit 4 of NC parameter 7401 is 0.) Enable or disable the following function and mode: D Language–by–language message function r.c Other Setting Screens D OVERRIDE mode of the forced input/output function ce nt e (1) Setting screen display Each setting screen can be displayed by pressing the [NEXT] or [PREV] soft key on the setting parameter screen. [YES] [NO] [ ] [PREV] [NEXT]] [ ] [PREV] [NEXT]] [ ] [PREV] [NEXT] nc Setting parameter [INPUT] [w w w .c Setting parameter (language–by–language message function) [YES] [NO Setting parameter (OVERRIDE mode of the forced input/output function) Each setting parameter can be set when the respective conditions are satisfied.
(a) Language–by–language message function D The PROGRAM ENABLE setting parameter is set to YES (bit 1 of K17 or bit 1 of K900 is set to 1). D The PMC model is RB6. (b) OVERRIDE mode of the forced input/output function D The PROGRAM ENABLE setting parameter is set to YES (bit 1 of K17 or bit 1 of K900 is set to 1) 401 4.
PMC PARAMETERS SETTING AND PMC OPERATION (CRT/MDI) DISPLAY (PMCPRM) B–61863E/10 D The PMC model is RA5, RB5, or RB6. D The editing function is provided. (2) Setting operations (a) Setting the language–by–language message function Position the cursor to the each setting item with the arrow keys, enter the desired data, then press the [INPUT] soft key.
MONIT RUN om PMC PRM (MESSAGE SHIFT) ce nt e r.c MESSAGE SHIFT VALUE = 0 MESSAGE SHIFT START ADDRESS = A0000.0 [INPUT] [ ] [ ] [PREV] [NEXT] (i) Parameters D MESSAGE SHIFT VALUE Enter a desired message display request bit shift amount. A value from 0 to 999 can be entered. The initially displayed value is 0. Entered data is maintained even after the power is turned off. w w w .c nc D MESSAGE SHIFT START ADDRESS Enter a shift start bit address in the message display request bit area. An address A value can be entered.
The initially displayed value is A0.0. Entered data is maintained even after the power is turned off. CAUTION Data entered for MESSAGE SHIFT START ADDRESS is valid only when a value other than 0 is entered for MESSAGE SHIFT VALUE. (b) OVERRIDE mode of the forced input/output function D Set OVERRIDE mode by using the soft key or by entering 0 or 1 followed by the key. 402 PMC OPERATION (CRT/MDI) B–61863E/10 4.
PMC PARAMETERS SETTING AND DISPLAY (PMCPRM) PMC PRM (OVERRIDE) MONIT RUN NO] [YES] [(0:NO 1:YES)] [PREV] [NEXT] r.c [= 0 om OVERRIDE ENABLE OVERRIDE ENABLE 0 : OVERRIDE mode is disabled. 1 : OVERRIDE mode is enabled. w w w .c nc ce nt e CAUTION If the setting of this parameter is modified, the new setting becomes effective when the power is next turned on.
After this parameter has been modified, the power must be turned off then back on. 403 4. PMC PARAMETERS SETTING AND PMC OPERATION (CRT/MDI) DISPLAY (PMCPRM) 4.5 B–61863E/10 If you make a keyboard without cursor keys, you must move cursor by searching the address or so. In case of TIMER,COUNTER and KEEP RELAY, press the soft key [TIMER],[COUNTR] or [KEEPRL] after typing the address(Ex.1,2). In case of Data Table Controlling Data, press the soft key [DATA](or [NO.SRH] if Data Table screen has already been displayed) after typing the group No.(Ex.3).
In case of the Data Table, press the soft key [SEARCH] after typing the address in the Data Table screen which contains the address you want to search(Ex.4). NOTE om Ex.1) In case of setting the TIMER NO.11(ADDRESS T20) 1 Press the soft key [TIMER] after typing T20(or T21;T can be omitted.). 2 Press the INPUT key after typing the value. PRESET ³ Press the soft key [COUNTER] after typing C4 (or C5;C can be omitted). ce nt e 1 r.c Ex.2) In case of setting PRESET and CURRENT values of the COUNTER NO.02(ADDRESS C04) CURRENT ³ Press the soft key [COUNTER] after typing C6 (or C7;C can be omitted). 2 Press the INPUT key after typing the value. nc CAUTION It is not the number(NO.) but the address(ADDRESS) that you type in searching. Ex.3) In case of the ADDRESS,PARAMETER,TYPE and NO. OF DATA of the Data Table Controlling Data NO.002. Press the soft key [NO.SRH] after typing 2, and the cursor is moved to the ADDRESS position. 2 Press the INPUT key after typing the ADDRESS(ex.D20;D must not be omitted), and the cursor is automatically moved to the next position(PARAMETER). The cursor is moved only by pressing the INPUT key. 3 In the same way, set the PARAMETER,TYPE and NO. OF DATA. If you finish setting the NO. OF DATA, the cursor is moved to the position(ADDRESS) in the same line. w w w .c 1 Ex.4) In case of setting D22 in the Data Table of the group 2 1 Press the soft key [G.DATA] on the Data Table Controlling Data screen, and the Data Table screen is displayed. 2 Press the soft key [G-SRCH] after typing 2 on the Data Table screen, and the Data Table of the group 2 is displayed. 3 Press the soft key [SEARCH] after typing D22(D can be omitted). 4 Press the INPUT key after typing the value. 404 PMC OPERATION (CRT/MDI) B–61863E/10 5 5. PMC LADDER DIAGRAM DISPLAY (PMCLAD) PMC LADDER DIAGRAM DISPLAY (PMCLAD) om Displaying the PMC ladder diagram on CRT/MDI panel is available. This ladder diagram display function offers functions effectively used for locating troubles in addition to the simple ladder diagram display. The following functions are done using the soft keys. r.c (1) Search and display of optional relay coil on ladder diagrams.
(2) Ladder diagram dynamic display. (3) Stop of ladder diagram display by trigger of signal (on or off). (4) Screen-dividing display. ce nt e (5) Monitor display of signal condition. (6) Monitor display of parameter in functional instructions. (7) ON LINE edit. For this operation, depress [PMCLAD] soft key of PMC basic menu to bring the following menu. A–TYPE nc PMCLAD PMCDGN PMCPRM RET .c TOP Sec. 5.5 Sec. 5.5 SRCH Sec. 5.5 F-SRCH w w w BOTTOM W-SRCH Sec. 5.5 N-SRCH Sec.5.5 NEXT ADRESS (SYMBOL) Sec. 5.5 Sec. 5.4 B–TYPE PMCLAD PMCDGN PMCPRM SEARCH ADRESS RET TRIGER WINDOW (SYMBOL) Sec. 5.5 DUMP Sec. 5.4 DPARA Sec. 5.6 Sec. 5.7 ONLEDT (NDPARA) Sec.
5.2 405 Sec. 5.3 Sec.
5.8 NEXT 5. PMC LADDER DIAGRAM DISPLAY PMC OPERATION (CRT/MDI) (PMCLAD) 5.1 B–61863E/10 The following functions can be done the ladder diagram display screen. LADDER DIAGRAM DISPLAY (a) Specified relay coil of ladder diagrams can be searched and displayed. (b) Ladder diagram dynamic display. The logical on-off states during a sequence program execution are displayed on a ladder diagram by changing the brightness in case of a monochrome CRT or by changing colors in case of a color CRT. om (1) Ladder diagram display Press [PMCLAD] soft key, then the ladder diagram will be displayed. Eight relay contacts and relay coils in total are displayed in the horizontal direction of the CRT screen. r.c If the number of relay contacts exceed the above value, they are displayed in 2 or more lines. Signal name LADDER MA MONIT RUN ce nt e (Within 6 characters) Address or symbol name SPDALM X2.4 MACHINE READY MACHINE ALARM nc 9 lines] [DUMP] [DPARA] [ ] .c [SEARCH] [ADRESS] [TRIGER] [WINDOW] [ ] [ONLEDT] [w w w 5.1 Ladder diagram display 406 Comments (within 30 characters) PMC OPERATION (CRT/MDI) B–61863E/10 5.
PMC LADDER DIAGRAM DISPLAY (PMCLAD) 5.2 Ladder diagram and signal status dump can be displayed together. DUMP DISPLAY ON LADDER DIAGRAM The dump is displayed over 2 lines at the last line of ladder diagram by pressing the [DUMP] soft key. PAGE°± keys or [SEARCH] soft key is used for changing of PMC address.
*TITLE DATA REMARKS 32 BYTES * X1000.0 X1000.1 NET 00001-00004 X1000.0 X1000.2 X1000.1 X1000.3 X1001.0 ACT X1001.1 X1001.2 0002 X1001.0 r.c SUB 3 TMR X1001.1 ce nt e X1001.3 G0000 G0016 [BYTE MONIT RUN om LADDER ADDRESS DUMP 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .... 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....] [WORD] [D.WORD] [ ] [ ] The [DUMP] soft key has the following functions. (1) [BYTE] : Byte type display (1 BYTE) nc "G0000 00 14 00 00 01 00 00 00 00 00 00 00 00 00 00 00" "G0016 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00" .c (2) [WORD] : Word type display (2 BYTE) "G0000 1400 0000 0001 0000 0000 0000 0000 0000" "G0016 0000 0000 0000 0000 0000 0000 0000 0000" (3) [D.WORD] : Long word type display (4 BYTE) w "G0000 00001400 00000001 00000000 00000000" w w "G0016 00000000 00000000 00000000 00000000" 407 5. PMC LADDER DIAGRAM DISPLAY PMC OPERATION (CRT/MDI) (PMCLAD) 5.3 B–61863E/10 The value of parameter of a functional instruction is displayed in the functional instruction of a ladder diagram. PARAMETER DISPLAY ON LADDER DIAGRAM LADDER *TITLE DATA REMARKS 32 BYTES * NET 00001-00004 X1000.0 MONIT RUN X1000.0 RST ABSDE ACT X1000.3 SUB36 ADDB 2 om X1000.0 D0000 [0]←(Content of D0) 1 DUMP] [DPARA] [ ] [ONLEDT] [ ] ce nt e [r.c D0000 [0] (NDPARA) The function of the soft key is as follows : (1) [DPARA] : The value of parameter is displayed in functional instruction. nc (2) [NDPARA] : The value of parameter is not displayed in functional instruction. 5.3.1 .c The Value of Functional Instruction Parameter w w w No. 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 Functional instruction instr ction Data no. no s END1 END2 TMR* DEC CTR** ROT COD MOVE COM JMP PARI 0 0 2 1 2 3 2 2 0 0 1 DCNV COMP COIN DSCH XMOV ADD SUB 2 2 2 3 3 3 3 408 Data length of instruction parameter (1: byte, 2: word, 4: d. word) Displaying form 1 2 3 4 5 6 4 1 2 4 2 2 1 2 2 1 1 Binary BCD Binary BCD BCD HEX 2 2 2 2 (Note 1) BCD BCD BCD BCD BCD BCD 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 PMC OPERATION (CRT/MDI) B–61863E/10 Functional instruction No. Data no. s Data length of instruction parameter (1: byte, 2: word, 4: d. word) Displaying form 1 2 3 4 5 6 21 MUL 3 2 2 2 BCD 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 DIV NUME TMRB* DECB ROTB CODB MOVOR COME JMPE DCNVB COMPB SFT DSCHB XMOVB ADDB SUBB MULB DIVB NUMEB DISPB EXIN MOVB MOVW MOVN 3 1 1 2 4 2 3 0 0 2 2 1 4 4 3 3 3 3 1 0 1 2 2 2 2 2 4 1/2/4 1/2/4 2 2 BCD BCD Binary END3 DISP PSGNL WINDR WINDW AXCTL TMRC* CTRC** 0 1 2 1 1 1 2 2 DIFU DIFD EOR AND OR NOT PSGN2 END CALL CALLU 0 0 3 3 3 2 1 0 0 0 JMPB LBL 0 0 SP SPE JMPC 0 0 0 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 4 1 2 4 1/2/4 1/2/4 1/2/4 1/2/4 1/2/4 1 2 4 4 1 2 2 2 1 4 4 2 2 1/2/4 1/2/4 1/2/4 1/2/4 MMC3R MMC3W FNC90 Binary HEX 1/2/4 1/2/4 1/2/4 1/2/4 1 Y Y 87 88 89 90 1/2/4 om 1 1 1/2/4 1/2/4 1/2/4 1/2/4 r.c 1 1/2/4 1 1 ce nt e nc .c w w w 5. PMC LADDER DIAGRAM DISPLAY (PMCLAD) (Note1) Binary HEX Binary Binary Binary Binary HEX Binary Binary Binary HEX HEX Binary Binary HEX Binary Binary HEX HEX HEX HEX HEX Y Y 4 4 1 409 2 2 2 2 2 2 2 2 2 Unsign Unsign Binary 5. PMC LADDER DIAGRAM DISPLAY PMC OPERATION (CRT/MDI) (PMCLAD) No. Functional instruction Data no. s B–61863E/10 Data length of instruction parameter (1: byte, 2: word, 4: d. word) Displaying form 1 2 3 4 5 6 FNC91 1 2 Binary 92 93 94 95 96 97 98 99 FNC92 FNC93 FNC94 FNC95 FNC96 FNC97 MMCWR MMCWW 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 Binary Binary Binary Binary Binary Unsign Unsign 2 2 om 91 w w w .c nc ce nt e r.c NOTE The data length of BCD is displayed for 1 is 2-figures, 2 is 4-figures. 1 The value of parameter is not displayed in this instruction. * The timer is displayed the content of timer number (3: TMR, 24: TMRB, 54: TMRC).
** The counter is displayed the content of counter number (5: CTR, 55: CTRC). 410 PMC OPERATION (CRT/MDI) B–61863E/10 5.4 5. PMC LADDER DIAGRAM DISPLAY (PMCLAD) If symbol data and comments are defined to the PMC address, a comment is displayed for symbol display and relay coil. SYMBOL AND COMMENT DISPLAY By pressing soft key [ADRESS], the symbol displayed relay is address-displayed.
By pressing soft key [SYMBOL], the symbol displayed relay is symbol-displayed. (See III. PMC programer, 5. 4 Symbol data setting) MONIT RUN SPDALM X2.4 MACHINE READY MACHINE ALARM ce nt e r.c MA om LADDER Signal name Address or symbol name mments [SEARCH] [ADRESS] [TRIGER] [WINDOW] [(SYMBOL) (1) [ADRESS] : is used to display the address name. w w w .c nc (2) [SYMBOL] : is used to display the symbol name. 411] Comments 5. PMC LADDER DIAGRAM DISPLAY PMC OPERATION (CRT/MDI) (PMCLAD) 5.5 SEARCH OF SPECIFIED RELAY COIL POINTS IN LADDER DIAGRAM B–61863E/10 Specified relay coil points of ladder diagrams can be displayed on the screen. For this operation, press [SEARCH] soft key to bring the following menu. SEARCH ADRESS TRIGER WINDOW TOP BOTTOM SRCH W- SRCH om RET N- SRCH NEXT r.c F- SRCH The function of the soft key is as follows : : Displays the first NET of the ladder from the beginning of the screen. ce nt e (1) [TOP] (2) [BOTTOM] : Displays the last NET of the ladder from the beginning of the screen. nc (3) [SRCH] : This is used for searching a relay coil. Press [W-SRCH] soft key after keying in an address and bit number or symbol name. If the same address and bit number or the same symbol name is detected, the screen containing it will be displayed. (5) [N-SRCH] : Displays the ladder with the specified NET number from the beginning of the screen. Moreover, when pressing the [N-SRCH] key without keying the NET number, the display is scrolled down by one NET. (6) [F-SRCH] : When the functional instruction name or functional instruction number is typed in and the [F-SRCH] key is pressed, the functional instruction is searched. w w w .c (4) [W-SRCH] : When the address and bit number or symbol name to be searched are typed in and the [SRCH] key is pressed, the specified address or symbol is searched from the top of the current screen.
If the specified relay cannot be found until the last NET of the ladder, the relay are searched again from the first ladder until the NET where they started being searched. 412 PMC OPERATION (CRT/MDI) B–61863E/10 5.
PMC LADDER DIAGRAM DISPLAY (PMCLAD) 5.6 Ę STOP OF LADDER DIAGRAM DISPLAY BY TRIGGER OF SIGNAL : Can be used : Cannot be us PA1 PA3 RA1 RA2 RA3 RA5 Ę Ę RB RB2 RB3 RB4 RB5 RB6 Ę Ę Ę Ę Ę RC Δ RC3 RC4 Δ Ę NB NB2 Ę Ę r.c om NOTE Δ: Can be used for the specific series of CNC (Series 16 : B005/11 to, B105/08 to, B305/04 to, B009/03 to, All serieses of model C) (Series 18 : BD03/12 to, BE03/09 to, BG23/03 to, BG03/06 to, BD09/02 to, BE09/14 to, All serieses of model C) PMC–PA3 can be used only with Power Mate–H. The ladder display can be stopped by manual operation or trigger of signal. ce nt e The former ladder diagram display renews signal status every moment. But by using this function, all the ladder diagram at the specified moment can be checked. The stop conditions as a trigger are specified by rising or falling edge detection of the designated signal. * Display of setting trigger The setting address, condition and counter are displayed at the title line.
: X0000.0 : 0 : 0001 " nc "MODE : ON w .c COUNT : Trigger checking number (default 1) : Trigger checking point (default 0) 0 1 2 3 the top of the 1st level after END1 execution after END2 execution after END3 execution ADR : Trigger setting address ON : Rising edge detection (TRGON) , OFF : Falling edge detection (TRGOFF) w w POINT * Setting form adr ; p1 ; p2 + [TRGON/TRGOFF] soft key Note) " ; " = "EOB" adr (trigger address) ; p1 (trigger point) ; p2 (trigger checking number (1 to 65535)) * Because parameters are stored in the nonvolatile memory, they are not lost even if the power is turned off. When bit 2 of keep relay K18 is set to 1 after parameters for sampling are specified, the trigger function automatically starts when the power is turned on. 413 5. PMC LADDER DIAGRAM DISPLAY PMC OPERATION (CRT/MDI) (PMCLAD) B–61863E/10 For this operation, press [TRIGER] soft key to bring the following menu. SEARCH ADRESS TRGON TRGOFF TRIGER WINDOW RET START (STOP) DUMP DPARA (NDPARA) om NEXT TRGSRC INIT r.c The function of the soft key is as follows : (1) [TRGON] : Trigger is set on condition that the ladder status stops when the status of designated signal is rising. ce nt e (2) [TRGOFF] : Trigger is set on condition that the ladder status stops when the status of designated signal is falling. (3) [START] : Change start/stop of trigger execution. While this function is executing, "TRG" is blinking. (4) [TRGSRC] : Search and blink the instruction stopped by trigger. : The setting of trigger is initialized. w w w .c nc (5) [INIT] 414 PMC OPERATION (CRT/MDI) DIVIDING DISPLAY OF LADDER DIAGRAM This function is used for dividing display of ladder diagram. The maximum number of division is 6. LADDER *TITLE DATA REMARKS 32 BYTES X1000.1 *ESP X1000.1 SMBL * NET 00001-00004 *NET NO. 00001 – 00001 X1000.1 X1000.1 X100.0 X100.0 X1000.2 MONIT RUN Y1000.0 X100.0 X1000.0 X1000.1 *NET NO. 00001 – 00001 X1000.2 Y1000.1 X1000.0 X1000.1 X1000.2 Y1000.2 ESP G0000 G0016 G0032 G0048 G0064 SMBL 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ADDRESS 00 00 00 00 00 – 00 00 00 00 00 00 00 00 00 00 00 00 DUMP 00 00 00 00 00 00 00 00 00 00 00 00 00 om 5.7 5. PMC LADDER DIAGRAM DISPLAY (PMCLAD) 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
....
r.c B–61863E/10 ce nt e [DIVIDE] [CANCEL] [DELETE] [SELECFT] [WIDTH] 5.7 Dividing display of ladder diagram NOTE For DUMP display, dump screen is displayed at the last part of screen. w w w .c nc For this operation, press [WINDOW] soft key to bring the following menu. SEARCH ADRESS TRIGER WINDOW DIVIDE CANCEL DELETE SELECT RET WIDTH The function of the soft key is as follows : (1) [DIVIDE] : The screen will be divided. The dividing display of ladder diagram can be displayed for the designated NET number. (NET number + [DIVIDE]) (2) [CANCEL] : The dividing display of ladder diagram display ends. (The screen returns to normal display.) (3) [DELETE] : The screen division subject to operation is ended. (4) [SELECT] : Change the screen subject to division operation. The screen in operation is displayed by "purple" title line, another screen is displayed by "blue" title line. In monochrome CRT, the screen is displayed by changing brightness. 415 5. PMC LADDER DIAGRAM DISPLAY PMC OPERATION (CRT/MDI) (PMCLAD) B–61863E/10 (5) [WIDTH] : Change the width of division by using [EXPAND] or [SHRINK] soft key. (6) [EXPAND] : The divided screen is expanded. (7) [SHRINK] : The divided screen is shrank. 5.8 Ę : Can be used Δ : Option : Cannot be used ON–LINE EDIT PA3 RA1 RA2 Δ RA3 RA5 Ę Ę RB RB2 RB3 RB4 Δ Δ RB5 RB6 RC RC3 RC4 NB NB2 Ę Ę Δ Δ Ę Ę Ę om PA1 ce nt e r.c NOTE Δ: Can be used for the specific series of CNC (Series 16 : B005/11 to, B105/08 to, B305/04 to, B009/03 to, All serieses of model C) (Series 18 : BD03/12 to, BE03/09 to, BG23/03 to, BG03/06, BD09/02 to, BE09/14 to, All serieses of model C) PMC–RA3 is usable only with the Power Mate–H. For the PMC MODEL PA, RA series and –RB series, the editing card (module) is necessary. When bit 1 in the keep relay K17 is 1, this function is available and [ONLEDT] soft key is displayed. When the ladder program is executing, a part of the ladder program can be changed.
w w w .c nc D Change the type of contact (A contact, B contact) D Change address of contact and coil. D Change address parameter of functional instruction. This function don't change the size. (Cannot be Addition, deletion and changable data size) When bit 3 of keep relay K18 is set to 1, the results of online editing are automatically reflected on the ladder program for editing. When bit 3 of keep relay K18 is set to 0, reflect the results of online editing on the ladder program for editing, using the COPY function for the I/O screen.
Otherwise, the results of editing will be lost upon power–off. For the Moreover, when the CNC being used is the Series 15–MODEL B, Series 16/18–MODEL B/C, Series 21/210–MODEL B, Series 16i/18i/21i–MODEL A, or Power Mate–MODEL H, write to flash ROM. How to store the results of editing PMC other than NB Press the COPY key on the I/O screen. NB Without DRAM Write the program into FROM. With DRAM Press the COPY key on the I/O screen. Write the program into FROM.
Operation Press the [ONLEDT] soft key to enable the editing of a ladder program. The editing procedure is the same as that using the programmer function, described in Part III. 416 PMC OPERATION (CRT/MDI) B–61863E/10 6 6. USER PMC SCREEN (PMCMDI) USER PMC SCREEN (PMCMDI) 6.1 This user PMC screen is open to users, and it employs function key . It is applicable only when C language programming has been made.
For details, see the PMC-RC/RC3/RC4/NB programming manual for C language (B-61863E-1). om FOR THE FS16 (PMC-RC OR PMC-RC3) ce nt e r.c NOTE Pressing the key several times changes the screen to the PMCMDI screen. Because the key is also used to execute other functions.
6.2 This user PMC screen is open to users. To display this screen, display the PMC screen and press the OTHERS key or call the pl–pcmdi function in C language. It is applicable only when the program has been written in C language. For details, see the PMC-RC/RC3/RC4/NB programming manual for C language (B-61863E-1). w w w .c nc FOR THE FS15 (PMC-NB) 417 om w w w .c nc ce nt e r.c III. PMC PROGRAMMER (CRT/MDI) 1 1. GENERAL PMC PROGRAMMER (CRT/MDI) B–61863E/10 GENERAL om This PMC programmer is used to set PMC system parameters and also generate and execute sequence programs by using soft keys on the CRT/MDI panel. For this operation, the PMC debugging RAM must be mounted in the CNC in advance. For the CRT/MDI panel keys, refer to PMC operation in PARTII, Chapter 1 and 2. r.c 1) Setting and display of PMC system parameters (SYSPRM) The following system parameters are available. a) Selection of counter data types (BCD or binary) ce nt e b) Selection of division/non-division of ladder program (only PMC-RC) c) Parameters for executing C language programs (only for PMC-RC) 2) Editing of sequence programs (EDIT) The following editing functions are provided. a) Clear of memory b) Title data input nc c) Input, insert, search, and delete of sequence programs by ladder diagram format d) Input, insert, delete, and search of symbol data e) Address setting to each module when I/0 unit is used .c f) Message data input w w w 3) Execution of sequence programs (RUN/STOP) The following function is provided to execute sequence programs a) Sequence program start and stop 4) To write, verify, and read of sequence programs and PMC data, and to write and read of I/0 sequence programs, followings are provided.
a) Input/output of sequence programs to and from FANUC floppy disk cassette b) Input/output of sequence programs to and from debugging RAM c) Input/output of sequence programs to and from ROM d) Input/output of PMC parameter data to and from FANUC FD cassette 5) Displaying the contents of memory for the user C program and debugging the user C program (MONIT) a) Displaying the GDT map of the user C program b) Displaying memory information for the user C program c) Debugging the user C program 421 2. COMPONENT UNITS AND CONNECTIONS 2 PMC PROGRAMMER (CRT/MDI) B–61863E/10 COMPONENT UNITS AND CONNECTIONS om This section describes only the 16/18 MODEL A. For other models, refer to the order list and the connection manual for each model. w w w .c nc ce nt e r.c The units required for generating a sequence program and connection methods are described below. 422 B–61863E/10 PMC PROGRAMMER (CRT/MDI) 2.1 1) PCB and module for PMC 2. COMPONENT UNITS AND CONNECTIONS This is PCB and module for PMC. The type of board is as follows; COMPONENT UNITS a) Series 16 i) PMC-RB (Main CPU board) D PMC control (A20B-2900-0560, -0143) D Debugging control (A20B-2900-0530) D PMC user ROM ii) PMC-RC (Option 3 board) D PMC control module om D Editing module (A02B-0120-C160) b) Series 18 r.c A20B-2900-0390 (When using language programs, work RAM is required.) A20B-2900-0391 A20B-2900-0143 i) PMC-RA1/RA2 (Main CPU board) (A20B-2900-0142) for PMC-RA1 (A20B-2900-0920) for PMC-RA2 ce nt e D PMC control module D Debugging RAM module (A20B-2900-0530) D PMC user ROM w w w .c nc D Editing module (A02B-0120-0160) 423 Common with PMC-RB 2. COMPONENT UNITS AND CONNECTIONS PMC PROGRAMMER (CRT/MDI) B–61863E/10 Configuration of the main CPU board (Series 16) CPU Module 1 2 3 4 5 6 7 8 Connector name name Application CRT JA1 CRT video signal MDI JA2 MDI keyboard R232-1 JD5A RS-232-C serial port R232-2 JD5B RS-232-C serial port MPG JA3 Manual pulse generator IOLINK JD1A FANUC I/O LINK SPDL-1 JA7A Serial spindle A-OUT1 JA8A Analog output APCBAT AMP1 AMP2 CNA F-bus backplane connector 9 10 11 12 APC battery JV1 1st axis servo amplifier JV2 2nd axis servo amplifier JV3 3rd axis servo amplifier AMP4 JV4 4th axis servo amplifier ENC1 JF1 1st axis pulse coder ENC2 JF2 2nd axis pulse coder ENC3 JF3 3rd axis pulse coder ENC4 JF4 4th axis pulse coder SCALE1 JF21 1st axis scale SCALE2 JF22 2nd axis scale SCALE3 JF23 3rd axis scale SCALE4 JF24 4th axis scale nc SERVO ROM JA4A ce nt e AMP3 om LED PMC-RB ROM 13 Connector r.c Drawing number : A16B-2200-0900 Layout of Parts on Main CPU Board (Series 16) .c 2.1 (a) No. w Table 2.1 (a) Module Modules of Main CPU board (Series 16) Drawing number Functional outline ROM module A20B-2900-0290 to 0293 ROM for CAP I or macros 2 ROM module A20B-2900-0290 to 0292 ROM for the CNC system 3 SRAM module A20B-2900-0530 RAM for debugging the PMC-RB 4 SRAM module A20B-2900-0530,-0531 A20B-2900-0540,-0541 RAM for part programs and parameters w w 1 5 PMC control module A20B-2900-0560 (For PMC-RB) A20B-2900-0143 (For PMC-RC) PMC operation control 6 CRT control module A20B-2900-0150 to 0152 CRT display control 7 System control module A20B-2900-0101 to 0103 Clear, battery backup, spindle control, etc. 8 I/O interface module A20B-2900-0110 MDI, MPG, RS-232-C, etc. 9 Servo control module A20B-2900-0160 Digital servo control of the 3rd and 4th axes 10 Servo control module A20B-2900-0160 Digital servo control of the 1st and 2nd axes 11 Servo interface module A20B-2900-0370,-0380 3rd/4th axis amplifier/pulse coder interface 12 Servo interface module A20B-2900-0370,-0380 1st/2nd axis amplifier/pulse coder interface 424 2. COMPONENT UNITS AND CONNECTIONS PMC PROGRAMMER (CRT/MDI) B–61863E/10 Configuration of the option 3 board (Series 16) Drawing number : A16B-2200-0940 (PMC-RC+CAP II) A16B-2200-0941 (only for PMC-RC) A16B-2200-0943 (only for CAP II) Connector Connector Name Name Application LED 1 2 3 5 4 6 JD1A2 FANUC I/O LINK CNA 7 F-bus backplane connector ce nt e r.c IOLINK om Module 8 9 nc CPU 2.1 (b) Layout of Parts on Option 3 Board (Series 16) .c Table 2.1 (b) No. Module Modules of Option 3 Board (Series 16) Drawing number Functional outline ROM module A20B-2900-0290 to 0293 User ROM for PMC-RC (Mount the RAM module during debugging.) 2 ROM module A20B-2900-0292 System ROM for PMC-RC 3 DRAM module A20B-2900-0553 Work RAM for PMC-RC 4 PMC control module A20B-2900-0560 PMC operation control and I/O Link control 5 PMC CPU module A20B-2900-0390 For ladder capacity 2400 steps or C language A20B-2900-0391 Other than the above w w w 1 425 2. COMPONENT UNITS AND CONNECTIONS PMC PROGRAMMER (CRT/MDI) B–61863E/10 Configuration of the Main CPU Board (Series 18) PMC-RA1/RA2 ROM 16 Module 1 2 3 4 5 6 7 8 Connector Name No. CRT JA1 CRT video signal MDI JA2 MDI keyboard R232-1 JD5A RS-232-C serial port R232-2 JD5B RS-232-C serial port MPG JA3 Manual pulse generator IOLINK JD1A SPDL-1 JA7A A-OUT1 JA8A APCBAT JA4A AMP1 JV1 AMP2 AMP3 AMP4 FANUC I/O LINK Serial spindle Analog output Battery for use with the APC Axis 1 servo amplifier JV2 Axis 2 servo amplifier JV3 Axis 3 servo amplifier JV4 Axis 4 servo amplifier JF1 Axis 1 pulse coder ENC2 JF2 Axis 2 pulse coder ENC3 JF3 Axis 3 pulse coder ENC4 JF4 Axis 4 pulse coder SCALE1 JF21 Axis 1 scale SCALE2 JF22 Axis 2 scale SCALE3 JF23 Axis 3 scale SCALE4 JF24 Axis 4 scale ce nt e ENC1 CNA F-BUS back plane connector nc 9 10 11 12 13 14 15 2.1 (c) Table 2.1 (c) No. Description om LED Connector r.c Drawing number : A16B-2201-0080 Module list for the main CPU board (Series 18) Drawing no. Function outline 1 ROM module A20B-2900-0290 to 0293 ROM for macros or CAP 1 2 ROM module A20B-2900-0290 to 0292 ROM for the CNC system 3 w .c Module name Parts layout for the main CPU board (Series 18) SRAM module A20B-2900-0530 RAM for PMC-RA1/RA2 debug SRAM module A20B-2900-0530,-0531 A20B-2900-0540,-0541 RAM for parameters and tape memory PMC control module A20B-2900-0142 (PMC-RA1) A20B-2900-0920 (PMC-RA2) PMC operation control 6 Main CPU module A20B-2900-0930 FS18 Main processor 7 System control module A20B-2900-0900 to 0902 Clear, battery backup, spindle control, servo/graphics software flash ROM 8 I/O interface module A20B-2900-0110 MDI, MPG, RS-232-C 9 Graphics control module A20B-2900-0310 Graphics display control 10 Graphics CPU module A20B-2900-0590 Graphics control CPU 11 CRT control module A20B-2900-0154 to 0156 CRT display control 12 Servo control module A20B-2900-0160 Digital servo control for axes 3 and 4 13 Servo control module A20B-2900-0160 Digital servo control for axes 1 and 2 14 Servo interface module A20B-2900-0380 Amplifier, pulse coder, and interface for axes 3 and 4 15 Servo interface module A20B-2900-0380 Amplifier, pulse coder, and interface for axes 1 and 2 4 w w 5 426 PMC PROGRAMMER (CRT/MDI) B–61863E/10 2. COMPONENT UNITS AND CONNECTIONS 2) Debugging RAM This is used for debugging sequence programs. Since this debugging RAM memory is backed up by the battery, the memory data contents are not erased even when turning off the power supply. om CAUTION If a RAM parity error occurs or when power is first turned on after installation, the RAM for debugging must be cleared. (Procedure) 3) Editing module r.c Turn on power to the CNC while pressing the X and O keys simultaneously. The contents of the RAM for debugging are then cleared. ce nt e This is a built-in programmer for PMC-RA1, PMC-RA2, RA3, PMC-RB, PMC-RB2, or RB3 that enables editing sequence programs. 4) ROM After debugging, write a sequence program into ROM. 5) ROM WRITER This unit is used for writing or reading out a sequence program to ROM. 6) Offline programmer This is used to transfer a sequence program. w w w .c nc By connecting the Offline programmer to PMC-RA1, -RA2, -RB, -RB2, -RB3, -RC, or -RC3, the storage of sequence programs in the floppy, and the output of a sequence program into printer can be done. 427 2. COMPONENT UNITS AND CONNECTIONS 2.2 PMC PROGRAMMER (CRT/MDI) B–61863E/10 (1) Connecting the debugging RAM module a) PMC-RB, -RB2 and RB3 : Connect the module to portion 3 shown in Fig. 2.1 (a). CONNECTING COMPONENT UNITS b) PMC-RC and PMC-RC3 : Connect the module to portion 1 shown in Fig. 2.1 (b). c) PMC-RA1, -RA2 and -RA3 : Connect the module to portion 3 shown in Fig.
2.1 (c). om (2) Connecting the editing module for PMC-RA1, -RA2, -RA3, -RB, -RB2 and -RB3 Connect the module to portion 3 shown in Fig. 2.1 (a). (3) Connecting ROM a) PMC-RB, -RB2 and -RB3 : Connect EPROM to portion 13 shown in Fig. 2.1 (a). r.c b) PMC-RC and PMC-RC3 : Connect the ROM module to portion 1 shown in Fig. 2.1 (b). c) PMC-RA1, -RA2 and -RA3 : Connect EPROM to portion 16 shown in Fig. 2.1 (c). ce nt e Ę : Enabled Δ : Enabled depending on the option : Disabled RA1 RA2 RA3 RB RB2 RB3 RC RC3 RAM module Ę Ę Ę Ę Ę Ę Ę Ę Editing module Ę Ę Ę Ę Ę Ę EPROM Ę Ę Ę Ę Ę Ę Δ Δ Ę Ę ROM module w w w .c nc NOTE 1 When 24,000 optional PMC-RB2 and PMC-RB3 ladder steps are available, 256K bytes of the ROM module can be used. In this case, connect the ROM module to portion 3 shown in Fig. 2.1 (a). 2 Either a RAM module, editing module, or ROM module can be connected to each board of PMC-RA1, -RA2, -RA3, -RB, -RB2 and -RB3. 3 Either a RAM module or ROM module can be connected to each board of PMC-RC and PMC-RC3. (4) Connecting the off–line programmer Connect the off–line programmer to the reader/punch interface on the CNC. There are several connectors for the reader/punch interface on the CNC. The connector to be used is specified during I/O processing for the PMC. For details, see Section 7. 428 PMC PROGRAMMER (CRT/MDI) B–61863E/10 3 3. SELECTION OF PROGRAMMER MENUS BY SOFTKEYS SELECTION OF PROGRAMMER MENUS BY SOFTKEYS om To operate the PMC programmer, set bit 1 in K17 of the keep relay area for PMC parameters to 1, enabling the programmer basic menu to be displayed. To display the programmer basic menu, press and [PMC] soft key on the MDI keyboard then, press the [NEXT] key. (1) Programmer basic menu r.c The programmer basic menu is displayed at the lower part of the CRT screen to signify the keys as shown in the following figure.
ce nt e The programmer basic menu and PMC basic menu are selected to each other alternately by pressing the [NEXT] key.
For the PMC basic menu and operation, see PMC operation in Chapter II. nc NOTE In the following description, the relation between soft keys and menus is described based on 9-inch CRT/MDI panel. The 14-inch CRT/MDI panel is different from the 9-inch CRT/MDI panel about the number of soft keys. Five soft keys are mounted on the 9-inch CRT/MDI panel, while ten soft keys are mounted on the 14-inch CRT/MDI panel. Address/numeric keys .c RESET key HELP key Edit keys w Cancel key w w INPUT keys Power on/off buttom Soft keys SHIFT key Cursor control keys Function keys Page keys 429 3. SELECTION OF PROGRAMMER PMC PROGRAMMER (CRT/MDI) MENUS BY SOFTKEYS B–61863E/10 (2) Relation between programmer menus and soft keys PMCLAD PMCDGN PMCPRM Chapter III.6 Chapter III.5 Chapter III.7 Chapter III.4 RUN or STOP EDIT I/O SYSPRM RET RET TITLE RET ce nt e MESAGE MONIT DBGLAD** LADDER SYMBOL NEXT r.c RET om The relation between programmer menus and soft keys are different according to each function as shown in the following figure. These menus are selected by pressing related keys.
For the menu contents, see the description given later.
Refer to this figure for operation. GDT* USRMEM* DEBUG* NEXT nc MODULE .c CROSS w w w CLEAR NOTE 1 Mark "*" is valid for PMC-RC/RC3/RC4/NB function. 2 Mark "**" is valid for PMC-RA3/RB3 with Editing module or PMC-RC/RC3 function. 430 PMC PROGRAMMER (CRT/MDI) B–61863E/10 4 4. SPECIFYING AND DISPLAYING SYSTEM PARAMETERS (SYSPRM) SPECIFYING AND DISPLAYING SYSTEM PARAMETERS (SYSPRM) om Display the system parameter screen by pressing soft key [SYSPRM] on the basic programmer menu. Move the cursor to necessary system parameters and specify them according to the menu displayed on the screen. When this function is selected, if the sequence program is in operation, the PMC management software automatically stops this function. (1) COUNTER DATA TYPE r.c Specifies whether the counter value is used in binary or BCD by functional instruction CTR. (2) LADDER EXEC (valid for PMCRC/RC3/RC4/NB/NB2) ce nt e CAUTION After changing a counter data type, set up the counter value again.
Specifies the increment or decrement of processing time of the 1st and 2nd level parts of the ladder program in the range of 1% to 150%. This increases or decreases the scanning time of the ladder program. This parameter influences the processing time of the 3rd level part of the ladder program and the language program. nc If 100% is specified, the time of 5 ms for an 8 ms cycle is used to process the 1st and 2nd level parts of the ladder program. The remaining 3 ms is used to process the 3rd level part of the ladder program, language program, and PMC screen display. w w w .c If 120% is specified, the time of 6 ms is used to process the 1st and 2nd level parts of the ladder program. This reduces the scanning time of the ladder program, thus enabling the ladder program to be executed at high speed. Note that the processing time required for the 3rd level part of the ladder program, language program, and PMC screen display is substantially reduced. If the undivided system is specified too, this parameter is validated.
If a value less than 40% is specified, 40% is assumed.
If a value greater than 120% is specified, 120% is assumed. The processing time of the 1st and 2nd parts of the ladder program is obtained by the following formula: Processing time of the 1st and (LADDER EXEC) 100 The processing time of the 3rd level part of the ladder program, language program, and PMC screen display 2nd parts of the ladder program=5 msec = 8 ms – (processing time of the 1st and 2nd level parts of the ladder program) 431 4. SPECIFYING AND DISPLAYING SYSTEM PARAMETERS (SYSPRM) PMC PROGRAMMER (CRT/MDI) 100% 120% 8 msec 1st and 2nd level parts of the ladder program B–61863E/10 8 msec 1st and 2nd level parts of the ladder program 5 ms 6 ms Others Others 3 ms 2 ms Specifies the division ratio of execution for PMC screen display and language program. (4) IGNORE DIVID CODE (valid for PMC-RB and-RC) (5) LANGUAGE ORIGIN (valid for PMC-RC/ RC3/RC4/NB/NB2) Specifies whether the ladder program is executed in the divided system (IGNORE DIVID CODE = NO) or in the undivided system. om (3) LANGUAGE EXEC RATIO (valid for PMC-RC/RC3/RC4/NB/ NB2) ce nt e r.c (0 to 99%) Since the execution priority of PMC screen display is higher than language program tasks, it is usually hard for the tasks to execute processing while displaying PMC screen. Then this parameter can be used to set the division ratio for each. Cyclic processing of language program is therefore possible during PMC screen display. Only language program tasks are running if PMC screen is not displayed. Specifies the first address of the link control statement data in the language program. Be sure to specify 0 when the language program is not stored.
LANGUAGE AREA and SIZE indicate the area where the language program is stored. Store the language program in the specified area.
nc When the language program is stored, the Language Origin is automatically set by moving the cursor to this item and pressing [ORIGIN] soft key.
Specify the maximum size of the ladder program. This parameter can be used to increase or decrease the size of the work area used by language programs. The setting of the parameter takes effect only after power is turned on. When the setting is to be changed, therefore, power must be turned off. w w .c (6) MAX LADDER AREA SIZE (valid for PMC-RC/ RC3/NB) w (7) FS0 OPERATOR PANEL For details, see the FANUC PMC-MODEL RC/RC3/RC4/NB PROGRAMMING MANUAL C LANGUAGE (B–61863E–1). The default is the size in kilobytes resulting from conversion of the ladder step option. Specifies whether the Series 0 operator's panel is connected. When YES is selected, specify the actual addresses of DI and DO connected to the operator's panel, the address of the key image transferred from the operator's panel, and the address of the LED image to be transferred to the operator's panel. (a) KEY DI ADDRESS Specify a PMC address representing the first address of the external DI actually connected (X0 to X127 or X1000 to X1019). (b) LED DO ADDRESS Specify a PMC address representing the first address of the external DO actually connected (Y0 to Y127 or Y1000 to Y1014). 432 PMC PROGRAMMER (CRT/MDI) B–61863E/10 4. SPECIFYING AND DISPLAYING SYSTEM PARAMETERS (SYSPRM) (c) KEY BIT IMAGE ADDRESS Specify a PMC address representing the first address of the key image to be referenced by the user program. Usually specify an arbitrary internal relay area. (d) LED BIT IMAGE ADDRESS (8) STEP SEQUENCE om Specify a PMC address representing the first address of the key image to be generated by the user program. Usually specify an arbitrary internal relay area. When creating new programs with the built–in editing function, set this parameter first, then execute CLEAR ALL or perform clear operation (turn on power while holding down X and O) at power on. r.c When selecting the step sequence method: STEP SEQUENCE = YES When selecting the ladder method: STEP SEQUENCE = NO = BINARY/BCD ce nt e PMC SYSTEM PARAMETER COUNTER DATA TYPE FS0 OPERATOR PANEL = YES/NO KEY DI ADDRESS = X100 LED DO ADDRESS = Y100 KEY BIT IMAGE ADDRESS = R900 nc LED BIT IMAGE ADDRESS = R910 > [BINARY] [BCD] [ ] 4(a) PMC-RA Series System Parameter Screen .c w] [PMC SYSTEM PARAMETER COUNTER DATA TYPE = BINARY/BCD IGNORE DIVIDE CODE = NO/YES w w] [> [BINARY] [BCD 4(b)] [ ] [ ] [ ] PMC-RB Series System Parameter Screen (1st Page) 433 4. SPECIFYING AND DISPLAYING SYSTEM PARAMETERS (SYSPRM) PMC PROGRAMMER (CRT/MDI) B–61863E/10 PMC SYSTEM PARAMETER MONIT STOP COUNTER DATA TYPE = BINARY /BCD LADDER EXEC = 100% (1-150) LANGUAGE EXEC RATIO = 50% (0-99) IGNORE DIVIDE CODE = NO/YES] [ ] [ ] [ ] r.c [BINARY] [BCD om LANGUAGE ORIGIN = 841000H (LANGUAGE AREA = 840000H,SIZE = 768KB) MAX LADDER AREA SIZE = 90KB (1-96) 4(c) PMC- RC, RC3 or NB System Parameter Screen (1st page) PMC SYSTEM PARAMETER (1/2) = BINARY/BCD = YES/NO ce nt e COUNTER DATA TYPE MONIT STOP IGNORE DIVIDE CODE nc > [BINARY] [BCD w w w .c 4(d)] [ ] [ ] [ ] PMC-RB4/RB6/RC4 System Parameter Screen (1st Page) 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 H, SIZE = STEP SEQUENCE > = [BINARY] [BCD] [KB) YES/NO] [ ] [ ] 4(e) PMC–RC4/NB2 System Parameter Screen (1 st Page) Press the [NEXT] key to select the following screen for PMC-RB series, PMC-RC series, and PMC-NB : 434 PMC PROGRAMMER (CRT/MDI) B–61863E/10 4. SPECIFYING AND DISPLAYING SYSTEM PARAMETERS (SYSPRM) PMC SYSTEM PARAMETER (2/2) MONIT STOP = YES/NO KEY DI ADDRESS = X100 LED DO ADDRESS = Y100 KEY BIT IMAGE ADDRESS = R900 LED BIT IMAGE ADDRESS = R910 > NO] [ ] [ ] [ ] PMC-RB Series, PMC-RC Series, or PMC-NB System Parameter Screen (2nd Page) w w w .c nc ce nt e 4(f)] [r.c [YES om FS0 OPERATOR PANEL 435 5.
EDITING OF SEQUENCE PROGRAM (EDIT) 5 PMC PROGRAMMER (CRT/MDI) B–61863E/10 EDITING OF SEQUENCE PROGRAM (EDIT) om Press soft key [EDIT] of the programmer basic menu to bring the following menu. For setting the CLEAR or I/O unit address, press the [NEXT] key to bring another menu. r.c Each menu of [EDIT] can be selected by EDIT key, or menu of other EDIT can be selected by each EDIT menu. When this function is selected, if the sequence program is in operation, the PMC management software automatically stops this function. (Operation) ce nt e Perform each operation by pressing necessary menu soft keys. Press [RETURN] key for resetting to the programmer basic menu.
RUN (STOP) EDIT I/O TITLE LADDER III.5.1 III.5.2 SYSPRM MONIT w w w .c nc RET SYMBOL MESAGE III.5.4 NEXT III.5.5 MODULE CROSS CLEAR III.5.3 III.5.7 III.5.6 PMC EDITION MENU MONIT STOP SELECT ONE OF FOLLOWING SOFT KEYS TITLE LADDER SYMBOL MESAGE MODULE CLEAR CROSS : : : : : : : TITLE DATA LADDER DIAGRAM SYMBOL & COMMENT DATA MESSAGE DATA I/O MODULE DATA CLEAR DATA CROSS REFERENCE [TITLE] [LADDER] [MESAGE] [ ] [[MODULE] [ ] [CLEAR]] [CROSS] [5 436 Editing basic menu] 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 5.1 D Machine tool builder name (32 characters) D Machine tool name (32 characters) D NC and PMC types (32 characters) D Sequence program number (16 characters) D Version (4 characters) D Sequence program drawing number (32 characters) D Date of sequence program creation (16 characters) D Sequence program programmer (32 characters) D ROM programmer (32 characters) D Comment (32 characters) The title for the 9" CRT consists of three screens. The screens are changed by pressing or . ce nt e r.c om SPECIFYING AND DISPLAYING TITLE DATA (TITLE) The title data refers to the title of the sequence program created by the machine tool builder. The data consists of the following ten items: RUN EDIT I/O SYSPRM TITLE LADDER SYMBOL MESAGE INSERT DELETE LADDER MONIT RETURN RETURN nc C LANG w w w .c NOTE When a C language board is installed in the Series 16i/18i, the title data for C can be edited. With the soft key [LADDER], the display can be switched to the ladder title data. With the soft key [C LANG], the display can be switched to the C title data. 5.1.1 Entering Title Data 5.1.2 Deleting Title Data (1) Move the cursor to the desired title data item. Use the cursor keys [°], [±], [³], [²] to move the cursor. (2) Press the address key and numeric keys to enter the title data, and press the key. (1) Move the cursor to the desired title data item. Use the cursor keys [°], [±], [³], [²] to move the cursor. (2) After keying in the title data by pressing the desired address keys and numeric keys, press the key. 437 5.
EDITING OF SEQUENCE PROGRAM (EDIT) 5.1.3 Editing Character Strings of Title Data PMC PROGRAMMER (CRT/MDI) B–61863E/10 When the length of the cursor is the same as the maximum number of characters, pressing the [INSERT] key enables the operator to edit character strings. Then, the length of the cursor is changed to that of one character. (1) Move the cursor to the desired insertion position with the cursor keys and enter a character string. Then, the character string is inserted. om (2) Pressing the [DELETE] key deletes the character at the cursor. PMC TITLE DATA #1 MONIT PMC PROGRAM NO. : EDITION NO.
: RUN 1234 12 r.c PMC CONTROL PROGRAM SERIES : 4061 EDITION : 01 : : : : : 44.0KB 32.0KB 10.2KB 01.8KB 048 MSEC ce nt e MEMORY USED LADDER SYMBOL MESSAGE SCAN TIME [INSERT] [DELETE] [5.1.3 (a)] [ ] [ ] Title Edit Screen 1 MONIT RUN nc PMC TITLE DATA #2 MACHINE TOOL BUILDER NAME : f · · · · · · · · · · · · f CNC & PMC TYPE NAME : f · · · · · · · · · · · · f PROGRAM DRAWING NO. : f · · · · · · · · · · · · f [INSERT] [DELETE] [5.1.3 (b) w w w .c MACHINE TOOL NAME : f · · · · · · · · · · · · f 438] [ ] [Title Edit Screen 2] 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) PMC TITLE DATA #3 MONIT RUN DATE OF PROGRAMING : f · · · · · · · · · · · · f PROGRAM DESIGNED BY : f · · · · · · · · · · · · f ROM WRITTEN BY : f · · · · · · · · · · · · f REMARKS : f · · · · · · · · · · · · f [INSERT] [DELETE] [ ] [ ] [r.c 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 om B–61863E/10 Title Edit Screen 3 w w w .c nc ce nt e 5.1.3 (c) 439] 5. EDITING OF SEQUENCE PROGRAM (EDIT) 5.2 PMC PROGRAMMER (CRT/MDI) B–61863E/10 Input, insert, delete, and search a sequence program as described below. The relation between these functions and soft keys is as shown below. SEQUENCE PROGRAM GENERATION (LADDER) RUN (STOP) EDIT I/O TITLE LADDER yj jy yj jy SYSPRM MONIT RET RET yĘyj FUNCTN r.c Ę j om SYMBOL MESAGE COMAND ce nt e ––––– NEXT y(S)yj III 5.3 y(R)yj III 5.7 INSNET DELNET INSERT NEXT RET MOVE SEARCH NEXT CHANGE nc COPY ADRESS w w w .c 5.2 Sequence program generation softkeys NOTE "y(S)yj"and "y(R)yj" are valid for PMC-PA3, -RA3, -RB3, -RB4, -RC3,-RC4, and -NB. Each of EDIT · LADDER software functional instruction keys can be selected by the [COMAND] key. Type in one of the following character strings and press software key [COMAND]. The character string within parentheses "[ ]" can be omitted. "n" after the character string indicates that a value can be input. For example, if the [COMMAND] key is pressed after "D2" is typed in, the operation can be performed in the same manner when the key is pressed after "2" is typed in. I[NSERT] A[DRESS] S[EARCH] M[OVE][n] D[ELNET][n] SY[MBOL] C[OPY][n] n:value Generate and search a program by pressing soft keys of the above menu. 440 PMC PROGRAMMER (CRT/MDI) B–61863E/10 5. EDITING OF SEQUENCE PROGRAM (EDIT) NOTE r.c om Soft keys ( ] or [ ])( ] or [ ) are used for producing or deleting an upper left vertical line or upper right vertical line on the ladder diagram. The solid line display vertical line indicates the production, while the dotted line display vertical line shows the deletion. Which one is available is determined by the ladder diagrams and cursor positions. ce nt e When the cursor is set to this position, the upper right vertical line is not produced yet, and the menu becomes [ ] When the cursor is set to this position, the upper right vertical line is already produced, and the menu becomes [ ] 5.2.1 If a sequence program is not input yet, the right and left vertical lines only of the ladder diagram are displayed on CRT/MDI. nc Sequence Program Input Press soft key [LADDER] for inputting a sequence program. The soft key menu changes as shown in Fig. 5.2. .c Start inputting a program with this screen condition. If a previous program remains unerased from RAM module for debug, clear it according to the instruction in 5.6 before starting the program input. The following description shows an example of the input of a program of the basic instruction and a program of the functional instruction. (1) In case of basic instruction program input; w w w Input a ladder diagram by moving the cursor to the desired input position by using the cursor key. 441 5.
EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) R0.1 R10.2 B–61863E/10 R20.2 R1.7 X2.4 9-inch CRT/MDI 10-inch CRT/MDI 7 contact + relay coil 14-inch CRT/MDI om The contacts and coils inputtable in one line are as specified below.

They cannot be input into one line more than specified.

If they exceed the specified range, provide a dummy relay coil halfway. r.c Press soft key [ ] after moving the cursor to the start position. Symbol [ ] is input to the cursor position and HORIZONTAL LINE ILLEGAL is displayed at the lower right part of the CRT screen.

This is a caution message to show that the ladder diagram horizontal line is not entered yet. Input address and bit data next. Press key after inputting R0.1 by using address key and numeric keys. The address is set on the contact, and cursor shifts rightward. Input A contact with address R10.2 by the above method 1, 2. (Note) The order of processes 1 and 2 are interchangeable. Input B contact R1.7 Press soft key [ ], input address R1.7, and then, press key. The address is set on the B contact and the cursor shifts rightward. Press soft key [ ]with the cursor kept as it is. A right horizontal line is automatically drawn, and a relay coil symbol is entered near the right vertical line. Press key after inputting. The cursor automatically shifts to the input start position of the next line. Input the OR condition next, Press soft key [ ], input address X2.4 and then, press INPUT key. The address is set on the B contact and the cursor shifts rightward. Press NEXT key, since the soft key of the right horizontal line of OR is necessary. Press soft key [ ] to input a horizontal line.], [ ]), key When inputting the horizontal bar key ([in a numerical value and press this bar key, and then the horizontal line for the frequency will be drawn. However, this horizontal line will not be drawn over the LINE. ce nt e 1 2 3 nc 4 w w w .c 5 6 7 8 9 10 Press soft key [line or OR. 442], and input necessary upper right vertical PMC PROGRAMMER (CRT/MDI) B–61863E/10 5. EDITING OF SEQUENCE PROGRAM (EDIT) nc ce nt e r.c om CAUTION 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. However, be careful since the program net (a block corresponding to a range from RD to WRT Instruction) containing an error is deleted when the screen is switched to an CNC screen. 2 7 contacts + a coil are specified to be inputtable per line from CRT/MDI, any more contacts exceeding the specified value are not inputtable. However, this limitation is not applicable to mnemonic sequence programs generated by Offline programmer. 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 usually, except when all programs from RD instruction to WRT instruction are erased. 3 If the power is turned off while a ladder program is being displayed in edit mode, that ladder program will be lost. Always save the program and exit the editing screen before turning off the power. 4 The termination processing of the ladder (JMP, COM, and other processing) is done when the EDIT screen is switched to another screen by pressing RET key, it takes several tens second until the screen is switched completely, if the ladder is large. 5 In the Series 15–MODEL B, Series 16/18–MODEL B/C, and Series 16/18/21–MODEL A CNC that use Flash Memory, the program is not automatically written into Flash Memory once editing ends. After editing, perform the processing for writing to Flash Memory (see 7.3.3, “FROM” in Chapter 7 of Part III). Otherwise, the editing results will be lost when the power is turned off. 6 When the user presses the RET key to switch from the edit screen to another screen, the parameters of functional instructions TMR, TMRB, CRT, DIFU, and DIFD are checked for a parameter number range error and duplicate parameter number in the ladder termination processing. If a range error is found, the editing cannot be terminated. If the use of a duplicate parameter number is found, the guidance message is displayed. w w w .c (2) In case of functional instruction program input; For inputting a functional instruction, press [FUNCTN] soft key, and then, input instruction symbol of the functional Instruction and SUB number. A function command can be input by pressing the [FUNCTN] key after keying the Function Command No. When pressing the [FUNCTN] key without keying in the Function Command No., the function command table is displayed. Key in Function Command No. and press [INPUT] key. If you don’t keep the instruction symbol and SUB number into mind, you can display a functional instruction table covering the correspondence between instruction on symbols and SUB numbers automatically by inputting a wrong instruction symbol or a wrong SUB number and then pressing the soft key [FUNCTN] key or by pressing soft key [FUNCTN] key only without inputting any other key. If an aimed functional instruction is not found in the displayed functional instruction table, press [NEXT] key or [PAGE] key to brings its subsequent table. Press [FUNCTN] key when resetting the functional instruction table to the original ladder diagram. 443 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 Functional instruction table FUNCTN When a wrong functional instruction is entered or [FUNCTN] key only is pressed; om [NEXT] key (to display the subsequent functional instruction table) Functional instruction generation soft key r.c (Press this key again when resetting the functional instruction table to the original ladder diagram) DECB ce nt e NOTE If the system is left undone without inputting any data after pressing soft key [FUNCTN], the other soft keys are not employ-able. In such a case, press [FUNCTN] key again.

Format specification ĘĘĘĘ ĘĘĘĘ ĘĘĘĘ ĘĘĘĘ Code data address Specification number Output data address nc SUB 25 Ę .c Input functional instruction parameters in the vertical direction as shown in the following figure when inputting functional instructions from CRT/MDI panel. w w w ACT DECB SUB 25 Ę ĘĘĘĘ ĘĘĘĘ ĘĘĘĘ ĘĘĘĘ 1 Format specification CRT/MDI Code data address Specification number Output data address Input a control condition.], input the address and bit data, and Press soft key [then, press key. The cursor shifts rightward. 2 Input an instruction. Press soft key [FUNCTN], input SUB number 25, and then, press key.

A functional instruction diagram appears as shown in the above figure. 3 Input an instruction parameter. 444 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 Input the first parameter, format specification, and then, press key. The cursor automatically lowers downward. Input three residual parameters in order. Alteration of Sequence Programs The method of altering a generated sequence program is the same as described in 5.2.1. Move the cursor to the program part to be altered and input change data. om 5.2.2 5.2.3 Insert of Sequence Program yj jy INSNET DELNET yĘyj FUNCTN ADRESS SEARCH ce nt e RETURN Ę j r.c yj jy INSNET INSLIN INSELM A sequence program is inserted in four ways on the ladder diagram as described below. (1) To insert a relay contacts in the horizontal direction. Insert Move the cursor to the position where a sequence program is to be inserted, and input the program by the method specified in 5.2.1. w w .c nc Simple horizontal insert w When a vertical line is to be deleted for the insert operation ; Insert ÅÅÅ ÅÅÅ Cursor 1 Set the cursor to the above position. 445 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) 2 B–61863E/10 Press soft key [ ] for erasing the upper left vertical line. The upper left vertical line to the cursor disappears. Press soft key [ ] to produce a upper right vertical line to the cursor, then, press soft key [ ]. Both verti-cal line and horizontal line are pro-duced. 4 Shift the cursor to a line of contact insert position. 5 Press soft key [ ] to add contacts. (2) For inserting vertical line; r.c om 3 ce nt e Insert w w w .c nc For inserting a vertical line as shown in the above figure, the inserting area is required, correspondingly. In order to produce the area, shift the entire part after the part to be inserted by one line by moving the cursor to the ladder diagram within the dotted line range (an optional part is allowable) and then pressing soft key [INSNET] (see Fig. 5.2). The lower ladder diagram shifts downward by one line, each time the [INSNET] key is pressed to produce the area to which a line is to be inserted. If a surplus insert area remains unused after the insert processing ends (if an area corresponding to 3 lines has been reserved when two lines have been inserted, for example), the area may be left as it is. No problem arises. 1 Move the cursor to the ladder diagram bounded by a dotted line. 2 Press soft key [INSNET]. The lower ladder diagram shifts downward by one line. 3 Pressing [INSNET] key without keying in numeric values will cause one line to be inserted. 4 Pressing [INSNET] key with keying in numeric values will cause the line to be inserted the number of numeric values input. 5 After setting the cursor to a position where the contacts is to be inserted, press soft key [ ]. After setting address data, press [INPUT] key. The cursor shifts rightward. 6 Press soft key [ ] to produce an OR circuit. (3) Inserting the 1 NET sequence program lines Space lines are inserted one by one. 1 Type in the number of lines to be inserted and press the [INSLIN] key. The lines corresponding to the input number are inserted.

(If the number of lines to be inserted is not typed in but the [INSLIN] key is pressed, one line is inserted.) 446 PMC PROGRAMMER (CRT/MDI) B–61863E/10 5. EDITING OF SEQUENCE PROGRAM (EDIT) om Cursor Cursor When the [INSELM] key is pressed when the cursor is positioned as shown in the above figure on the left, the element is inserted as shown in the above figure on the right. w w w .c nc ce nt e r.c If the [INSLIN] key is pressed when the cursor is in the position specified as shown in the above figure on the left, the line is inserted as shown in the above figure on the right. (4) Inserting the 1 NET sequence program elements Elements can be inserted one by one. 1 Type in the number of elements to be inserted and press the [INSELM] key. The elements corresponding to the input number are inserted. If the number of elements prefixed by character ”A” is typed in and the [INSELM] key is pressed, the elements are inserted after the cursor. (If the number of elements to be inserted is not typed in but the [INSELM] key is pressed, one element is inserted.) Cursor If ”A” is typed in when the cursor is positioned as shown in the above figure on the left and the [INSELEM] key is pressed, the element is inserted as shown in the above figure on the right. 447 5. EDITING OF SEQUENCE PROGRAM (EDIT) 5.2.4 Delete of Sequence Program PMC PROGRAMMER (CRT/MDI) B–61863E/10 (1) Delete a part of sequence program by using three kinds of soft keys after setting the cursor to the portion from which the sequence program is to be deleted. [ ] : Delete of horizontal lines, relay contacts, relay coils, etc. [ ] : Delete of upper left vertical line to the cursor [ ] : Delete of upper right vertical line to the cursor om (2) Delete a net of the sequence program (the part from the RD instruction to the WRT instruction) with the [DELNET] key. (3) Deleting NETs one by one yj jy INSNET DELNET ce nt e RETURN Ę j EXEC 1 yĘyj FUNCTN ADRESS SEARCH C-DOWN C-UP r.c yj jy CANCEL SEARCH Deletion Move the cursor to the NET to be deleted and press the [DELETE] key.

The NET to be deleted brightly displayed on the screen. Deleting multiple NETs Move the cursor with the cursor DOWN key, [C-DOWN] key, or [SEACH] key to blink the NETs to be deleted. Type in a 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. Cancel 4 Press the [CANCEL] key. If the NET to be deleted is already known, move the cursor to the first NET, type in the number of NETs, and press the [DELNET] key to omit steps 1 and 2. w w w .c nc 2 448 5.

EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 R20.1 R6.4 Y 1.2 R0.5 R0.5 For searching the same address as specified here, set the cursor to this position, and press soft key [SRCH]. R20.1 X 4.2 R21.0 Y 2.0 R20.1 R2.2 om X 2.0 R0.4 r.c The same address is searched, and the cursor shifts to this position.

R10.5 5.2.5 Search a sequence program by using the following soft keys. For the following soft keys, see Fig. 5.2. yj jy yj jy INSNET DELNET TOP BOTTOM nc Search of Sequence Program ce nt e R0.5 Ę j yĘyj FUNCTN ADRESS SEARCH W-SRCH N-SRCH w w w .c RETURN SRCH NEXT F-SRCH C-DOWN C-UP (1) Soft key [TOP] When this key is pressed, the start of the sequence program is displayed on the screen and the cursor shifts to this 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 shifts to this position. (3) Soft key [SRCH] This key is used to search a specified address.

It searches the specified address from the program of the cursor part to the last program of this screen, and displays the address on the screen. 449 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 a) Method of specifying the address by the cursor Set the cursor to the relay contact part of the address to be searched and press soft key [SRCH]. The system searches the same address as the address specified by the cursor from the cursor program on the presently displayed screen to the end of the program (SUB 48). When the same address is found, the program part is displayed on the screen, and the cursor shifts to the address part. If the same address is not found as a result of this search, an error is displayed. om b) Method of specifying the address by inputting it Input an address to be searched by using address and numeric keys and press soft key [SRCH]. The same address as specified is searched from the program of the cursor part on the presently displayed screen to the last of the program (SUB 48). r.c When the same address is found, the program part is displayed on the screen, and the cursor shifts to the address part. If the same address is not found as a result of this search, an error is displayed. ce nt e (4) Soft key [W-SRCH] This key specified an address of the relay coil to be searched, and searches the relay coil of the specified address from the program at the cursor part to the end of the program (SUB 48) 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 nc Set the cursor to the relay contact of the relay coil to be searched, and press soft key [W-SRCH].

w w w .c The corresponding relay coil is searched from the program of the cursor part to the end of the program (SUB 48). 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 search, an error occurs. b) Method of specifying the address by inputting it Input the address of the relay coil to be searched by both address and numeric keys, and then, press soft key [W-SRCH]. The specified address relay coil is searched from the program of the cursor part on the presently displayed screen to the end of program (SUB 48). 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 search, an error is displayed. (5) Soft key [N-SRCH] Display the ladder with the specified NET number from the top of the screen. If the number is not typed in but the [N-SRCH] key is pressed, the display is scrolled down by one NET. 450 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 r.c om (6) Soft key [F-SRCH] Type in the functional instruction number and press the [F-SRCH] key to start searching the functional instruction. When the [F-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 (, , ,) D Type in the address or symbol and press the cursor key to start searching the address. When the ”³” key is pressed, the operation is performed in the same manner when the [SRCH] key is pressed. D Type in NET NO. and press the cursor key to start searching the NET NO. D Type in the functional instruction name or functional instruction number with ”S” and press the cursor key to start searching the functional instruction. Example) Type in ”END1” or ”S1” and press the cursor key to search functional instruction END1. Copying the Sequence Program The sequence program with multiple NETs can be copied in NETs. Specify the NET to be copied and the copy position with the cursor. The number of copies can be also specified.

ce nt e 5.2.6 yj jy yj jy Ę j yĘyj FUNCTN COPY MOVE UNTIL CANCEL SEARCH C-DOWN C-UP TO CANCEL SEARCH C-DOWN C-UP 1 Copying 2 Move the cursor to the NET to be copied and press the [COPY] key. The NET to be copied blinks on the screen. Copying multiple NETs 3 Move the cursor with the cursor UP/DOWN key, [C-UP] key, [C-DOWN] key, or [SEARCH] key to blink the NETs to be copied. Type in a value and press [C-UP] or [C-DOWN] key to scroll up or down the screen by the number of times specified by this value.

Setting the NET to be copied 4 Press the [UNTIL] key. Specifying the copying address w w w .c nc RETURN Press the [TO] key to start copying a NET. The NET is copied into the position above the cursor. If the number of copies is typed in before the [TO] key is pressed, the NET is copied the specified number of times. 451 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) 5 B–61863E/10 If the cursor is moved to the first NET and the number of NETs is typed in when the NETs to be copied are known, steps 1 through 3 can be omitted by pressing the [COPY] key. yj jy yj jy COPY MOVE Ę j ce nt e Moving the Sequence Program The sequence program with multiple NETs can be moved in NETs. Specify the NET to be moved and the move position with the cursor. The number of times of moving NETs can be also specified. r.c 5.2.7 om NOTE An error NET cannot be copied. yĘyj FUNCTN RETURN CANCEL SEARCH C-DOWN C-UP TO CANCEL SEARCH C-DOWN C-UP nc UNTIL 1 Moving Move the cursor to the NET to be copied and press the [MOVE] key. The NET to be moved blinks on the screen. 3 Moving multiple NETs Moving the cursor with the cursor UP/DOWN key, [C-UP] key, [C-DOWN] key, or [SEARCH] key to blink the NETs to be moved. Type in a value and press [C-UP] or [C-DOWN] key to scroll up or down the screen by the number of times specified by this value. Setting the NET to be moved Press the [UNTIL] key. 4 w w w .c 2 Specifying the copying address Press the [TO] key to start copying a NET. The NET is moved to the position above the cursor. 5 If the cursor is moved to the first NET and the number of NETs is typed in when the NETs to be moved are known, steps 1 through 3 can be omitted by pressing the [MOVE] key. NOTE An error NET cannot be copied.

452 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 5.2.8 (1) The symbol data and comment assigned to undefined address can be edited. a) Move the cursor to the position where a contact or coil is to be inputted. b) Enter an address, enter the symbol and comment enclosed in characters other than alphanumeric characters, then press the soft key of [contact or coil]. (Example) When the contact X8.4 is assigned the symbol “*ESP” and the comment “EMERGENCY STOP”. Operation : Depress [contact] soft key after entering “X8.4/ *ESP/EMERGENCY STOP/”. (2) The symbol data and comment assigned to the address already defined can be edited. a) Move the cursor on the address part where symbol data or comment will be edited. b) Enter the symbol and comment enclosed in characters other than alphanumeric characters, then press the key. (Example) When the contact X8.4 is assigned the symbol “*ESP” and the comment “EMERGENCY STOP”. Operation : Depress the key after entering “/*ESP/ EMERGENCY STOP/”. (3) The symbol data and comment only can be edited by the similar operation to the above (1) and (2). a) For entering “X8.4/*ESP/” or “/*ESP/” with the “INPUT” key, the symbol data only can be edited. b) For entering “X8.4//EMERGENCY STOP/” or “//EMERGENCY STOP/” with the “INPUT” key, the comment only can be edited. .c nc ce nt e r.c om Editing Symbol Data and Comment at Once While editing a sequence program, symbol data and comment can be edited. 5.2.9 w w w Address Change of Sequence Program The address in a sequence program can be replaced with another address by the procedure below. COPY MOVE O-ADR CANCEL N-ADR CANCEL EXEC CANCEL CHANGE SEARCH C- DOWN C-UP (1) Changing the address while checking it one by one a) Press the [CHANGE] key. 453 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 b) Input the original address and press the [O-ADR] key. c) Input the new address and press the [N-ADR] key. d) Press the [EXEC] key for executing the change. After completion, the cursor will shift downward to the nearest address to be changed. If the [EXEC] key is pressed again at the point, the address change can be continued. (2) Changing the address within the specified range om a) Press the [CHANGE] key, and move the cursor to the address to be changed. b) Input the original address and press the [O-ADR] key.

c) Input the new address and press the [N-ADR] key. r.c d) The specified range will be brightened by using the [C-DOWN] or [C-UP] key. All the addresses within the specified range can be changed. e) Press the [EXEC] key for executing the change. (3) Address designation by a wild card ce nt e The address to be changed can be designated by using the “*” code as a wild card. (Example) “X*.0” means X0000.0, X0001.0, ...

, X9999.0. “X0000.*” means X0000.0, ... , X0000.7. “X*” means X0000, X0001, ... , X9999. The wild card can be used for both of the original address (O-ADR) and new adress (N-ADR).

The following are examples by wild card. w w w .c nc a) “X0.*” to “D100.*” X0000.0 ³ D0100.0 X0000.1 ³ D0100.1 : X0000.7 ³ D0100.7 b) “X*.0” to “X*.7” X0000.0 ³ X0000.7 X0001.0 ³ X0001.7 : X9999.0 ³ X9999.7 [Limit items] D The address of data part in Functional instruction “DISP” cannot be changed. D If the original address (O-ADR) and new address (N-ADR) are different in address name and the byte part of new address (N-ADR) is specified by a wild card, the change can not be done. Example) 454 D1234.0 ³ X*.7, D* ³ X* 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 5.3 I/O UNIT ADDRESS SETTING (MODULE) Set and delet the address of each module in I/O unit as follows. The relation between these functions and soft keys is as shown in the following figure. RUN EDIT I/O TITLE LADDER SYSPRM MONIT om RETURN SYMBOL MESAGE NEXT MODULE INPUT SEARCH DELETE ce nt e RETURN r.c CLEAR NEXT IOSTRT 5.3 Address setting for I/O unit (1) Address setting for each module 1 Press the soft key [MODULE]. nc The following module address setting screen is displayed. Move the cursor to the address to be set, and input data in the order of GROUP, BASE, SLOT, and NAME and press INPUT key. Input a dot (.) as a delimiter of each data.

Example) When setting the AID16A module with group = 0, base = 0, and slot = 5 0. 0. 5. ID16A Table 3.2.2 in Section I-3.2 lists the necessary names for the NAME column. 3 Set all data of the module employed to aimed addresses by using the cursor key and page key. w w w .c 2 455 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 PMC I/O MODULE NAME ADDRESS GROUP BASE SLOT ID16C Y000 3 0 1 ID16C Y001 3 0 5 #2 Y002 #2 Y003 #2 Y004 2 0 1 #2 Y005 2 0 1 #2 Y006 2 1 1 #2 Y007 2 1 1 #2 Y008 2 1 1 #2 Y009 2 1 1 ## Y010 2 0 3 ## Y011 2 0 3 ## Y012 ## Y013 FS04A Y014 0 0 1 NAME #1 #1 OD16C OD16C OD32D OD32D OD32D OD32D OD16C OD16C I/O Unit MODEL B I/O Unit MODEL A om ADDRESS GROUP BASE SLOT X000 2 0 2 X001 2 0 2 X002 3 0 9 X003 3 0 9 X004 3 0 30 X005 3 0 30 X006 3 0 10 X007 3 0 10 X008 3 0 20 X009 3 0 20 X010 3 0 0 X011 3 0 0 X012 3 0 0 X013 3 0 0 X014 0 0 1 FS04A Power Mate ce nt e r.c GROUP.BASE.SLOT.NAME = >2.0.4.OD08C nc CAUTION I/O module data items are made valid in the power-on sequence. When changing settings, be sure to turn off the power and turn it on to validate the settings. However, the power need not be turned off and on again to validate settings when the programmer function version displays the soft key IOSTRT, (described later (item 4).

Press the IOSTRT key after changing data. (2) Delete of address w w w .c A preset address of each module can be deleted as follows: 1 Move the cursor to the address to be deleted, and press soft key [DELET] (see Fig. 5.3). 2 The preset address data are deleted. (3) Soft key [SEARCH] Searches the type-in address. 1 Type in the address to be searched and press the [SEARCH] key. 2 The typed-in address starts being displayed from the top of the screen. (4) Validate the assignment data. I/O module data is validated when the power is turned on. If I/O module data is changed without changing the configuration of the I/O devices, the new I/O module data is validated when the IOSTRT key is pressed. The conditions where the IOSTRT key is displayed depend on the version of the programmer function. 456 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 (5) Error and warning messages issued during the editing of assignment data No. Error or warning message Description ERR : GROUP NO. (0-15) The group number must be from 0 to 15. 2 ERR : BASE NO. (0-3) The base number must be from 0 to 3.

3 WARN : BASE NO. MUST BE 0 The base number must be 0 for the I/O Unit-B. It is forcibly set to 0. 4 ERR : SLOT NO. (1-10) The slot number must be from 1 to 10 for the I/O Unit-A. 5 ERR : SLOT NO. (0, 1-30) The slot number must be from 0 to 30 for the I/O Unit-B. 6 ERR : SLOT NO. MUST BE 0 The slot number must be 0 to set the power-on/off information for the I/O Unit-B. 7 ERR : ILLEGAL NAME An invalid or unsupported assignment name has been entered. Enter a correct name. 8 INPUT INVALID An invalid character string has been entered. Reenter with a correct format. 9 INPOSSIBLE WRITE An attempt has been made to edit ROM data. ROM data cannot be edited. 10 ERR : ADDRESS ALREADY ASSIGNED The specified address is already assigned. Assign another address or retry after deleting the existing data. 11 ERR : ADDRESS OVER An address exceeds the upper limit (X127, Y127). Check the addresses used for the unit to be set. 12 ERR : SLOT ALREADEY DEFINED The specified slot is already assigned. Check the existing data. 13 WARN : SLOT ALREADY DEFINED The specified slot is already assigned. Check the existing data. 14 ERR : UNIT TYPE MISMATCH (IN OR OUT) An X address cannot be assigned to an output module. A Y address cannot be assigned to an input module. 15 ERR : UNIT TYPE MISMATCH (MODEL) ce nt e r.c om 1 w w w .c nc I/O Unit-A and I/O Unit-B are assigned in the same group. I/O Unit-A and I/O Unit-B cannot exist in the same group. 457 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) 5.4 B–61863E/10 A signal name (within 6 alphanumeric characters) can be attached to I/O signals and internal relays employed in sequence programs. SYMBOL DATA SETTING (SYMBOL) Also, a comment (within 30 alphanumeric characters) can be attached to the relay coils in addition to the symbol name. Symbol data and comment are displayed together with a ladder diagram on the CRT/MDI screen as follows. om Signal name (within 6 characters) MA SPDALM X2.4 R2.2 Y4.3 ATCALM R100.1 MRDY APCALM R5.4 MACHINE READY MALM Comment within 30 characters r.c MACHINE ALARM 10 lines APC R52.1 ce nt e R120.1 TIND D20.7 nc If symbol data and comment are defined in signal addresses of the program, the signal name and comment are displayed as Shown in the above figure. If neither symbol data nor comment is defined at an address, the address is displayed as it is. w w w .c A maximum of 64 KBytes can be used for the ladder, symbol, comment, and data. After the program is initialized, the capacity of the symbol area and that of the comment area are usually 32KB (28KB for PMC–NB) each. When additional data is entered causing the total amount of data in either area to exceed 32KB (28KB for PMC–NB), the area is automatically extended in 1KB units. STOP EDIT I/O SYSPRM TITLE LADDER SYMBOL MESAGE INPMOD DELETE SEARCH MONIT RETURN RETURN COPY 5.4(a) Setting and display of symbol data 458 PMC PROGRAMMER (CRT/MDI) B–61863E/10 5. EDITING OF SEQUENCE PROGRAM (EDIT) When soft key [SYMBOL] is pressed, the following screen is displayed, and the soft key operation is done hereafter. r.c om SYMBOL & COMMENT 001 ADDRESS SYMBOL COMMENT *---------* *------*-------*--------------* ce nt e 5.4(b) Symbol data screen 5.4.1 w w .c nc Symbol Data and Comment Input Refer to Fig. 5.4 (a) and Fig. 5.4 (b). Input symbol data and comment on the screen shown in Fig. 5.4 (b). 1 Press key after inputting an address where a symbol and a comment are to be set. The input address is set to the ADDRESS column of Fig. 5.4 (b), and the cursor shifts to the address. The input addresses are arrange and set in the alphabetic sequence, and they can be inserted halfway. 2 For setting a symbol, shift the cursor rightward by using the cursor key. 3 After setting symbol data (within 6 alphanumeric characters), press INPUT key. The symbol data are set in the SYMBOL column of Fig. 5.4 (b), and the cursor shifts to the COMMENT column. 4 For inputting a comment, set comment data (within 30 alphanumeric characters), and press INPUT key. Comment data are set to the COMMENT column in Fig. 5.4 (b). Repeat steps 1 to 4 hereafter. w 5.4.2 Symbol Data Search (SRCH) 5.4.3 Delete of Symbol data and Symbol data and Comment Display the screen shown in Fig. 5.4 (b) and search symbol data (1) After setting an address or symbol data to be searched, press soft key [SRCH]. Specified address or symbol data is searched and displayed on the screen. The cursor shifts to the corresponding address part. Move the cursor to the address to be deleted in the ADDRESS column of Fig. 5.4 (b), and press soft key [DELETE]. 459 5. EDITING OF SEQUENCE PROGRAM (EDIT) 5.4.4 Editing Character Strings of Symbol Data and Comment Data PMC PROGRAMMER (CRT/MDI) B–61863E/10 The edit modes can be changed by pressing the [INPMOD] soft key as follows: Character string edit status Maximum character input Insertion mode (INSERT) Replacement mode (ALTER) D When the key is pressed (1) In the character string edit status om ”INSERT” is displayed on the screen in the insertion mode.

“ALTER” is displayed on the screen in the replacement mode. ce nt e r.c Insertion mode : The entered character is inserted at the cursor. If the [INPUT] key is pressed after no character is entered, one space is inserted. Replacement mode : The character at the cursor is replaced with the entered character. If the [INPUT] key is pressed after no character is entered, the character at the cursor is replaced with one space. (2) When the length of the cursor is the same as the number of characters that can be entered. The original character string are replaced with the entered characters. D When the key is pressed (1) In the character string edit status w w w .c nc Insertion mode : The character at the cursor is deleted. Replacement mode : The character at the cursor is replaced with a space. (2) When the length of the cursor is the same as the number of characters that can be entered The character string at the cursor is deleted. NOTE The cursor having the size equivalent to the total size of the maximum number of characters that can be entered is displayed in the address section. 5.4.5 An address, symbol, and comment can be entered at one time. Function for editing symbol data and comment data at one time (1) Editing the symbol and comment assigned to address not defined Enter an address, enter the symbol and comment enclosed in characters other than alphanumeric characters, then press the key. The cursor may be located anywhere. A comment can be omitted. Example) G0.4 / *EMG / EMERGENCYSW/ key Address Symbol Comment 460 PMC PROGRAMMER (CRT/MDI) B–61863E/10 5. EDITING OF SEQUENCE PROGRAM (EDIT) (2) Editing the symbol and comment assigned to address already defined Move the cursor to the desired line of the address whose symbol and comment are to be edited. Next, enter the symbol and comment enclosed in characters other than alphanumeric characters. Then, press the key. A comment can be omitted. 5.4.6 Function of Copying Symbol and Comment Data INPMOD DELETE ALL UNTIL EXEC COPY CANCEL ce nt e SYMBOL SEARCH r.c RETURN om Copy the specified data to re-edit and register it.

INPMOD DELETE CANCEL CANCEL (1) Press the [COPY] soft key. nc (2) Select data to be copied with the corresponding soft key. [ALL] : The address, symbol data, and comment data are copied. .c [SYMBOL] : The address and symbol data are copied. When either of the above soft keys is pressed, the line at the cursor is specified as the beginning of the range of the data to be copied. w w w (3) Specify the range with the [±] and [°] cursor keys.

D A range of more than 15 lines cannot be specified. Up to 15 lines can be displayed on one screen. D A range cannot be specified at a position above the cursor position. When the copy range of the data is specified, the data is displayed differently. (4) Press the [UNTIL] soft key to determine the copy range. (5) Edit the address and symbol data according to the procedure described in Sections 5.4.1 and 5.4.4. (6) When updating the data is completed, press the [EXEC] soft key to register the copied data. 461 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 5.5 Message data are used for PMC functional instruction DISPB (SUB 41). MESSAGE DATA SETTING (MESSAGE) The setting and display methods are as shown below. RUN (STOP) EDIT I/O TITLE LADDER INPMOD DELETE SYSPRM MONIT RET RET DSPMOD r.c COPY NEXT D.CHAR ce nt e ETC SRCH om SYMBOL MESAGE 5.5(a) When soft key MESAGE is pressed, the next screen is displayed, and setting operation can be done hereafter. 001 MESSAGE A 01.1 w w w .c nc MESSAGE NO. A 00.0 A 00.1 5.5(b) Message data screen After initialization, the capacity of the message area is approx. 2.1KB. When additional data is entered causing the total amount of data in the area to exceed 2.1KB, the message area is automatically extended in 1KB units to a maximum of 64KB.

462 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 5.5.1 Message Data Input Refer to Fig. 5.5(a) and Fig. 5.5 (b). Display the screen shown in Fig. 5.5 (b). 1 Display a number to set a message data by using PAGE key. 2 Shift the cursor to this number by the cursor key. 3 After setting message data, press INPUT key. om If the message data has already been entered, it is deleted and the set data is entered. Repeat steps 1 to 3 hereafter. 5.5.2 (1) Specify an address to be searched for, and press the [SRCH] soft key. Then, the specified address is searched for and displayed on the screen. r.c Searching for an Address (SRCH) Editing a Character String in Message Data Edit modes can be changed by pressing the [INPMOD] soft key as follows: ce nt e 5.5.3 Character string edit status Maximum character input (32 characters) Insertion input (INSERT) Replacement input (ALTER) The edit mode is changed every time the [INPMOD] soft key is pressed. nc [INSERT] is displayed on the screen in the insertion mode. [ALTER] is displayed on the screen in the replacement mode. D When the key is pressed w w w .c (1) In the character string edit status Insertion mode : The entered character is inserted at the cursor. Replacement mode : The character at the cursor is replaced with the entered character. (2) When the maximum number of characters are entered The original character string at the cursor is replaced with the entered characters. D When the key is pressed (1) In the character string edit status Insertion mode : The character at the cursor is deleted. Replacement mode : The character at the cursor is replaced with a space.

(2) When the maximum number of characters are entered The character string at the cursor is deleted. (3) When the cursor is located at the address field The entire message data specified at the address is deleted. 463 Input with a Katakana Identification Code 5.5.5 Copying Message Data (COPY) 5.5.6 Inputting a Multi-byte Character (D.CHAR) B–61863E/10 If no ”@” key on the MDI key, pressing the [ETC] soft key enables the operator to enter the data enclosed between at signs (@). When the soft key is pressed, ”ETC CODE” is displayed on the screen. Move the cursor to the message number to be copied and press the [COPY] key. Then press the [EXEC] key after moving the cursor to the message number in which it is copied. om 5.5.4 PMC PROGRAMMER (CRT/MDI) The input mode becomes multi-byte character by pressing the [D.CHAR] key.

r.c 5. EDITING OF SEQUENCE PROGRAM (EDIT) (@02, 01@ are added to input data automatically.) 5.5.7 The ASCII code enclosed with @ characters is displayed in the form of screen display by pressing the [DSPMOD] key. Example) Katakana : “@B6C532@” ³ “ ” is displayed. Multi-byte character : “@” ” is displayed. ³“ w w w .c nc Displaying Input Code (DSPMOD) ce nt e For example, “4873 [INPUT]” is processed as “@02487301@”. 464 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 5.6 CLEARING THE SEQUENCE PROGRAM AND CONDENSATION OF THE SEQUENCE PROGRAM RUN (STOP) EDIT I/O TITLE LADDER SYSPRM RET SYMBOL MESAGE MODULE CLRLAD CLEAR CLRSYM CLRMSG CLRLNG r.c CLRTTL CROSS om NEXT RET CLRMDL CONDNS CLRPRM CLRTMR CLRCNT ce nt e CLRALL CLRKPR NEXT CLREXT CLRDT NEXT nc CLRALL 5.6.1 Clears each data in the sequence program w w w .c Clearing the Sequence Program The function of the key is as follows: (1) [CLRTTL] : Clears the title data. (2) [CLRLAD] : Clears the ladder program. (3) [CLRSYM] : Clears the symbol and comment data. If the extend symbol and comment data is cleared, the field is restored to the original size. (4) [CLRMSG] : Clears the message data. If the extend message data is cleared, the field is restored tothe original size. (5) [CLRLNG] : The C language area is cleared. Clear the C language area before transferring a C program. When a C language board is installed in the Series 16i/18i, this item is displayed. (6) [CLRALL] : Clear all data described in the above (1) to (4). Clear also the C language programs for models which create C language programs, such as models PMC-RC, RC3, RC4, NB, and NB2. Press this soft key when the message “PLEASE CLEAR ALL” is displayed. 465 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 (7) [CLRMDL] : Clears the I/O module data. (8) [CONDNS] : Compress the sequence program in 1KB units. The detail will be explained chapter 5.6.2. (9) [CLRPRM] : Clears each parameter data. The detail will be explained chapter 5.6.3. (10) [CLREXT] : Clears the expand nonvolatile memory (valid for PMC-RC/RC3/RC4/NB/NB2) r.c om NOTE When using a system that incorporates flash ROM, clear the flash ROM before writing to it. If the power is turned on again without performing this operation, sequence program data is not cleared. Compresses the sequence program in 1KB units. Compress the Sequence Program (1) [CONDNS] : Compresses the unused area in the message, symbol, or comment area in the sequence program in 1KB units when the capacity of the unused area extends 1KB. The unused area, which is the size less than 1KB, will not be compressed. 5.6.3 Clears each PMC parameter. Clearing the PMC Parameter The function of the key is as follows: ce nt e 5.6.2 nc (1) [CLRTMR] : Clears timer data. : Clears counter data. (3) [CLRKPR] : Clears keep relay data. (4) [CLRDT] : Clears data table. (5) [CLRALL] : Clear all data described in the above (1) to (4).

NOTE These functions require the same condition as PMC data setting in operation For [CLRALL], all conditions are required. See “Chapter II.4. PMC PARAMETER SETTING AND DISPLAY” w w w .c (2) [CLRCNT] 466 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 5.7 CROSS REFERENCE DISPLAY The cross reference is displayed for PMC address and functional instruction used in a sequence program. Cross reference display has the following functions. (1) Display NET number by specifying the PMC address.

(2) Display the address list by specifying PMC address name (G, F, Y, · ·).

(3) Display a functional instruction list. om (4) Display NET number by specifying the functional instruction number. 5.7.1 Pressing the [CROSS] key displays the cross reference screen for setting parameters. r.c Operation ce nt e Press soft key [CRSREF] in the parameter setting screen for displaying the cross reference of address and functional instruction in use. Press soft key [CRSPRM] to return to the parameter setting screen from cross reference display. RUN (STOP) EDIT TITLE LADDER I/O SYSPRM MONIT nc RET MODULE SYMBOL MESAGE CLEAR CROSS CRSPRM w w w .c RET CRSREF 467 INPUT NEXT INIT 5.

EDITING OF SEQUENCE PROGRAM (EDIT) 5.7.2 PMC PROGRAMMER (CRT/MDI) B–61863E/10 To display cross reference, the address and reference type need to be specified on the parameter screen. Display the reference of addresses which are used. 1. In “SELECT CROSS TYPE”, input “1”. 2. In “1: REFERENCE ADDRESS”, input addresses which should be displayed. (maximum number of input: 8) 3. Press [CRSREF] key. om Parameter Setting Screen The address, symbol, relay and the NET number will be displayed as shown in Fig. 5.7.2 (b). ce nt e r.c PMC CROSS REFERENCE SELECT CROSS TYPE = 1 (1:ADDRESS 2:ADRS KIND 3:FUNCTION.NO) 1:REFERENCE ADDRESS 1 = X0000.0 5 = 2 = 6 = 3 = 7 = 4 = 8 = 2:ADRS KIND = (G /F /Y /X /A /R /K /C /D /P /L) 3:FUNCTION.NO = (ALL=0) [CRSREF] [INPUT] [nc 5.7.2 (a)] [INIT] [ ] Cross Reference Setting (TYPE1) X0000.0 ( ) ABCDE : : 1 4 2 32 w w w .c PMC CROSS REFERENCE [CRSPRM] [5.7.2 (b) 468] [ ] [ ] [Cross Reference Display (TYPE1)] 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 Display the reference of each address name 1. In “SELECT CROSS TYPE”, input “2”. 2. In “2: ADRS KIND”, input the address name. 3. Press [CRSREF] key. The bit/byte addresses and the related symbol in the sequence program will be displayed as shown in Fig.5.7.2 (d). r.c om PMC CROSS REFERENCE SELECT CROSS TYPE = 2 (1:ADDRESS 2:ADRS KIND 3:FUNCTION.NO) 1:REFERENCE ADDRESS 1 = X0000.0 5 = 2 = 6 = 3 = 7 = 4 = 8 = 2:ADRS KIND = X (G /F /Y /X /A /R /K /C /D /P /L) 3:FUNCTION.NO = (ALL=0)] [INIT ce nt e [CRSREF] [INPUT] [5.7.2 (c)] [ ] Cross Reference Setting (TYPE2) PMC CROSS REFERENCE HEAD CHARACTER = = = X0000.0 ABCDE X0000 SYMBOL NOTHING [CRSPRM] [5.7.2 (d) w w w .c nc USED ADDRESS SYMBOL NAME X 469] [ ] [ ] [Cross Reference Display (TYPE2)] 5. EDITING OF SEQUENCE PROGRAM (EDIT) Display a functional instruction list in use PMC PROGRAMMER (CRT/MDI) B–61863E/10 1. In “SELECT CROSS TYPE”, input “3”. 2. In “3: FUNCTION. NO”, input “0”. 3.

Press [CRSREF] key. The functional instruction name and the functional instruction number in the sequence program will be displayed as shown in Fig.5.7.2 (f). ce nt e r.c om PMC CROSS REFERENCE SELECT CROSS TYPE = 3 (1:ADDRESS 2:ADRS KIND 3:FUNCTION.NO) 1:REFERENCE ADDRESS 1 = X0000.0 5 = 2 = 6 = 3 = 7 = 4 = 8 = 2:ADRS KIND = (G /F /Y /X /A /R /K /C /D /P /L) 3:FUNCTION.NO = (ALL=0) [CRSREF] [INPUT] [5.7.2 (e)] [INIT] [ ] Cross Reference Setting (TYPE3) nc PMC CROSS REFERENCE END1(COD([CRSPRM] [ ] [1) 7) END2(2) w w w .c FUNCTION NAME 5.7.2 (f) 470] [ ] [Cross Reference Display (TYPE3)] 5. EDITING OF SEQUENCE PROGRAM (EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 Display the reference of functional instruction (FUNCTION. NO = number of the functional instruction) 1. In “SELECT CROSS TYPE”, input “3”. 2. In “3: FUNCTION. NO”, input the functional instruction number. 3. Press [CRSREF] key. The functional instruction name, functional instruction number and NET number in the sequence program will be displayed as shown in Fig.5.7.2 (g).

7 COD 6 14 ce nt e r.c FUNCTION NO. = FUNCTION NAME = USED NET NO. = om PMC CROSS REFERENCE [CRSPRM] [w w w .c nc 5.7.2 (g) 471] [ ] [ ] [ ] Displaying of Cross Reference (TYPE3) 6. EXECUTION OF A SEQUENCE PROGRAM B–61863E/10 EXECUTION OF A SEQUENCE PROGRAM w w w .c nc ce nt e r.c om 6 PMC PROGRAMMER (CRT/MDI) 472 PMC PROGRAMMER (CRT/MDI) B–61863E/10 6.1 6. EXECUTION OF A SEQUENCE PROGRAM Start and Stop of a sequence program are described as follows. START AND STOP OF A SEQUENCE PROGRAM PMCLAD PMCDCN PMCPPM RETURN RUN or STOP EDIT I/O SYSPRM MONIT om 6.1 The sequence program execution software key r.c An operable sequence program is usually automatically started when power is turned on if the program is stored in ROM. However, the program is not started if it is stored in RAM. (1) Start of a sequence program (RUN) ce nt e When a sequence program is at the stopped state, pressing the [RUN] key causes the sequence program to run displaying the software key as [STOP]. The ladder program starts from the beginning. However, whether C-language programs start from the beginning depends on the function selected in advance. a) When a C-language program starts from the beginning Functions selected in advance: Ladder editing, reading the system parameter, reading a sequence program using input/output processing nc b) When a C-language program does not start from the beginning but restarts from the next step after stopping Function selected in advance: Functions other than the function in item (a) (2) Stop of a sequence program (STOP) When a sequence program is at the run state, pressing the [STOP] key causes the sequence program to stop displaying the software key as [RUN]. w w w .c NOTE Both ladder and C-language programs start from the beginning immediately after the power is turned on. 473 6. EXECUTION OF A SEQUENCE PROGRAM 6.2 PMC PROGRAMMER (CRT/MDI) B–61863E/10 The sequence program can be automatically started immediately after power–on, when bit 2 of keep relay K17 (PMC parameter) is set to 1.

STARTING THE SEQUENCE PROGRAM om NOTE For the Series 16/18–MODEL B/C, Series 16i/18i/21i–MODEL A, automatic start is specified when bit 2 of K17 is set to 0. 6.3 To forcibly stop starting the sequence program in ROM or RAM, at power-on, turn on the power by pressing the [Z] key while pressing the [CAN] key. (Except for PMC–NB/NB2) FORCIBLY STOPPING THE SEQUENCE PROGRAM r.c This method is effective for locating the error when a system error occurs after power is turned on and when the error may be caused by the sequence program.

w w w .c nc ce nt e Never perform this operation in a normal state. 474 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PARAMETER DATA PMC PROGRAMMER (CRT/MDI) B–61863E/10 7 WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PARAMETER DATA EDIT EXEC CANCEL SYSPRM (NO.) w w w .c nc ce nt e RETURN I/O r.c RUN om When the [I/O] key is pressed, the sequence program and PMC data are written, read, or collated for the specified device. Operations are performed with cursor keys and soft keys. 475 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PROGRAMMER (CRT/MDI) PMC PARAMETER DATA 7.1 B–61863E/10 When the [I/O] key is pressed, the sequence program and PMC data are written, read, or collated for the specified device. Operations are performed with cursor keys and soft keys. OVERVIEW RUN EDIT I/O EXEC CANCEL (NO.) SYSPRM om RETURN NEXT r.c SPEED MONIT STOP ce nt e PMC I/O PROGRAM CHANNEL DEVICE = 1 = HOST DATA KIND = (ALL:LADDER + LANGUAGE) FUNCTION = nc > ALM .c [EXEC] [CANCEL] [HOST] [FDCAS] [F-ROM] PMC I/O PROGRAM CHANNEL w w [Case of FS16/18] DEVICE MONIT STOP = 1 = HOST w DATA KIND = FUNCTION = > [EXEC] [CANCEL] [HOST] [FDCAS] [ROMWRT] [ ] [OTHERS] [SPEED] [476] [COPY] 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PARAMETER DATA PMC PROGRAMMER (CRT/MDI) B–61863E/10 om NOTE The sequence program can be output while the ladder is being executed, but the output speed is low. When the sequence program is input while the ladder is being executed, the execution of the ladder is automatically stopped.

7.1.1 Conventionally, C programs have been managed together with ladder programs. For this reason, when only a ladder program is to be modified as shown in Fig. 7.1.1 (a), the program linked with a C program needs to be replaced. With the FANUC Series 16i/18i/160i/180i, ladder programs and C programs are managed separately. So, each program can be edited and replaced independently of the other programs as shown in Fig. 7.1.1 (b). ce nt e r.c C Input/Output Ladder + C nc Download Upload 7.1.1 (a) Ladder and C Structure for FANUC Series 16B/16C NC Ladder Download Upload Ladder C Download Upload C w w Ladder + C Download w .c C NC 7.1.1 (b) Ladder and C Structure for FANUC Series 16i/18i/160i/180i 477 7.

WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PROGRAMMER (CRT/MDI) PMC PARAMETER DATA 7.2 B–61863E/10 (1) CHANNEL Specify which connector the reader/punch interface (such as RS-232C) is connected to. CHANNEL must be set when HOST, FDCAS, or OTHERS is selected for DEVICE. (2) DEVICE Select the device with which the PMC inputs or outputs data, using soft keys.

SET ITEMS Soft key Description Transfers data with a FAPT LADDER (P-G, P-G Mate, or personal computer). (See Subsection 7.3.1 for details.) FDCAS Transfers data with a FANUC FD cassette. (See Subsection 7.3.2 for details.) F-ROM Transfers data with flash EEPROM. This is where the sequence program is stored. (See Subsection 7.3.3 for details.) M-CARD Transfers data with a memory card. (See Subsection 7.3.4 or details.) OTHERS Transfers data with other input/output devices. (See Subsection 7.3.5 for details.) SPEED Used to set transfer conditions when RS-232C is used. (See Subsection 7.3.6 for details.) ce nt e r.c om HOST ROMWRT Transfers data with a ROM WRITER. (See Subsection 7.3.6 for details) NOTE Some functions cannot be used with some PMC models. See Section 7.3. nc (3) DATA KIND Select the type of output data using soft keys. DATA KIND must be set when FDCAS, M-CARD, or OTHERS is selected for DEVICE. DATA KIND ALL Output the data of sequence program and C program LADDER Output the data of sequence program (Ladder, Symbol, Comment, Message, etc.) PARAM (Note 1) Output PMC Parameters (TIMER, COUNTER, KEEP RELAY, DATA TABLE, etc.) C–LANG (Note 2) Outputs a C program. .c w w w CONTENTS NOTE 1 The conditions of outputting PMC parameters 1) When sequence program is stopped You can input/output them. 2) When sequence program is executed You must satisfy the following conditions. Output (WRITE) . Set NC to “EDIT” mode. Input (READ) . . . . Set NC to “Emergency Stop” status, and, set “PWE” of NC parameters to 1. 2 When a C language board is installed in the Series 16i/18i, the item of C–LANG is displayed, and C program output is enabled.

478 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PARAMETER DATA PMC PROGRAMMER (CRT/MDI) B–61863E/10 (4) FUNCTION Select the direction of data transfer between the PMC and input/output device. Item Description Outputs data from the PMC to an input/output device. READ Inputs data from an input/output device to the PMC.

COMPARE Collates data in the PMC and an input/output device. (Note) PMC data cannot be collated. DELETE Deletes a file in FDCAS or M-CARD. LIST Lists the files in FDCAS or M-CARD. BLANK Performs blank check for F-ROM. ERASE Clears the data in F-ROM. FORMAT Initializes M-CARD (clears all data). r.c (5) FILE NO. om WRITE ce nt e FILE NO.

is displayed when FDCAS or M-CARD is selected for DEVICE. Specify the file number or file name for WRITE, READ, COMPARE, or DELETE processing. Note the following restrictions on the file name when FDCAS or M-CARD is selected for DEVICE: M-CARD (MEMORY CARD) Number of characters in the file name Up to 17 characters following @ or #. The file is written after the existing files. Up to 8 characters following @ or #(*1). When the same name as an existing file is specified An error occurs. Delete the existing file and reoutput the new file. The new file is written over the existing file (the contents of the existing file are lost). When -1 is specified for the file name The file is written after the existing files. The system names the file and writes it(*2).

When 0 is specified for the file name The file is written and all the existing files are deleted. The system names the file and writes it(*2). nc FDCAS (FANUC FD CASSETTE) w w w .c NOTE 1 Name the file in the MS-DOS format (up to eight characters for the file name with up to three characters for the extension). Example) FILE NO. = @12345678.123 FILE NO. = @LADDER.EXE 2 If the file name is not specified, the system names the file as follows: DATA KIND File name ALL model-name.ALL LADDER model-name.LAD PARAM model-name.PRM 479 The model name is PMC-NB for the PMC-NB and PMC-RA for the PMC-RA1 or PMC-RA3.

7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PROGRAMMER (CRT/MDI) PMC PARAMETER DATA B–61863E/10 7.3 OPERATIONS 7.3.1 (a) Setting the channel Move the cursor to ”CHANNEL = .” Check that an RS-232C cable is connected to the main board. Enter the number (1 or 2) corresponding to the connector. The correspondence between the connector and CHANNEL is as follows: om Transfer to and from a FAPT LADDER CHANNEL = 1 : JD5A (main board) CHANNEL = 2 : JD5B (main board) (b) Setting the transfer conditions r.c Press the [SPEED] soft key and set each condition. See Subsection 7.3.6 for details. (c) Writing, reading, or collating the sequence program Item Operation Press the [HOST] soft key. ce nt e DEVICE Press the [EXEC] soft key and to make the NC ready for operation.

Select necessary items on a FAPT LADDER and start transfer. nc NOTE WRITE, READ, or COMPARE is automatically switched by operation on a FAPT LADDER. 7.3.2 PMC I/O PROGRAM CHANNEL w .c Transfer to and from a FANUC FD Cassette Reads or writes the sequence program, Pascal or C programs, or PMC data. w w DEVICE MONIT STOP = 1 = FDCAS DATA KIND = ALL (ALL:LADDER + LANGUAGE) FUNCTION = WRITE FILE NO. = -1 (-1:ADD,0:INIT,OR @ NAME) > ALM [EXEC] [CANCEL] [HOST] [FDCAS] [F-ROM] (a) Setting the channel Enter the number of the channel used at ”CHANNEL = .” See (a) in Subsection 7.3.1 for details. 480 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PARAMETER DATA PMC PROGRAMMER (CRT/MDI) B–61863E/10 (b) Setting the transfer conditions Press the [SPEED] soft key and set each condition. See Subsection 7.3.6 for details. (c) Writing a file Item Operation Press the [FDCAS] soft key. FUNCTION Press the [WRITE] soft key. DATA KIND Select the type of data to be output (see (3) in Section 7.2). FILE NO. Name the file within 17 characters. -1 is displayed if no name is entered (see (5) in Section 7.2). om DEVICE Press the [EXEC] soft key to start outputting the file. (d) Reading a file Operation r.c Item DEVICE Press the [FDCAS] soft key. FUNCTION Press the [READ] soft key. FILE NO. Enter the number or name of the file to be input. ce nt e Press the [EXEC] soft key to start inputting the file. (e) Collating a file Item Operation DEVICE Press the [FDCAS] soft key. FUNCTION Press the [COMPAR] soft key. FILE NO. Enter the number or name of the file to be collated. nc Press the [EXEC] soft key to start collating the file.

.c Note PMC data cannot be collated. The data the file is to be collated with depends on the file. w w w (f) Deleting a file Item Operation DEVICE Press the [FDCAS] soft key. FUNCTION Press the [DELETE] soft key. FILE NO. Enter the number or name of the file to be deleted. Press the [EXEC] soft key to start deleting the file. (g) Listing the files Item Operation DEVICE Press the [FDCAS] soft key. FUNCTION Press the [LIST] soft key. Press the [EXEC] soft key to start listing the files. 481 7.

WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PROGRAMMER (CRT/MDI) PMC PARAMETER DATA B–61863E/10 7.3.3 Ę : Supported : Not supported Storage to Flash ROM Power Power Mate-H FS21/ Mate-D/F/G FS20 210MB FS16–A Ę FS21i FS16–B FS16–C FS16i FS18–B FS18–C FS18i Ę Ę Ę FS15B Ę om Ę FS18 Formerly, a RAM module or ROM module was necessary for storing programs. Using Flash Memory, however, programs can be ROM-stored on the PMC board. nc ce nt e r.c CAUTION 1 If the power is turned off without performing the writing operation, the updated sequence program is not stored. 2 The CNC must be placed in the emergency stop state when data is read from or written to Flash Memory. 3 Even if the sequence program is cleared with the X and O keys at power-on, the contents of Flash Memory are not cleared.

Therefore, when the power is turned on again, the sequence program in Flash Memory is read. To clear the contents of Flash Memory, write Flash Memory after clearing the sequence program with X and O. PMC I/O PROGRAM .c CHANNEL DEVICE = 1 = F-ROM DATA KIND = (ALL:LADDER + LANGUAGE) FUNCTION = WRITE w w w MONIT STOP RAM SIZE = A (MAX SIZE = B) PROGRAM ALREADY EXISTS (EXEC?) > ALM [EXEC] [CANCEL] [HOST] [FDCAS] [F-ROM] (a) Writing data to Flash Memory Item Operation DEVICE Press the [F-ROM] soft key. FUNCTION Press the [WRITE] soft key.

Press the [EXEC] soft key to start outputting data. 482 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PARAMETER DATA PMC PROGRAMMER (CRT/MDI) B–61863E/10 ce nt e r.c om NOTE 1 Operation in PMC–NB (a)If data is stored in Flash Memory, a message is displayed to confirm writing. (b)RAM SIZE indicates the size of the sequence program. MAX SIZE indicates the size of the ROM option in the PMC. A: 64K bytes, B: 128K bytes, C: 256K bytes, D: 512K bytes, E: 1M bytes 2 When a C board is installed in the Series 16i/18i, ladder programs and C programs must be written into Flash Memory separately. (a)When a ladder program is written into Flash Memory, select [LADDER] from the DATA KIND item. When a C program is written into Flash Memory, select [C–LANG] from the DATA KIND item. (b)Select [WRITE] from the FUNCTION item. (c) Press the [EXEC] soft key to start writing to Flash Memory. (b) Reading data from Flash Memory Item Operation DEVICE Press the [F-ROM] soft key. FUNCTION Press the [READ] soft key. Press the [EXEC] soft key to start inputting data. (c) Comparing data in Flash Memory Operation nc Item DEVICE Press the [F-ROM] soft key. FUNCTION Press the [COMPAR] soft key. w w w .c Press the [EXEC] soft key to start comparing data. (d) Checking if data is stored in Flash Memory Item Operation DEVICE Press the [F-ROM] soft key.

FUNCTION Press the [BLANK] soft key. Press the [EXEC] soft key to check if data is stored in Flash Memory. NOTE Operation in PMC–NB When data is stored in Flash Memory : BLANK ERROR is displayed. When no data is stored in Flash Memory : BLANK COMPLETE is displayed.

(e) Deleting data in Flash Memory Item Operation DEVICE Press the [F-ROM] soft key. FUNCTION Press the [ERASE] soft key. 483 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PROGRAMMER (CRT/MDI) PMC PARAMETER DATA B–61863E/10 Press the [EXEC] soft key to start deleting data. 7.3.4 om NOTE In FS16B/18B, [READ], [COMPAR], [BLANK] and [ERASE] functions are unavailable. Ę : Supported : Not supported Storage to a Memory Card Ę Ę FS18 FS16–A FS21i FS16–B FS16–C FS16i FS18–B FS18–C FS18i Ę Ę r.c Power Power Mate-H FS21/ Mate-D/F/G FS20 210MB Ę FS15B Ę ce nt e NOTE This function is not supported on DPL/MDI of Power Mate–H. Sequence programs and data are input from or output to a memory card as described below. The memory card to which data is input from or output to can directly send or receive data to or from the programming unit (FAPT LADDER). The supported function and the kind of memory card is shown as below. .c nc Any kind of card has to be conformed to TYPE 1 to 2 of PCMCIA (The Personal Computer Memory Card International Association) 2.0 (or later) or TYPE 1 to 2 of JEIDA (Japanese Electronics Development Association) 4.0 (or later) .

And the format is based on MS–DOS FAT file system. The size of memory–card that can be used is up to 32Mbytes. w w w The case of FS20,FS18B,FS16B FS18C, FS16C PMC Ę : Supported : Not supported SRAM Card Flash Memory Card Read of a file Ę Ę Format of a card Ę Write of a file Ę Delete of a file Ę List of a file Ę 484 Ę 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PARAMETER DATA PMC PROGRAMMER (CRT/MDI) B–61863E/10 The case of FS16i, FS18i, FS15B(PMC–NB) Ę : Supported : Not supported Flash Memory Card SRAM Card Supported Card Unsupported Card Ę Ę Ę Format of a card Ę Ę Write of a file Ę Ę Delete of a file Ę List of a file Ę om Read of a file Ę (1) Flash memory card writing Ę The following kinds of flash memory card are supported. r.c S Intel Series 2 Flash Memory Cards (or compatible cards) Attribute memory is needed for any card. ce nt e Files can be written on the card that is formatted by MS–DOS. But there are following limitations.

S It is impossible to alter a file that is already written. S A card that is formatted by Flash File System can not be used. (Neither Read nor List) S The data can not be written in the last 128Kbyte of the card. So, w w w .c nc available size of a card is (Card_size – 128Kbyte). Please refer to the following figure. 485 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PARAMETER DATA PMC PROGRAMMER (CRT/MDI) PMC PARAMETER DATA B–61863E/10 Before Writing File–A File–B om File–C Add File–D r.c 128Kbyte File–A File–B File–C w w w .c 128Kbyte nc File–D ce nt e After writing ”CLOSE ERROR” is displayed and File–D cannot be saved. ,the data of File–D is written. But

"read" and In the part of the "list" functions are not available for File–D. After this operation, any file cannot be written to this card. There are following limitations due to the system that formats the flash memory card. (a) When the card formatted and written files by FANUC products is used by other systems.
Ramu–zou Note1) Ę Read of a file Add of file Not supported function List of file Ę 486 CardPro Note2) Ę Ę 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PARAMETER DATA PMC PROGRAMMER (CRT/MDI) B–61863E/10 NOTE 1 Ramu–zou is a memory card reader/writer that is made by ADTEK SYSTEM SCIENCE.
2 CardPro is a memory card reader/writer that is made by Data I/O.
Ramu–zou Ę Add of file Ę List of file Ę CardPro Note3) Ę Ę r.c Read of a file om (b) When the card formatted and written files by other system is used by FANUC products.
ce nt e NOTE If you use the CardPro to format a flash memory card, type the following command. CPFORMAT drive–name: /F:FLASHFAT /NOCIS (2) Operation The operation is almost the same as Subsection 7.3.2 except that steps (a) and (b) are not necessary for a memory card. nc PMC I/O PROGRAM CHANNEL DEVICE 1 = M–CARD DATA KIND = PARAM (ALL:LADDER + LANGUAGE) FUNCTION = WRITE .c w w w = MONIT STOP FILE NO. = –1 (–1:ADD, 0:INIT, OR@ NAME) > ALM [M–CARD] [OTHER] [ ] [ ] [(a) Formatting the memory card Item Operation DEVICE Press the [M-CARD] soft key. FUNCTION Press the [FORMAT] soft key. Press the [EXEC] soft key to start formatting. 487] 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PROGRAMMER (CRT/MDI) PMC PARAMETER DATA B–61863E/10 (b) Writing a file Item Operation DEVICE Press the [M–CARD] soft key. FUNCTION Press the [WRITE] soft key. DATA KIND Select the type of data to be output. (See (3) in Section 7.2) FILE NO. Name the file within 8 characters. –1 is displayed if no name is entered. (See (5) in Section 7.2) om Press the [EXEC] soft key to start outputting the file. If the file name is not specified, the system names the file as follows: DATA KIND File name model-name.ALL LADDER model-name.LAD PARAM model-name.PRM (c) Reading a file Operation ce nt e Item The model name is PMC-NB for the PMC-NB and PMC-RA for the PMC-RA1 or RA3. r.c ALL DEVICE Press the [M–CARD] soft key. FUNCTION Press the [READ] soft key. FILE NO. Enter the number or name of the file to be input. Press the [EXEC] soft key to start inputting the file. (d) Collating a file Item Operation Press the [M–CARD] soft key. FUNCTION Press the [COMPAR] soft key. FILE NO. Enter the number or name of the file to be collated.
nc DEVICE .c Press the [EXEC] soft key to start collating the file. w w w NOTE PMC data cannot be collated. (e) Deleting a file Item Operation DEVICE Press the [M–CARD] soft key. FUNCTION Press the [DELETE] soft key. FILE NO. Enter the number or name of the file to be deleted. Press the [EXEC] soft key to start deleting the file. (f) Listing the files Item Operation DEVICE Press the [M–CARD] soft key.
FUNCTION Press the [LIST] soft key. Press the [EXEC] soft key to start listing the files. 488 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PARAMETER DATA PMC PROGRAMMER (CRT/MDI) B–61863E/10 7.3.5 Data Input to and Output from other Devices Ę : Supported : Not supported Power Mate FS20 FS21/ 210MB FS18 FS16 Ę Ę Ę Ę Ę FS18B FS16B Ę Ę FS21i FS16–C FS16i FS18–C FS18i Ę Ę FS15B Ę om Reads or writes the sequence program, Pascal or C programs, or PMC data. CHANNEL DEVICE MONIT STOP = 1 r.c PMC I/O PROGRAM = OTHERS ce nt e DATA KIND = ALL (ALL:LADDER + LANGUAGE) FUNCTION = WRITE > [M-CARD] [OTHERS] [SPEED] [ALM] [ ] nc (a) Setting the channel Enter the number of the channel used at "CHANNEL = ." See (a) in Subsection 7.3.1 for details. .c (b) Setting the transfer conditions Press the [SPEED] soft key and set each condition. See Subsection 7.3.6 for details. w w w (c) Outputting data (PMC to input/output device) Item Operation DEVICE Press the [OTHERS] soft key. FUNCTION Press the [WRITE] soft key. DATA KIND Select the type of data to be output (see (3) in Section 7.2). Set the input/output device so that it is ready to accept data (wait state). Press the [EXEC] soft key to start outputting data. (d) Inputting data (input/output device to PMC) Item Operation DEVICE Press the [OTHERS] soft key. FUNCTION Press the [READ] soft key. Press the [EXEC] soft key and wait until data input finishes. The input/output device starts outputting data. 489 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PROGRAMMER (CRT/MDI) PMC PARAMETER DATA B–61863E/10 om CAUTION If DATA ERROR is displayed when a C program is written with the Series 16i/18i, perform the following: 1 Clear the C language area by pressing soft keys [EDIT], [CLEAR], [CLRLNG], then [EXEC]. 2 Read the C program again. 3 On the system parameter screen, set LANGUAGE ORIGIN. 4 Write the C program into flash ROM. (e) Collating data Item Operation Press the [OTHERS] soft key. FUNCTION Press the [COMPAR] soft key. r.c DEVICE Press the [EXEC] soft key and wait until data collation finishes. The input/output device starts outputting data. 7.3.6 Power Mate FS20 FS21/ 210MB FS18 FS16 Ę Ę Ę Ę Ę Ę FS21i FS16–C FS16i FS18–C FS18i Ę Ę Ę FS15B Ę MONIT STOP BAUD RATE = 3 (0:1200,1:2400,2:4800,3:9600,4:19200) PARITY BIT = 0 (0:NONE,1:ODD,2:EVEN) w w Ę : Supported : Not supported FS18B FS16B PMC SPEED OTHERS w .c nc Setting the Transfer Speed ([SPEED] Soft Key) ce nt e NOTE PMC data cannot be collated.
The data the file is to be collated with depends on the file. STOP BIT = 1 (0:1BIT,1:2BIT) WRITE CODE = 1 (0:ASCII,1:ISO) > ALM [INPUT] [ ] [ ] [ ] [INIT] The items shown above must be set when RS-232C is used for communication. Move the cursor to each item and enter a number. Pressing the [INIT] soft key sets each item to the initial value.
"WRITE CODE = " is displayed only when OTHERS is selected for DEVICE.
490 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PARAMETER DATA PMC PROGRAMMER (CRT/MDI) B–61863E/10 The table below lists the setting for communication with a FAPT LADDER. Setting on the personal computer Item Baud rate (bps) 9600 Character length Setting on the PMC (SPEED screen) BAUD RATE = 3 (9600bps) 8 bits No parity Number of stop bits 2 bits X parameter None PARITY BIT = 0 (NONE) STOP BIT = 1 (2BIT) om Parity check Transfer to and from a ROM WRITER FS20 FS21/ 210MB FS18 FS16 FS18B FS16B Ę Ę : Supported : Not supported FS21i FS16–C FS16i FS18–C FS18i ce nt e Power Mate r.c 7.3.7 FS15B Ę Reads or writes the sequence program, Pascal or C programs, or PMC data. This function is valid for the built-in programer function.
(a) Writing a file Item Operation Press the [ROMWRT] soft key. nc DEVICE FUNCTION Press the [WRITE] soft key.
Press the [EXEC] soft key to start outputting the data.
w w w .c (b) Reading a file Item Operation DEVICE Press the [ROMWRT] soft key. FUNCTION Press the [READ] soft key. Press the [EXEC] soft key to start inputting the data. (c) Collating a file Item Operation DEVICE Press the [ROMWRT] soft key. FUNCTION Press the [COMPAR] soft key. Press the [EXEC] soft key to start collating the data. 491 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PROGRAMMER (CRT/MDI) PMC PARAMETER DATA 7.3.8 When the machine tool builder creates a MDI keyboard which has no cursor keys on the PMC-MODEL RA1/RA3 of the FS 20, note the following methods of operation. Ladder diagrams cannot be edited using the ladder-diagram-edit memory card. On each setting screen, when an item is specified, the cursor automatically moves to the next item to be specified. When the item at the cursor need not be modified, specify the same value again.
When the item at the bottom of the screen has been specified, the cursor automatically moves to the item at the top of the screen (CHANNEL setting). When the return key (leftmost soft key) is pressed to exit from the I/O screen, the cursor automatically moves to the top of the screen. Examples of setting items are shown below. Example 1) When a ladder program is output to an off-line programmer (such as the P-G or a personal computer) CHANNEL setting : Enter the desired channel number, then press the key or [(NO.)] key. To use the current value, just press the key or [(NO.)] key. ce nt e 1 r.c om Notes on Using an MDI Keyboard without Cursor Keys (when using the FS20 PMC–MODEL RA1/RA3) B–61863E/10 DEVICE setting : Press the [HOST] key. The cursor returns to the CHANNEL setting position to enable CHANNEL setting. Example 2) When a ladder program is written into an F-ROM 1 CHANNEL setting : No specification required. To move the cursor, perform the operation described in 1 of Example 1. 2 DEVICE setting nc 2 : Press the [F-ROM] key. FUNCTION setting : No specification required. To change the CHANNEL setting, press the [WRITE] key to return the cursor to the CHANNEL setting position. Example 3) When a ladder program or a PMC parameter is read from or written into an FDCAS (M-CARD) w w w .c 3 1 CHANNEL setting : See 1 of Example 1 (or 1 of Example 2).
2 DEVICE setting 3 DATA KIND setting: Press the [LADDER] key for ladder operation. Press the [PARAM] key for PMC-parameter operation.
4 FUNCTION setting : Press the [READ]/[WRITE] key. 5 FILE NO. setting : Press the [FDCAS] ([M-CARD]) key. : Enter the desired file number or file name, then press the key or [EXEC] key. When the current value is used, just press the key. The cursor automatically returns to the CHANNEL setting position. The setting can be modified. In each example, pressing the [EXEC] key after setting data executes the corresponding processing. 492 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PARAMETER DATA PMC PROGRAMMER (CRT/MDI) B–61863E/10 7.4 I/O ERROR MESSAGES Message CONTENTS ³ OPERATION A program is already stored in the FLASH ROM (during blank check). PROGRAM ALREADY EXISTS (EXEC ?) A program is already stored in the FLASH ROM (during writing or deleting data).
Action) Press the EXEC key again when the message is displayed. Data is then written or deleted. PROGRAM NOTHING No program is in the FLASH ROM. ERASE ERROR The FLASH ROM is faulty and must be replaced. Consult your FANUC service office. om PROGRAM ALREADY EXISTS F L S H WRITE ERROR ANOTHER USED The FLASH ROM is being used by a device other than the PMC. R O M MUST BE IN EMG STOP NOT EMG STOP The CNC is not in the emergency stop state. F–ROM WRITE ERROR 37 There is no ROM cassette option. SIZE ERROR The size of the program exceeds the FLASH ROM size (during writing of the sequence program). Action) Use the CONDENSEM function (EDIT/CLEAR screen). If the error persists, the FLASH ROM size must be increased. The size of the program exceeds the RAM size (during reading of the sequence program). Action) The RAM size must be increased. ce nt e NO OPTION I/O OPEN ERROR nn nn = -1: The RS-232C interface is being used by a device other than the PMC. Action) Check if another device is using the RS-232C interface.
Check that, on the online setting screen (Section 8.5.1 in Part III), NOT USE is set for the RS–232C item. nn = 6: There is no RS-232C option. nn = 20: The RS-232C interface is connected incorrectly. Action) Check that the connection and the setting, such as channel and baud rate, are correct.
nn = 20: The RS-232C interface is connected incorrectly. Action) Check that the connection and the setting, such as channel and baud rate, are correct. nn = 22: Communication cannot be performed normally. Action) Check if a cable is disconnected. .c I/O READ ERROR nn nn = 20: The RS-232C interface is connected incorrectly. Action) Check that the connection and the setting, such as channel and baud rate, are correct. nn = 22: Communication cannot be performed normally. Action) Check if a cable is disconnected. nc I/O WRITE ERROR nn w ADDRESS IS OUT OF RANGE (xxxxxx) DATA ERROR w H O S T . F D C A S .
O T H E R S r.c READ ERROR w PROGRAM DATA ERROR Data other than that stored in the PMC debugging RAM area has been transferred. xxxxxx: Transfer address Invalid data was read. Action) Check the cable and setting (SPEED). When a C program is read into the Series 16i/18i: Action) Clear the C language area by pressing soft keys [EDIT], [CLEAR], [CLRLNG], then [EXEC]. Data output contains an error Action) On the alarm screen, check the details of the alarm. 493 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PROGRAMMER (CRT/MDI) PMC PARAMETER DATA Message NO MORE SPACE WRITE ERROR The memory card capacity is insufficient.
Action) Replace the memory card or delete unnecessary files and retry. NOT READY The memory card is not mounted. Action) Confirm if the memory card is mounted correctly. MOUNT ERROR The memory card is not formatted. Action) Format the memory card (see (a) of Subsection 7.3.4). WRITE PROTECT The memory card is protected. Action) Remove the protection of the memory card. BATTERY ALARM The battery of the memory card is not enough. Action) Exchange the battery of the memory card. FILE NOT FOUND Specified file number or file name is not found.
Action) Confirm the file number or the file name by LIST. DELETE ERROR The file cannot be deleted.
Action) Change the attribute of the file.
PROGRAM ALREADY EXISTS The file name already exists. Action) Change to other file name. I/O WRITE ERROR nn I/O READ ERROR nn I/O COMPARE ERROR nn I/O DELETE ERROR nn I/O LIST ERROR nn I/O FORMAT ERROR nn nn=30 : Action) nn=31 : Action) r.c om The file name is invalid. Action) Name the file is the MS–DOS format (see(5) of Section 7.2). ce nt e The memory card is not mounted. Confirm if the memory card is mounted correctly. The data cannot be written to the memory card. Remove the protection of the memory card. Exchange the memory card for the S–RAM card. The battery of the memory card is not enough. Exchange the battery of the memory card. The memory card capacity is insufficient. Replace the memory card or delete unnecessary files and retry. The memory card is not formatted. ditto Format the memory card. Specified file is not found. Confirm the file number or the file name by LIST. Specified file is protected. Confirm the attribute of the file. nn=32 : Action) nn=102: Action) nn=135: nn=105: Action) nn=114: Action) nn=115: Action) COMPARE ERR XXXXXX = AA:BB CONT?(Y/N) The data between DEVICE and PMC is different. XXXXXX : Address aa : The data in PMC bb : The data in DEVICE Action) If you continue it, press Y key. Otherwise, press N key. DATA ERROR Invalid data was read. Action) Check the cable and setting (SPEED).
When a C program is read into the Series 16i/18i: Action) Clear the C language area by pressing soft keys [EDIT], [CLEAR], [CLRLNG], then [EXEC]. w C o m m o n CREATE ERROR nc C A R D ³ OPERATION .c M E M O R Y CONTENTS B–61863E/10 Data output contains an error. Action) On the alarm screen, check the details of the alarm. w w PROGRAM DATA ERROR 494 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PARAMETER DATA PMC PROGRAMMER (CRT/MDI) B–61863E/10 7.5 SEQUENCE PROGRAM COPY FUNCTION The data items of the sequence program stored in EPROM can be copied into the debugging RAM module for PMC-RA1, PMC-RA2, PMC-RB, and PMC-RB2. The following shows the relationship between the function and soft keys. RUN EDIT I/O EXEC CANCEL (NO.) SYSPRM MONIT om RETURN NEXT CPYLAD CPYSYM CPYMSG ce nt e CPYTTL CPYALL 7.5.1 CPYMDL 7.5.2 nc Copies title data. Copy Title Data [CPYTTL] .c Copies a ladder program. Copies symbol data and comment data. w w Copy a Ladder Program [CPYLAD] 7.5.3 w Copy Symbol Data and Comment Data [CPYSYM] 7.5.4 Copies message data. Copy Message Data [CPYMSG] 495 COPY r.c ROM NEXT 7. WRITING, READING, AND VERIFYING THE SEQUENCE PROGRAM AND PMC PROGRAMMER (CRT/MDI) PMC PARAMETER DATA 7.5.5 B–61863E/10 Copies all the sequence programs into the debugging RAM. Copy the Sequence Programs [CPYALL] 7.5.6 Copies I/O module data. om NOTE If the I/O module data is different from the currently selected data during copying in Subsection 7.5.5 or 7.5.6, turn off the power and restart the system. r.c Copy I/O Module Data [CPYMDL] 7.6 Two channels cannot be used for the reader/punch interface at the same time. Before performing these I/O operations, be sure to terminate the system other than the PMC and processing through the reader/punch interface in the PMC program.
w w w .c nc ce nt e RESTRICTIONS 496 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING THE PROGRAM (MONIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 8 FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING THE PROGRAM (MONIT) Δ PA3 RA1 RA2 RA3 RA5 RB RB2 RB3 f : Can be used : Cannot be used : Can be used (with some restrictious) RB4 RB5 RB6 RC RC3 r.c PA1 om Press the [MONIT] soft key on the basic programmer menu to display the basic monitor menu shown in Fig. 8.
Pressing an appropriate soft key enables the user to display memory areas used for a user program written in the C language or to debug a program. Δ Δ Ę Ę RC4 NB NB2 Ę Ę Ę Work RAM is necessary (A02B-0120-H987 for the PMC-RC and RC3 and A02B-0162-J151 or A02B-0162-J152 for the PMC-NB). ce nt e These functions facilitate debugging a user program created by the machine tool builder in the C language. If the settings are erroneous, a system error may occur or the system may be shut down. Specify the settings correctly. For details of operation, refer to the "PMC-RC/RC3/NB Programming Manual (C language)" (B-61863-1). The following figure shows the soft key related to these functions. EDIT I/O SYSPRM MONIT DBGLAD GDT USRMEM DEBUG III 8.4 III 8.1 III 8.2 III 8.3 nc RUN (STOP) w w w .c RET PMC MONITOR MENU MONIT STOP SELECT ONE OF FOLLOWING SOFT KEYS DBGLAD GDT USRMEM DEBUG [DBGLAD] [: : : : DEBUG LADDER DIAGRAM DESCRIPTOR TABLE MAP USER MEMORY INFORMATION DEBUG FUNCTION] [GDT 8 497] [USRMEM] [DEBUG] Basic Monitor Menu 8.
FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING PMC PROGRAMMER (CRT/MDI) THE PROGRAM (MONIT) B–61863E/10 8.1 Information of a User Program Coded in C DISPLAYING THE GDT (GLOBAL DESCRIPTOR TABLE) GDT Nos. 32 to 95 defined in a user program can be displayed. The specified GDTs can also be dumped.
The following figure shows the soft keys related to this function. GDT USRMEM DEBUG om RET NO.SRH M.DUMP SEARCH INPUT WORD NEXT D.WORD ce nt e BYTE r.c RET 8.1.1 (1) Press the [GDT] soft key to display the user GDT information shown in Fig. 8.1.1 (a). Operation (2) Use the [NO. SRH] key to search for the GDT table with a desired number. nc (3) Press the [M. DUMP] key to dump the data of the GDT number which is displayed at the top. w w w .c (4) Press the [NEXT] key on the memory dump screen. Pressing the [BYTE] key displays the data in units of bytes. Pressing the [WORD] key displays the data in units of words, where one word equals two bytes. Pressing the [D. WORD] key displays the data in units of double words, or four bytes. See Fig. 8.1.1 (b). (5) When bit 4 of keep relay K17 is set to 1, the contents of RAM can be changed in units of the specified length on the memory dump screen by moving the cursor to the data to be changed. CAUTION Depending on the settings, a user program may operate erroneously, causing a system error. Be sure to specify the correct settings. 498 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING THE PROGRAM (MONIT) PMC PROGRAMMER (CRT/MDI) PMC DESCRIPTOR TABLE(GDT) NO. ACCESS USE BASE 032 RW 16 0016000AH 033 RW 16 0016005AH 034 RW 16 00160300H 035 RW 16 00160340H 036 ER 16 00823000H 037 ER 16 0084FB7CH 038 NULL DESCRIPTOR 039 ER 16 0084FF88H 040 RW 16 00160A6CH 041 RW 16 00160600H >] [LIMIT 0000056FH 0000023FH 00000040H 00000234H 00000058H 0000070AH 0000292FH 0000005AH 00000402H] [M.DUMP] [ ] r.c [NO.SRH] [MONIT RUN om B–61863E/10 8.1.1(a) User GDT Information MONIT RUN ce nt e PMC DESCRIPTOR TABLE(GDT) NO.
ACCESS USE 032 RW 16 033 RW 16 BASE 0016000AH 0016005AH LIMIT 0000056FH 0000023FH Y nc [NO.SRH] [ ] [ ] [M.DUMP] [PMC MEMORY DUMP(GDT NO.032) Y] MONIT RUN .c 103:0000 0000 0000 0000 0000 0000 ...... 103:0010 0000 0000 0000 0000 0000 ...... 103:0020 0000 0000 0000 0000 0000 ..... Dumped information of GDT NO.32 w w w 8.1.1(b) Memory Dump 499 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING PMC PROGRAMMER (CRT/MDI) THE PROGRAM (MONIT) B–61863E/10 8.1.2 Descriptions of Displayed Items NO. 32 ACCESS ER USE 16 BASE 00862340H LIMIT 0000523FH om Segment limit Segment base Segment type Segment attribute r.c GDT NO. (1) Access attribute of a segment Code Description Read-only data segment Read/write data segment ROD Read-only downward-expansion data segment RWD Read/write downward-expansion data segment ce nt e RO RW EO Execute-only code segment ER Execute/read code segment (2) Segment type Code Description 16-bit segment 32 32-bit segment nc 16 (3) Undefined segment NULL DESCRIPTOR is displayed for an undefined segment. w w w .c NOTE A user program created with the IC286 compiler is segmented in 16-bit units. 500 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING THE PROGRAM (MONIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 8.2 DISPLAYING THE MEMORY ALLOCATION INFORMATION OF A USER PROGRAM CODED IN C. The segment information of the following areas defined by a user program for each task can be displayed and the contents of the areas can be dumped. D Data area D Stack area D Common memory area The PMC management software dynamically allocates the areas mentioned above at locations which are different from those defined by the user program.
om The system allocates the data area at activation. When the system is not activated after the user program has been loaded, the data area is located at the address defined by the user program. Be sure to refer to the area after the system starts.
GDT USRMEM DEBUG ce nt e RET r.c The following figure shows the soft keys related to this function. TASK.D TASK.S SEARCH INPUT COM.D M.DUMP RET 8.2.1 w w w Operation .c nc BYTE WORD NEXT D.WORD (1) Press the [USRMEM] soft key. Depending on which soft key is pressed next (see below), the task memory information of a user program is displayed on the screen as shown in Fig.
8.2.1 (a) to (c). Soft keys [TASK. D] : Displays the information of allocating task data. [TASK. S] : Displays the information of allocating task stacks. [COM.D] : Displays the information of common memory allocation. (2) A task data area and stack area are displayed for each task ID. The information for all the common memory defined by user link control statements is displayed. (3) Pressing the [M. DUMP] key on each allocation information screen enables the contents of the memory related to the item which is displayed at the top to be dumped.
(4) Operation on the memory dump screen is the same as that described in Section 8.1. (5) When bit 4 of keep relay K17 is set to 1, the contents of RAM can be changed in units of the specified length on the memory dump screen by moving the cursor to the data to be changed. 501 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING PMC PROGRAMMER (CRT/MDI) THE PROGRAM (MONIT) B–61863E/10 CAUTION Depending on the settings, a user program may operate erroneously, causing a system error. Be sure to specify the correct settings. NAME TASK-O01 TASK-O02 TASK-O03 TASK-O04 TASK-O05 TASK-O06 GDT 039 040 041 042 043 044 BASE 00160050H 00160060H 00160070H 00160080H 00160210H 00160110H LIMIT 00010100H 00004100H 00005100H 00000160H 00000170H 00000110H r.c ID 10 11 12 13 14 15 MONIT RUN om PMC USER MEMORY(TASK DATA) > ce nt e [TASK.D] [TASK.S] [COM.D] [M.DUMP] [ ] 8.2.1(a) Information of a Task Data Area PMC USER MEMORY(TASK STACK) NAME TASK-O01 TASK-O02 TASK-O03 TASK-O04 TASK-O05 TASK-O06 GDT 239 240 241 242 243 244 BASE 00161050H 00161060H 00161070H 00161080H 00161210H 00161110H LIMIT 00010100H 00004100H 00005100H 00000160H 00000170H 00000110H > [TASK.D] [TASK.S] [COM.D] [M.DUMP] [8.2.1(b) Information of a Task Stack Area w w w .c nc ID 10 11 12 13 14 15 MONIT RUN 502] 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING THE PROGRAM (MONIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 PMC USER MEMORY(COMMON DATA) GDT 042 045 047 048 BASE 00162010H 00162020H 00162030H 00162040H LIMIT 00000100H 000A0100H 0000D000H 0000A100H om NO. 01 02 03 04 MONIT RUN >] r.c [TASK.D] [TASK.S] [COM.D] [M.DUMP] [8.2.2 ce nt e 8.2.1(c) Information of a Common Memory Data Area (1) Items displayed for a task data area and stack area Displayed Items ID 10 NAME TASK-001 GDT 032 BASE 00160010H LIMIT 00000100H Segment limit GDT No. Task name Task ID .c nc Segment base w (2) Items displayed for a common memory area GDT 032 BASE 00160010H LIMIT 00000100H Segment limit Segment base w w NO 01 GDT NO. Common memory No. 503 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING PMC PROGRAMMER (CRT/MDI) THE PROGRAM (MONIT) 8.3 B–61863E/10 There are two ways to check if a user program operates as intended. One is to execute the program while displaying the sequence on an external unit such as a display monitor. The other is to execute the program to a specified point (breakpoint), and check if the internal data items such as program work areas are correct. DEBUGGING 8.3.1 (1) Number of breakpoints: Up to 4 Specifications (2) Number of portions to be traced: 8 om This PMC debugging function checks programs using breakpoints. r.c (3) Capacity of memory used for storing traced data: Up to 256 bytes, up to 32 bytes for each traced portion 8.3.2 Press the [DEBUG] soft key to display the parameter screen for debugging. Pressing the [D.DUMP] key on the parameter screen displays the contents of the CPU registers and specified internal data items at the breakpoint. ce nt e Operation To return from the data display screen to the parameter screen, press the [D.PRM] soft key.
nc After the parameters are set, but before the program is interrupted, DBG blinks at the bottom right of the PMC screen. The breakpoint numbers BP1 to BP4 are also displayed at the bottom of the debug function screen. When the program is interrupted at a breakpoint, BRK blinks at the bottom right of the PMC screen. At this time, the breakpoint number, from BP1 to BP4, is displayed in reverse at the bottom of the debug function screen. w w GDT USRMEM DEBUG D.DUMP BRK.NO EXEC D.PRM BRK.NO RET w .c The following figure shows soft keys related to this function. RET 504 INIT 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING THE PROGRAM (MONIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 8.3.3 When the debug function is used, it is necessary to set the break conditions on the parameter screen. When using a 9" screen, press the key to set a trace data area for a breakpoint. Parameter Screen om (1) Setting parameters (a) BREAK SEG.ADR Specify the effective address of the breakpoint using a segment address. When data is accessed, specify the break address using a segment address. Use a key, such as EOB, to delimit a segment and an offset. Do not use alphanumeric keys. ce nt e r.c NOTE In data access, an even boundary or 4–byte boundary is assumed according to the type of ACCESS LENGTH, described later. Example) When the break address is GDT.NO = 32, OFFSETADDRESS = 101, specify 103; 101, obtained using the following formula: 32 (GDT.NO) x 8 + 3 = 259 = 103 (Hex) S When ACCESS LENGTH = WORD is specified with BREAK SEG.ADR = 103; 101 An access to 103; 100–101 causes a break. S When ACCESS LENGTH = D.WORD is specified with BREAK SEG.ADR = 103; 101 An access to 103; 100–103 causes a break. w w w .c nc (b) BREAK COND.
Specify a break condition. 0 (EXEC) : A program is interrupted at the specified effective address. 1 (WRITE) : A program is interrupted when it writes data to the specified address. 2 (READ/WRITE) : A program is interrupted when it writes data to or read data from the specified address. (c) ACCESS LENGTH Specify the address type of a breakpoint. 0 (BYTE) : An address is specified in units of bytes for read/write operation at the specified address and for when a program is interrupted at the specified effective address. 1 (WORD) : An address is specified in units of words for read/write operation at the specified address.
2 (D.WORD) : An address is specified in units of two words for read/write operation at the specified address. (d) PASS COUNT Specify the number of times a break condition is satisfied before the program is interrupted, in the range of 1 to 65535. 505 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING PMC PROGRAMMER (CRT/MDI) THE PROGRAM (MONIT) B–61863E/10 om (e) TASK ID Specify the task ID of a program. This parameter is convenient for identifying the program when it is to be interrupted at a breakpoint located in a function called from multiple tasks or is located in common memory. (f) TASK STATUS Specify how to handle the task when a program is interrupted. 0 (PASS) : The task continues after the program is interrupted. 1 (STOP) : The user task stops when the program is interrupted. The ladder program does not stop. r.c NOTE To restart the user program, press the [STOP] key to stop the sequence program and then press the [RUN] key to start the program on the basic menu using the RUN/STOP function. nc ce nt e (g) BREAK AVAIL. Specify whether the parameters for each breakpoint are valid or invalid. (h) NO. TRACE ADR. Using segment addresses, specify up to eight addresses from which data is traced when a program is interrupted at a breakpoint. Up to 32 bytes are stored for each address. Use a key, such as EOB, to delimit a segment and an offset. Do not use alphanumeric keys. To initialize these addresses only, enter 0; 0. w w w .c NOTE If the addresses are specified erroneously, the following two items, TYPE and LENGTH, cannot be specified. (i) TYPE Specify an address type with which traced data is displayed. 0 (BYTE) : Data is displayed in units of bytes. 1 (WORD) : Data is displayed in units of words. 2 (D.WORD) : Data is displayed in units of double words. (j) LENGTH Specify the length of traced data to be displayed. (2) Starting processing for a breakpoint When the parameters for each breakpoint are correctly specified, press the [EXEC] soft key on the parameter screen to start the processing for the currently selected breakpoint.
The breakpoint number, from BP1 to BP4, is displayed at the bottom of the screen. (3) Initializing data used for debugging To initialize the parameters and memory used for storing traced data, press the [INIT] soft key on the parameter screen. The parameter and memory for the currently selected breakpoint are then initialized. 506 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING THE PROGRAM (MONIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 (4) Changing a breakpoint Up to four breakpoints can be specified. For each breakpoint, parameters are specified and traced data is stored. To select a desired breakpoint, press the [BRK.NO] soft key on the parameter screen. The breakpoint is selected in the order of BP1, BP2, BP3, and BP4. PMC DEBUG (PARAM) MONIT RUN BREAK POINT NO.1 om = 0000:00000000 = 0 (0:E 1:W 2:RW) = 0 (0:B 1:W 3:D) = 32767 = 1 (0:ALL / 10-25) = 0 (0:PASS 1:STOP) = 0 (0:NO 1:YES) r.c BREAK SEG.ADR BREAK COND. ACCESS LENGTH PASS COUNT TASK ID TASK STATUS BREAK AVAIL. >] ce nt e [D.DUMP] [BRK.NO] [EXEC] [INIT] [8.3.3 (a) Screen for Specifying a Break Condition PMC DEBUG (PARAM) MONIT RUN [D.DUMP] [BRK.NO] [EXEC] [INIT] [8.3.3 (b) w w w .c nc BREAK POINT NO.1 NO. DUMP ADR. TYPE LENGTH (0:B/1:W/2:D) (32BYTE) 01 0000:00000000 0 10 02 0000:00000000 1 9 03 0000:00000000 2 8 04 0000:00000000 0 7 05 0000:00000000 1 6 06 0000:00000000 2 5 07 0000:00000000 0 4 08 0000:00000000 1 3 > 507] Screen for Specifying Data to Be Traced 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING PMC PROGRAMMER (CRT/MDI) THE PROGRAM (MONIT) 8.3.4 Screen for Displaying Traced Data B–61863E/10 When a program is interrupted under the break condition specified on the parameter screen, BRK blinks at the bottom right of the PMC screen. The breakpoint number at which the program has been interrupted is displayed in reverse at the bottom of the debug function screen. To display the traced data, press the [D.DUMP] soft key on the parameter screen, then press the [BRK.NO] key to select the screen for displaying the traced data corresponding to the breakpoint. om The following items are displayed. (1) REGISTER Displays the contents of the CPU registers.
(2) MEMORY r.c Displays the contents of memory at addresses of the traced data specified on the parameter screen. When the contents are displayed on multiple pages, scroll the screen, if necessary, using the , , , or, key. nc ce nt e PMC DEBUG (DUMP) MONIT RUN BREAK POINT NO.1(0000:00000000) REGISTER EAX=00000000 EBX=00000000 ECX=00000000 EDX=00000000 ESI=00000000 EDI=00000000 EBP=00000000 ESP=00000000 IEP=00000000 DS=0000 ES=0000 FS=0000 GS=0000 SS=0000 CS=0000 EFLAGS=00000000 CONTENS OF MEMORY 01 0000:00000000 00000000 00000000 00000000 02 0000:00000000 00000000 00000000 03 0000:00000000 00 00 00 00 00 00 00 00 04 0000:00000000 0000 0000 0000 0000 > w 8.3.5 w .c [D.PRM] [BRK.NO] [w Enabling Automatic Debugging at Power–on 8.3.6 Notes] [ ] [ ] 8.3.4 Screen for Displaying Traced Data As parameters used for debugging and traces data are stored in the retained memory, they are not lost when the power is turned off. When bit 1 of keep relay K18 is set to 1 after break condition parameters are correctly specified, debugging is automatically enabled when the power is turned on.
(1) Specify a break address (BREAK SEG.ADR) in the area used by the user program. If a break address is specified in the area which is used by the PMC management software, the system may hang up. (2) Debug function is incorporated in the CPU, reduces the CPU speed. Do not use the function during normal system operation. 508 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING THE PROGRAM (MONIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 8.4 Ę : Can be used LADDER DEBUGGING FUNCTION : Cannot be used Δ : To use this function, a ladder editing module is required PA1 PA3 RA1 RA2 Δ RA3 RA5 Δ Ę RB RB2 RB3 RB4 RB5 RB6 Δ Ę Ę Ę RC RC3 RC4 NB NB2 Ę Ę Ę om NOTE PMC–PA3 is usable with the Power Mate–H. r.c Using this function, Step Operations and Stop Operations listed below are possible. Step Operations are to execute ladder by specified step (single instruction, single net, and specified block). Stop Operations are to stop the execution of ladder when specified condition becomes true. (1) Step Operation to execute one instruction from current position. (2) Step Operation to execute one net (one circuit) from current position. nc ce nt e (3) Step Operation to execute from current position to specified contact or coil instruction. (4) Stop Operation to execute from the first step and stop the execution at specified contact or coil instruction. (5) Stop Operation to stop the execution of ladder by a trigger of signal condition. (Optionally, a trigger counter can be specified.) (6) Stop Operation to stop the execution of ladder after executing one scan. (Optionally, a scan counter can be specified.) 8.4.1 LADDER *TITLE DATA REMARKS 32 BYTES * NET 00001-00004 w w w .c Screen of Ladder Debugging Function MONIT RUN X1000.0 Y1000.0 X1000.1 Y1000.1 ABSDE X1000.0 RST FGHI ACT SUB36 ADDB 2 Y1000.3 D0000 [0] 1 D0000 [0] X1000.0 ACC=1 Y1000.5 STK=0000 0011 OF=0 SF=1 ZF=1 [SEARCH] [STEP] [BRKCTL] [ ] [ADRESS] [RUN] [DPARA] [RESET] [DUMP] [ONLEDT 8.4.1 Screen of Ladder Debugging Function 509] 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING PMC PROGRAMMER (CRT/MDI) THE PROGRAM (MONIT) 8.4.2 Soft key menu of Ladder Debugging Function B–61863E/10 For this operation, press [DBGLAD] soft key to bring the following menu. DBGLAD ONLINE GDT SEARCH STEP BRKCTL USRMEM DEBUG RET ADRESS DPARA (SYMBOL) (NDPARA) NEXT ONLEDT RESET r.c RUN (STOP) om Chap. 8.4.3 Chap. 8.4.3 DUMP The function of the soft key is as follows. : is used to specify several types of search functions. (2) [STEP] : is used to specify several types of Step Functions. This function can not be used when the ladder program is being executed. (3) [BRKCTL] : is used to specify Stop Functions. This function is to stop the execution of ladder when specified condition becomes true. This function can not be used when the ladder program is being executed. (4) [DUMP] : is used to display the contents of PMC address in the 2 lines at the bottom of CRT where the last NET is normally displayed. nc ce nt e (1) [SEARCH] : is used to switch the monitor mode from STOP to RUN, or vice versa. (6) [ADRESS] : is used to switch the symbol display mode from SYMBOL to ADDRESS, or vice versa. w w w .c (5) [RUN] (7) [DPARA] : is used to switch the mode for displaying the contents of functional instruction parameters from NDPARA (No Display Parameter) mode to DPARA (Display Parameter) mode, or vice versa. (8) [ONLEDT] : is used to edit the ladder program without stopping the execution. Editing is limited within the operations which do not change the size of ladder. (9) [RESET] : is used to initialize the Step Function and Stop Function. NOTE See Chapter II. 5.3 and 5.4 for details of (7) or (8). 510 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING THE PROGRAM (MONIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 8.4.3 Step Operation [STEP] Using this function, Step Operations such as single step, single net, and block steps until specified instruction are possible. [Function] om (1) Step Operation to execute one instruction from current position. (2) Step Operation to execute one net (one circuit) from current position. (3) Step operation to execute from current position to specified contact or coil instruction. [Displaying of Step] See "Fig. 8.4.1" r.c "ACC=1 STK=0000 0011 OF=0 SF=0 ZF=1" ACC : result of operation STK : contents of stack (1 byte) OF : overflow (0=NO, 1=YES) SF : sign (0=NO, 1=YES) ZF : zero (0=NO, 1=YES) For this operation, press [STEP] soft key to bring the following menu. STEP BRKCTL ELMMNT NETMNT BLOCK SEARCH START ce nt e SEARCH DUMP nc RET B.SRCH TRNS.B The function of the soft key is as follows.
: A blinking cursor shows the current position at which the execution is stopped. y moving the cursor, a position at which the execution is to be stopped can be specified. (2) [ELMMNT] : is used to execute one instruction from current position. (3) [NETMNT] : is used to execute one net from current position. (4) [BLOCK] : is used to execute from current position to specified instruction. If specified instruction is not executed because it is skipped by conditional JMP or CALL instructions, the execution will stop at the END instruction, END1 (SUB 1), END2 (SUB 2), or END3 (SUB 48), of the current level. (5) [B.SRCH] : is used to search the instruction at which the execution is currently stopped. (6) [TRNS.B] : is used to transfer the current status of input signals to the synchronous buffer so that succeeding instructions could operate on refreshed inputs when the execution is continued from current position. (For more about the synchronous buffer, see Chapter I.2.5 Processing I/O Signal) w w w .c (1) [STEP] 511 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING PMC PROGRAMMER (CRT/MDI) THE PROGRAM (MONIT) B–61863E/10 NOTE Normally, transferring to the synchronous buffer is automatically performed at the beginning of the 2nd level ladder. 8.4.4 Stop Function of Break with Condition [BRKCTL] om Using this function, the execution of the ladder can be stopped when specified condition becomes true. Then, the signal condition can be checked. [Function] r.c (1) Stop operation to execute from the first step and then to stop at specified contact or coil instruction. (Optionally, a trigger counter can be specified to stop after the instruction is executed specified times.) ce nt e (2) Stop operation to stop the execution of ladder when a trigger condition specified by signal becomes true. (Optionally, a trigger counter can be specified to stop after the trigger becomes true specified times.) (3) Stop operation to stop the execution of ladder after executing a scan. (Optionally, a scan counter can be specified to stop after executing specified times of scans.) The execution is started by pressing [START] key. [Displaying of specified trigger] :ON :X0000. 0 : 0 : 0000/12345" .c nc "MODE POINT : trigger point at which the trigger condition is checked (default is 0) 0 the top of the 1st level seq 1 after END1 2 after END2 3 after END3 w ADR : specified trigger address ON : signal turn on (TRGON), OFF : turn off [Displaying of specified scan] w w COUNT : trigger counter (present counter/specified counter) "SCAN COUNT : 00000/00003" counter of scan (present counter/specified counter) 512 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING THE PROGRAM (MONIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 SEARCH STEP TRIGER SCAN TRGON TRGOFF BRKCTL DUMP RET START (STOP) om INIT The function of [BRKCTL] soft key is as follows. r.c (1) [TRIGER] : is used to specify the trigger condition by signal. Trigger condition has to be specified according to the following syntax.
And then, the execution is started by pressing [START] key. ce nt e "ADR ; PONIT (0-3) ; COUNT + [TRGON/ TRGOFF]" ADR POINT : specified trigger address : trigger point at which the trigger condition is checked (default is 0) 0 the top of the 1st level sequence 1 after END1 2 after END2 3 after END3 COUNT : counter of checked trigger (default is 1) (1 to 65535) nc (2) [TRGON] : is used to specify "turn on" trigger to stop the execution when the signal is transitioned from off to on status. (3) [TRGOFF] : is used to specify "turn off" trigger to stop the execution when the signal is transitioned from on to off status. : is used to display the contents of PMC address in the 2 lines at the bottom of CRT where the last NET is normally displayed. (5) [SCAN] : is used to specify a stop function by scan counter. To specify a scan counter, input as follows. "counter + [SCAN]". (counter: 1 to 65535) When the counter is not specified, it is recognized as 1. After specifying the scan counter, the execution is started by pressing [START] key. (6) [INIT] : is used to initialize the stop function with break condition.
(7) [START] : is used to start the execution after specifying the condition to stop. w w w .c (4) [DUMP] 513 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING PMC PROGRAMMER (CRT/MDI) THE PROGRAM (MONIT) B–61863E/10 8.5 ONLINE FUNCTION Δ f : Usable : See Note : Not usable PA1 PA3 RA1 Δ Δ RA2 RA3 RA5 Δ Ę RB RB2 RB3 RB4 RB5 RB6 Ę Ę Ę Ę RC RC3 RC4 NB NB2 Ę Ę Ę Ę om NOTE PMC–PA3 is usable with the Power Mate–D/H. PMC–RA1 is usable with the loader control function of the Series 21i. PMC–RA3 is usable with the FANUC NC Board. r.c With the online function of the FAPT LADDER–II or ladder editing package, the following can be performed using the personal computer: nc ce nt e D Ladder monitor display D Online ladder editing D PMC parameter display and editing D Signal state monitor display and modifications D Input/output to and from the PMC (loading from the PMC, storing to the PMC) D Writing to flash ROM Before this online function can be used, communication conditions must be set in the PMC built into the CNC. 8.5.1 Online Setting Screen w w w .c PARAMETERS FOR ONLINE MONITOR MONIT STOP CPU ID = RS–232C = USE/ NOT USE CHANNEL = 1 BAUD RATE = 300/600/1200/2400/4800/9600/19200 PARITY = NONE/ODD/EVEN STOP BIT = 1 BIT/2 BITS TIMER 1 = 0 TIMER 2 = 5000 TIMER 3 = 15000 MAX PACKET SIZE = 1024 F–BUS = USE/ NOT USE RS–232C = INACTIVE : 0 F–BUS = INACTIVE : 0 > EMG ST INIT 8.5.1 Online Monitor Setting Screen EMG ST : Terminates communication forcibly. Use this key when communication becomes abnormal, such that the connection cannot be terminated normally. 514 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING THE PROGRAM (MONIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 INIT : Initializes the parameters to their default values. om NOTE When the configuration disables the use of the ladder editing package, the F–BUS item is not displayed. When the 9–inch CRT is being used, two pages are used for screen display. Switch between the pages with the and keys.
8.5.2 Setting Method r.c 1. For communication with FAPT LADDER–II (1) Check that NOT USE is set for the RS–232C item.
(2) Set the CHANNEL and BAUD RATE parameters. ce nt e (3) Move the cursor to the RS–232C item with the and keys.
(4) Select USE with the "²" or "³" key. 2. For communication with the ladder editing package (1) Move the cursor to the F–BUS item with the and keys. (2) Select USE with the "²" or "³" key. This completes the setting for communication. The online function is operated from the FAPT LADDER–II or ladder editing package. .c nc For details of operation, refer to the operator's manual for the FAPT LADDER–II or ladder editing package.
8.5.3 w w w Setting on the NC Parameter Screen When the NC (such as the Series 160i or 180i) does not support the display of the PMC screen, or if communication is to be started automatically at power–on without screen manipulation, select a communication destination with NC parameter No. 24. When the value of NC parameter No. 24 is changed, the PMC online monitor screen is set as indicated below. NC parameter No. 24 Setting on the PMC online monitor screen RS–232C F–BUS Description 0 NOT USE USE Selects the ladder editing package. 1 USE (Channel 1) NOT USE Selects FAPT LADDER–II (channel 1). 2 USE (Channel 2) NOT USE Selects FAPT LADDER–II (channel 2). NOT USE NOT USE Terminates communication forcibly (EMG ST). 255 515 8. FUNCTIONS FOR DISPLAYING MEMORY AREAS AND DEBUGGING PMC PROGRAMMER (CRT/MDI) THE PROGRAM (MONIT) B–61863E/10 w w w .c nc ce nt e r.c om NOTE With NC parameter No. 24, the PMC online monitor screen setting is changed when the power is turned on. After this parameter has been modified, the power must be turned off then back on for the new setting to become effective. When the setting of the online monitor screen is to remain unchanged, a value other than 0, 1, 2, and 255 must be set for this parameter. 516 9. ERROR MESSAGES (FOR EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 9 ERROR MESSAGES (FOR EDIT) Error messages (For EDIT 1) Contents and solution om Message The address of the relay/coil is not set. FUNCTION NOT FOUND There is no functional instruction of the input number. COM FUNCTION MISSING The functional instruction COM(SUB9) is not correctly dealt with. Correspondence of COM and COME(SUB29) is incorrect. Or, the number of coil controlled by COM is specified by the model which the number cannot be specified.(It is possible to specify the number of coil only on PMC-RB/RC.) EDIT BUFFER OVER There is no empty area of the buffer for the editing. (solution) Please reduce NET under editing. END FUNCTION MISSING Functional instruction END1, END2, END3 and END do not exist Or, there are error net in END1, END2, END3, END. Or, order of END1, END2, END3, and END is not correct. ERROR NET FOUND There is an error net. ILLEGAL FUNCTION NO. The wrong number of the functional instruction is searched. FUNCTION LINE ILLEGAL The functional instruction is not correctly connected. HORIZONTAL LINE ILLEGAL The horizontal line of the net is not connected. ILLEGAL NET CLEARED Because the power had been turn off while editing LADDER, some net under editing was cleared. ILLEGAL OPERATION Operation is not correct. The value is not specified and only INPUT key was pushed. The address data is not correctly inputted. Because the space to display the instruction on screen is not enough, the functional instruction cannot be made. SYMBOL UNDEFINED The symbol which was inputted is not defined.
ce nt e nc NET TOO LARGE There is an incorrect input data. Non-numerical value was inputted with COPY,INSLIN,C-UP, C-DOWN etc.
The input address was specified for write coil. An illegal character was specified for the data table. The input net is larger than the editing buffer. (solution) Please reduce the net under editing. .c INPUT INVALID r.c ADDRESS BIT NOTHING w JUMP FUNCTION MISSING The functional instruction JMP(SUB10) is not correctly dealt with.
Correspondence of JMP and JMPE(SUB30) is incorrect.
The number of coil to jump is specified by the model which the number of coil cannot specified.
(It is possible to specify the coil number only on PMC-RB/RC.) LADDER is broken. LADDER ILLEGAL There is an incorrect LADDER. IMPOSSIBLE WRITE You try to edit sequence program on the ROM.
OBJECT BUFFER OVER The sequence program area was filled. (solution) Please reduce the LADDER. w w LADDER BROKEN PARAMETER NOTHING There is no parameter of the functional instruction. PLEASE COMPLETE NET The error net was found in LADDER. (solution) After correcting the error net, please continue operating.
PLEASE KEY IN SUB NO. Please input the number of the functional instruction. (solution) If you do not input the functional instruction, please push soft key "FUNC" again. PROGRAM MODULE NOTHING You tried to edit though there was neither RAM for debugging nor ROM for sequence program. RELAY COIL FORBIT There is an unnecessary relay or coil. RELAY OR COIL NOTHING The relay or the coil does not suffice. PLEASE CLEAR ALL It is impossible to recover the sequence program. (solution) Please clear the all data.
517 9. ERROR MESSAGES (FOR EDIT) PMC PROGRAMMER (CRT/MDI) B–61863E/10 Error messages (For EDIT 2) Message Contents and solution The same symbol name is defined in other place. COMMENT DATA OVERFLOW The comment data area was filled. (solution) Please reduce the number of the comment. SYMBOL DATA OVERFLOW The symbol data area was filled. (solution) Please reduce the number of the symbol. VERTICAL LINE ILLEGAL There is an incorrect vertical line of the net. MESSAGE DATA OVERFLOW The message data area was filled. (solution) Please reduce the number of the message. 1ST LEVEL EXECUTE TIME OVER The 1st level of LADDER is too large to complete execution in time. (solution) Please reduce the 1st level of LADDER.
PARA NO.RANGE ERR : functional–instruction name A parameter number used for a functional instruction falls outside the allowable range. (solution) Change the parameter number to a value within the allowable range. PARA NO.DUPLICATE : functional–instruction name EXIT ? A parameter number used for a functional instruction is specified more than once. (solution) Change the number to a number that has not yet been used if duplicate execution causes an error. w w w .c nc ce nt e r.c om SYMBOL DATA DUPLICATE 518 PMC PROGRAMMER (CRT/MDI) B–61863E/10 10 10. ERROR MESSAGES (FOR I/O) ERROR MESSAGES (FOR I/O) Error messages (For I/O 2) Contents and solution om Message An error occurs when the reader/puncher interface was started. nn = –1 Because the interface is used with NC etc., the interface is not able to be opened by PMC side. (solution) After other functions finishes using the line, please execute again. 6 There is no option for the interface. 20 The interface cannot be opened. (solution) Please confirm the connection of the cable. Please confirm setting of the baud rate etc. I/O WRITE ERROR nn An output error occurred in the reader/puncher interface.
nn = 20 The state of the interface is not correct. (solution) Please confirm the connection of the cable. Please confirm setting the baud rate etc. 22 Opponent side is not ready to receive. (solution) Please confirm the power supply on the opponent side. Or, please initialize the interface. I/O READ ERROR nn An input error occurred in the reader/puncher interface. nn = 20 The state of the interface is not correct. (solution) Please confirm the connection of the cable. Please confirm setting the baud rate etc. 21 The data is not sent from the opponent side. (solution) Please confirm the power supply on the opponent side. I/O LIST ERROR nn An error occurred in directory read processing from FD Cassette. nn = 20 The state of the interface is not correct. (solution) Please confirm the connection of the cable. Please confirm setting of the baud rate etc. COMPARE ERR xxxxxx = aa:bb CONT?(Y/N) A compare error occurred. xxxxxx : The Address where the compare error occurred. aa : The data on PMC side bb : The data on device side Enter 'Y' to continue processing. ADDRESS IS OUT OF RANGE(xxxxxx) The data transferred to the address out of the PMC debugging RAM area. xxxxxx : Transferred address. (solution) Please confirm the address of the transferring data. LADDER : Please confirm the model setting. C language : Please confirm setting the address in the link control statement and build file. .c nc ce nt e r.c I/O OPEN ERROR nn An error occurred in the ROM writer. w w w ROM WRITER ERROR nnnnnn 519 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power PMC PROGRAMMER (CRT/MDI) Mate–D/F/H) 11 B–61863E/10 PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power Mate–D/F/H) om The DPL/MDI panel is used to set PMC system parameters and create and execute the sequence program.
(1) Setting and displaying PMC system parameters (SYSTEM PARAM) – The type of counter data (BCD or binary) can be selected. – r.c (2) Editing the sequence program (EDIT) The sequence program can be edited (input, addition, search, and deletion) by using the ladder mnemonics display. ce nt e (3) Executing the sequence program (RUN/STOP) – The execution of the sequence program can be started and stopped. (4) Storing the sequence program into flash EEPROM (I/O) – The sequence program can be stored into flash EEPROM (only for the Power Mate–H). nc The DPL/MDI panel is shown below. O w w w .c Y F F N K/A M RD X AXIS X DGNOS PARAM POS S T PRGRM WRT NOT OPR ALARM &@ NO. MENU VAR I READ INSRT J WRITE DELET G G H # / EOB SUB OR AND STK 7 8 9 4 5 6 1 2 3 D/R K ALTER 0 –/+ & CAN INPUT P P Q T/C R Fig.11 DPL/MDI panel for Power Mate 520 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power Mate–D/F/H) PMC PROGRAMMER (CRT/MDI) B–61863E/10 om NOTE 1 With the PMC programmer (DPL/MDI) function, the characters indicated at the lower–left part of each key are used. 2 When the key is pressed once, the left–hand character is valid. When the key is pressed twice, the right–hand character is valid. Example : When the key is pressed once, D is keyed in. When the key is pressed twice, R is keyed in. When a password is cleared, however, only the characters on the left side are valid. PMC programmer menu or or or or PMC editing menu PMC EDIT >LADDER 1/1 or or Editing ladder mnemonics N0001 RD or .c w PMC PRG MENU 3/3 >SYSTEM PARAM w PMC PRG MENU 4/4 >I/O w LADDER RUN/STOP MONITOR (STOP) nc PMC PRG MENU 2/3 >EDIT Sequence program start and stop ce nt e PMC PRG MENU 1/3 >RUN/STOP r.c The screen configuration for the PMC programmer (DPL/MDI) function is as follows: or Setting and displaying PMC system parameters CTR TYPE=BCD (BINARY=0/BCD=1) The Power Mater–H supports the following screen: Screen for storing the sequence program into flash ROM or DEVICE=F–ROM >WRITE Y/N[YES] 521 X0000.0 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power PMC PROGRAMMER (CRT/MDI) Mate–D/F/H) 11.1 SELECTING THE PMC PROGRAMMER MENU B–61863E/10 To operate the PMC programmer, set K17#1 of the keep relay area for PMC parameters to 1, then press the key two times on the DPL/MDI (press the key further when the program screen is selected), thus causing the PMC programmer menu to be displayed. Program screen PMC programmer menu screen PMC PRG MENU 1/3 >RUN/STOP om N010!G90 G01 G43 X10 ; Can be switched only when K17#1 = 1. PMC programmer screen (PMC programmer menu) (PMC editing menu) Current Position screen Program screen ce nt e r.c To return to the CNC screen, press the , , , , or key. (K17#1=1) nc Alarm/Message screen Parameter/Diagnostic screen Offset/Setting/Macro Variable screen w w w .c The following keys on the DPL/MDI panel are used for PMC operation: 1 , , , , key Returns to the CNC screen.
2 key Shifts the cursor upward. 3 key Shifts the cursor downward. 4 , key Selects a function when the PMC programmer menu or PMC editing menu is displayed. 5 , key Returns to the previous menu from the PMC programmer menu or PMC editing menu. 522 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power Mate–D/F/H) PMC PROGRAMMER (CRT/MDI) 11.2 SETTING AND DISPLAYING SYSTEM PARAMETERS (SYSTEM PARAM) Selecting SYSTEM PARAM on the PMC programmer menu displays the system parameter screen. If the sequence program is running, selecting this function automatically stops the program.
1 Display the PMC programmer menu. 2 Display the SYSTEM PARAM item by pressing the or key. PMC PRG MENU >SYSTEM PARAM 3 3/3 Press the or key. The system parameter screen appears. om B–61863E/10 r.c CTR TYPE = BIN (BINARY=0/BCD=1) 4 The current counter data type is displayed on the screen. ce nt e (a) Specify the type of the counter value to be used for the CTR functional instruction, as binary or BCD (enter for binary or for BCD). (b) Press the key. The counter data type is set. 5 Pressing the or key displays the PMC programmer menu. nc NOTE If the PMC parameter keep relay K19#0 is set to 1, the screen for writing a sequence program into Flash Memory is displayed upon the completion of editing. (This is applicable to the Power Mate–H only). Write a sequence program into Flash Memory as explained in Section 11.7. w w w .c DEVICE=F–ROM >WRITE Y/N [YES] 523 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power PMC PROGRAMMER (CRT/MDI) Mate–D/F/H) EDITING THE SEQUENCE PROGRAM (EDIT) Selecting EDIT on the PMC programmer menu displays the editing menu. 1 Display the PMC programmer menu. 2 Display the EDIT item by pressing the or key. PMC PRG MENU >EDIT 3 2/3 Press the or key. The PMC editing menu appears. PMC EDIT >LADDER 1/1 om 11.3 B–61863E/10 w w w .c nc ce nt e r.c To end editing and display the PMC programmer menu, press the or key. 524 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power Mate–D/F/H) PMC PROGRAMMER (CRT/MDI) 11.4 EDITING LADDER MNEMONICS 11.4.1 Starting Ladder Mnemonics Editing When ladder mnemonic editing (LADDER) is selected from the PMC edit menu, the ladder mnemonic edit screen is displayed. When this function is selected, the sequence program stops. When ladder mnemonic editing (LADDER) is selected from the PMC edit menu, the ladder mnemonic edit screen is displayed. When this function is selected, the sequence program stops. 1 Display the PMC edit menu screen. 2 Display the LADDER item by pressing the or key.
1/1 r.c PMC EDIT >LADDER om B–61863E/10 Press the or key.
When a password is set for the ladder: Proceed to step 4. When no password is set for the ladder: Proceed to step 6. 4 If a password is set, a password clear request is displayed. ce nt e 3 PASSWORD (R/W) nc NOTE For a ladder for which a password has been set, the ladder mnemonic editing function cannot be started unless the correct password is entered. Once the password is cleared, the password remains cleared until the power is turned off then back on. .c 5 Enter the password, then press the key. w w NOTE The entered password is not displayed (not echoed back on the screen). w If the entered password is incorrect, the following error message is displayed. FALSE PASSWORD If the key is pressed at this time, the screen display returns to the password clear request screen.
If the key is pressed at this time, the screen display returns to the PMC edit menu. If the entered password is correct, the password is cleared. 6 A sequence program is displayed. 525 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power PMC PROGRAMMER (CRT/MDI) Mate–D/F/H) B–61863E/10 Step number > N0001 RD X0000.0 Instruction 11.4.2 Cursor scroll (scroll per step) Pressing the cursor key displays the instruction one step before that currently displayed. Pressing the cursor key displays the instruction one step after that currently displayed.
2 Specifying the step number Entering , , then displays the instruction having the entered step number. (The cursor key can be used instead of the key.) (Example) , , om 1 ce nt e N0123 SUB 50 PSGNL r.c Confirming the Ladder Mnemonics 3 Relay search Entering then searches for the relay including the entered address. (Example) , N0105 AND nc

X0000.2 w w w .c 4 5 Relay coil search Entering , , then searches for the relay coil including the entered address.

(Example) , , N0187 WRT. NOT Y0033.5 Functional instruction search Entering , , then searches for the entered functional instruction. (Example) , , N0123 SUB 50 PSGNL 526 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power Mate–D/F/H) PMC PROGRAMMER (CRT/MDI) B–61863E/10 NOTE 1 Relay search, relay coil search, and functional instruction search are started from the current screen. If the relay, relay coil, or instruction is not found by the end of the ladder program, search is performed from the beginning of the ladder program to the step at which search was started. If still not found, "NOT FOUND" is displayed. om N0105 NOT FOUND AND X0000.2 r.c 2 Display of some instructions may differ from that for FAPT LADDER. P–G, personal–computer FAPT LADDER (a) RD.NOT.STK Ladder mnemonics editing RD.N.STK SUB 03 TMR P001 timer–number ce nt e (b) TMR timer–number (c) DEC code–signal–address (PRM) decode–instruction SUB 04 DEC P001 code–signal–address P002 decode–instruction The above also applies when modifying the ladder mnemonics. 11.4.3 1 Changing an instruction (a) Display the instruction to be changed. nc Modifying the Ladder Mnemonics (b) Enter a new instruction. w .c (c) Press the key. (Example) , , N1234 AND R0123.4 Before change w N1234 OR Y0032.4 w After change NOTE If changing the instruction causes the memory capacity to be exceeded, the key is ignored without changing the instruction. 2 Deleting an instruction (a) Display the instruction to be deleted. (b) Press the key.

The instruction is deleted and the next instruction is displayed. 527 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power PMC PROGRAMMER (CRT/MDI) Mate–D/F/H) 3 B–61863E/10 Inserting an instruction (a) Display the instruction after which an instruction is to be inserted. (b) Enter the instruction to be inserted. (c) Press the key.

(Example) , , N1234 AND R0123.4 om Before insertion N1234 AND.STK r.c After insertion ce nt e NOTE If inserting the instruction causes the memory capacity to be exceeded, the key is ignored without inserting the instruction.

4 Deleting the ladder program (a) Enter . (b) Press the key. The whole ladder program is deleted. 1 Press the or key. 2 "EXECUTING" is displayed. nc 11.4.4 .c Ending Ladder Mnemonics Editing The PMC editing menu appears.

w w w 3 N0001 EXECUTING 528 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power Mate–D/F/H) PMC PROGRAMMER (CRT/MDI) B–61863E/10 NOTE 1 If the sequence program contains an error, the PMC editing menu is not displayed but an error message appears on the screen. (Example) Error message END FUNCTION MISSING ce nt e DEVICE=F–ROM >WRITE Y/N [YES] r.c om Pressing the or cursor key displays the ladder mnemonics editing screen. 2 Pressing the , , , , or key during the editing of the sequence program displays the CNC screen by forcibly terminating editing even if the program contains an error. 3 If the PMC parameter keep relay K19#0 is set to 1, the screen for writing a sequence program to flash ROM is displayed upon the completion of editing. (This is applicable to the Power Mate–H only). w w w .c nc Write a sequence program into flash ROM as explained in Section 11.7. 529 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power PMC PROGRAMMER (CRT/MDI) Mate–D/F/H) 11.5 B–61863E/10 Selecting RUN/STOP on the PMC programmer menu displays the sequence program start/stop screen. 1 Display the PMC programmer menu. 2 Display the RUN/STOP item by pressing the or key. PMC PRG MENU >RUN/STOP 3 1/3 Press the or key. start/stop screen appears.

LADDER RUN/STOP MONITOR [RUN] The sequence program om STARTING AND STOPPING THE SEQUENCE PROGRAM (RUN/STOP) The current execution state of the sequence program is displayed on the screen. Pressing the or key switches the state between running and stopped. 5 Pressing the or key displays the PMC programmer menu. ce nt e r.c 4 w w w .c nc NOTE When the sequence program cannot be started(RUN), the alarm of PMC occurred. Please confirm the alarm status referring to "11.11 Error List". 530 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power Mate–D/F/H) PMC PROGRAMMER (CRT/MDI) B–61863E/10 11.6 Displayed error message Error description (operator action) COIL NOTHING No coil is specified for a functional instruction using a coil. 2 COM FUNCTION MISSING The use of the COM (SUB9) functional instruction is incorrect. 3 END FUNCTION MISSING The END1 or END2 functional instruction is missing (or ERROR NET). 4 JUMP FUNCTION MISSING The use of the JMP (SUB10) functional instruction is incorrect. 5 LADDER BROKEN The ladder program is corrupted. 6 OBJECT BUFFER OVER The user program RAM is full. (Note) (Perform condensation or reduce the size of the ladder program.) 7 PLEASE CLEAR ALL The sequence program has become unrecoverable due to power–off during editing. 8 1ST LEVEL EXEC TIME OVER The ladder first level is too great. om 1 r.c ERROR MESSAGES (FOR LADDER MNEMONICS EDITING) w w w .c nc ce nt e NOTE Use a memory card for ladder diagram editing or the CONDENSE function of FAPT LADDER (for personal computers). These methods may, however, not be effective.

531 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power PMC PROGRAMMER (CRT/MDI) Mate–D/F/H) STORING THE SEQUENCE PROGRAM INTO FLASH EEPROM (I/O) (ONLY FOR THE Power Mate–H) Selecting I/O on the PMC programmer menu displays the screen for storing the sequence program into flash EEPROM. Before attempting to store the sequence program into flash EEPROM, place the CNC in the emergency stop state. (1) Display the PMC programmer menu. (2) Display the I/O item by pressing the or key. PMC PRG MENU >I/O 4/4 om 11.7 B–61863E/10 (3) Press the or key. The sequence program storage screen appears. Pressing the or key switches display between [YES] and [NO]. DEVICE=F–ROM >WRITE!Y/N[NO] r.c DEVICE=F–ROM >WRITE!Y/N[YES] ce nt e (4) When [NO] is displayed, pressing the key displays the sequence program storage screen. When [YES] is displayed, pressing the key starts writing the sequence program into flash EEPROM.

"EXECUTING" is displayed during writing. WRITE TO F–ROM EXECUTING "EXECUTING" BLINKS. Once the sequence program has been written normally, "COMPLETE" is displayed. nc WRITE TO F–ROM COMPLETE w .c NOTE If an error occurs, an error message appears on the screen. w w Example Example error message NOT EMG STOP To return to the sequence program storage screen, press the or key. (5) Pressing the key displays the PMC programmer menu. 532 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power Mate–D/F/H) PMC PROGRAMMER (CRT/MDI) B–61863E/10 11.8 The table below lists the details of the errors which may occur during storage into F–ROM using the DPL/MDI (only for the Power Mate–H). ERROR DETAILS Error message Description The ladder data in RAM is invalid. Alternatively, there is no RAM or ROM. SIZE ERROR The program exceeds the maximum size which can be written into F–ROM. NOT EMG STOP The CNC is not in the emergency stop state. OPEN ERROR The OPEN processing has failed.

ERASE ERROR The ERASE processing has failed. The F–ROM cannot be erased. Alternatively, the F–ROM is defective. WRITE ERROR The WRITE processing has failed. The F–ROM cannot be written. Alternatively, the F–ROM is defective. w w w .c nc ce nt e r.c om PROGRAM DATA ERROR 533 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power PMC PROGRAMMER (CRT/MDI) Mate–D/F/H) B–61863E/10 11.9 INPUT/OUTPUT LADDER/PMC–PARA METER BY MDI/DPL D Method of Inputting/Outputting Ladder (1) Select "Diagnose screen" by key in key. (2) Key in key or key. om Input/Output Method to Office Programmer (P–g Mate/Mark II) (Fixed 9600bit/Sec.) (3) Turn on key from the office programmer menu screen, and key in menu number "5" or "3". r.c 11.9.1 D Method of Inputting Ladder and PMC–Parameter. Input/Output Method to FANUC FLOPPY CASSETE (Fixed 4800bit/sec.) (1) Select "Diagnose screen" by key in key. (2) Key in key and optionally key in [File No.]. (3) Key in key. ce nt e 11.9.2 CAUTION In case of input PMC–Parameter, it is necessary to set following conditions. (a)Emergency stop condition, and NC–Parameter PWE=1. (b)Stop condition the Ladder program. nc D Method of Outoutting Ladder. (1) Select "Diagnose screen" by key in key. (2) Key in key and optionally key in [Files No.]. .c (3) Key in . D Method of Outputting PMC–Patameter. w w w (1) Select "PMC STATUS screen" by key in key. (2) Key in key and optionally key in [File No.]. (3) Key in . CAUTION In case of output PMC–Parameter, it is necessary to set following condition.

(a)Edit mode. (b)Stop condition the Ladder program. 534 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power Mate–D/F/H) PMC PROGRAMMER (CRT/MDI) 11.10 ON–LINE DEBUGGING FUNCTION (ONLY FOR Power Mate–H) The on–line debugging function enables the monitoring and modification of ladder programs and signal status on personal computer's screen using a personal computer connected to the Power Mate through an RS–232C cable. FANUC FAPT LADDER–II is necessary to use the on–line debugging function.

(This software is a programming system for developing FANUC PMC sequence programs which operate on IBM PC/AT and compatible computers.) Software name FAPT LADDER–II Specification A08B–9201–J503 Personal computer om B–61863E/10 IBM PC/AT and compatible Name of Manual r.c In this section, only the parameter of on–line monitor driver for Power Mate–H and attention in use is described. Other points(connection of cable with personal computer, details of the operation, etc.) are described in the following manual. Spec.No. 11.10.1 Starting and Stopping the On–line Debugging Function Reference Items B–66184EN On–line function ce nt e FAPT LADDER–II OPERATOR'S MANUAL When using the on–line debugging function to connect a personal computer to the PMC, first start the driver that provides the communication function of the PMC. When starting or stopping the driver, it is necessary to set either of the following parameters. nc D Parameter screen for on–line monitor(dPARAMETERS FOR ONLINE MONITORc) Pressing the [MONIT] then [ONLINE] soft keys on the PMC menu screen causes the on–line monitor parameter screen to appear. ParameterdRS–232Cc = dUSEc : On–line monitor driver is used. w w w .c dNOT USEc: On–line monitor driver is not used. NOTE The CRT/MDI is necessary when the parameter is set on the "PARAMETERS FOR ONLINE MONITOR" screen. D Parameter in the Power Mate–H (No.0101#6) #7 #6 #5 #4 #3 #2 #1 #0 0101 #6 = 0 : On–line monitor driver is not used. 1 : On–line monitor driver is used. When either of the following conditions consists, the on–line monitor driver is started. DParameter "RS–232C" is "USE" DBit 6 of parameter No.0101 is "1" 535 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power PMC PROGRAMMER (CRT/MDI) Mate–D/F/H) B–61863E/10 w w w .c nc ce nt e r.c om CAUTION 1 The on–line monitor driver occupies the line while it is operating. In this state, other input/output functions cannot use the line. If other input/output functions use the line, it is necessary to display the above–mentioned parameter and stop the on–line monitor driver.

2 While the on–line monitor driver is operating, the following functions cannot be used. D[PMCLAD], [I/O], [EDIT], [SYSPRM] on CRT/MDI D[EDIT], [SYSTEM PARAM], [I/O] on DPL/MDI 3 In case of operating NC, the screen display of NC(Position, etc.) might be slow when using input/output functions(Load from PMC, Store to PMC, etc.). There is no problem in the operation of NC. It is recommended to using input/output functions while NC is not operating. 4 When the screen made by C language executor is displayed, the communication speed decreases. It is recommended to use input/output functions after moving to other screens(Position, etc.). 536 11. PMC PROGRAMMER (DPL/MDI) (ONLY FOR THE Power Mate–D/F/H) PMC PROGRAMMER (CRT/MDI) B–61863E/10 11.11 If an alarm is issued in the PMC, the alarn message is displayed on the CRT (PMC ALARM MESSAGE screeen). But in case of DPL/MDI, it is displayed only by R–relay status (ON or Off). Refer to the "APPENDIX L.ALARM MESSAGE LIST" for more information. ERROR LIST (1) Error ststus at power on or PROGRAM DOWN LOAD. 5 4 3 2 1 0 R9044 0 1 2 om 6 r.c 7 ER01 PROGRAM DATA ERROR (RAM) ER03 PROGRAM SIZE ERROR (OPTION) 4 ER04 LADDER OBJECT TYPE ERROR ce nt e 3 5 6 6 4 3 2 1 0 ER16 RAM CHECK ERROR (PROGRAM RAM) 1 ER17 PROGRAM PARITY 2 ER18 PROGRAM DATA ERROR BY I/O 3 ER19 LADDER DATA ERROR 4 ER20 SYMBOL/COMMENT DATA ERROR 5 ER21 MESSAGE DATA ERROR 6 ER22 PROGRAM NOTHING 7 ER23 PLEASE TURN OFF POWER 0 w w w .c R9046 5 ER07 NO OPTION (LADDER STEP) nc 7 7 537 om w w w .c nc ce nt e r.c IV. STEP SEQUENCE FUNCTION GENERAL w w w .c nc ce nt e r.c om 1 1. GENERAL STEP SEQUENCE FUNCTION B–61863E/10 541 1. GENERAL STEP SEQUENCE FUNCTION 1.1 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.

r.c om STEP SEQUENCE METHOD B–61863E/10 Ladder method ce nt e 1.1 (a) 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. w w w .c nc 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. CALL CALL Subprogram Subprogram 1.1 (b) Module method 542 1. GENERAL STEP SEQUENCE FUNCTION B–61863E/10 om 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. Step 1 Transition r.c Step 2 ce nt e Subprogram of ladder diagram Drawing flow 1.1(c) Step sequence method The step sequence method has the following features: w w .c nc (1) Increased programming efficiency D Since the flow of processes can be programmed directly, simple, correct programming is enabled, reducing the time required for programming. D 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. D Structured modules can be used again easily. (2) Easy debugging and maintenance D Graphical display enables the operator to easily understand the execution state of a program visually. D Erroneous steps in a program can be found easily. D A part of a program can be easily modified. (3) High-speed program D Since only the subprograms required for a certain process are executed, the cycle time is reduced. (4) Transition from ladder programs D 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. 543 1. GENERAL STEP SEQUENCE FUNCTION] Step A Starts execution. Waits for machining request.

(Process 1) Machining request?

±When machining is requested Step C Holds a workpiece on the pallet. (Process 2) Transition D Loading completed? ±Once loading has been completed Step E Machines the workpiece.

(Process 3) Transition F Machining completed? ±Once machining has been completed Step G Unloads the workpiece to the pallet. (Process 4) Transition H Unloading completed? ±Once unloading has been completed Step I om Transition B r.c [B–61863E/10 Moves the pallet. (Process 5) Example of machining the workpiece ce nt e 1.1 (d) 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. w w w .c nc 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. 544 1. GENERAL STEP SEQUENCE FUNCTION B–61863E/10 1.2 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. GRAPHICAL SYMBOLS Table 1.2 List of graphical symbols Display Display of programming manual CNC Device Step Sn Sn] Sn [Transition Pn] Sn [ ] Sn [Pn Pn ce nt e Divergence of Selective Sequence Sn r.c Initial Step FAPT LADDER of Personal Computer om Contents Convergence of Selective Sequence nc Divergence of Simultaneous Sequence .c Convergence of Simultaneous Sequence w w w Jump → Ln > Ln > Ln Label Ln → < Ln < Ln Block Step] Sn Initial Block Step [ ] Sn End of Block Step 545] Sn [ ] Sn] Sn [ ] Sn 1. GENERAL STEP SEQUENCE FUNCTION 1.3 B–61863E/10 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. PROGRAMMING (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) om (5) Write to the FlashROM (6) Execute (7) Diagnosis and debugging r.c (8) Correct a subprogram of ladder diagram (editing) RS232C CNC device PMC- RB4/RB6/ RC4/NB2 ce nt e Personal Computer Memory card FAPT LADDER software ± (4) Transfer to the CNC device FlashROM (1) Create Step Sequence program (editing) (6) Execute (7) Diagnosis and debugging (2) Create a subprogram of ladder (8) Correct a subprogram of ladder (editing) nc diagram (editing) (3) Compile (5) Write to the FlashROM Table1.3 lists the step sequence functions supported by a personal computer (on which the FAPT LADDER software package is installed) and CNC. w w w .c 1.3 Programming to create a program 546 1. GENERAL STEP SEQUENCE FUNCTION B–61863E/10 Table 1.3 Step sequence functions Ę : usable Functions PMC-RB4/ RB6 PMC-RC4 PMC-NB2 Ę Ę Ę FAPT LADDER of PERSONAL COMPUTER Display and edit of a program Display of subprogram list Delete a subprogram Edit a subprogram of StepSequence form Edit a subprogram of ladder diagram Ę Compile Decompile Input and output Input and output with RS232C Write to a FlashROM Execution of program execution of a ladder diagram ce nt e execution of Step Sequence program Ę Ę Ę r.c Input and output with a memory card Ę Ę Ę om Create a new subprogram Diagnosis and debugging (note1) Diagnosis of Step Sequence program Diagnosis of a ladder diagram Set and display a monitoring timer w w w .c nc 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). 547 2. STEP SEQUENCE BASICS B–61863E/10 STEP SEQUENCE BASICS w w w .c nc ce nt e r.c om 2 STEP SEQUENCE FUNCTION 548 STEP SEQUENCE FUNCTION B–61863E/10 2.1 2.

STEP SEQUENCE BASICS 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. TERMINOLOGY (Block) [ ] S1 (Initial Step) om P100 (Transition) S2 (Step) P101 (Transition) r.c L1 (Label) S3 (Divergence of Selective Sequence) ce nt e (Divergence of Simultaneous Sequence) nc (Convergence of Simultaneous Sequence) .c (Convergence of Selective Sequence) w] S10 (Block Step) w w L1 (Jump) 2.1(a) Step sequence elements 549 2. STEP SEQUENCE BASICS STEP SEQUENCE FUNCTION B–61863E/10 (1) Step Sn (Pm) om 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 ce nt e r.c 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. Table 2.1 Step state State Active Transition from execution to halt. The action program (subprogram) is executed once only, then the step automatically transits to halt. Halt Not activated state.

The action program (subprogram) has not yet been executed. .c w Inactivate (halt status) w w Activated step. The action program (subprogram) is being executed. Transition to halt nc Inactive Processing Execution Activate (active status) Inactivate (transition to halt) Fig.2.1(b) Step state transition 550 Display Sn Sn 2. STEP SEQUENCE BASICS STEP SEQUENCE FUNCTION B–61863E/10 (3) Transition Pn Executing step1 Executing step2 Executing step3 S1 (step1) P10 (Condition is true) S2 (step2) P20 S3 (step3) S1 (step1) ce nt e S1 (step 1) r.c om 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. P10 P10 S2 (step2) S2 (step2) P20 (Condition is true) P20 S3 (step3) S3 (step3) w w w .c nc 2.1(c) Transition of step state by the transition 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. Executing step1 Executing step1 S1 (step1) S1 (step1) P10 P10 S2 (step2) S2 (step2) P20 (Condition is true) P20 S3 (step3) S3 (step3) 2.1(d) Transition of step state by transition 551 2. STEP SEQUENCE BASICS STEP SEQUENCE FUNCTION B–61863E/10 (4) Initial Step [ ] Sn (Pm) Stopping program (STOP)] S1 (step1) P10 S2 (step2) [ce nt e P20 Executing program (RUN)] S1 (step1) r.c [om 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). S3 (step3) P10 S2 (step2) P20 S3 (step3) 2.1(e) Activate of initial step nc 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. 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. w w w .c (5) Divergence and Convergence of Selective Sequence 552 2. STEP SEQUENCE BASICS STEP SEQUENCE FUNCTION B–61863E/10 S1 (Divergence of selective sequence) P21 S21 P22 S22 P23 S23 When transition P22 evaluates true ± S1 S2 (true) (true) S22 S23 S21 S22 ce nt e S21 r.c When transition P21 evaluates true ± om (Convergence of selective sequence) S23 2.1(f) Selective sequence (6) Divergence and Convergence of Simultaneous Sequence w w w .c nc 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. 553 2. STEP SEQUENCE BASICS STEP SEQUENCE FUNCTION B–61863E/10 S1 P10 S21 S22 S31 S32 (Divergence of simultaneous sequence) S23 S33 om (Convergence of simultaneous sequence) S4 r.c When transition P10 evaluates true ± ce nt e S1 P10 (true) S21 S22 S23 S31 S32 S33 nc S4 2.1(g) Simultaneous sequence 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. w w w .c (7) Jump and Label 554 STEP SEQUENCE FUNCTION B–61863E/10 2.

STEP SEQUENCE BASICS Executing step1 Executing step3 S1 (step1) P10 P10 S2 (step2) S2 (step2) P20 P20 S3 (step3) S3 (step3) ³³³ P30 r.c P30 (Condition is true) om L1 (Label) L1 S1 (step1) L1 L1 (Jump) ce nt e 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. .c nc A block is executed as the main program in a step sequence, or called from another step sequence program as a subprogram.

Block (P1)] [ ] w w w [Block (P2) 2.1(i) Block (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. 555 2. STEP SEQUENCE BASICS STEP SEQUENCE FUNCTION Ladder (Second level) B–61863E/10 Block2 (P2) [ ] CALLU P2 om CALLU P3 Block3 (P3)] ce nt e r.c [2.1(j) Calling block (10) Block step (calling step sequence program) 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. w .c nc] Sn (Pm) Block (P1) Block (P2) [[w w] S1] S231 S232 S21 S22] S23 (P2) S233 S3 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. 556 2. STEP SEQUENCE BASICS STEP SEQUENCE FUNCTION B–61863E/10 Block (P1) [ ] S1 S21 S22 S231 om S232 S3 r.c S233 2.1(l) Program without block step ce nt e (11)End of block step w w w .c nc Use an end block step to terminate nested-block-step calling and to return to the calling sequence.

557 2. STEP SEQUENCE BASICS STEP SEQUENCE FUNCTION B–61863E/10 2.2 EXECUTION OF STEP SEQUENCE Editing (source program) Compile Executing (ROM format program) END1 (SUB 1) function Second level (Ladder diagram) om First level (Ladder diagram) CALL CALL ce nt e Third level (Ladder diagram) r.c END2 (SUB 2) function END3 (SUB 48) function Subprogram P1 (Ladder diagram)] Subprogram P2 nc [Subprogram P3 (Ladder diagram) w w .c (Step sequence) w [ ] Subprogram P4 (Step Sequence) S S Subprogram Pn END (SUB 64) function 2.2(a) Structure of program 558 STEP SEQUENCE FUNCTION B–61863E/10 2. STEP SEQUENCE BASICS In the step sequence method, a program is created (edited) in units of subprograms.

The edited source program is compiled andconverted 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. tn-8 tn+0 T msec First level Second level tn+8 8msec 8msec T msec T msec °division °finished nc °division Third level (depends on the PMC model) tn+16 ce nt e 8msec r.c om 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. w w w .c 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 is not determined in advance, the ladder being automatically divided upon the determined period elapsing. A PMC which allows step sequence programming executes the second level ladder in undivided system. 559 2. STEP SEQUENCE BASICS STEP SEQUENCE FUNCTION B–61863E/10 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. r.c om 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. LEVEL1 (Ladder diagram) ce nt e LEVEL2 (Ladder diagram) CALLU P2 CALL P1 R0. 0 nc LEVEL3 (Ladder diagram) P2 (Step sequence) L1] S1 (P3) [w .c P1 (Ladder diagram) P4 w w S2 (P5) P6 L1 P3 (Ladder diagram) P4 (Ladder diagram) 2.2(c) Execution of step sequence 560 2. STEP SEQUENCE BASICS STEP SEQUENCE FUNCTION B–61863E/10 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). 8msec om T msec LEVEL1 First level r.c LEVEL2 Second level Subprogram P1 ce nt e P2 Step P3 P4 Transition LEVEL3 nc Third level 2.2(d) Timing of execution of step sequenceprogram w w w .c 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. 561 3. CONFIGURATION AND OPERATION OF STEP-SEQUENCE STEP SEQUENCE FUNCTION PROGRAMS CONFIGURATION AND OPERATION OF STEPSEQUENCE PROGRAMS w w w .c nc ce nt e r.c om 3 B–61863E/10 562 3. CONFIGURATION AND OPERATION OF STEP-SEQUENCE PROGRAMS STEP SEQUENCE FUNCTION B–61863E/10 3.1 A step is a unit of processing in a program. STEP [Display] Sn (Pm) [Contents] D Assign a step number to a step.

om D Define a step number (Sn), necessary for controlling execution, and subprogram number (Pm) specifying actua processing, for a step. D The same step number cannot be used twice in a program. r.c D 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. Contents of operation ce nt e State Activate Inactivate Execution Sn 1 Transition to Transition from execution to halt. halt The action program (subprogram) is executed once only, then the step automatically transits to halt. Sn 0 Sn 0 nc Not activated state. The action program (subprogram) has not yet been executed. .c NOTE Sn. 0 Refer to 4.(2) w State transition of Step B w Transition A Inactivate (halt state) Step B w Sn.0 note1) Activated step. The action program (subprogram) is being executed. Stop Example) Display Transition A Activate (execution state) Transition C Transition C Inactivate (transition to halt) (Execute one time) 563 3. CONFIGURATION AND OPERATION OF STEP-SEQUENCE STEP SEQUENCE FUNCTION PROGRAMS B–61863E/10 [Example] After the M7 code is decoded, control is transferred to the next step using a DEC functional instruction. Subprogram P1 MF R0.0 DEC F0 F7.0 Subprogram P101 P101 R0.0 w w w .c nc ce nt e TRSET r.c 711 f om S1 (P1) 564 3. CONFIGURATION AND OPERATION OF STEP-SEQUENCE PROGRAMS STEP SEQUENCE FUNCTION B–61863E/10 3.2 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. INITIAL STEP [Display]] Sn (Pm) om [[Contents] D Define a step number (Sn), necessary for controlling execution, and subprogram number (Pm) specifying the actual processing, for an initial step. r.c D All initial steps are activated when the other steps are not activated. D Each block must contain at least one initial step.

No limit is applied to the number of initial steps contained in a block. D A block having no initial step cannot be executed if called.

ce nt e D Assign a step number to an initial step. D The same step number cannot be used more than once in a program. D In parallel branch, one initial step is required for each path. (See example 2.) [Example1]] L1 S1 When a program is executed, step P1, specified by an initial step, is activated first. P101 S2 P102 L1 w w w .c nc [565 Initial step S1 is executed in the same way as normal step once S1 has been executed. 3. CONFIGURATION AND OPERATION OF STEP-SEQUENCE STEP SEQUENCE FUNCTION PROGRAMS B–61863E/10 Example2 L1 When a program is executed, steps S3 and S4, specified by an initial step, are activated first. S1 P101]] S3 P102 P103 S4 S5 P110 ce nt e L1 Once steps S3 and S4 have been executed, initial steps S3 and S4 are executed in thesame way as normal step when the program starts from step S1. r.c [[om S2 3.3 A transition specifies the conditions governing the transition from the step to the next step. TRANSITION [Display] nc Pn [Contents] D Only one transition is required between steps. .c D Transition between steps is performed as described below. w S1 w P101 While S1 is activate, only S1 and P101 are executed. Other steps and transition are not executed. When the transition P102 evaluates true unles S2 is not being executed, the state is ignored. S2 w P102 When the transition P101 evaluates true, control passes fro S1 to S2. In this case, when the condition i true, S1 is terminated regardless of the state of S1, and S2 is activated. D 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). 566 3. CONFIGURATION AND OPERATION OF STEP-SEQUENCE PROGRAMS STEP SEQUENCE FUNCTION B–61863E/10 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] om [Contents] D Transitions are placed after a divergence of selective sequence. D The step connected to the transition for which the conditions are true is first activated. r.c D When the conditions for any transition are true simultaneously, the leftmost step is activated. D A selective sequence can create up to 16 paths. ce nt e [Example] [3.5] S1 P100 P101 S2 S3 When the conditions for P101 are satisfied earlier than those of P102, step S3 is activated. nc It combines two or more divergent paths to the main sequence. [Display] w w w .c CONVERGENCE OF SELECTIVE SEQUENCE [Contents] The number of divergent paths must match that of the convergent paths.

[Example] S2 S3 P102 P103 S4 567 While step S3 is executed, the transition P103 evaluates true, thus step S4 is activated. 3. CONFIGURATION AND OPERATION OF STEP-SEQUENCE STEP SEQUENCE FUNCTION PROGRAMS DIVERGENCE OF SIMULTANEOUS SEQUENCE A simultaneous sequence branches to two or more sequences, and all steps are activated simultaneously. [Display] om 3.6 B–61863E/10 [Contents] D A transition must be placed before a divergence of simultaneous sequence. D All branched steps are activated simultaneously, then executed.

r.c D A simultaneous sequence can create up to 16 paths. [Example]] S1 ce nt e [When the transition P101 evaluates true, step S2 and S3 are activated simultaneously. P101 S3 w w w .c nc S2 568 3. CONFIGURATION AND OPERATION OF STEP-SEQUENCE PROGRAMS STEP SEQUENCE FUNCTION B–61863E/10 It combines two or more divergent paths to the main sequence. CONVERGENCE OF SIMULTANEOUS SEQUENCE [Display] om 3.7 [Contents] S10 S20 When the transition P120 evaluates true, step S10 and S20 are terminated and step S21 is activated. ce nt e P120 r.c D A convergence of simultaneous sequence is processed as follows. S21 D Wait processing is processed as follows. case1) S15 P110 P115 nc S10 w w w .c S11 S16 When the transition P109 evaluates true unless both of step S11 and S16 are active, control does not pass to step S20. When the transition P109 evaluates true while both of S11 and S16 are active, S11 and S16 are terminated and S20 is activated. In the case, P109 provides the termination conditions for both S11 and S16. P109 S20 case2) To specify the termination conditions for S11 and S16 separately, place the conditions in P111 and P116 and specify two dummy steps, S12 and S17, as shown S11 S16 P111 P116 S12 (dummy) A dummy step also requires a step number S17 and subprogram number. Alsospecify a (dummy) dummy transition condition, which becomes always true, in P110. P110 S20 569 3. CONFIGURATION AND OPERATION OF STEP-SEQUENCE STEP SEQUENCE FUNCTION PROGRAMS 3.8 B–61863E/10 A jump controls the execution of steps non-sequentially, together with a transition. JUMP [Display] Ln om [Contents] D Specify a jump destination label (Ln). D The step to which control is transferred (jumped) is activated. D The jump destination must be within the same program. r.c D A jump cannot be performed from outside a simultaneous sequence to within the simultaneous sequence, or from within a simultaneous sequence to outside. D A jump cannot be performed between parallel-branched paths. ce nt e [Example] [ ] L1 S1 When steps S4 and S5 ar executed and the transition P110 evaluates true, the program is repeated from initial step S1. .c nc P101 S2 S3 P102 P103 S4 S5 P110 w w L1 A label specifies the jump destination. LABEL [Display] w 3.9 Ln [Contents] Specify the jump destination label (Ln). [Example] Refer to an example described on the jump function (3.8).

570 3. CONFIGURATION AND OPERATION OF STEP-SEQUENCE PROGRAMS STEP SEQUENCE FUNCTION B–61863E/10 3.10 A block step specifies the step sequence subprogram to be executed. BLOCK STEP [Display]] Sn (Pm) [Contents] S1 P101] S2 (P2) ce nt e r.c NOTE 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. Example) om 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.

S1 P101 P2 S20 P120 S20 equal P120 S21 P121 P121 .c nc S21 P102 S3 w w S3 w 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. 571 3. CONFIGURATION AND OPERATION OF STEP-SEQUENCE STEP SEQUENCE FUNCTION PROGRAMS B–61863E/10 3.11 This is an initial step on the block step. INITIAL BLOCK STEP [Display] [ ] Sn (Pm) [Contents] om D Define a step number (Sn), necessary for controlling execution, and subprogram number (Pm)specifying the actual processing, for an initial step. r.c D This step has the same function and graphical symbol asan initial step. This terminates a block step. END OF BLOCK STEP [Display] ce nt e 3.12 [Contents] D Use this step to terminate a block step. D Each block requires at least one end block step. No limit is applied to the number of end block steps.

nc [Example]] S1 P100 S2 P102 w w w .c [572 P103 4 4. EXTENDED LADDER INSTRUCTIONS STEP SEQUENCE FUNCTION B–61863E/10 EXTENDED LADDER INSTRUCTIONS w w w .c nc ce nt e r.c om 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. 573 4. EXTENDED LADDER INSTRUCTIONS STEP SEQUENCE FUNCTION B–61863E/10 4.1 [Function] FUNCTIONAL INSTRUCTION TRSET D This instruction describes that the conditions for a transition have been true. D This instruction is used in a subprogram which is called from a transition. [Format] r.c TRSET (SUB122) om ACT [Contens] PMC ADDRESS (S ADDRESS) D This address is used to read the logical state of a specified step. 0 : Transition to halt state, or halt state 1 : Execution state D 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 D This address allows any subprogram to reference the state of any step. D Data cannot be written into state signal Sn. 0. D 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. w w w .c nc ce nt e 4.2 In case of starting the executing of S2 and S6 synchronously: S1 S4 P1 P4 S2 S5 P2 P1 S3 S6 574 Sub program S5.0 TRSET STEP SEQUENCE FUNCTION B–61863E/10 SPECIFICATION OF STEP SEQUENCE w w w .c nc ce nt e r.c om 5 5. SPECIFICATION OF STEP SEQUENCE 575 5. SPECIFICATION OF STEP SEQUENCE STEP SEQUENCE FUNCTION B–61863E/10 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 Number of paths Up to 16 paths r.c om 32 columns @@@@ @@@@ ce nt e @@@@ Up to 16 paths @@@@ w w w .c nc @@@@ 576 Up to 16 paths 5.

SPECIFICATION OF STEP SEQUENCE STEP SEQUENCE FUNCTION B–61863E/10 D One transition must exist between step and step. 5.2 GENERAL RULES S1 (P10) S1 (P10) sub prog Correct CALL P10 P1 or S1 CALL P11 S2 (P11) The step S1 calls to subprogram P10, P11. om S2 (P11) r.c D The transition shall never be repeated even at the point of the divergence and the convergence. S1 S1 Correct ce nt e P1 P2 P10 P1 P10 S3 S10 S3 S10 S2 S10 S2 P10 P2 P11 P2 P11 Correct nc P12 w w w .c S12 577 S12 5. SPECIFICATION OF STEP SEQUENCE STEP SEQUENCE FUNCTION B–61863E/10 D When a simultaneous sequence is specified in another simultaneous sequence, one convergence must not be used for each sequence. S1 S2 S1 P2 S3 om P2 S2 S3 S4 r.c Correct S4 P1 P3 S5 (dumy) ce nt e S5 P1 S6 .c nc D When a selective sequence is specified in a simultaneous sequence, dummy steps must be required both after the divergence and before convergence. P2 w w w S1 P4 S2 S3 P3 P5 S1 Correct S2 (dummy) P2 P4 S3 S4 P3 P5 P1 S5 (dummy) S4 P1 S6 578 5. SPECIFICATION OF STEP SEQUENCE STEP SEQUENCE FUNCTION B–61863E/10 D In case of branching again immediately after the convergence, a step/transition is required between the divergence and convergence. P2 P10 S3 S10 S1 Correct S2 P100 (dummy) S100 (dummy) S2 S10 P2 P11 S12 S13 om S2 P1 P10 S3 S10 S2 S10 r.c S1 Correct ce nt e P2 P11 S100 (dummy) P100 (dummy) S12 S13 nc D Immediately after the block step, a dummy transition which is always true is needed. S10 [ ] S100 P100] S11 When block step S11 is used, transition P11 and P101 cannot be omitted S101 note) P11 P101 S12 w w w .c P10 579 P11 is a dummy transition.

The transition condition of P11 must always be true. 5. SPECIFICATION OF STEP SEQUENCE STEP SEQUENCE FUNCTION B–61863E/10 D The divergence must be terminated with the same type of convergence. P2 P4 S2 P3 P4 S3 S2 S3 P5 P3 P5 P2 P2 S2 r.c Correct om P2 Correct S2 S3 P3 S3 ce nt e P3 D The number of convergences must match that of divergences. P2 S2 S3 L1 Correct .c P3 P1 nc P1 w S4 L1 P3 S4 P2 P1 S2 P3 S3 D The number of convergences must match that of divergences, even at the end of a block step. w w P1 S2 P2 S2 Correct 580 P3 P2 STEP SEQUENCE FUNCTION B–61863E/10 5. SPECIFICATION OF STEP SEQUENCE D It is not possible to jump to the other subprogram. Sub–program P1 Sub–program P2 S10 S100 P10 L1 S101 S11 S20 P11 P20 S12 L1 P101 r.c S102 om P100 ce nt e D It is not possible to jump from a simultaneous sequence to another simultaneous sequence. S1 .c nc P1 S2 S10 P2 P10 S3 L1 P3 w w w L1 S4 581 5. SPECIFICATION OF STEP SEQUENCE STEP SEQUENCE FUNCTION B–61863E/10 D It is not allowed to jump from inside of the simultaneous sequence to outside. L1 S1 S10 P2 P10 S3 S11 S20 P20 L1 w w w .c nc ce nt e r.c S2 om P1 582 5. SPECIFICATION OF STEP SEQUENCE STEP SEQUENCE FUNCTION B–61863E/10 5.3 The use of the following functional instructions is restricted in steps and transitions.

Group A B Functional instructions Description The instructions operate when a signal changes Condition Multiple functional instructions having the same number are used. Problem Not activated. Correct operation cannot be guaranteed. Restriction due to the interface. Condition Data is input or output by using two subprograms. Problem Invalid return value. Not terminated. WINDR (SUB51) WINDW (SUB52) DISP (SUB49) DISPB (SUB41) EXIN (SUB40) r.c (1) Functional instructions of group A CTR (SUB5) CTRC (SUB60) TMR (SUB3) TMRB (SUB24) TMRC (SUB54) DIFU (SUB57) DIFD (SUB58) om EXCLUSIVE CONTROL FOR FUNCTIONAL INSTRUCTIONS Since these functional instructions operate when the corresponding signals change, they may not operate correctly when called from multiplesteps. ce nt e 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. S1 (P100) Subprogram P100 CTR nc P1 .c S2 (P100) X1.0 w Subprogram P1 X1.0 w w TRSET 583 1 R0.0 ( ) 5. SPECIFICATION OF STEP SEQUENCE STEP SEQUENCE FUNCTION B–61863E/10 Correct program Divide the subprogram so that ACT of CTR is called after it is set to off. S1 (P100) Subprogram P100 CTR S2 (P102) X1.0 r.c P103 S3 (P100) R0.0 ( ) om P101 1 Subprogram P101 X1.0 P101 ce nt e TRSET S4 (P102) Subprogram P102 CTR P103 1 R0.0 ( ) Subprogram P103 R9091.1 w w w .c nc R9091.0 TRSET (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.

584 5. SPECIFICATION OF STEP SEQUENCE STEP SEQUENCE FUNCTION B–61863E/10 Example) Subprogram P100 S1 (P100) R9091.1 WINDR P1 S2 (P101) R10 R0.0 ( ) Subprogram P1 TRSET Subprogram P101 R9091.0 R10 R0.0 ( ) r.c WINDR om R0.0 ce nt e Correct program Correct the program so that ACT is set to on and off within one subprogram. S1 (P100) Subprogram P100 R9091.1 P1 S2 (P101) w w w .c nc CALLU R0.0 ( ) P2 Subprogram P1 R0.0 TRSET Subprogram P101 R9091.0 CALLU R0.0 ( ) P2 Subprogram P2 R0.1 WINDR 585 R10 R0.0 ( ) STEP SEQUENCE FUNCTION 6. CRT/MDI OPERATION 6 B–61863E/10 CRT/MDI OPERATION (1) Displaying the sequence diagram om The following operations are supported to enable the diagnosis and debugging of a step sequence program. (2) Displaying the run time of the step sequence program w w w .c nc ce nt e r.c (3) Monitoring the run time of the step sequence program 586 6. CRT/MDI OPERATION STEP SEQUENCE FUNCTION B–61863E/10 6.1 The diagnosis and debugging of a step sequence program have four screens. DISPLAYING OF SEQUENCE PROGRAM (1) Program configuration list (main screen) (2) Step sequence screen (3) List screen 6.1.1 om (4) Ladder screen Press the [STPSEQ] key and display the program configuration list. STPSEQ > LEVEL2 V P0002 V P0009 V P0022 · · V P0202 LEVEL3 V P0004 V P0014 V P0024 · · V]P0304 V P0005 V P0015 V P0025 · · V]P0405 V P0006 V P0016 V P0026 · · V]P0406 ce nt e LEVEL1 V P0001 V P0008 V P0021 · · · · V P0101 PROGRAM:(STEP SEQUENCE DEMO PROGRAM) P0001 ( ) SUB PROGRAM NO.1 [UP] [MONIT RUN r.c Program Configuration List (Main Screen) DOWN] [TIME] [P–ADRS V P0007 V P0017 V] P0027 V]P0407] [ZOOM] Items displayed on the screen Display Contents Display by [ZOOM] key Ladder first level Ladder diagram LEVEL2 Ladder second level Ladder diagram LEVEL3 Ladder third level note1) Ladder diagram V Pxxx Subprogram Ladder diagram V] Pxxx Subprogram Step sequence diagram nc LEVEL1 .c Pxxx indicates a subprogram number. w w w 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. 587 STEP SEQUENCE FUNCTION 6.1.2 Step Sequence Screen B–61863E/10 V], then press the (1) Position the cursor to a program indicated by [ZOOM] key. STPSEQ > LEVEL1 V P0001 V P0008 V P0021 · · · · V P0101 [UP PROGRAM:(STEP SEQUENCE DEMO PROGRAM) P0407 (MAIN) STEP SEQUENCE NO.1 LEVEL2 V P0002 V P0009 V P0022 · · V P0202] [LEVEL3 V P0004 V P0014 V P0024 · · V]P0304 DOWN] [V P0005 V P0015 V P0025 · · V]P0405 TIME V P0006 V P0016 V P0026 · · V]P0406 MONIT RUN V P0007 V P0017 V] P0027 om 6. CRT/MDI OPERATION] [V]P0407 P–ADRS] [ZOOM] r.c 6.1.2(a) Program configuration list (main screen) Example) ce nt e When the cursor is positioned to V] P0407 and press the [ZOOM] key, the subprogram P407 isdisplayed. (2) Displayed Step Sequence Activated steps are indicated by red V (highlighted V on a monochrome display). (In this manual, activated steps are indicated by H.) nc PCLAD > PROGRAM:(STEP SEQUENCE DEMO PROGRAM) MONIT RUN S0001 P0001(ROTATE) ROTATE THE WORK TIP P0407 1– 1 S2 S20 P10 P13 S3 S11 S13 P20 P3 S4 P11 P14 S14 P15 P21 S23 P22 [MAIN] [CHANGE P30 S31 P23 –> L2 P31 S32] [TIME] [P–ADRS] [6.1.2(b) Step sequence screen 588 L4 S30 S24 S15 w w P2 w .c] ZOOM] 6. CRT/MDI OPERATION STEP SEQUENCE FUNCTION B–61863E/10 Meaning of display Display Contents Display by [ZOOM] key Ladder diagram V Sxxx Step Ladder diagram V] Block step Step sequence diagram Transition Ladder diagram Selective sequence Cannot zoom. Simultaneous sequence Cannot zoom. L2 Jump Cannot zoom. L2 Label Sxxx Sxxx + Pxxx om Initial step [V] Cannot zoom. w w w .c nc ce nt e r.c 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. PCLAD <> PROGRAM:(STEP SEQUENCE DEMO PROGRAM) MONIT RUN S0001 P0001 (ROTATE) ROTATE THE WORK TIP V] P0001 V] P0002 V] P0004 V] P0005 V] P0006 V] P0007 V P0008 V P0021 · · · · V P0101 V P0009 V P0022 · · V]P0202 V P0014 V P0024 · · V]P0304 V P0015 V P0025 · · V]P0405 V P0016 V P0026 · · V]P0406 V P0017 V]P0027 V]P0407 [UP] [DOWN] [ ] [ ] [[MAIN] [CHANGE] [ ] [ ] [589 ZOOM]] STEP SEQUENCE FUNCTION 6. CRT/MDI OPERATION B–61863E/10 [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 om Press the [TIME] key to display the time display screen (Fig. 6.2.1). [P–ADRS/P–SYMB] key r.c 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. 6.1.3 V, then press the ce nt e (1) Position the cursor to a program indicated by [ZOOM] key. Ladder Screen STPSEQ > LEVEL2 V P0002 V P0009 V P0022 · · V]P0202 nc LEVEL1 V P0001 V P0008 V P0021 · · · · V P0101 PROGRAM:(STEP SEQUENCE DEMO PROGRAM) [UP] [LEVEL3 V P0004 V P0014 V P0024 · · V]P0304 DOWN] [V P0005 V P0015 V P0025 · · V]P0405 TIME] [V P0006 V P0016 V P0026 · · V]P0406 MONIT RUN V P0007 V P0017 V] P0027 V]P0407 P–ADRS] [ZOOM] Example) When the cursor is positioned to LEVEL1, press the[ZOOM] key, the first level ladder is displayed. PCLAD > PROGRAM:(STEP SEQUENCE DEMO PROGRAM) S0001 P0001 (ROTATE) ROTATE THE WORK TIP w w w .c 6.1.3(a) Program configuration list (main screen)] S2 [S10 S20 P2 P10 P13 S3 S11 S13 MAIN] [CHANGE] [P20 TIME] [L4 S30 P30 S31 P–ADRS] [6.1.3(b) Step Sequence screen 590 MONIT RUN P0100 4– 2 ZOOM] 6. CRT/MDI OPERATION STEP SEQUENCE FUNCTION B–61863E/10 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). LADDER * STEP SEQUENCE DEMO PROGRAM * NET 0031–0033 MONIT RUN MOVN 20 D10 R10 WINDR R10 FIN / om R9091.1 FIN ( ) r.c FIN MOVN 20 R10 SUB 45 D10 ce nt e END1 SUB 1 SEARCH [DUMP] [ ] [ADRESS DPARA] [ ] [TRIGER] [ ] [WINDOW ONLEDT] [ ] [ ]] 6.1.3(c) Ladder screen [SEARCH] key Used for search within a subprogram. SEARCH ADRESS TRIGER WINDOW TOP BOTTOM SRCH RET W–SRCH N- SRCH w w w .c nc [NEXT F–SRCH [TOP] key Displays the top of a subprogram. [BOTTOM] key Displays the bottom of a subprogram. 591 STEP SEQUENCE FUNCTION 6. CRT/MDI OPERATION B–61863E/10 [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 [F–SRCH] key om Displays the ladder having the specified net number, at the top of the screen. Displays the specified functional instruction, at the to of the screen. [ADRESS/SYMBOL] key [TRIGER] key r.c 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. ce nt e 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 nc Displays the contents of addresses at the bottom of the screen. [DPARA] key Displays the data specified with functional instructions. While a sequence program is executing, a part of the ladder diagram can be changed. w w w .c [ONLEDT] key 592 6. CRT/MDI OPERATION STEP SEQUENCE FUNCTION B–61863E/10 6.2 The elapsed time of a step sequence program is displayed. TIMER SCREEN 6.2.1 Time Screen STEP NO. STATUS ELAPSE MONITOR S0001( ) EXEC 1000000 T(1) OVER S0002( ) EXEC 100 T(3) S0003( ) EXEC 10000 T(4) S0004( ) 1000000 · · · · · · · · UP] [DOWN] [SEARCH STEP NO. S0010(TILE S0011(S0012(S0013(· ·] [RESET STATUS. EXEC))))] [MONIT] r.c [MONIT RUN om STPSEQ > PROGRAM:(STEP SEQUENCE DEMO PROGRAM) ce nt e 6.2.1 Time screen Meaning of display Display STEP NO. Contents Step number S0001 : Step number (123456) : symbol display Step state EXEC : Active ELAPSE Actual elapsed time (per msec) The time is increasing during active state. MONITOR Monitor time T (1) : monitoring time number OVER : An elapsed time is over monitoring time nc STATUS space : Inactive w w w .c [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) 593 STEP SEQUENCE FUNCTION 6. CRT/MDI OPERATION 6.2.2 Monitoring Elapsed Time B–61863E/10 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. om (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. r.c "ER48 STEP SEQUENCE TIME OVER (xxH)" "xx" displays the content of address R9118 in hexadecimal code. Time N b Number Corresponding Add Address Time N b Number Corresponding Add Address R9118.0 5 R9118.4 2 R9188.1 6 R9118.5 3 R9188.2 7 R9188.6 4 R9188.3 8 R9188.7 w w w .c nc ce nt e 1 594 6. CRT/MDI OPERATION STEP SEQUENCE FUNCTION B–61863E/10 6.3 Operation time limits can be specified for a step sequence program. Up to eight steps can be monitored. MONITOR TIME SCREEN STPSEQ> PROGRAM:(STEP SEQUENCE DEMO PROGRAM) [DELETE STEP NO. S0001(S0010(MOVE S0002(S0003(ELAPSE 1000000 100 100 10000))))] [MONITOR 2000 1000 2000 20000 om NO. T(1) T(2) T(3) T(4) T(5)

T(6) T(7) T(8) MONIT RUN] [ ] [ ] [ ] r.c 6.3 monitor time screen Meaning of display Display Meaning Monitor time number T (1) : means monitor time 1. STEP NO. Step number S0001 : Step number ce nt e NO. (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. nc Operation Definition of monitor w w w .c (1) Position the cursor at the input position and input a step (or symbol). NO. T(1) T(2) STEP NO.

S0001( ) S0010(MOVE) ELAPSE 1000000 MONITOR 2000 Key in “MOVE” and push [INPUT] key. (2) Position the cursor at the input position and define a monitor time. NO. T(1) T(2) STEP NO. S0001( ) S0010(MOVE) ELAPSE 1000000 1000 MONITOR 2000 100 Keyin “100” and push [INPUT] key. 595 STEP SEQUENCE FUNCTION 6. CRT/MDI OPERATION B–61863E/10 Deletion of monitor Position the cursor at the deletion and press [DELETE] key. NO. T(1) T(2) STEP NO. S0001( ) S0010(MOVE) ELAPSE 1000000 100 MONITOR 2000 1000 om Alteration of monitor Position the cursor at the alteration position and input again. STEP NO. S0001(S0100( )) ELAPSE 1000000 2000 MONITOR 2000 1000 r.c NO.

T(1) T(2) w w w .c nc ce nt e Key in “S100” and push key. 596 6. CRT/MDI OPERATION STEP SEQUENCE FUNCTION EDITING FUNCTION OF LADD ER DIAGRAM 6.4.1 Program Configuration List (Main Screen) The display and editing of a step sequence program per subprogram aresupported. 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. STPSEQ > LEVEL1 V P0001 V P0008 V P0021 · · · · V P0101 UP PROGRAM:(STEP SEQUENCE DEMO PROGRAM) P0001 ( ) SUB PROGRAM NO.1 LEVEL2 V P0002 V P0009 V P0022 · · V P0202] [LEVEL3 V P0004 V P0014 V P0024 · · V]P0304 DOWN V P0005 V P0015 V P0025 · · V]P0405 V P0006 V P0016 V P0026 · · V]P0406] [ ] [P–ADRS ce nt e [om 6.4 MONIT STOP V P0007 V P0017 V] P0027 r.c B–61863E/10 V]P0407] [ZOOM] Items displayed on the screen Display Contents Display by [ZOOM] key LEVEL1 Ladder first level Ladder diagram LEVEL2 Ladder second level Ladder diagram LEVEL3 Ladder third level (Note) Ladder diagram V Pxxx Subprogram Ladder diagram V] Pxxx Subprogram Step sequence diagram nc Pxxx indicates a subprogram number. .c NOTE The third level ladder can be omitted. w w w [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. 597 STEP SEQUENCE FUNCTION 6.4.2 Step Sequence Screen B–61863E/10 (1) Position the cursor to a program indicated by ¢¢], then press the [ZOOM] key. LADDER > LEVEL1 V P0001 V P0008 V P0021 · · · · V P0101 [UP PROGRAM:(STEP SEQUENCE DEMO PROGRAM) P0407 ( ) STEP SEQUENCE NO.1 LEVEL2 V P0002 V P0009 V P0022 · · V P0202] [LEVEL3 V P0004 V P0014 V P0024 · · V]P0304 DOWN TIME V P0006 V P0016 V P0026 · · V]P0406 V P0007 V P0017 V] P0027] [P–ADRS V]P0407] [ZOOM] Program configuration list (main screen) r.c 6.4.2 (a)] [V P0005 V P0015 V P0025 · · V]P0405 MONIT STOP om 6. CRT/MDI OPERATION Example) ce nt e When the cursor is positioned to j] P0407 and press the [ZOOM] key , the subprogram P407 is displayed. (2) Displayed Step Sequence Activated steps are indicated by red j (highlighted j on a monochrome display). (In this manual, activated steps are indicated by J.) S10 P2 P10 P13 S3 S11 S13 P20 P3 S4 P11 P14 S14 P15 P21 S23 P22 S20 [MAIN] [CHANGE P30 S31 P31 P23 –> L2 S32] [ ] [P–ADRS 6.4.2 (b) Step sequence screen 598 L4 S30 S24 S15 w w] S2 w .c nc LADDER > PROGRAM:(STEP SEQUENCE DEMO PROGRAM) MONIT STOP S0001 P0001(ROTATE) ROTATE THE WORK TIP P0407 1– 1] [ZOOM] 6. CRT/MDI OPERATION STEP SEQUENCE FUNCTION B–61863E/10 Meaning of display Display Contents Display by [ZOOM] key Ladder diagram V Sxxx Step Ladder diagram V] Block step Step sequence diagram Transition Ladder diagram Selective sequence Cannot zoom. Simultaneous sequence Cannot zoom. L2 Jump Cannot zoom. L2 Label Sxxx Sxxx + Pxxx om Initial step [V] Cannot zoom. w w w .c nc ce nt e r.c 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 <> PROGRAM:(STEP SEQUENCE DEMO PROGRAM) MONIT STOP S0001 P0001 (ROTATE) ROTATE THE WORK TIP V] P0001 V] P0002 V] P0004 V] P0005 V] P0006 V] P0007 V P0008 V P0021 · · · · V P0101 [UP V P0009 V P0022 · · V P0202] [V P0014 V P0024 · · V]P0304 CHANGE] [V P0015 V P0025 · · V]P0405] [V P0016 V P0026 · · V]P0406 P–ADRS V P0017 V]P0027 V]P0407] [ZOOM] [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. 599 STEP SEQUENCE FUNCTION 6. CRT/MDI OPERATION B–61863E/10 [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 om 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. Ladder Screen LADDER > PROGRAM:(STEP SEQUENCE DEMO PROGRAM) LEVEL2 V P0002 V P0009 V P0022 · · V P0202 LEVEL3 V P0004 V P0014 V P0024 · · V]P0304 V P0005 V P0015 V P0025 · · V]P0405 V P0006 V P0016 V P0026 · · V]P0406 ce nt e LEVEL1 V P0001 V P0008 V P0021 · · · · V P0101 r.c (1) Position the cursor to a program indicated by j, then press the [ZOOM] key. 6.4.3 [UP] [DOWN 6.4.3 (a)] [ ] [P–ADRS MONIT STOP V P0007 V P0017 V] P0027 V]P0407] [ZOOM] Program configuration list (main screen) nc Example) When the cursor is positioned to LEVEL1, press the [ZOOM] key, the first level ladder is displayed.

w .c LADDER > PROGRAM:(STEP SEQUENCE DEMO PROGRAM) MONIT STOP S0001 P0001 (ROTATE) ROTATE THE WORK TIP P0100 4– 2 w w] S2 [S10 S20 P2 P10 P13 S3 S11 S13 P20] [ ] [MAIN] [CHANGE 6.4.3 (b) L4 S30 P30 S31 P–ADRS] [ZOOM] Step Sequence screen Example) When the cursor is positioned to ” P2”, press the [ZOOM] key, the subprogram P2 is displayed. 600 6. CRT/MDI OPERATION STEP SEQUENCE FUNCTION B–61863E/10 (2) Ladder Screen LADDER NET 0031–0033 MONIT STOP R9091.1 MOVN 20 D10 R10 WINDR R10 FIN / FIN ( ) om FIN MOVN 20 R10 SUB 45 D10 END1 ce nt e r.c SUB 1 6.4.3(c) Ladder screen Please refer to the following manual about the operations of editing a ladder diagram. FANUC PMC MODEL PA1/PA3/RA1/RA2/RA3/RB/RB2/RB3/ RB4/RC/RC3/RC4/NB LADDER LANGUAGE (B–61863E) PROGRAMMING MANUAL w w w .c nc III PMC PROGRAMMER(CRT/MDI) 5.2 Sequence Program Generation(LADDER) 601 STEP SEQUENCE FUNCTION 6. CRT/MDI OPERATION 6.5 CORRESPONDING FUNCTION B–61863E/10 The following ladder diagnosis and debugging functions can be used together with the step sequence functions. Ę : can be used Δ : can be used on condition : cannot be used Functions PMC-RB4/ RB6 PMC-RC4 PMC-NB2 Ę Ę Ę Ę Ę Δ caution1 Ę Δ Ę 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) r.c PMC Parameters screen (PMCPRM) Timer screen (TIMER) Counter screen (COUNTR) Keep relay screen (KEEPRL) Data table screen (DATA) Simple setting screen (SETING) om PMC Ladder diagram display (PMCLAD) ce nt e 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) Ę Ę Ę 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) caution1 Δ Δ Δ Δ Δ Δ Δ Ę Ę Ę Ę Ę Ę Ę Ę Ę Ę Ę Ę Ę Ę Ę Ę Input and output FAPT LADDER (HOST) Floppy cassette (FDCAS) FlashROM (F-ROM) Memory card (M-CARD) Other I/O device (OTHERS) Ę Ę Ę Δ caution2 Ę Ę Ę Ę Ę Ę Ę Ę Ę Ę Ę System Parameter (SYSPRM) Δ caution1 Ę Ę Ę Ę Ę Ę Ę Ę w w w .c nc Execute or stop the sequence program (RUN) Debug function (MONIT) Ladder debug function (DBGLAD) Descriptor table screen (GDT) User memory screen (USRMEM) User program debug function (DEBUG) caution1 CAUTION 1 An Editor card is needed. 2 It is possible to use while an Editor card is not mounted. 602 6. CRT/MDI OPERATION STEP SEQUENCE FUNCTION B–61863E/10 6.6 COMPATIBILITY OF LADDER DIAGRAM The PMC-RB4 and PMC-RC4 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. om 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. r.c When the ladder method is used: STEP SEQUENCE = NO.

PMC SYSTEM PARAMETER (1/2) = BINARY / BCD STEP SEQUENCE = ce nt e COUNTER DATA TYPE MONIT STOP YES / NO > 6.6 (a) BCD] [w] [ ] [ ] PMC-RB4/RB6 System parameter screen (first page) PMC SYSTEM PARAMETER w w .c nc [BINARY] [(2/2) FS0 OPERATOR PANEL = KEY DI ADDRESS = LED DO ADDRESS = MONIT STOP YES / NO KEY BIT IMAGE ADDRESS = LED BIT IMAGE ADDRESS = > [YES 6.6 (b)] [NO] [ ] [ ] [ ] PMC-RB4/RB6 System parameter screen (second page) 603 STEP SEQUENCE FUNCTION B–61863E/10 PMC SYSTEM PARAMETER (1/2) COUNTER DATA TYPE = BINARY / BCD LADDER EXEC = % (1–150) LANGUAGE EXEC RATIO = LANGUAGE ORIGIN (LANGUAGE AREA = = H H, SIZE = STEP SEQUENCE = % (0–99)] [ ] [ ] [ ] r.c BCD KB) YES / NO > [BINARY] [MONIT STOP om 6. CRT/MDI OPERATION 6.6 (c) PMC-RC4/NB2 System parameter screen (first page) (2/2) MONIT STOP ce nt e PMC SYSTEM PARAMETER FS0 OPERATOR PANEL = KEY DI ADDRESS = LED DO ADDRESS = YES / NO KEY BIT IMAGE ADDRESS = nc LED BIT IMAGE ADDRESS = > .c [YES NO] [ ] [ ] [ ] PMC-RC4/NB2 System parameter screen (second page) w w w 6.6 (d)] [604 om w w w .c nc ce nt e r.c V. PMC PROGRAMMER (SYSTEM P series) 1. GENERAL PMC PROGRAMMER (SYSTEM P series) B–61863E/10 1 GENERAL om The FAPT LADDER system can easily prepare sequence programs, symbol data, titles, and message of PMC-RB and PMC-RC, 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. Machine tool nc series 16/18/20/21 /Power Mate-D ce nt e r.c (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. .c PMC card DI/DO card System floppy loading Paper tape input/output PPR SYSTEM P series List output (FAPT LADDER) w w FAPT LADDER PMC-RA1/RA2 PMC WRITER FAPT LADDER PMC-RB/RC EPROM for PMC input/output w or FA WRITER EPROM for PMC input/output ROM for PMC module input/output Floppy input/output PRINTER Sequence program figure output 1 607 2.

FUNCTIONS OF PROCESSING 2 PMC PROGRAMMER (SYSTEM P series) B–61863E/10 FUNCTIONS OF PROCESSING (1) Input of sequence programs a) SYSTEM P series keyboard b) PPR tape reader (paper tape) d) PMC memory r.c c) Floppy om Input sequence programs using the following units when sequence programs are loaded into the memory of the SYSTEM P series. e) EPROM for PMC or ROM module (2) Sequence program display ce nt e 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. nc a) Alteration b) Insertion c) Deletion w w w .c (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 608 2. FUNCTIONS OF PROCESSING PMC PROGRAMMER (SYSTEM P series) B–61863E/10 (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.

w w w .c nc ce nt e r.c om Error codes are displayed at the lower right of the screen as ALARM=XXX. Refer to list of error codes in Appendix. 609 3. COMPONENT UNITS AND CONNECTIONS B–61863E/10 COMPONENT UNITS AND CONNECTIONS w w w .c nc ce nt e r.c om 3 PMC PROGRAMMER (SYSTEM P series) 610 PMC PROGRAMMER (SYSTEM P series) B–61863E/10 3.1 3. COMPONENT UNITS AND CONNECTIONS (1) SYSTEM P series This system serves as a programmer to generate and edit sequence programs. COMPONENT UNITS (2) Series 16 This system transfers a generated sequence program to CNC. (3) FANUC PPR (4) FANUC printer om This PPR inputs/outputs a sequence program by using a paper tape, and also output a source list to the printer. This printer prints out the sequence program. (5) FANUC PMC writer (6) FANUC FA Writer r.c This unit is used for writing a sequence program to the EPROM for PMC or ROM module when the sequence program has been completed. w w w .c nc ce nt e This unit is used for writing data to the EPROM or ROM module for the PMC after a sequence program has been created.

611 3. COMPONENT UNITS AND CONNECTIONS 3.2 CONNECTIONS OF UNITS PMC PROGRAMMER (SYSTEM P series) B–61863E/10 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 nc ce nt e r.c om This chapter mainly describes the connections between SYSTEM P series and I/O devices. w w w .c 3.2(a) External view of SYSTEM P Mark II 612 PMC PROGRAMMER (SYSTEM P series) 3. COMPONENT UNITS AND CONNECTIONS ce nt e r.c om B–61863E/10 nc 3.2(b) External view of SYSTEM P Mate 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. w w w .c 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). 613 3. COMPONENT UNITS AND CONNECTIONS PMC PROGRAMMER (SYSTEM P series) FANUC ROM WRITER B–61863E/10 (FANUC PMC WRITER) Adaptor for ROM modules (FANUC FA WRITER) CN2 CN1 EPROM for the PMC CN4 CN3 r.c Adaptor for 40-pin ROMs EPROM for the PMC Adaptor for 1M-bit EPROMs om Rear panel of the SYSTEM P Mark II FANUC PPR ROM module for the PMC 1 Tape input 2 Tape printout 3 List output ce nt e PMC-RA1/RA2/RB/RC FANUC PRINTER Prints sequence program figures. nc 3.2(c) Connection of SYSTEM P series with each unit (1) Connect FANUC PPR to connector CN1. .c (2) Connect FANUC PMC writer or FANUC FA writer to connector CN2. (4) Connect connector CN4 to PMC-RA1/RA2/RB/RB2/RC. 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 ³ 1 CHANNEL Connector JD5B on MAIN PCB ³ 2 CHANNEL w w w (3) Connect FANUC printer to connector CN3. 614 PMC PROGRAMMER (SYSTEM P series) B–61863E/10 3. COMPONENT UNITS AND CONNECTIONS 3.3 Figs. 3.3(a) - (b) show the panel of the SYSTEM P series keyboard. KEYBOARD OF SYSTEM P SERIES 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. om In this chapter, you should understand an outline of functions of these keys. Soft keys r.c Function keys Function keys ce nt e Numeric keypad Standard keyboard 3.3(a) Panel of the SYSTEM P Mark II key board w w .c nc Function keys w Data input keys 3.3(b) Panel of the SYSTEM P Mate key board 615 3. COMPONENT UNITS AND CONNECTIONS 3.3.1 LOAD Key (System Program Loading Key) 3.3.2 PMC PROGRAMMER (SYSTEM P series) B–61863E/10 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. F key is used to select an I/O device among I/O devices connected at that time. F Keys (F1 to F0) om These F keys are provided with an LED. When depressing a key, the LED 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 not lighting. r.c The correspondence between F keys and I/O devices is as shown below. (I) shows an input, while (O) shows an output. key: FANUC PPR paper tape reader (I) (2) key: Floppy disk input (I) ce nt e (1) key: Not used (4) key: Display of ladder diagram on SYSTEM P series screen (O) (5) key: FANUC PPR printer (O) (6) key: FANUC PPR paper tape puncher (O) (7) key: Floppy disk output (O) (8) key: PMC-PA1/PA2/RA1/RA2/RB/RB2/RC (I/O) nc (3) (9) key: FANUC PMC writer, FANUC FA writer (I/O) (10) key: FANUC printer (O) .c (The ladder diagram is printed on the printer.) (12) key: FANUC Floppy Cassette/FANUC FA Card adapter (O) Combination of F key and menu number of FAPT LADDER decided which function is to be executed. w w w (11) key: FANUC Floppy Cassette/FANUC FA Card adapter (I) 616 PMC PROGRAMMER (SYSTEM P series) B–61863E/10 3.3.3 3. COMPONENT UNITS AND CONNECTIONS Four R keys to are provided. The meaning of these keys differ according to the screen conditions at their operating time, even in case of the same key. R Keys (R0 to R3) (1) R key menu screen This screen is obtained just after loading a FAPT LADDER system program (1/2) or when pressing key only in a menu screen. Refer to Fig. 4.2.2. . A menu screen appears. om FAPT ladder start. Editing a ladder diagram starts. Not used in FAPT ladder. (Not accepted when pressing these keys) r.c Request key (see 4.8) Press NL keys, if a wrong key was pressed by mistake. The screen is reset to the condition before pressing the wrong R key. ce nt e (2) Other than R key menu screen 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. 1 When this key is pressed during printing of a ladder diagram on an external printer, the printer stops every page to be ready for key entry. 3 The signal display in a sequence program is alternately selected to symbols and addresses, each time this R1 is pressed during the display of the sequence program on the screen. Data on the last page are displayed, each time this key is pressed on the EDIT screen. 1 Data on the next page are displayed, each time this key is pressed on the EDIT screen.

2 Transfer is aborted when this key is pressed during ROM data transfer between SYSTEM P series and PMC-WRITER or floppy. w w w .c nc 2 When this key is pressed during data transfer between SYSTEM P series and PMC-RB/RC, data transfer is stopped. 617 3. COMPONENT UNITS AND CONNECTIONS 3.3.4 Data Keys and Screen Scroll Key PMC PROGRAMMER (SYSTEM P series) B–61863E/10 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. om (1) key Data entry from the SYSTEM P series keyboard are input into SYSTEM P series by depressing key. Two keys are located on the keyboard for easily operation. (2) key r.c 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. ce nt e (4) Arrow keys These keys are accepted only when a ladder diagram is being displayed on the screen, and used for scrolling the ladder diagram. NOTE w w w .c nc None of [INS] [DEL] [CHG] [AUX] keys and K key is employable in the FAPT LADDER.

618 PMC PROGRAMMER (SYSTEM P series) B–61863E/10 3.4 3.

COMPONENT UNITS AND CONNECTIONS (1) SYSTEM P series Mate An initial I/O device setting of ’FAPT LADDER’ for SYSTEM P series Mate is as follows. SETTING OF I/O DEVICE Table 3.4(a) FAPT LADDER (Mate) of table I/O device Channel F key PMC-RAM F8 CN2 PMC WRITER FA WRITER F9 CN3 External printer om CN1 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. r.c (Setting method of IO command) ’REQUEST =’ is displayed in the left bottom of screen and becomes the state which can be typed in. ii) Type in IO PPR, CN1 . ce nt e 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 (2) SYSTEM P Mark II The initial setting of I/O devices of FAPT LADDER for the SYSTEM P Mark II is as follows. nc Table 3.4(b) FAPT LADDER (Mark II) I/O device w w w .c Channel 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: i) Press the [R3] key on the menu screen for the [R] keys. 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 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 key. (4) Setting of the ROM writer The PMC-RA1, -RA2 or -RB uses one of 1MB EPROM (27C1024). The PMC-RC uses one of ROM module (128KB, 256KB, or 512KB). 619 3. COMPONENT UNITS AND CONNECTIONS PMC PROGRAMMER (SYSTEM P series) B–61863E/10 For this reason, when the PMC-RA1, -RA2 or -RB is used, both the FA Writer and PMC Writer can be used.

When the PMC-RC is used, only the FA Writer is available. When the PMC-RA1, -RA2 or -RB 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 key. 2 The following message appears. To select the FA Writer, enter 0 or press the key. To select the PMC Writer, enter 1.

om 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 ce nt e r.c The PMC Writer is required when the PMC-RA1, -RA2, -RB or RB2 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 w w w .c nc When the FA Writer is used with the PMC-RA1, -RA2 or -RB, 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. 620 OPERATION w w w .c nc ce nt e r.c om 4 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 621 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 4.1 Various operations of FAPT ladder are done onthe specified screen. GENERAL Fig. 4.1 shows the relation between various operations and corresponding screens. SYSTEM P series power on SYSTEM P Visual identification characters FAPT ladder system loading Key-in NL Key-in R0 REQUEST= Parameter screen Key-in NL Visual identification characters R key menu screen Key-in R1 Ladder diagram direct editing ce nt e Key-in R3 FAPT ladder system floppy loading r.c FAPT ladder R0, R1, R2, R3 om Power on Key-in R2 Unused 00 or NL key-in Menu screen 02 key-in 03 key-in 04 key-in 05 key-in 06 key-in 07 key-in 08 key-in 09 key-in 10 key-in Edit screen Source program input screen (PTR, FD, KB) ROM program input screen (FD, RAM cassette) Source program output screen (PTP, FD, PRT) ROM program output screen (FD, RAM cassette) Source program verify screen (PTR, FD) ROM program verify screen (FD, RAM cassette) Unused Clear screen E NL key-in when an alarm occurred or when processing was aborted halfway. w Automatic return when processing has been terminated normally. w Parameter setting screen .c w E NL key-in nc 01 key-in 4.1(a) Relation between various operations and screens 622 00 or NL key-in after parameter set-ting. 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 START SYSTEM P power on Load system floppy om Set necessary system parameters. r.c Sequence program input, title, symbol, comment, message I/O module ce nt e Editing Debug using PMC-RAM NO Is dedug completed? YES nc Write a sequence program into EPROM for PMC or ROM module w w w .c SYSTEM P series power off Operate the system after loading EPROM for PMC or ROM module to CNC END 4.1(b) Outline of operation 623 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 4.2 PREPARATION BEFORE OPERATION 4.2.1 System Floppy 4.2.2 To apply the FAPT LADDER system for PMC-RA1/RA2/RB/RB2/RC to the SYSTEM P Mate will overlay each of the following functions. D The function to display the ladder diagram on the screen and output it on an external printer, which is operated using the menu numbers 03 and F4 or 04 to F10. r.c Limitations with the SYSTEM P Mate om The system floppy disk contains the system of FAPT LADDER for PMC-RA1/RA2/RB/RB2/RC.

D 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. ce nt e D Ladder diagram direct editing, which is operated by pressing key on the R key menu screen and executing ladder diagram direct editing. 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: nc SET SYSTEM FD & KEY I ’OK’ OR ’NO’ FDD =OK ODRIVEJ (VOL =01) FDD = w 4.2.3 .c 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. w w 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 624 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 Drive No. 0 Drive No. 1 r.c om CRT 4.2.4 (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 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. w w w .c nc FAPT LADDER System Floppy Loading ce nt e 4.2.3 Loading direction of floppy 4.2.4 R key menu screen 625 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION 4.2.5 B–61863E/10 A programmer menu screen (hereinafter called menu screen) is displayed by pressing 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 key just after loading the 1/2 system floppy. Programmer Menu Screen Set parameters as required, referring to 4.2.6. Proceed to the menu screen by pressing key. om Parameters are displayable and settable from the menu screen, too. Programmer menu screen (The programmer menu and function keys are displayed.) r.c The following figure shows the screen to be displayed when the programmer key (R0) is pressed. ce nt e The programmer menu, function keys with I/O indication, and statuses are displayed on this screen. 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 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. w .c nc FD, PMC-RAM OR ROM. F1 : PTR (I) . F4 F2 : FD F5 : PRT (O) F6 : PTP (O) F7 : FD : GRP (O) (I) . F10 : FPRT (O) w w A period of the sequence program SYMBOL =00.0KB SCAN TIME=OOOMS LADDER =00.0KB MESSAGE=00.0KB ROM MODULE=A (O) : PMC (I/O) ERR SEQ.NO=00000 F9 : ROM (I/O) ERR BLOCK =00000 Capacity of a ladder program Capacity of message data The number of the last step in the sequence program END SEQ.NO=00000 F8 NO. Capacity of area used for symbols and comment data ALARM=00 = The number of a step at which an error occurred The number of an error occurred The number of blocks in which an error was detected Enter a menu number. 4.2.5 Programmer menu screen 626 4.

OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 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 ” from the menu screen. om 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. ITEMS CURRENT PARAMETERS 01 02 03 (UNUSED) COUNTER DATA TYPE OPERATOR PANEL KEY/LED ADDRESS KEY/LED BIT IMAGE ADRS. PMC TYPE LANGUAGE ORIGIN (UNUSED) LADDER EXEC. (UNUSED) IGNORE DIVIDE CODE (UNUSED) NOTHING TO SET r.c NO. BINARY NO / / PMC-RC 000000H ce nt e 04 05 06 07 08 09 10 00 ; ; ; ; ; ; ; ; ; ; ; ; 100% (1-150%) NO ; ROM WRITER=FA WRITER NO.= nc 4.2.6 Parameter setting screen (PMC-RC) Key in “00 ” 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. w w w .c 2 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 ” 2 Select a binary or BCD notation, and key in the corresponding number “@@”. 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 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 key. To enable the operator’s panel, enter 1 and press the key.

627 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION 4 B–61863E/10 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 key. 6 Entering data as shown above displays the following message: 7 om SET KEY/LED IMAGE ADDRESS (KEY ADRS, LED ADRS.) ADDR= Enter addresses other than X- and F-addresses. For example, enter R0,R10 and press the key. c) Select the type of PMC. r.c The initial value has been set to the PMC-RB or -RA1.

1 Type 04 and press the key. 2 The following message appears at the lower left of the screen: ce nt e EXAMPLE 0:PMC-RB, 1:PMC-RC PMC TYPE= or EXAMPLE 0:PMC-RA1, 1:PMC-RA2 PMC TYPE= To select the PMC-RB or -RA1, enter 0 and press the key. To select the PMC-RC or -RA2, enter 1 and press the key. 4 When the type of PMC is changed, all data items including ladder data are cleared. The following message is displayed for confirmation: w w w .c nc 3 5 CLEAR ALL DATA TO CHANGE PMC TYPE (0:NO, 1:YES) CLEAR/KEEP= To cancel changing the type of the PMC, enter 0 and press the key. To change the type of the PMC, enter 1 and press the key.

d) LANGUAGE ORIGN (for PMC-RC only) The initial value is already set to 0. 1 Enter 05 and press the key. 2 Enter @@@@@@ (hexadecimal) and press the key to specify the first address of the TCB in a C program. e) LADDER EXEC (only for PMC-RC) 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 ”.

2 Key in “@@@ ” by numeric characters within a range of 1% to 150%. After setting, key in “@@ ” or “” to set the menu screen. This parameter is not supported for PMC-RA1, PMC-RA2, PMC-RB or PMC-RB2. 628 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 f) IGNORE DIVIDE CODE (only for PMC-RB/RC) 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-RB and PMC-RC: Enter 09 and press the key. 2 To execute the ladder program by dividing it into smaller units, enter 0 and press the key. To execute it without dividing it, enter 1 and press the key. om 1 This parameter is not supported for PMC-RA1 or PMC-RA2. w w w .c nc ce nt e r.c The ladder program is always executed without being divided. 629 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 4.3 PROGRAM EDITING Data Display and Setting (Title, Symbol, Ladder Program, Comment, Message, I/O Module) Display the EDIT screen by keying in ”1 ” from the menu screen. Press 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 key is pressed. Individual screens are reset to the menu screen by “E ”. In this paragraph, only the input and editing operation of each data from the keyboard is described. om 4.3.1 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).

r.c Set the following data on sequence program as a comment. Key in ’1 NL ’ ce nt e Menu screen Ladder program list screen R0 R0 R0 Symbol & comment list screen nc Title data list screen EDIT screen I/O module data list screen R0 Message data list screen .c R0 w w w 4.3.1(a) Switching sequence of data display screen *** TITLE DATA LIST *** 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 PROGRAMMING 08 PROGRAM DESIGNED BY 09 ROM WRITTEN BY 10 REMARKS PMC CONTROL PROGRAM FANUC PMC-MODEL RB & F16MA SERIES : 4061 EDITION : 01 MEMORY USED : 00.0 KBYTE SCAN TIME : 008 MSEC 0003 ALTERED EDIT 4.3.1(b) Title data list screen 630 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 a) MACHINE TOOL BUILDER NAME Set the machine tool builder name (max. 32 characters). Key in “A1 @@@..........@@@ NL”. Machine tool builder name to be set Example) “A1 ***MACHINE(LTD) NL” b) MACHINE TOOL NAME om Set the machine tool name (max. 32 characters). Key in “A2 @@@..........@@@ NL”. Machine tool name to be set Example) “A2 ***MACHINE NL” r.c c) CNC & PMC NAME Set the CNC and PMC name (max. 32 characters).

Key in “A3 @@@..........@@@ NL”. ce nt e 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 “A4 0001 NL” nc Example) e) EDITION NO. Set the edition number (max. 2 characters). w w w .c Key in “A5 @@ 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 @@@...........@@@ 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 @@..........@@ NL”. Date to be set Example) 631 “A7 1990.10.23 NL” PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 h) PROGRAM DESIGNED BY Set the sequence program designer name (max. 32 characters). Key in “A8 @@@..........@@@ NL”. Name to be set Example) i) “A8 MR.***&MISS *** NL” ROM WRITTEN BY om Set the name of the programmer who wrote a program into ROM cassette (max. 32 characters). Key in “A9 @@@..........@@@ NL”. Name to be set j) REMARKS “A9 MR.***&MISS *** NL” r.c Example) Set remarks (memo) (max. 32 characters). ce nt e 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. nc (2) Symbol and comment data (SYMBOL & COMMENT LIST screen). w w w .c 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 “G0.1 SYMNAM COMMENT NL”.

Comment data Symbol name Address 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).

632 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 d) Delete i) Delete every line Key in “D@@@ NL”. Line number to be deleted ii) Sequential delete Key in “D@@@@,@@@ NL”. r.c e) Search i) Search by line number Key in “L@@@@ NL”. om Delete end line number Delete start line number Line number to be searched ce nt e ii) Search by symbol name Key in “L@@@@ SYMNAM NL”. Symbol name to be searched Line number with which the search is to be started iii) Address search Key in “L@@@@ F0.1 NL”. nc Address to be searched Line number with which the search is to be started .c (3) Message data (MESSAGE DATA LIST screen) Message data are alarm and operator message data to be displayed by using functional instruction DISPB (SUB 41). Message data w ***MESSAGE DATA LIST*** 0001 A00.0 0123456·········89XY 0002 ABCD······ 0003 0004 0005 0006 0007 0008 0009 A00.1 · · · EDIT = w w Line number Address 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. 633 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 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) om A means alter. b) Delete Delete message data every line in the following format. Key in “D@@@ NL”. r.c Line number to be deleted ce nt e 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. w w w .c nc ***I/O MODULE DATA LIST*** ADDRESS GROUP BASE SLOT NAME X000 X001 X002 X003 · · · · 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, ...) 634 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 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, ...) address group base slot name a. 2 0 1 FS08A X001 2 0 1 FS08A X002 2 0 1 FS08A X003 2 0 X004 2 0 X005 2 0 X006 2 0 X007 2 X009 FS08A 1 FS08A 1 FS08A 1 FS08A .c FS08A 1 2 1 8 ID16C 2 1 8 ID16C w X010 b. 1 w X008 If an attempt is made to set a module to X014 like c., alarm No. 88 occurs due to the reason in 1). nc X000 0 When two IO modules, b are set as shown in the following figure; ce nt e Example) r.c om NOTE 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. d. If an attempt is made to set an output module to X006 like d., alarm No. 87 occurs due to the reason in 2). If an attempt is made to set input module ID32B to X006 like d., alarm No. 81 occurs due to the reason in 3). In this case, this input module must be set after deleting a. and b. modules once. w X011 X012 X013 X014 X015 c. 2 1 8 ID16C 2 1 8 ID16C The module names (FS08A, CT01A, etc.) used for input and output in common are out of the objects of check in 1) and 2). 635 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 4.3.2 Input a sequence program from the keyboard. Programming from Keyboard Set the EDIT screen (LADDER PROGRAM LIST screen). Press menu number ”1 ” on the menu screen, or press key on the symbol or I/O module screen. Turn off all F keys at this time. Key in ”IS0 ” (Insert Succession) to set the sequential insert mode, and then, input a sequence program. (Key in sequence) om ”*IS MODE*” is displayed at the lower right part of the screen. key in desired instructions sequentially in the following format.

1 IS0 (Sequential input start command) 2 R X0.1 3 W R1.1 r.c ³ *IS MODE* is displayed at the lower right part of the screen.

4 IE (Sequential input end command) ce nt e ³ *IS MODE* display disappears from the lower right part of the screen. Usually do not program as shown below: w w .c nc 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 ”. 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. w Wrong program 636 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 Table 4.3.2 Keyboard input format and screen display format Input format from keyboard (Simple symbol) Display format on screen (Full name) R X0. 1 RN RD X0. 2 X0. 1 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 Y1.

1 OS AG2. 0 AN G2. 1 AS T 5 D F0 S 5 Y1. 0 OR.NOT Y1.1 OR. STK AND G2. 0 AND.NOT G2. 1 AND. STK TMR 5 DEC F0 SUB 5 (Parameter) ce nt e P 1234 OR om Y1. 0 r.c O ON 4.3.3 Correct a generated sequence program by alter operation. Alter 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 “A@@@@@ R X0.1 NL” .c nc Instruction to be altered Line number of the instruction to be altered (maximum 5 digits) A means alter. w w w 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 ” format. iii) After sequential alter, key in “AE ” (Alter End). Example) Example of sequential alter of sequence program For altering all step numbers 20 to 23; (Key in sequence) 1 AS20 (Sequential alter start command) ³ *AS MODE* is displayed at the lower right part of screen. 00020 RD Y0.1 2 R Y0.1 637 PMC PROGRAMMER (SYSTEM P series) 4.

OPERATION 00021 WRT R0.1 3 00022 RD F1.1 4 00023 WRT R1.1 5 6 B–61863E/10 W R0.1 R Y1.2 W R1.2 AE (Sequential alter end command) ³ *AS MODE* display disappears from the lower right part of the screen. c) Wiring change function om All of address used in Ladder Program is changed to a new address independently of a command. Only bit address can be changed. r.c Type in ’CA Address 1 Address 2 (Symbol can not be changed) A new address A previous address to be changed Abbreviation of CHANGE ALL ’CA R0.1 R1.2 ’ — All ”R0.1” used in Ladder Program is changed to ”R1.2”. ce nt e Example) w w w .c nc NOTE 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.

638 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 SUB1 00001 SUB1 00002 RD X0.1 00002 RD X1.7 00003 WRT R1.3 00003 WRT R1.2 00004 RD R1.2 00004 RD R1.2 00005 WRT R1.3 00005 WRT R1.3 00006 RD X0.1 00006 RD X1.7 00007 WRT R1.4 00007 WRT R1.4 00008 RD R1.3 00008 RD R1.2 00009 SUB 8 00009 SUB 8 00010 1111 00010 1111 00011 1111 00011 1111 00012 X0 00012 X0 00013 R2 00014 SUB 2 00015 SUB 48 00014 SUB 2 00015 SUB 48 r.c R2 This does not apply to byte address. ce nt e 00013 om 00001 ’CA X0.1 X1.7 NL ’ EDIT= _ EDIT= _ Normal end of changing. i. ii. SUB1 00001 SUB1 00001 SUB1 00002 RD X0.1 00002 RD X0.1 00002 RD R1.2 00003 WRT R1.5 00004 RD R1.2 00003 WRT R1.5 00003 WRT R1.5 00004 RD X0.1 00004 RD R1.2 00005 WRT R1.3 00005 WRT R1.3 00005 WRT R1.3 00006 RD X0.1 00006 RD X0.1 00006 RD R1.2 00007 WRT R1.2 00007 WRTR1.2 00007 WRT R1.2 00008 RD R1.2 00008 RD R1.2 00008 RD R1.2 SUB 8 00009 SUB 8 00009 SUB 8 1111 00010 1111 00010 1111 1111 00011 1111 00011 1111 00012 R1 00012 R1 00013 R2 00013 R2 00013 R2 .c nc 00001 iii. 00009 00010 w w 00011 SUB 2 00014 SUB 2 00014 SUB 2 00015 SUB 48 00015 SUB 48 00015 SUB 48 w 00014 EDIT= _ EDIT= _ EDIT= _ Abnormal end of changing. 1 ‘CA R1.2 X0.1 NL’ Alarm No. 09 Normal end of changing? 2 ‘CA X0.1 R1.2 NL’ 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. 639 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 Then, it may be impossible to return a ladder program ii to a ladder program i by an operation 2.

(Special use of wiring change function) All address used in ladder program of specified line number of subsequent, is changed a new address independently of a command. Only bit address can be changed. Operate carefully with enough recognition of above. om ’C@@@@@ Address 1 Address 2 NL’ (Symbol cannot be changed) A new address A previous address to be changed Change start line number ’C7 R0.1 R1.2 ’ r.c Example) ce nt e — All ”R0.1” used in ladder program of 7th line or subsequent, are changed to ”R1.2”. 4.3.4 Insert a new program to the generated sequence program. Insert Set the EDIT screen (LADDER PROGRAM LIST screen) first. a) Insert every instruction Key in “I@@@@@ R X0.1 NL”. Instruction to be inserted nc Line number just before the instruction to be inserted (maximum 5 digits) I means insert. w w w .c b) Sequential insert i) Key in ”IS@@@@@ ” (Insert Succession) to set the sequential insert mode, and *IS MODE* is displayed at the lower right part of the screen. @@@@@:Line number just before the instruction to be inserted (maximum 5 digits) ii) Instructions are sequentially inserted starting with the line next to the line specified by @@@@@, each time the key-in operation is done in the ”R X0.1 NL” format. iii) After sequential insert, key in ”IE ” (Insert End). Example) Sequential insert of a sequence program For inserting multiple instructions after step number 20 (Key in sequence) 1 IS20 (Sequential insert start command) ³ *IS MODE* is displayed at the lower right part of screen. 640 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 4.3.5 om 00020 RD Y200.0 2 R.S R200.1 00021 WRT R300.7 3 R.S R200.2 4 R 5 5 P 9 6 IE (Sequential insert end command) ³ *IS MODE* display disappears from the lower right part of the screen. i) Delete every instruction Delete Key in “D@@@@@ NL”. D means delete. ce nt e ii) Sequential delete r.c Line number to be deleted (maximum 5 digits) Key in “D@@@@@,@@@@@ NL”. Line number to complete delete (maximum 5 digits) Line number to start delete Search a sequence number by a line number or instruction Location Search i) Search by line number nc 4.3.6 w w w .c 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 NOTE 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. 641 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 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 om NOTE 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. r.c 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. ce nt e Type in ’L@@@@ ????? NL’.

Bit address or its symbol name Search start line number ’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. nc Example) v) Continuous search .c A specified command, address (only bit address) or its symbol name is searched from 2nd line displayed on the screen. w Command, bit address or its symbol name to be searched Type in ’F NL’. w w Type in ’L ????? NL’. FIND: Example) Search the same command, bit address or its symbol name as that searched just before, from 2nd line displayed on the screen. ’L R R0.1 ’ — Search the command ”RD R0.1” from 2nd line displayed on the screen. ’L R0.1 ’ — Search the bit address ”R0.1” from 2nd line displayed on the screen. ’L SYMBOL ’ — Search the symbol name ”SYMBOL” defined at bit address from 2nd line displayed on the screen. 642 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 4.3.7 The ladder diagram can be displayed on the programmer function EDIT screen. Display of Ladder Diagram Set the screen to EDIT screen (LADDER PROGRAM LIST) a) Turn on F4 key. b) Depress key The ladder diagram is displayed on the screen. om For displaying the sequence program in the mnemonic format from the ladder diagram, turn off F4 key, and depress key. The ladder diagram at an optional point can be displayed by the step number search or instruction search method. c) r.c 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. Edition during LADDER diagram display ce nt e 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. Key ² 4 NL Right (Right ladder on screen is displayed.) 6 ³ NL Upper (Upper ladder on the screen is displayed.) Lower .c Left nc (Left ladder on screen is displayed.) w Scroll direction (Lower ladder on the screen is displayed.) w Upper half page w Lower half page ° 8 NL 2 ± NL R2 NL *** LADDER DIAGRAM *** X0.0 X0.1 Y 0.0 X1.0 X1.1 Y 0.1 TMR X2.0 X2.1 EDIT= R2 NL 4.3.7 Ladder diagram display screen 643 1 R 40.7 PMC PROGRAMMER (SYSTEM P series) 4.

OPERATION B–61863E/10 Example) LADDER DIAGRAM * From ’EDIT=’ IN the left figure, key in as follows.

’ISO NL’ ’R X1.0 NL’ ’W Y1.0 NL’ ’IE NL’ 2 Press R0 key to display the ’LADDER PROGRAM LIST’ screen.

Two lines ’RDX1.0’ AND ’WRT Y1.0’ are added before ’SUB1 (END1)’. 3 When the LADDER diagram is displayed again, the diagram after adding the above two lines is displayed. END1 Y0.0 X0.0 END2 ~ ~ om 1 ce nt e EDIT= r.c * 4.3.8 Editing command explanation screen can be displayed from Ladder program edition screen.

(LADDER PROGRAM LIST or LADDER DIAGRAM) Help Screen nc key in ’H ’ to display the following screen.

*** HELP LIST *** w w w .c I@@@@@ OPERATION CODE IS@@@@@ IE A@@@@@ OPERATION CODE AS@@@@@ AE D@@@@(,@@@@@) L@@@@@ L@@@@@ OPERATION CODE L@@@@@ L OPERATION CODE L F CA ADDRESS1 ADDRESS2 (@@@@@=SEQUENCE NO.) : INSERT : INSERT SUCCESSION START : INSERT SUCCESSION END : ALTER : ALTER SUCCESSION START : ALTER SUCCESSION END : DELETE (SUCCESSION END) : : : : : : SEQUENCE NO. SEARCH OPERATION CODE SEARCH ADDRESS OR SYMBOL SEARCH SEARCH FROM DISPLAY 2ND LINE SEARCH FROM DISPLAY 2ND LINE FIND FROM DISPLAY 2ND LINE :CHANGE ALL ADDRESS1 TO ADDRESS2 Key in ”” to return it to ladder program editing screen. 4.3.9 Editing end Key is ”E ” (End) after editing a sequence program, and the EDIT screen is reset to the menu screen. 644 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 4.4 INPUT OF PROGRAM 4.4.1 Source Program % @0 ±±±±± CR % @1 CR ce nt e b) Title date Feed % CR Feed r.c Feed om 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) Paper tape format of source programs Paper tape format of source programs is of ISO code. No EIA code paper tape can be used. a) Parameter date 1 MACHINE TOOL ±±±±± CR % CR Feed c) Symbol date % @2 CR F0.1 SYMBOL COMMENT CR ±±±± % CR Feed nc Feed % @3 CR RD w Feed .c d) Ladder program w w ~ ±± WRT WORK01 CR ±± % CR Feed e) Message date Feed % @4 CR WRT WORK01 CR CR X0.1 CR A0.0 MESSAGE = 1 CR ±± A24.7 MESSAGE = 24 CR % Feed f) I/O module date Feed % @5 CR X 0 0 0 5 ID32C CR 645 ±±±±± % CR Feed PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 (2) Input method from PPR reader 1 Turn on F1 key.

Key in menu number ”2 ”.

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.

om 2 Key in menu number ”2 ”. 3 The following message is displayed at the lower part of the screen. r.c 2 ce nt e SET FD & KEY IN ”OK” ”KILL” OR ”NO” FD0=OK 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. Example Key in file names to be input from floppy as shown in the example. w w w .c nc 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 SCAN TIME-OO8MS F2 : FD (I) . F10 : EPRT (O) LADDER =00.0KB F5 : PRT (O) MESSAGE=00.0KB F6 : PTP (O) F7 : FD (O) END SEQ.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’ NO.= 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. 646 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 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. 4.4.2 (1) Transfer of sequence program from the PMC-RA1/RA2/RB/RC The created sequence program is transferred from the PMC-RA1/RA2/RB/RC. 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. w w w .c nc ROM Format Program ce nt e r.c om 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. 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. 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-RA1/RA2/RB/RC 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.) 647 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 ce nt e r.c om 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-RA1/RA2/RB/RC 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 FD0=OK FD0= 4 Insert the floppy into the disk and enter the following data 5 'OK @LADDER2 [NL] nc File name 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 ' is keyed in, the message shown below will be displayed. PMC-RA1/RA2/RB/RC 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.= w w w .c 6 648 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 5 The screen is switched and the message shown below is displayed. SET EPROM OR ROM MODULE & KEY IN 'OK' OR 'NO' KEY IN= Check the above message. For the PMC-RA1/RA2/RB, 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 ' or 'NO '. When 'OK ' 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 ' is keyed in, the screen returns to the menu screen. ce nt e r.c om 6 w w w .c nc NOTE 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.

649 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 4.5 OUTPUT OF PROGRAM 4.5.1 By selecting '04 ' (OUTPUT LADDER PROGRAM) from menu no.4, the following detail menu is displayed. Source Program om SET I/O KEY & KEY IN ONE OF THE FOLLOWING NO.S WHICH YOU WANT. ce nt e r.c NO. ITEMS 01 OUTPUT ALL DATA. 02 OUTPUT SYSTEM PARAMETER. 03 OUTPUT TITLE DATA 04 OUTPUT SYMBOL DATA. 05 OUTPUT MESSAGE DATA. 06 OUTPUT I/O MODUL DATA. 07 OUTPUT LADDER PROGRAM (MNEMONIC). 08 OUTPUT LADDER DIAGRAM (ONLY FANUC PRINTER). 09 OUTPUT CROSS REFERENCE (SEQUENCE NO.) 00 END F5 : PRT (O) , F10 : FANUC PRINTER (O) F6 : PTP (O) , F13 : CROSS REFERENCE (NO.8) F7 : FD (O) NO.

= Select a desired data and device from the above details menu screen by combining the menu numbers and F keys. w w w .c nc (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 '. 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 '. (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 '.

(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 '. The screen is switched and the following display appears. OUTPUT = 'L@@@@ (,@@@@)' OUTPUT =_ Specify the output range by line numbers as follows. 650 4.

OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 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 r.c om (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 '. 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' ce nt e Output end address (If this parameter is omitted, data are output to the last one.) Output start address w w w .c nc (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 '. (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 '. 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 ', 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.) 651 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 Example) Key in 'L1,100 ' Output end line number (If this parameter is omitted, data are output to the last one.) Output start line number * * * * * ALL ADDRESS HEAD CHARACTER ADDRESS 'G14.6' ADDRESS TO ADDRESS ADDRESS TO END 'ALL' 'G' 'G14.6,R142.5' 'G14.6-END' * END ; PUSH 'NL' KEY ce nt e r.c ADDR= om ITEMS .c nc NOTE D 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. D If R1 key is pressed when each data is being output to the FANUC printer (External printer), the output is cancelled. w w w (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 ' 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 All addresses (G,F,Y,X,A,R,T,K,C,D in order) Address initial R All address with the specified initial Bit address X1.0 Only bit address specified address Byte address R58 Bit 0 - 7 of specified Address range specification F8.0, X7.2 Specified addresses in order of G,F,Y,X,A,R,T,K, C,D X0.2-END All address after specified address 652 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 *** CROSS REFERENCE LIST *** PAGE=1 COMMENT DATA *IT *CST *ESP 568 *SP 2802 ERS 3512 *AIT 901 912 1177 1189 1288 2800 r.c G0000.0 653 G0000.1 653 G0000.4 22 G0000.5 45 G0000.7 3435 G0001.0 656 SYMBOL om ADDRESS ce nt e 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 A 12-inch chart is also applicable to the FANUC printer (external printer).

Paper Command (The standard chart size is 11 inches.) Enter the command for changing the chart by the following operation. nc (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 '. KEY IN NUMBER OF PAPER LENGTH EXAMPLE 11-INCH;0,12-INCH;1.

LINE NUM.= (5) Key in '0 ' for 11-inch chart, or '1 ' for 12-inch chart. w w w .c (4) The following message is displayed at the lower left part of the screen. 653 PMC PROGRAMMER (SYSTEM P series) 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. 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]. ce nt e ROM Format Program (1) Transfer of sequence program into PMC-RA1/RA2/RB/RC A generated sequence program is transferred into PMC-RA1/RA2/RB/RC. 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. 1 Pressing soft keys 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. om 4.5.3 B–61863E/10 r.c 4. OPERATION 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.) 8 Key in menu number "5 ". nc 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. i) When an alarm 31 occurs on SYSTEM P series screen; Cause 1 : Reader/Puncher interface cable is defective. Remedy : Use the specified cable. Cause 2 : Reader/Puncher interface connector is not connected to correct channel SYSTEM P series. Remedy : Connect the connector correctly. 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; Cause 1 : Reader/Puncher interface connector is not connected to CNC. Remedy : Connect it correctly.

Cause 2 : CNC screen is not set to "I/O of PMC" screen. Remedy : Set the I/O screen by the soft key. Cause 3 : An error occurs in ACI channel due to a certain cause. Remedy : Turn off the power supply once, and turn it on again.

(2) Output method to floppy 1 Turn on F7 key. (Also turn on F12 key when the C language program is included.) w w w .c *Procedure when a program cannot be transferred from SYSTEM P series to RAM of PMC 654 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 2 Set the floppy to the disk. 3 Key in menu number "5 ". 4 The screen is switched and the following message is displayed: SET FD & KEY IN "OK","KILL" OR "NO". FD0= OK @NAME FD0 = om 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 ". (3) Method of outputting data to FA writer or PMC writer (EPROM for PMC/ROM module write) 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. ce nt e r.c 1 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 = .c nc Check the above message. For the PMC-RA1/RA2/RB, 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. Key in "OK " or "NO ". When "OK " 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. w w w 6 655 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 *** TRANSFER ROM DATA TO PMC WRITER *** PMC CONTROL PROGRAM SERIES : 4061 EDITION :01 MEMORY USED : 00.0 KBYT SCAN TIME : 008 MSEC ce nt e SET EPROM OR ROM MODULE & KEY IN 'OK' OR 'NO' KEY IN = om MACHINE TOOL BUILDER NAME MACHINE TOOL NAME PMC & NC NAME PMC PROGRAM NO EDITION NO PROGRAM DRAWING NO DATE OF PROGRAMING PROGRAM DESIGNED BY ROM WRITTEN BY REMARKS r.c 01 02 03 04 05 06 07 08 09 10 *** TRANSFER ROM DATA TO PMC WRITER *** 01 02 03 04 05 06 07 08 09 10 MACHINE TOOL BUILDER NAME MACHINE TOOL NAME PMC & NC NAME PMC PROGRAM NO EDITION NO PROGRAM DRAWING NO DATE OF PROGRAMING PROGRAM DESIGNED BY ROM WRITTEN BY REMARKS nc PMC CONTROL PROGRAM SERIES : 4061 EDITION :01 MEMORY USED : 00.0 KBYT SCAN TIME : 008 MSEC ROMH=EF MEM FF .c ROML=10 MODE=BLANK AD=000000 ALARM=083 Display mode BLANK : Blank check PROGRAM : Write VERIFY : Compare Error number w OUTPUT= w w Enter 'E NL', and restart from menu. 656 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 4.6 COLLATION OF PROGRAM 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. (1) Comparison with PTR 1 Turn on F1 key. 2 Key in menu number "6 NL". r.c (2) Comparison with FD 1 Turn on F2 key. om 4.6.1 2 Key in menu number "6 ".

3 The screen is switched, and the following message is displayed.

4.6.2 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. nc ROM Format Program ce nt e SET FD & KEY IN "OK", "KILL" OR "NO", FD0=OK FD0= 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 (1) Comparison with FD 1 Turn on F2 key. Key in menu number "7 ". 3 The following operation is the same as in 4.6.1 2)– 3 and later. w w w .c 2 (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 ". 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 ".

The screen is switched, and the comparison of ROM program is started. After normal end, the screen is automatically reset to the menu screen. 657 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION 4.7 B–61863E/10 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 ". 3 The screen is switched, and the following message is displayed at the lower left part of the screen. See Fig. 4.7. om 4 KEY IN "1,2,3,4 OR 5" OR "NO" CLEAR/KEEP = Key in data number of the data to be deleted or key in "NO ", if it is not desired to delete any data. After processing, the screen is automatically reset to the programmer menu screen. r.c DELETION OF PROGRAMS KEY IN ONE OF THE FOLLOWING NO.S WHICH YOU WANT TO CLEAR DATA ITEMS TITLE DATA SYMBOL DATA LADDER DATA MESSAGE DATA I/O MODULE DATA ALL DATA CLEAR ce nt e NO. 01 02 03 04 05 06 nc KEY IN '1. 2. 3. 4. 5 OR 6 OR 'NO' CLEAR/KEEP = 4.7 Deletion of sequence programs w w w .c Example) i) When all title data are to be deleted; Key in "1 ". ii) When all symbol data are to be deleted; Key in "2 ". iii) When all ladder programs are to be deleted; Key in "3 ". iv) When all message data are to be deleted; Key in "4 ".

v) When I/O module data are to be deleted; Key in "5 ". vi) When all titles, symbols, ladders, messages and I/O module data are to be deleted; Key in "6 ". vii) When no data are to be deleted; Key in "NO ". 658 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 4.8 SPECIAL USES OF THE R3 KEY Key in 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 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 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 . 4 The file name will appear on the CRT display. To print out the file name, turn on the F5 (printer) key in advance. r.c om 1 (2) Change of I/O devices (for output to a printer other than that of PPR) Key in IO PRT, CN3, F5 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.

w w w .c nc ce nt e 1 659 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 4.9 DIRECT EDITING BY LADDER DIAGRAM 4.9.1 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. Outline om 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 r.c is displayed. (In systems where [UNUSED] is displayed, it cannot be used.) The following items are present in the edit function. ce nt e D Ladder diagram direct editing by software key and cursor (input, addition, deletion and substitution) D Copying, moving and deletion of multiple lines of the ladder D Optional relay and coil reference D Comment display on ladder diagram 4.9.2 nc Limitations in SYSTEM P Mate (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.) w 4.9.3 w .c (2) The function 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. w Selection of Program Menu by Soft Keys The program menu appears in order to operate this function. The program menu is displayed when the 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 between 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.

660 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 R0 R1 R2 R3 : : : : PROGRAMMER EDIT UNUSED REQUEST Press the key or FUNCTN om R keys menu screen or COMAND [COMAND] 1 2 [DELNET] EXEC 2 INSERT ADRESS SEARCH CANCEL SEARCH C-DOWN [INSERT] INSLIN 5 C-UP INSELM SYMBOL SEARCH COPY MOVE w INSERT w [SEARCH] w TOP 5 MOVE [ADRESS] INSNET DELNET 4 4 .c INSNET 3 3 COPY nc 1 DELNET ce nt e INSNET r.c [COMAND] BOTTOM SRCH W-SRCH N-SRCH F-SRCH C-DOWN [COPY/MOVE] UNTIL CANCEL SEARCH C-DOWN C-UP TO CANCEL SEARCH C-DOWN C-UP 661 C-UP PMC PROGRAMMER (SYSTEM P series) 4. OPERATION 4.9.4 Sequence Program Input B–61863E/10 In order to input the sequence program, press the 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. (1) Basic instruction program input R10.2 R20.2 ce nt e X2.4 R1.7 r.c R0.1 om The following description shows an example of the input of a program of basic instruction and a program of functional instruction. 1 Press the soft key [position.] after moving the cursor to the start nc 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. w w w .c 2 Press the 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 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 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 key. The address is set on the B contact and the cursor shifts rightward. 662 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 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. ce nt e r.c om 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. .c nc 1 Case of multiple nets on 1 LINE Net is repeated Downward from the net is erased A B C w w w 2 Case of multiple WRT results in 1 NET difference as shown in the diagram below. Section C is erased 663 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 NOTE 3 Case of exceeding the highest rank WRT in 1 NET B A C om Section B is erased (2) Case of functional instruction program input nc ce nt e r.c 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. MOVE ACT (1) (2) (3) (4) .c (SUB 8) w w w Control condition Output address Input address Low rank 4 bit logic data High rank 4 bit logic data When inputting a functional instruction with this function, the functional instruction parameters are input vertically as shown in the diagram below. MOVE (1) (SUB 8) (2) (3) (4) 664 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 1 Input a control condition.], input the address and bit data, and then Press soft key [press the key. The cursor shifts rightward. 2 Input an instruction Press the soft key [FUNCTN], input SUB number 8, and then press the key. A functional instruction diagram appears as shown in the above figure. 3 Input an instruction parameter om Input the high rank 4 bit logic data of the first parameter, and then press the key. The cursor automatically lowers downwards. Input the three residual parameters in order. 4.9.5 r.c The method of substituting a created sequence program is the same as that described earlier in Section 4.9.4. Substitution of Sequence Programs 4.9.6 Additions to Sequence Programs ce nt e Move the cursor to the program part you want to alter and input the change data. From the soft key program menu, press the soft key [COMAND] and operate with the soft keys shown below.

When you want to end the program menu shown below, press the soft key at the extreme left.

DELNET INSERT ADRESS SEARCH INSNET INSLIN INSELM COPY MOVE 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 w w .c nc INSNET w When the addition is horizontal Case of adding Move the cursor to the position where you want to add, and input te program by the method described in Section 4.9.4. 665 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 When a vertical line influences the addition ÕÕÕ ÕÕÕ om Case of adding Cursor Move the cursor to the above position. 2 Press the soft key [ ] in order to erase the upper left vertical line. r.c 1 The upper left line, vertical to the cursor disappears. 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.

ce nt e 3 4 Shift the cursor to a line of contact addition position. 5 Press the soft key [ ] to add contacts. (2) Adding a vertical line nc 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]. 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. w w w .c 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. Addition 1 Move the cursor to the ladder diagram bounded by a dotted line. 2 Press the soft key [INSNET]. 666 4. OPERATION PMC PROGRAMMER (SYSTEM P series) 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 key. The cursor shifts rightward. 6 Press the shift key [ ] to create an OR circuit. om B–61863E/10 (3) Inserting the 1 NET sequence program LINE. Space lines are inserted in units of 1 LINE.

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.) r.c 1 ÕÕÕ ÕÕÕ ce nt e a Crusor .c nc b ÕÕÕ ÕÕÕ If the [INSLIN] key is pressed with the cursor in the above position, the state shown in the diagram on the right will occur. w w w (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.) 667 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 ÕÕÕ ÕÕÕ a Cursor ÕÕÕ ÕÕÕ om b 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. r.c ÕÕÕ ÕÕÕ ce nt e a Cursor b ÕÕÕ ÕÕÕ nc 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.

(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. .c 4.9.7 w w Deleting a Sequence Program EXEC DELNET [ ] : Deletion of upper left vertical line to the cursor [: Deletion of upper right vertical line to the cursor] (2) For the deletion of a program net (part corresponding to the section from RD instruction to WRT instruction), use the [DELNET] key. (3) Deleting multiple NETs in NET units w INSNET [– – – –] : Deletion of horizontal lines, relay contacts coils, etc. INSERT ADRESS SEARCH CANCEL SEARCH C-DOWN C-UP 668 COPY MOVE 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 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 3 om 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. Execution . . .

. .

. Press the [EXEC] key Cancellation . . . . Press the [CANCEL] key 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. r.c 4 Search a sequence program by using the following soft keys. Searching a Sequence Program (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. w w w .c nc ce nt e 4.9.8 669 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 DELNET INSERT TOP BOTTOM SRCH X2.0 W-SRCH N-SRCH ÕÕÕ ÕÕÕ R20.1 Y1.2 .c R0.5 ADRESS SEARCH nc INSNET ce nt e r.c om 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. R5.0 COPY MOVE F-SRCH C-DOWN R6.4 R20.2 X4.2 Y2.0 R20.1 R2.2 w R21.0 w ÕÕÕ ÕÕÕ C-UP When you want to search the same address as specified here. Set the cursor to this position and press the soft key [SRCH]. w R0.4 R0.5 R10.5 The same address is searched and the cursor shifts to this position. (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. 670 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 4.9.9 Copying a Sequence Program ce nt e r.c om 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 (,) D Key in the address or symbol and press the cursor to start searching the NET No. D Key in the NET NO. and press the cursor key to start searchng the NET NO. D Key in the functional instruction name or functional instruction number starting with "S" and press the cursor key to start searching the functional instruction. Example) Key in "END1" or "S1" and press the cursor to search functional instruction END1. 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 cursor. When copying, the number of copies can also be specified. Copying nc 1 .c 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). w w w 2 3 Copying 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 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 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 specified number of times. 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. 671 PMC PROGRAMMER (SYSTEM P series) INSNET DELNET INSERT ADRESS SEARCH UNTIL CANCEL SEARCH C-DOWN C-UP TO CANCEL SEARCH C-DOWN C-UP COPY B–61863E/10 MOVE 4.9.10 Moving a Sequence Program ce nt e r.c NOTE An error NET cannot be copied.

om 4. OPERATION A sequence program with multiple NETS is moved in units of NETs. Specify the NET to be moved and specify the move position with the cursor. When moving, the number of moves can also be specified. 1 Moving nc 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.) w w w .c 2 3 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. 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. 672 4. OPERATION PMC PROGRAMMER (SYSTEM P series) INSNET DELNET INSERT ADRESS SEARCH UNTIL CANCEL SEARCH C-DOWN C-UP TO CANCEL SEARCH C-DOWN C-UP COPY MOVE 4.9.11 ce nt e r.c NOTE An error NET cannot be copied. om B–61863E/10 (1) Symbol and comment data display Symbol Data Display Symbol data and comment are displayed together with a ladder diagram on the screen as follows. nc 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]. 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. w w w .c (2) Symbol input and search in the sequence program 673 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 Signal name (within 6 characters) MA SPDALM X2.4 R2.2 Y4.3 ATCALM R100.1 MRDY MACHINE READY APCALM R5.4 Comment (within 30 characters) MALM om MACHINE ALARM 10 lines R120.1 TIND D20.7 R52.1 ce nt e r.c APC An address is displayed if a symbol is not defined. nc Signal name (within 6 characters) 4.9.12 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. w w w .c Compressed Input by [COMAND] Key The main function of each soft key can be directly selected from the [COMAND] 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. 674 4. OPERATION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 4.9.13 om NOTE The software keys [or] and [or] 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. Ending Edit of a Sequence Program or ce nt e FUNCTN r.c In the program menu shown below, press the extreme left software key. or COMAND w w w .c nc NOTE When an error NET exists, ERROR NET NO. is displayed and you cannot end the edit.

End after correcting the erroneous NET. 675 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION B–61863E/10 4.10 INPUT/OUTPUT OF LADDER PROGRAM WITH P–G AND FLOPPY CASSETTE/FA CARD 4.10.1 om 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 P–G into PMC RAM. The usable adapters are as follows: r.c General 4.10.2 When using the FANUC floppy cassette adapter/FA card adapter, change the settings of the input/output devices by the following 'IO commands'.

1 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. nc Setting I/O Commands ce nt e D FANUC cassette adapter 3 (A13B–0131–B001)/cassette F1 (A87L–0001–0038) D FANUC floppy cassette adapter (A13B–0150–B001)/floppy cassette (A87L–0001–0039) D FA card adapter (A13B–0148–B001)/FA card (A87B–0001–0108) w 4.10.3 .c 3 1 Turn on F13 key. (Turn on F12 too, when C language program is included.) 2 Key in the menu No. '3 [NL]'. 3 (For PMC–RC 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. = w w Program Input To return the assignment to channel 2 to PMC WRITER, key in 'IO AUX, CN2, F9 [NL]'. 4 The message is displayed lower left on the screen. SET BC & KEY IN 'OK' OR 'NO' BC = OK BC = 5 Set the floppy cassette/FA card in the adapter, and enter the following data. 676 4. OPERATION PMC PROGRAMMER (SYSTEM P series) 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. om B–61863E/10 ce nt e r.c NOTE 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 1 Program Output Turn on F14 key. (Turn on F12 too, when C language program is included.) Key in the menu No. '5 [NL]'. 3 The message is displayed lower left on the screen. SET BC & KEY IN 'OK' OR 'NO' BC = OK BC = nc 2 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.). .c 4 w w w 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. 677 PMC PROGRAMMER (SYSTEM P series) 4. OPERATION 4.10.5 1 B–61863E/10 Turn on F13 key. Program Collation (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'. w w w .c nc ce nt e r.c om NOTE For the program which is output from PMC–RA1/RA2/RB/RC 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. 678 PMC PROGRAMMER (SYSTEM P series) B–61863E/10 FILE EDITING FUNCTION w w w .c nc ce nt e r.c om 5 5. FILE EDITING FUNCTION 679 5. FILE EDITING FUNCTION 5.1 PMC PROGRAMMER (SYSTEM P series) B–61863E/10 This function edits floppy disk data in the unit of file. When key in only 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. GENERAL The file attributes are as shown below. (1) File number r.c (2) File name om NOTE The format for file designation is as follows: [drive No.] @ file name : file No. (3) File creation date (4) Identification of protection file (protect) ce nt e (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. File editing command table Name of command (Instruction) nc Contents of jobs FDLIST Change of file name, date, etc. RENAME Deletion of file SCRATCH w w w .c Display of file name, or file size File area condensation CONDENSE Copy of file (This command is effective for SYSTEM P series with 2–floppy disk unit.) REMOVE Contents inputted from keyboard (NL key is inputted at the end of a command) FDLI [D, [P,] [S,] [F,] [L,] File designation RENA file designation [, [,@ new file name] SCRA file designation COND [drive No.] M ,] [file designation] A [, INT] [, P] [,/date] ADD NP [,@ new. file name] REMO [Set FD, and key in 'OK', KILL, or 'NO'. FD=OK FD=_ Set the floppy disk and key in as follows. 680 P] [,/date] NP PMC PROGRAMMER (SYSTEM P series) B–61863E/10 OK INT. P , ADD NP 5. FILE EDITING FUNCTION [/date] [drive] [@ file name] NO KILL Set FD, and key in 'OK', KILL, or 'NO'. FD=OK om When reading, the following floppy set request message is displayed.

r.c FD=_ ce nt e Set floppy disk and key in as follows. OK [drive number] @ file name : file number NO KILL nc 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. .c OK . . . After instructing execution of read and write, specify parameter. NO . . . Cancel read/write to floppy only. Parameters instructing details of read and write is as follows. w w w KILL . . Cancel the specified process. 681 5. FILE EDITING FUNCTION PMC PROGRAMMER (SYSTEM P series) Function Notes INT When writing, write from the head of the floppy ADD When writing, add after exising fie When omitted, it is regarded as ADD. If INT is specified to protection file, file an error generaters. generaters P Prepare as protection file NP Prepare as ordinary file 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. When omitted, it is regarded as NP.

Ready files can be changed by RENAME command. r.c om Parameter B–61863E/10 ce nt e NOTE When specifying drive number and file name or file number, specify without separating, as follows.

Example) 0 @ ABC or 1 : 5 w w w .c nc 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). 682 PMC PROGRAMMER (SYSTEM P series) B–61863E/10 5.2 5. FILE EDITING FUNCTION (1) General form of command CONFIGURATION OF COMMAND Operation Command name or its abbreviated form (4 leading characters) Space Operand _ 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. om 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. w w w .c nc ce nt e r.c 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.) 683 5. FILE EDITING FUNCTION 5.3 FDLIST COMMAND — FILE ATTRIBUTE DISPLAY PMC PROGRAMMER (SYSTEM P series) B–61863E/10 This command displays the attributes of files in the floppy disk, such as file name, file size, etc. a) Input format [D,] [P,] [S,] [F,] FDLIST [L,] om @ file name [Drive No.] : file No. b) Operand Display of file creation date consisting of 6 characters Identification display of protection files Display of file size Display of size of unoccupied area Executes all display by parameters D, P, S, F. r.c D: P: S: F: L: Displays a file having the designated file name or designated : file number only. If this designation is omitted, all files are treated as processing objects. ce nt e @ file name c) file No. This command displays the information (attributes) on the floppy disk files. w w w .c nc 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 NAME V. DATE SIZE 001 DATA1 830928 72 002 DATA2 831028 60 003 DATA3 831028 8 **** DELETED FILE **** 10 005 DATA4 901022 10 006 DATA5 901022 5 FILE USED AREA = 155 DELETED FILE AREA = 10 FREE AREA = 1019 P. P P P NOTE The numeric characters shown in SIZE, FILE USED AREA, DELETED FILE AREA, and FREE AREA are displayed assuming that 256 characters are 1. 684 5. FILE EDITING FUNCTION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 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 RENAME [drive No.] @ file name : file No. Designation of file to be changed P NP [,/date] [@ new file name] Designation of file to be changed om b) Function r.c 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.

ce nt e 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 , for example) Old attributes (B) and new attributes (A) are displayed by executing this command as shown below, for example. RENAME :3, @ NEWNAME NL> RENA : 3,@NEWNAME NO. FILE NAME V. DATE P. B: 003 DATA3 901020 A: 003 NEWNAME 901020 w w w .c nc Example) 685 5. FILE EDITING FUNCTION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 5.5 This command deletes files of floppy disk. SCRATCH COMMAND — DELETION OF FILES a) Input format @ file name : file No. SCRATCH [drive No.] b) Function om 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 This command releases the deleted file area to be employable. CONDENSE COMMAND — RELEASE OF DELETED AREA a) r.c Input format CONDENSE [drive No.] b) Function w w w .c nc ce nt e 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. 686 5. FILE EDITING FUNCTION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 5.7 REMOVE COMMAND — FILE COPY REMOVE M A, This command copies files to another floppy disk by using two floppy disk units. a) Input format input drive No. @ file name file No. INT ADD , P , [/date] NP , @ new file name Output designation Input designation om Designation of copying method b) Operand No. M,A designation: M (manual): r.c Specified file is copied with specified file attributes. Copies designated files one by one. ce nt e Floppy disk mounting request message is displayed every file to ask if the file is to be copied or not. If input file is not designated, all files becomes the object to be copied. If new file name is not designated, input file name is produced. Accordingly, the output designation of REMOVE command is meaningless, because the request for output designation is performed every file. When a file name is designated as an input; All coincident files having the designated length are treated as processing objects. If @A is desig–nated, for example, all files starting with A are asked.

ii) When a file number is designated as an input; Files from a file having the designated file number to the last file are treated as processing objects. A (auto): w w w .c nc i) All files conforming to the designated conditions are copied. The output designation file name is meaningless. However, P, NP and / date are designated to all copied files with new attributes. i) If the file name or file number is not designated by input designation, all files of the input floppy disk are copied. ii) If the file name or file number is designated in input designation, processing is done in the same way as in M designation. c) Function This command copies floppy disk files to another floppy disk. Files are output to a floppy disk opposite to the drive number (0, if omitted) designated by input designation.

These files are copied by the following three methods. i) Copy of one file only (Neither M nor A is designated.) (Ex. 1) REMOVE : 3,P, @ NEW In this example, the input/output designations are as follows; Input designation : 3rd file of drive No.0 floppy disk 687 5. FILE EDITING FUNCTION PMC PROGRAMMER (SYSTEM P series) B–61863E/10 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 In this example, the system asks to copy or not every file with file name starting with "A" of drive No.1 floppy disk. om 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 r.c 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. ce nt e 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 '. nc NO. FILE NAME I:001 T01 ZX 1. O:001 T01 ZX 1. I:002 T02 ZX 1. O:002 T02 ZX 1. I:003 T04 ZX 1. O:003 T04 ZX 1. I:004 T05 ZX 1. O:004 T05 ZX 1. 100/40 100/40 150/50 150/50 100/50 100/50 20/50 20/50 V.DATE P. 830920 830920 P 830920 830920 P 830920 830920 P 830920 830920 P w w w .c 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.) 0 @ file name REMOVE INT , 1 : file No. P , ADD NP [,/ date] [, @ new file name] ii) With M designation (request message is displayed for each objected file.) 0 @ file name 1 : file No REMOVE M, iii) With A designation (All subjected files are copied.) 0 @ file name REMOVE 1 : file No. [,/ date] 688 INT , P , ADD NP r.c ce nt e nc .c w w w om APPENDIX A A. ERROR CODES LIST (FOR FAPT LADDER P–G) APPENDIX B–61863E/10 ERROR CODES LIST (FOR FAPT LADDER P–G) Error codes Details of errors 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.

31 Input/output unit error 32 Read error END1 is not specified. A data sent from PMC–RB/RC is in error. R1 key is pressed during data transmission between SYSTEM P series and PMC–RB/RC. w w 29 r.c ce nt e .c nc 30 27 om 01 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 w 33 691 A. ERROR CODES LIST (FOR FAPT LADDER P–G) APPENDIX Error codes B–61863E/10 Details of errors Symbol name (max. 6 characters) over 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 79 r.c ce nt e nc Slot number error I/O module name error I/O port data are not prepared yet. w 80 .c 78 om 51 I/O port data were doubly specified. 82 Specified symbol or address is missing. w 81 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. w 83 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–RA1 and PMC–RA2.) 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. 692 A. ERROR CODES LIST (FOR FAPT LADDER P–G) APPENDIX B–61863E/10 Details of errors 153 Blank check error (Ultraviolet ray is not sufficiently irradiated or the EPROM, ROM module is defective.) 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. w w w .c nc ce nt e r.c om Error codes 693 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B APPENDIX B–61863E/10 WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) om B.1 FUNCTION r.c This window function is a functional instruction by which the data on the CNC is read or is written. B.2 In the way to process, there are window function high speed and one processed at low speed. ce nt e LOW–SPEED RESPONSE AND HIGH–SPEED RESPONSE OF WINDOW FUNCTION 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). nc In a high–speed response, it is not necessity for take the interlock because the data is directly read. w w w .c 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 694 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) w w w .c nc ce nt e r.c om NOTE 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. 695 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.3 LIST OF WINDOW FUNCTIONS Number Description Function code R/W 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 :Low–speed response 15 R 5 Write work origin offset *PM :Low–speed response 16 W 6 Read parameters *RB56 :Low–speed response 17 R 7 Write parameters :Low–speed response 18 W 8 Read setting data *RB56 :Low–speed response 19 R 9 Write setting data :Low–speed response 20 W 10 Read custom macro variables *RB56 :Low–speed response 21 R :Low–speed response 22 W 23 R r.c om 1 Write custom macro variables Read the CNC alarm state 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 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 38 R 25 Reading tool life management data (number of tool group s) *PM *21T 39 R 26 Reading tool life management data (number of tools) *PM *21T 40 R Reading tool life management data (usable life of tool) *PM *21T 41 R Reading tool life management data (tool usage counter) *PM *21T 42 R Reading tool life management data (tool length compensation No. (1): Tool No.) *PM *21T 43 R Reading tool life management data (tool length compensation No. (2): Tool order No.) *PM 44 R 28 nc .c w 29 *RB56 :Low–speed response w 27 ce nt e 11 12 w 30 *21T 31 Reading tool life management data (cutter compensation No. (1): Tool No.) *PM *21T 45 R 32 Reading tool life management data (cutter compensation No. (2): Tool order No.) *PM *21T 46 R 33 Reading tool life management data (tool information (1): Tool No.) *PM *21T 47 R 34 Reading tool life management data (tool information (2): Tool order No.) *PM *21T 48 R 35 Reading tool life management data (tool No.) *21T 49 R 36 Reading the actual spindle speed 50 R *PM 696 Number 37 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 Description :low–speed response *PM *21T Entering data on the program check screen Function code R/W 150 W Reading clock data (date and time) 151 R 39 Writing torque limit data for the digital servo motor: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 an operator message 83 R 49 Reading value of the P– code macro variable 59 R 50 Writing value of the P– code macro variable 51 Reading the Tool life management data (Tool life counter type) 52 r.c om 38 *RB56 :low– speed response :low– speed response W R Registering the Tool life management data (Tool group) :low– speed response 163 W 53 Writing the Tool life management data (Tool life) :low– speed response 164 W 54 Writing the Tool life management data (Tool life counter) :low– speed response 165 W 55 Writing the Tool life management data (Tool life counter type) :low– speed response 166 W 56 Writing the Tool life management data (Tool length offset number (1): Tool number) :low– speed response 167 W 57 Writing the Tool life management data (Tool length offset num–ber (2): Tool operation sequence number) :low– speed response 168 W 58 Writing the Tool life management data (Cutter compensation number (1): Tool number) :low– speed response 169 W 59 Writing the Tool life management data (Cutter compensation nu–mber (2): Tool operation sequence number) :low– speed response 170 W 60 Writing the Tool life management data (Tool condition (1): Tool number) :low– speed response 171 W 61 Writing the Tool management data (Tool condition (2): Tooloperation sequence number) :low– speed response 172 W Writing the Tool life management data (Tool number) :low– speed response 173 W 211 R nc .c Reading the Estimate disturbance torque data w 63 w 62 ce nt e 60 160 w *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 *RB56 support high–speed window response for the RB5/RB6. *6 Functions marked with *C are not provided for the RB5/RB6.

697 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 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. om (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. Meaning r.c Completion code Normal termination 1 Error (invalid function code) 2 Error (invalid data block length) 3 Error (invalid data number) ce nt e 0 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. Top address +0 nc Function code 2 Completion code 4 .c 6 Data length (M) (Byte length of data area) These data set as input data are remain unchanged in the output data. Data number w 8 Data attribute w 10 *Data length Depends on the function.

X w X Data area 698 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.1 [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. om Reading CNC System Information [Input data structure] Top address + 0 r.c (Function code) 0 2 (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) — (Need not be set) 8 (Data attribute) — (Need not be set) w w w .c nc 10 (Data area) — (Need not be set) X X 42 [Completion codes] 0 : CNC system information has been read normally. 699 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Output data structure] Top address + 0 (Function code) 0 2 (Completion code) 0 (Always terminates normally.) 4 om (Data length) 14 6 8 (Data attribute) — r.c (Data number) — Value 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, . . .) nc ce nt e 10 w w w .c 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. 700 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 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. om [Input data structure] Top address + 0 (Function code) 13 2 r.c (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) N (N = offset number) 8 (Data attribute) M (M = offset type) nc 10 (Data area) — (Need not be set) X X .c 42 w w w (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 9 701 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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.

om 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. r.c [Output data structure] Top address + 0 ce nt e (Function code) 13 2 (Completion code) ? (See the explanation of the completion codes.) 4 (Data length) L (Normally set to 4) (L: Byte length of offset value) 6 8 nc (Data number) N (N = offset number) .c (Data attribute) M (M = offset type),, Tool offset value Signed binary (A negative value is represented in 2's complement.) Upper 3 bytes are always "0" for virtual tool tip Output data unit w w w 10 Value Input system Machining center system y Power Mate–D, F Radius specification Diameter Lathe specification system Radius specification Diameter specification 702 Increment system IS–B Increment system IS–C mm, deg system 0.001 0.0001 inch system 0.0001 0.00001 0.001 0.0001 0.002 0.0002 0.0001 0.00001 0.0001 0.00001 g mm, deg system inch system B.

WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 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. om [Input data structure] Top address + 0 (Function code) 14 2 r.c (Completion code) — (Need not to be set) 4 6 ce nt e (Data length) 4 (Data number) N (N = offset number) 8 (Data attribute) M (M = offset type) Value Tool offset value Signed binary (A negative value is represented in 2's complement.) Upper 3 bytes are always "0" for virtual tool tip nc 10 w w w .c (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. 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 9 703 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 Input data unit Input system 0.001 0.0001 inch system 0.0001 0.00001 0.001 0.0001 0.002 0.0002 Diameter specification g mm, deg system inch system [Completion codes] om mm, deg system Radius specification Diameter Lathe specification system Radius specification Increment system IS–C 0.0001 0.00001 0.0002 0.00002 r.c Machining center system y Power Mate–D, F Increment system IS–B 0 : The tool offset has been written normally. ce nt e 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. nc 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) [Output data structure] .c Top address + 0 (Function code) 14 w w w 2 (Completion code) ? (See the explanation of the completion codes.) 4 (Data length) L (L: Input data) 6 (Data number) N (N = Input data) 8 (Data attribute) M (Input data) 10 Tool offset value: Input data 704 Value Signed binary (A negative value is represented in 2's complement.) APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.4 [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] Top address + 0 r.c (Function code) 15 om Reading a Workpiece Origin Offset Value (Not Supported by the Power Mate–D or –F) 2 ce nt e (Completion code) — (Need not be set) 4 N = 0: External workpiece origin offset N = 1: G54 · · · · N = 6: G59 With "addition of workpiece coordinate system pair" N=7 : G54.1 P1 · · N=306 : G54.1 P300 (Data length) — (Need not be set) 6 (Data number) N (N = 0 to 6) 8 nc (Data attribute) M (M = 1 to n or –1) 10 .c (Data area) — (Need not be set) X w w X w 42 705 M = 1 to n: Workpiece origin offset number of a specific axis. n is the axis number. M = –1: Read for all axes B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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. om 6 : There is no workpiece coordinate shift option added. [Output data structure] r.c Top address + 0 (Function code) 15 4 (Completion code) ? (See the explanation of the completion codes.) ce nt e 2 (Data length) L (L: Byte length of the workpiece origin offset value) 6 L = 4: The workpiece origin offset value for a specific axis is read. L = 4*n: Workpiece origin offsets for all axes are read. (Data number) N (N = Input data) 8 10 nc (Data attribute) M (M = Input data) Signed binary number (A negative value is represented in 2's complement.) w w w .c Workpiece origin offset value Value Output data unit Input system Machining center system y Power Mate–D, F Radius specification Diameter Lathe specification system Radius specification Diameter specification 706 Increment system IS–B Increment system IS–C mm, deg system 0.001 0.0001 inch system 0.0001 0.00001 0.001 0.0001 0.002 0.0002 0.0001 0.00001 0.0001 0.00001 g mm, deg system inch system B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.5 [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. om Writing a Workpiece Origin Offset Value (:Low–Speed Response) (Not Supported by the Power Mate–D or –F) r.c [Input data structure] Top address + 0 (Function code) 16 L = 4 : Workpiece origin offset value for a specific axis is written. ce nt e 2 (Completion code) — (Need not be set) 4 L = 4*n Workpiece origin offset values for all axes are written. (Data length) L (L: Byte length of the workpiece origin offset value) 6 N = 0: External workpiece origin offset N = 1: G54 · · · · N = 6: G59 nc (Data number) N (N = 0 to 6) 8 (Data attribute) M (M = 1 to n, or –n) .c 10 w Value Signed binary (A negative value is represented in 2's complement.) Input data unit w w Workpiece origin offset value M = 1 to n: Workpiece origin offset number of a specific axis. n is the axis number. M = –1: Write for all axes Input system Machining center system y Power Mate–D, F Radius specification Diameter Lathe specification system Radius specification Diameter specification 707 Increment system IS–B Increment system IS–C mm, deg system 0.001 0.0001 inch system 0.0001 0.00001 0.001 0.0001 0.002 0.0002 0.0001 0.00001 0.0002 0.00002 g mm, deg system inch system B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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. om 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.

[Output data structure] Top address + 0 (Function code) 16 4 (Completion code) ? (See the explanation of the completion codes.) ce nt e 2 r.c 6 : There is no workpiece coordinate shift option added. (Data length) L (L: Input data) 6 (Data number) N (N = Input data) nc 8 (Data attribute) M (M = Input data) Workpiece origin offset value w w w .c 10 708 Value Signed binary number (A negative value is represented in 2's complement.) APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.6 Reading a Parameter (:Low–Speed Response) [Description] Parameter data in the CNC can be read. om 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. r.c 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). ce nt e For details of parameter data, refer to the Operator's manual of the CNC. [Input data structure] Top address + 0 (Function code) 17 2 (Completion code) — (Need not be set) nc 4 (Data length) — (Need not be set) 6 .c (Data number) N (N = parameter number) M = 0: w 8 w w 10 (Data attribute) M (M = 1 to n or –1) M = 1 to n: A specific axis M = –1: (Data area) — (Need not be set) X No axis All axes When all axes are specified by spindle parameters (parameters 4000 to 4799), only two axes are specified.

X 42 709 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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). om 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] Top address + 0 2 r.c (Function code) 17 When no axis or one axis is specified L = 1: Bit or byte parameter L = 2: Word parameter L = 4: Double word parameter (Completion code) ? (See the explanation of the completion codes.) 4 ce nt e (Data length) L (L = 1, 2, 4, 1*n, 2*n, 4*n) 6 When all axes are specified L = 1*n : Bit or byte parameter L = 2*n: Word parameter L = 4*n: Double word parameter (Data number) N (N = Input data) 8 (Data attribute) M (M = Input data) nc 10 Value Parameter data Parameter–dependent form w w w .c For the RB5/RB6, macro executor parameters 9000 to 9011 cannot be read. 710 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.7 Writing a Parameter (:Low–Speed Response) [Description] Data can be written in a parameter in the CNC. om 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. r.c 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. ce nt e 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 (Function code) 18 2 When no axis or one axis is specified L = 1: Bit or byte parameter L = 2: Word parameter L = 4: Double word parameter 4 nc (Completion code) — (Need not be set.) (Data length) L (L = 1, 2, 4, 1*n, 2*n, 4*n) .c When all axes are specified L = 1*n: Bit or byte parameter L = 2*n: Word parameter L = 4*n: Double word parameter 6 w (Data number) N (N = parameter number) M = 0: No axis M = 1 to n: A specific axis M = –1: All axes (Data attribute) M (M = 0, 1 to n, or –1) When all axes are specified by spindle parameters (parameters 4000 to 4799), only two axes are specified. w w 8 Value 10 Parameter data Parameter–dependent form 711 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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). Top address + 0 (Function code) 18 4 (Completion code) ? (See the explanation of the completion codes.) ce nt e 2 r.c [Output data structure] om 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. (Data length) L (L = Input data) 6 (Data number) N (N = Input data) 8 nc (Data attribute) M (M = Input data) Parameter data: Input data w w w .c 10 712 Value Parameter–dependent form B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.8 Reading Setting Data (:Low–Speed Response) [Description] The CNC setting data can be read. om 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. r.c 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] ce nt e Top address + 0 (Function code) 19 2 (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) nc 6 (Data number) N (N = Setting data number) M=0: .c 8 w w w 10 (Data attribute) M (M = 1 to n or –1) M = 1 to n: A specific axis M = –1: (Data area) — (Need not be set) X X 42 713 No axis All axes B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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). om [Output data structure] Top address + 0 (Function code) 19 When no axis or one axis is specified L = 1: Bit or byte parameter L = 2: Word parameter L = 4: Double word parameter (Completion code) ? (See the explanation of the completion codes.) r.c 2 4 (Data length) L (L = 1, 2, 4, 1*n, 2*n, 4*n) ce nt e 6 When all axes are specified L = 1*n: Bit or byte parameter L = 2*n: Word parameter L = 4*n: Double word parameter (Data number) N (N = Input data) 8 (Data attribute) M (M = Input data) 10 Parameter–dependent form w w w .c nc Setting data Value 714 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) 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] om Top address + 0 (Function code) 20 When no axis or one axis is specified L = 1: Bit or byte parameter L = 2: Word parameter L = 4: Double word parameter 2 r.c (Completion code) — (Need not be set.) 4 (Data length) L (L = 1, 2, 4, 1*n, 2*n, 4*n) ce nt e 6 When all axes are specified L = 1*n: Bit or byte parameter L = 2*n: Word parameter L = 4*n: Double word parameter (Data number) N (N = Setting data number) 8 M = 0: (Data attribute) M (M = 0, 1 to n, or –1) No axis M = 1 to n: A specific axis M = –1: All axes Value nc Setting data w w w .c 10 715 Setting data–dependent form B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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). Top address + 0 (Function code) 20 4 (Completion code) ?

(See the explanation of the completion codes.) ce nt e 2 r.c [Output data structure] om 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. (Data length) L (N = Input data) 6 (Data number) N (N = Input data) nc 8 (Data attribute) M (M = Input data) Setting data: Input data w w w .c 10 716 Value Setting data–dependent form APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [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. B.4.10 om Reading a Custom Macro Variable (:Low–Speed Response) ce nt e r.c NOTE 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. (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. NOTE For the RB5/RB6, system variables cannot be read. nc [Input data structure] w w w .c Top address + 0 (Function code) 21 2 (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) 6 (Data number) N (N = Custom macro variable number) 8 (Data attribute) M (M: Number of decimal places) 10 (Data area) — (Need not be set) X X 42 717 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [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. om 6 : The custom macro option is not provided. The specified variable number is out of range.

(Power Mate–D, F) [Output data structure] Top address + 0 2 (Completion code) ?

(See the explanation of the completion codes.) ce nt e 4 r.c (Function code) 21 (Data length) L (L: Byte length of custom macro variable data) 6 (Data number) N (N = Input data) 8 nc .c Custom macro variable data (4 bytes) Mantissa (custom macro B) The number of decimal places is not specified. M = 1x nx 7: The number of decimal places is specified. n stands for the number of decimal places. Value Signed binary (A negative value is represented in 2's complement.) Custom macro variable data (2 bytes) Exponent (custom macro B): The num– Signed binary 0 to 8 (no negative values) ber of decimal digits w w w 14 Custom macro B The mantissa of a floating– point number is indicated in 4 bytes, and the exponent is indicated in 2 bytes. M = 0: (Data attribute) M (M: Number of decimal places) 10 L = 6: 718 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.11 Writing a Custom Macro Variable (:Low–Speed Response) [Description] Data can be written in a custom macro variable in the CNC. For details of common variables, refer to the Operator's manual of the CNC. om [Input data structure] Top address + 0 (Function code) 22 2 4 r.c (Completion code) — (Need not be set) (Data length) L (L: Byte length of custom macro variable data) 6 (Data number) N (N = Custom macro variable number) 8 (Data attribute) — (Need not be set) Custom macro variable data (4 bytes) Mantissa (custom macro B) Signed binary (A negative value is represented in 2's complement.) Custom macro variable data (2 bytes) Signed binary Exponent (custom macro B): The num– (A negative value is represented in 2's ber of decimal digits complement.) w w w .c 14 Value nc 10 Custom macro B The mantissa of a floating–point number is indicated in 4 bytes, and the exponent is indicated in 2 bytes. ce nt e L = 6: 719 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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. om The specified variable number is out of range. (Power Mate–D, F) [Output data structure] (Function code) 22 4 (Completion code) ? (See the explanation of the completion codes.) ce nt e 2 r.c Top address + 0 (Data length) L (L: Input data) 6 (Data number) N (N = Input data) 8 nc (Data attribute) — (Need not be set) Custom macro variable data: Input data Signed binary (A negative value is represented in 2's Mantissa (custom macro B) complement.) .c 10 Custom macro variable data: Input data Exponent (custom macro B): The num– Signed binary ber of decimal digits w w w 14 Value 720 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.12 Reading the CNC Alarm Status B.4.12.1 [Description] When the CNC is in the alarm status, the alarm status data can be read. om Except Power Mate–D and –F The following alarm status data can be read: (1) First byte of alarm status data 6 5 4 3 2 1 0 ce nt e r.c 7 PS1 PS2 PS3 PS OTS OH SV MALM 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.) nc PS1 : w w w .c PS3 : PS : 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.) A P/S alarm other than the above alarm is generated OTS : Stroke limit alarm OH : Overheat alarm SV : Servo alarm MALM: Memory alarm 721 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 (2) Second byte of alarm status data 7 6 5 4 3 2 1 0 APC alarm SPA : Spindle alarm ce nt e [Input data structure] r.c APAL: om APAL SPA (Not used) (Not used) (Not used) (Not used) (Not used) (Not used) Top address + 0 (Function code) 23 2 (Completion code) — (Need not be set) nc 4 (Data length) — (Need not be set) (Data number) — (Need not be set) 8 (Data attribute) — (Need not to be set) 10 w w .c 6 (Data area) — (Need not be set) w X X 42 722 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 0 : This alarm status in the CNC has been read normally.

[Output data structure] Top address + 0 2 (Completion code) ? (See the explanation of the completion codes.) 4 r.c (Data length) 2 6 ce nt e (Data number) — 8 om (Function code) 23 (Data attribute) — Value 2 byte bit data. For the meanings of the bits, see [Description] in this section. CNC alarm status data w w w .c nc 10 723 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.12.2 (1) Overview For Power Mate–D and –F PMC application programs can read CNC alarm information. (2) Alarm information 1) Alarm status Information concerning the alarm type om 2) Detailed alarm Information concerning the alarm number and axis information (3) Input data configuration +2 Function code +6 Data length +8 Data number +10 Data attribute Data area : 23 (fixed) ce nt e Function code +4 Completion code r.c Top address+0 Completion code : No specification required. : No specification required. Data number : Number of alarms which can be stored. (Up to 30). If 31 or more are specified, the value is assumed to be 30. Data attribute : Other than 0 : Alarm status information nc Data length .c Data area w w w 15 Bit 0 14 13 12 : Detailed alarm information, indicated in two–byte bit–type data described below (multiple bits can be specified.) : No specification required. 11 10 9 8 7 6 5 4 3 2 1 0 bit : 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) 724 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 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) Data attribute : 0 om Bit 11 to Bit 15 : Not used : Alarm status information (4) Output data configuration +2 Function code Completion code Function code +4 Data length +6 +8 Data number Data attribute r.c Top address+0 +10 Data area : 23 (fixed) ce nt e 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). : Same as that for the input data.

Data attribute : Same as that for the input data. Data area attribute information : Two–byte bit–type data when the input data is set to 0 (each bit indicates the same as that for the input data). nc Data number .c (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. 2byte 2byte w 2byte 7 6 5 4 3 2 1 0 n 15 14 13 12 11 10 9 8 w 4byte Alarm number Axis information w 4byte ·· Bit 0 = 1 : When an alarm is is sued for the first axis Bit 1 = 1 : When an alarm is issued for the second axis Bits 2 to 15 are always set to 0. ·· ·· Number of alarms First alarm Second alarm n–th alarm (n: Number of alarms issued) (5) Completion code 0 : CNC alarm status has been read normally. 725 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 0 : This alarm status in the CNC has been read normally. [Output data structure] Top address + 0 2 (Completion code) ? (See the explanation of the completion codes.) 4 r.c (Data length) 2 6 ce nt e (Data number) — 8 om (Function code) 23 (Data attribute) — Value 2 byte bit data. For the meanings of the bits, see [Description] in this section.

CNC alarm status data w w w .c nc 10 726 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.13 Reading the Current Program Number [Description] The program number of a machining program being executed on the CNC can be read. om When a subprogram is executed on the CNC, the program number of the main program can also be read. Note that the main program is the first loop program from which the subprogram was called (even in nested). [Input data structure] Top address + 0 r.c (Function code) 24 2 (Completion code) — (Need not be set) ce nt e 4 (Data length) — (Need not be set) 6 (Data number) — (Need not be set) 8 nc (Data attribute) — (Need not to be set) (Data area) — (Need not be set) X X 42 w w w .c 10 727 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 0 : The program number of the program currently being executed has been read normally. [Output data structure] Top address + 0 2 om (Function code) 24 (Completion code) ? (See the explanation of the completion codes.) 4 r.c (Data length) L (L = 4) 6 8 ce nt e (Data number) — (Data attribute) — Value 10 Current program number: ON 12 Program number of the main program: OMN Unsigned binary, 2 bytes long nc (a) Current program number (ON) The program number of the program being executed is set. When the currently executed program is a subprogram, the number of its main program (first loop main program) is set. When the currently executed program is not a subprogram, 0 is set. w w w .c (b) Program number of the main program (OMN) 728 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) 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. om [Input data structure] Top address + 0 (Function code) 25 r.c 2 (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) — (Need not be set) 8 (Data attribute) — (Need not to be set) (Data area) — (Need not be set) X 42 w w w .c nc 10 729 X B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 0 : The current sequence number has been read normally. [Output data structure] Top address + 0 (Function code) 25 2 om (Completion code) ? (See the explanation of the completion codes.) 4 (Data length) L (L = 4) r.c Note that the data length must be set to 4 bytes even though the current program number is 2 bytes long (the sequence number is indicated by 5 digits). 6 (Data number) — ce nt e 8 (Data attribute) — 10 Value Unsigned binary w w w .c nc Current sequence number 730 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.15 [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. om Reading the Actual Velocity of Controlled Axes [Input data structure] r.c Top address + 0 (Function code) 26 2 ce nt e (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) 6 (Data number) — (Need not be set) 8 nc (Data attribute) — (Need not to be set) (Data area) — (Need not be set) X X 42 w w w .c 10 731 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 0 : The actual velocity for the controlled axes has been read normally. [Output data structure] Top address + 0 (Completion code) ? (See the explanation of the completion codes.) 4 (Data length) L (L = 4) 6 (Data number) — 8 Actual velocity for controlled axes w w w .c nc 10 ce nt e (Data attribute) — r.c 2 om (Function code) 26 732 Value Unsigned binary SInput in mm 1 mm/min SInput in inches 0.01 inch/min. B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.16 [Description] The absolute coordinates of the CNC–controlled axes for movement can be read. The read absolute coordinates equal the absolute coordinates (absolute position) indicated on the current position display screen in the CNC. (The screen is displayed by pressing function button POS.) om Reading the Absolute Position (Absolute Coordinates) of Controlled Axes [Input data structure] r.c Top address + 0 (Function code) 27 2 ce nt e (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) 6 (Data number) — (Need not be set) 8 M = 1 to n: Absolute coordinate of a specific axis. n is the axis number. nc (Data attribute) M (M = 1 to n or –1) 10 .c w Coordinates of all axes X w X w M = –1: (Data area) — (Need not be set) 42 [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.

733 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Output data structure] Top address + 0 (Function code) 27 4 (Completion code) ? (See the explanation of the completion codes.) (Data length) L (L = 4*n, n is the number of axes specified.) om 2 6 (Data number) — r.c 8 (Data attribute) L (L: Input data) Absolute coordinate of the controlled axis specified (4 bytes) Signed binary (A negative value is represented in 2's complement.) ce nt e 10 Value When the number of controlled axes is 4 Value 10 Absolute coordinate of the first axis (4 bytes) 14 Absolute coordinate of the second axis (4 bytes) nc Signed binary (A negative value is represented in 2's complement.) Absolute coordinate of the third axis (4 bytes) 22 Absolute coordinate of the fourth axis (4 bytes) w w w .c 18 Output data unit Input system Machining center y system Power Mate–D, F Radius specification Diameter Lathe specification system Radius specification Diameter specification 734 Increment system IS–B Increment system IS–C mm, deg system 0.001 0.0001 inch system 0.0001 0.00001 0.001 0.0001 0.001 0.0001 0.0001 0.00001 0.0001 0.00001 g mm, deg system inch system B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.17 [Description] 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.) om Reading the Machine Position (Machine Coordinates) of Controlled Axes [Input data structure] Top address + 0 r.c (Function code) 28 2 ce nt e (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) 6 (Data number) — (Need not be set) 8 M = 1 to n: Machine coordinate of a specific axis. n is the axis number. nc (Data attribute) M (M = 1 to n or –1) 10 M = –1: Coordinates of all axes .c (Data area) — (Need not be set) X w w X 42 [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. 735 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Output data structure] Top address + 0 (Function code) 28 4 (Completion code) ? (See the explanation of the completion codes.) om 2 (Data length) L (L = 4*n, n is the number of axes specified.) 6 r.c (Data number) — 8 (Data attribute) M (M: Input data) ce nt e 10 Value Machine coordinate of the controlled axis specified (4 bytes) Signed binary (A negative value is represented in 2's complement.) When the number of controlled axes is 4 Value 10 Machine coordinate of the first axis (4 bytes) 14 Machine coordinate of the second axis (4 bytes) nc Signed binary (A negative value is represented in 2's complement.) Machine coordinate of the third axis (4 bytes) .c 18 Machine coordinate of the fourth axis (4 bytes) Output data unit w w w 22 Input system Machining center system y Power Mate–D, F Radius specification Diameter Lathe specification system Radius specification Diameter specification 736 Increment system IS–B Increment system IS–C mm, deg system 0.001 0.0001 inch system 0.0001 0.00001 0.001 0.0001 0.001 0.0001 0.0001 0.00001 0.0001 0.00001 g mm, deg system inch system APPENDIX B–61863E/10 B.

WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) 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. om [Input data structure] Top address + 0 r.c (Function code) 29 2 (Completion code) — (Need not be set) ce nt e 4 (Data length) — (Need not be set) 6 (Data number) — (Need not be set) 8 M = 1 to n: Skip coordinate on a specific axis. n is the axis number. (Data attribute) M (M = 1 to n or –1) Coordinates on all axes nc M = –1: (Data area) — (Need not be set) .c 10 X w w w X 42 [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. 737 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Output data structure] Top address + 0 (Function code) 29 4 (Completion code) ? (See the explanation of the completion codes.) om 2 (Data length) L (L = 4*n, n is the number of axes specified.) 6 8 10 Value ce nt e (Data attribute) M (M: Input data) r.c (Data number) — Skip coordinate of the controlled axis specified(4 bytes) Signed binary (A negative value is represented in 2's complement.) When the number of controlled axes is 4 Value Skip coordinate of the second axis (4 bytes) 14 Skip coordinate of the third axis (4 bytes) Signed binary (A negative value is represented in 2's complement.) nc 10 Skip coordinate of the fourth axis (4 bytes) .c 18 Skip coordinate of the first axis (4 bytes) Output data unit w w w 22 Input system Machining center system y Power Mate–D, F Radius specification Diameter Lathe specification system Radius specification Diameter specification 738 Increment system IS–B Increment system IS–C mm, deg system 0.001 0.0001 inch system 0.0001 0.00001 0.001 0.0001 0.001 0.0001 0.0001 0.00001 0.0001 0.00001 g mm, deg system inch system APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) 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] om Top address + 0 (Function code) 30 2 r.c (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) ce nt e 6 (Data number) — (Need not be set) M = 1 to n: Servo delay for a specific axis 8 (Data attribute) M (M = 1 to n or –1) M = –1: 10 nc (Data area) — (Need not be set) .c X w w 42 739 X Servo delay for all axes B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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. (Function code) 30 4 6 (Completion code) ? (See the explanation of the completion codes.) (Data length) L (L = 4*n, n is the number of axes specified.) ce nt e 2 r.c Top address + 0 om [Output data structure] (Data number) — 8 (Data attribute) M (M: Input data) Servo delay for the controlled axis speSigned binary cified (4 bytes) (A negative value is represented in 2's complement.) nc 10 Value .c When the number of controlled axes is 4 Servo delay for the first axis (4 bytes) 14 Servo delay for the second axis (4 bytes) w w w 10 18 Servo delay for the third axis (4 bytes) 22 Servo delay for the fourth axis (4 bytes) 740 Value Signed binary (A negative value is represented in 2's complement.) APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.20 Reading the Acceleration/ Deceleration Delay on Controlled Axes [Description] 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.

om [Input data structure] Top address + 0 r.c (Function code) 31 2 (Completion code) — (Need not be set) ce nt e 4 (Data length) — (Need not be set) 6 (Data number) — (Need not be set) 8 M = 1 to n: Acceleration/deceleration delay for a specific axis (Data attribute) M (M = 1 to n or –1) nc M = –1: (Data area) — (Need not be set) .c 10 X w w w X Acceleration/deceleration delay for all axes 42 [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. 741 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Output data structure] Top address + 0 (Function code) 31 4 (Completion code) ? (See the explanation of the completion codes.) om 2 (Data length) L (L = 4*n, n is the number of axes specified.) 6 r.c (Data number) — 8 (Data attribute) M (M: Input data) ce nt e 10 Value Acceleration/deceleration delay for the Signed binary controlled axis specified (4 bytes) (A negative value is represented in 2's complement.) When the number of controlled axes is 4 Acceleration/deceleration delay for the Signed binary first axis (4 bytes) (A negative value is represented in 2's complement.) 14 Acceleration/deceleration delay for the second axis (4 bytes) nc 10 Acceleration/deceleration delay for the third axis (4 bytes) .c 18 Acceleration/deceleration delay for the fourth axis (4 bytes) Output data unit w w w 22 Input system Machining center system y Power Mate–D, F Radius specification Diameter Lathe specification system Radius specification Diameter specification 742 Increment system IS–B Increment system IS–C mm, deg system 0.001 0.0001 inch system 0.0001 0.00001 0.001 0.0001 0.001 0.0001 0.0001 0.00001 0.0001 0.00001 g mm, deg system inch system B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.21 [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. 6 7 5 S 4 3 om Reading Modal Data 2 1 Code in a group 0 r.c 1byte 1byte 0: Not specified in the block ce nt e 1: Specified in the block 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. Data type G code for machining center (M) w w w 0 0 1 2 3 4 .c G00 G01 ::G02 ::G03 :G33 Code in a group Data type nc Identificati on code Data 1 G17 G18 G19 0 8 4 2 G90 G91 0 1 3 (1/2) A series G00 G01 G02 G03 G32 G33 G34 G90 G92 G94 G71 G72 G73 G74 G96 G97 Data G code for lathe (T, G) B series C series G series only G00 G01 G02 G03 G33 G00 G01 G02 G03 G33 G34 G77 G78 G79 G71 G72 G73 G74 G34 G20 G21 G24 G72 G73 G74 G75 G series only G series only Code in a group 0 1 2 3 4 8 9 5 6 7 10 11 12 13 G96 G97 G96 G97 1 0 G90 G91 G90 G91 0 1 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 1 743 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 (2/2) Data type Data Identificati on code G code for machining center (M) Code in a 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 10 11 12 0 1 2 3 4 5 6 7 8 9 G22 G23 G22 G23 8 G73 G74 G76 G80 G81 G82 G83 G84 G85 G86 G87 G88 G89 0 1 G80 G83 G84 G85 G87 G88 G89 9 G66 G67 13 :G54 :G55 :G56 :G57 :G58 :G59 om r.c G98 G99 0 1 G54 G55 G56 G57 G58 G59 0 1 2 3 4 5 G50.2 G51.2 G50.2 G51.2 G50.2 G51.2 0 1 G13.1 G12.1 G13.1 G12.1 G13.1 G12.1 0 1 G40.1 G41.1 G42.1 1 2 0 18 G25 G26 0 1 w G98 G99 G54 G55 G56 G57 G58 G59 17 w 0 1 2 3 5 6 7 G54 G55 G56 G57 G58 G59 0 1 w G80 G83 G84 G85 G87 G88 G89 0 1 2 3 4 5 :G15 :G16 19 G13.1 G12.1 G80 G83 G84 G85 G87 G88 G89 1 0 1 0 20 1 0 G66 G67 :G68 :G69 16 G22 G23 G66 G67 1 2 3 0 15 Code in a group G66 G67 :G61 :G62 :G63 :G64 14 G code for lathe (T, G) B series C series 1 0 nc 11 0 1 .c :G50 :G51 A series Data ce nt e :G98 :G99 10 Data type 0 1 744 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 (2) Format and types of modal data for other than the G function 6 5 4 bytes FLAG1 1 byte FLAG2 1 byte 4 3 2 1 0 – Number of input digits om 7 Data 0: Positive 1: Negative 0: A decimal point not specified r.c 1: A decimal point specified 0: Not specified in the current block 1: Specified in the current block – – – – Number of decimal places ce nt e – 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 decimal places may not be 0. w w w .c nc Data type Identification code Specified address –2 Enter identification codes 100 to 126 at one time. 100 B (second auxiliary function) 101 D 102 E (reserved) 103 F 104 H 105 L 106 M 107 S 108 T 109 R 110 P 111 Q 112 A 113 C 114 I 115 J 116 K 117 N 118 O 119 U 120 V 121 W 122 X 123 Y 124 Z 125 M2 126 M3 NOTE The Power Mate–D or –F is not provided with the second auxiliary function. 745 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Input data structure] Top address + 0 (Function code) 32 2 (Completion code) — (Need not be set) (Data length) — (Need not be set) 6 N = 0 to : (Data number) N (N: Data type) (Data attribute) M (M: Specified block) 10 N = –1: See the list of data explained above. All data for G function N = –2: All data for other than G function r.c 8 om 4 Current block M=1: Next block M=2: Block after the next block ce nt e (Data area) — (Need not be set) M=0: X X nc 20 When all data items are specified to be read, the data items are all output w w w .c simultaneously in the order specified in the above data table. 746 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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 2 (Completion code) ?

(See the explanation of the completion codes.) 4 L=2 : 6 (Data number) N (N: Input data) L=6: Other than G function Value Modal data for G function (2 bytes) Or 8 See the data format for the G function. The upper byte must always be set to 0.

Value Data part of modal data for other than G function(4 bytes) See the data format for other than the G function. nc (Data attribute) M (M: Input data) .c w 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. w w All data for G function L = 6*m : All data for other than G function (n: Number of groups for the G function) (m: Number of types other than for the G function) (Data attribute) M (M: Input data) 14 L = 2*n : ce nt e 8 10 G function r.c (Data length) L (L = 2, 6, 2*n, 6*m) 10 om (Function code) 32 (See the explanation above 747 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.22 Reading Diagnosis Data (:Low–Speed Response) [Description] The information displayed on the diagnosis data screen in the CNC can be read. (Function code) 33 2 (Completion code) — (Need not be set) 4 6 ce nt e (Data length) — (Need not be set) r.c Top address + 0 om [Input data structure] (Data number) N (N: Diagnosis No.) 8 M=0: (Data attribute) M (M: 0, 1 to n, or –1) M = 1 to n: One axis M = –1: (Data area) — (Need not be set) w w w .c nc 10 748 No axis All axes B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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). om 6 : An option required for reading the specified diagnosis data, such as the remote buffer option, is not provided. [Output data structure] Top address + 0 2 r.c (Function code) 33 (Completion code) ? (See the explanation of the completion codes.) ce nt e 4 When no axis or one axis is specified L = 1 : Bit or byte parameter L = 2: Word parameter L = 4: Double Word parameter (Data length) L (L = 1, 2, 4, 1*n, 2*n, 4*n) 6 When all axes are specified L = 1*n Bit or byte parameter L = 2*n: Word parameter L = 4*n: Double word parameter (Data number) N (N: Input data) 8 (Data attribute) M (M: Input data) nc Diagnosis data w w w .c 10 Value 749 Data–dependent form B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.23 Reading A/D Conversion Data [Description] The load current for the CNC control axis can be converted to analog voltage, and input to the A/D converter in the CNC to obtain digital data. The output of the A/D converter can then be read. om [Input data structure] Top address + 0 (Function code) 34 2 r.c (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) N (N: Type of analog voltage) 8 (Data attribute) M (M – 1 to 8: Axis specification) 10 nc (Data area) — (Need not be set) w w w .c (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 NOTE Only one–path control of the Power Mate MODEL–D is provided with one channel of general–purpose analog voltage information. 750 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) (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: Specification of CNC controlled–axis Specified data JV1 (MAIN BOARD) JV2 (MAIN BOARD) JV3 (MAIN BOARD) JV4 (MAIN BOARD) JV5 (OPTION BOARD) JV6 (OPTION BOARD) JV7 (OPTION BOARD) JV8 (OPTION BOARD) [Completion codes] r.c om 1 2 3 4 5 6 7 8 Connector in the CNC 0 : A/D conversion data has been read normally. ce nt e 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] Top address + 0 nc (Function code) 34 (Completion code) ? (See the explanation of the completion codes.) .c 2 4 w (Data length) 2 w w 6 (Data number) N (Input data) 8 (Data attribute) M (Input data) Value 10 A/D conversion data (2 bytes) AD For general purpose analog Binary number from 0 to 255 10 A/D conversion data (2 bytes) AD For CNC controlled axis load information Binary number from 0 to "6554 751 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 (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. (AD * 128) N + Load current [A peak] 7282 om The value actually set in the AD field is obtained from the following formula: AD = A/D conversion data [Value read by the window function (")] N = Nominal current limit for the amplifier corresponding to the motor For the nominal current limits, see the table below or the descriptions of the control motor. r.c (b) A/D conversion data (A/D) for general–purpose analog voltage information ce nt e 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. (AD * 128) N + Load current [A peak] 128 AD = A/D conversion data [Value read by the window function (")] N = Nominal current limit for the amplifier corresponding to the motor For the nominal current limits, see the table below or the descriptions of the control motor. 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 w w 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 w 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 SVM1–40S Output current at rated output Nominal current limit 2.9Arms 12Ap 5.8Arms 20Ap 5.8Arms 40Ap Connected axis .c Model Applicable motor model nc Servo amplifier module 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 α α α α α α α α α 752 APPENDIX B–61863E/10 Servo amplifier module Applicable motor model Output current at rated output Nominal current limit α 3/3000 α 6/2000 α 12/2000 α M3/3000 α L3/3000 α C22/1500 12.2Arms 40Ap α 3/3000 α 6/2000 α 12/2000 α 22/1500 α M3/3000 α L3/3000 α C22/1500 12.2Arms 40Ap M axis M axis N axis N axis N axis SVM1–40L SVM2–40/80 L axis 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 18.4Arms ce nt e SVM2–12/40 SVM2–20/40 SVM2–40/40 SVM3–12/12/40 SVM3–12/20/40 SVM3–20/20/40 om Connected axis r.c Model B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) α 30/2000 α 40/2000 26.7Arms α 22/3000 α 30/3000 α 40/2000 (with a fan) α L25/3000 α L50/2000 51.0Arms nc SVM1–130 80Ap 130Ap (Note3) w w w .c NOTE 1 The rated output is guaranteed at the rated input voltage. If an input voltage fluctuation occurs even within the allowable fluctuation range, the rated output may not be obtained. 2 The current limits (peak values) are standard values. The operation value variation due to a circuit constant is about +10%. 3 SVM1–130 requires forced air cooling when the α 22/3000, α 30/3000, α 40/2000 (with a fan), α L25/3000, or α L50/2000 is driven.

At this time, the rated output current is 51.0 Arms.

753 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.24 Reading Tool Life Management Data (Tool Group No.) (Not available for Power Mate–D/F, Series 21–TA) [Description] By specifying a tool No., the No. of the tool group to which the specified tool belongs can be read from tool life management data. om [Input data structure] Top address + 0 (Function code) 38 2 r.c (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) — (Need not be set) 8 (Data attribute) M (M : Tool No.) w w w .c nc 10 (Data area) — (Need not be set) X X 42 NOTE 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. 754 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 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. om [Output data structure] Top address + 0 (Function code) 38 2 r.c (Completion code) ? (See the explanation of the completion codes.) 4 (Data length) L (L = 4 n) ce nt e L = 4 to 4 n n is the number of tool groups to which the specified tool belongs. 6 (Data number) — 8 (Data attribute) M (M: Input data) Tool group No. (4 bytes) Unsigned binary nc 10 Value When the specified tool belongs to two or more tool groups Tool group No. (4 bytes) .c 10 Tool group No. (4 bytes) Tool group No. (4 bytes) w 14 w w 18 755 Unsigned binary. B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.25 Reading Tool Life Management Data (Number of Tool Groups) (Not available for Power Mate–D/F, Series 21–TA) [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. GS2 om Parameter 6800 Number of tools The numbers in parentheses apply when the additional option is used GS1 M series T series 0 1 to 16 (1 to 64) 1 to 16 (1 to 16) 0 1 1 to 32 (1 to 128) 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) r.c 0 1 to 32 (1 to 32) ce nt e M series: For Machining Centers [Input data structure] Top address + 0 (Function code) 39 2 nc (Completion code) — (Need not be set) (Data length) — (Need not be set) (Data number) — (Need not be set) 8 (Data attribute) — (Need not be set) 10 (Data area) — (Need not be set) w w w .c 4 X X 42 756 T series: For Lathes APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 0 : The number of tool group Nos. has been read normally.

6 : The tool life management option has not been added. [Output data structure] (Function code) 39 2 (Completion code) ? (See the explanation of the completion codes.) 4 r.c (Data length) 4 6 ce nt e (Data number) — 8 (Data attribute) — Number of tool groups (4 bytes) w w w .c nc 10 om Top address + 0 757 Value Unsigned binary. B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.26 Reading Tool Life Management Data (Number of Tools) (Not available for Power Mate–D/F, Series 21–TA) [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. GS2 om Parameter 6800 Number of tools The numbers in parentheses apply when the additional option is used GS1 M series T series 0 1 to 16 (1 to 64) 1 to 16 (1 to 16) 0 1 1 to 32 (1 to 128) 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) r.c 0 ce nt e 1 to 32 (1 to 32) M series: For Machining Centers [Input data structure] Top address + 0 (Function code) 40 2 nc (Completion code) — (Need not be set) (Data length) — (Need not be set) 6 (Data number) N (N: Tool group No.) 8 (Data attribute) — (Need not be set) 10 (Data area) — (Need not be set) w w w .c 4 X X 42 758 T series: For Lathes APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) NOTE 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] om 0 : The number of tools has been read normally.

3 : The specified tool group No. is invalid.

[Output data structure] Top address + 0 2 4 ce nt e (Function code) 40 r.c 6 : The tool life management option has not been added. (Completion code) ? (See the explanation of the completion codes.) (Data length) 4 6 nc (Data number) N (N: Input data) 8 (Data attribute) — .c Number of tools (4 bytes) w w w 10 759 Value Unsigned binary B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.27 Reading Tool Life Management Data (Tool Life) (Not available for Power Mate–D/F, 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. om 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] Top address + 0 (Function code) 41 r.c 2 (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) N (N: Tool group No.) 8 (Data attribute) — (Need not be set) w w w .c nc 10 (Data area) — (Need not be set) X X 42 NOTE 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. 760 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 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] 2 (Completion code) ? (See the explanation of the completion codes.) 4 (Data length) 4 6 8 ce nt e (Data number) N (N: Input data) (Data attribute) — Value Unsigned binary Unit: Time (minutes) or number of cycles Tool life (4 bytes) w w w .c nc 10 r.c (Function code) 41 om Top address + 0 761 B.

WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.28 Reading Tool Life Management Data (Tool Life Counter) (Not available for Power Mate–D/F, Series 21–TA) [Description] By specifying a tool group No., the tool life counter for the specified tool group can be read from tool life management data. om [Input data structure] Top address + 0 (Function code) 42 r.c 2 (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) N (N: Tool group No.) 8 (Data attribute) — (Need not be set) w w w .c nc 10 (Data area) — (Need not be set) X X 42 NOTE 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. 762 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 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] 2 (Completion code) ? (See the explanation of the completion codes.) 4 (Data length) 4 6 8 ce nt e (Data number) N (N: Input data) (Data attribute) — Value Unsigned binary Unit: Time (minutes) or number of cycles Tool life counter (4 bytes) w w w .c nc 10 r.c (Function code) 42 om Top address + 0 763 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.29 [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. om Reading Tool Life Management Data (Tool Length Compensation No. (1): Tool No.) (Not available for Power Mate–D/F, Series 21–TA) [Input data structure] Top address + 0 r.c (Function code) 43 2 ce nt e (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) 6 (Data number) N (N: Tool group No.) 8 nc (Data attribute) M (M: Tool No.) (Data area) — (Need not be set) X X w w w .c 10 42 NOTE 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. 764 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 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. Top address + 0 (Function code) 43 2 (Completion code) ? (See the explanation of the completion codes.) 4 6 ce nt e (Data length) 4 r.c [Output data structure] om 6 : The tool life management option has not been added. (Data number) N (N: Input data) 8 Value Tool length compensation No. (4 bytes) Unsigned binary w w w .c nc 10 (Data attribute) M (M: Input data) 765 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.30 [Description] 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. om Reading Tool Life Management Data (Tool Length Compensation No. (2): Tool Order No.) (Not available for Power Mate–D/F, Series 21–TA) [Input data structure] Top address + 0 r.c (Function code) 44 2 (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) N (N: Tool group No.) 8 (Data attribute) M (M: Tool order No.) w w w .c nc 10 (Data area) — (Need not be set) X X 42 NOTE 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.

766 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 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. Top address + 0 (Function code) 44 2 (Completion code) ? (See the explanation of the completion codes.) ce nt e 4 r.c [Output data structure] om 6 : The tool life management option has not been added. (Data length) 4 6 (Data number) N (N: Input data) 8 (Data attribute) M (M: Input data) nc 10 w w w .c Tool length compensation No. (4 bytes) 767 Value Unsigned binary B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.31 [Description] By specifying a tool group No. and a tool 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. om Reading Tool Life Management Data (Cutter Compensation No. (1): Tool No.) (Not available for Power Mate–D/F, Series 21–TA) [Input data structure] Top address + 0 r.c (Function code) 45 2 (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) N (N: Tool group No.) 8 (Data attribute) M (M : Tool No.) w w w .c nc 10 (Data area) — (Need not be set) X X 42 NOTE 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. 768 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 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. Top address + 0 (Function code) 45 2 (Completion code) ? (See the explanation of the completion codes.) 4 6 ce nt e (Data length) 4 r.c [Output data structure] om 6 : The tool life management option has not been added. (Data number) N (N: Input data) 8 (Data attribute) M (M: Input data) Cutter compensation No. (4 bytes) w w w .c nc 10 769 Value Unsigned binary B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.32 [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. om Reading Tool Life Management Data (Cutter Compensation No. (2): Tool Order No.) (Not available for Power Mate–D/F, Series 21–TA) [Input data structure] Top address + 0 r.c (Function code) 46 2 (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) N (N: Tool group No.) 8 (Data attribute) M (M: Tool order No.) w w w .c nc 10 (Data area) — (Need not be set) X X 42 NOTE 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. 770 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 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. om 6 : The tool life management option has not been added. [Output data structure] Top address + 0 2 (Completion code) ? (See the explanation of the completion codes.) ce nt e 4 r.c (Function code) 46 (Data length) 4 6 (Data number) N (N: Input data) 8 (Data attribute) M (M: Input data) nc 10 w w w .c Cutter compensation No. (4 bytes) 771 Value Unsigned binary B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.33 [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. om Reading Tool Life Management Data (Tool Information (1) : Tool No.) (Not available for Power Mate–D/F, Series 21–TA) [Input data structure] Top address + 0 r.c (Function code) 47 2 (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) N (N: Tool group No.) 8 (Data attribute) M (M: Tool No.) w w w .c nc 10 (Data area) — (Need not be set) X X 42 NOTE 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. 772 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 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. Top address + 0 (Function code) 47 2 (Completion code) ? (See the explanation of the completion codes.) 4 6 ce nt e (Data length) 4 r.c [Output data structure] om 6 : The tool life management option has not been added. (Data number) N (N: Input data) 8 (Data attribute) M (M: Input data) Number of tools (4 bytes) w w w .c nc 10 773 Value 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. B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.34 Reading Tool Life Management Data (Tool Information (2): Tool Order No.) (Not available for Power Mate–D/F, Series 21–TA) [Description] By specifying a tool group No. and a tool order No., the information for the specified tool can be read from tool life management data.

om [Input data structure] Top address + 0 r.c (Function code) 48 2 (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) N (N: Tool group No.) 8 (Data attribute) M (M: Tool order No.) (Data area) — (Need not be set) X X 42 w w w .c nc 10 NOTE 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. 774 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 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. Top address + 0 (Function code) 48 2 (Completion code) ? (See the explanation of the completion codes.) ce nt e 4 r.c [Output data structure] om 6 : The tool life management option has not been added. (Data length) 4 6 (Data number) N (N: Input data) 8 (Data attribute) M (M: Input data) nc 10 w w w .c Tool information (4 bytes) 775 Value 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. B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.35 Reading Tool Life Management Data (Tool No.) (Not available for Power Mate–D/F, Series 21–TA) [Description] By specifying a tool group No. and a tool order No., the No. of the corresponding tool can be read from tool life management data. om [Input data structure] Top address + 0 (Function code) 49 2 r.c (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) N (N: Tool group No.) 8 (Data attribute) M (M: Tool order No.) 10 nc (Data area) — (Need not be set) X w w w .c X 42 NOTE 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.

776 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 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. (Function code) 49 2 (Completion code) ? (See the explanation of the completion codes.) 4 6 ce nt e (Data length) 4 r.c Top address + 0 om [Output data structure] (Data number) N (N: Input data) 8 (Data attribute) M (M: Input data) Value Unsigned binary Tool No. (4 bytes) w w w .c nc 10 777 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.36 Reading the Actual Spindle Speed [Description] The actual speed of the spindle can be read from the CNC. [Input data structure] Top address + 0 om (Function code) 50 2 (Completion code) — (Need not be set) r.c 4 (Data length) — (Need not be set) 6 ce nt e (Data number) — (Need not be set) 8 (Data attribute) — (Need not be set) 10 nc (Data area) — (Need not be set) .c X w w w 42 778 X APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 0 : The actual speed of the spindle has been read normally. [Output data structure] Top address + 0 (Function code) 50 (Completion code) ? (See the explanation of the completion codes.) om 2 4 (Data length) 4 (Data number) — 8 Value Unsigned binary rpm Actual spindle speed w w w .c nc 10 ce nt e (Data attribute) — r.c 6 779 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.37 [Description] On the program check screen of the CNC, data can be entered for the spindle tool No. and the next tool No. This function is available only with the M series CNCs. [Input data structure] Top address + 0 (Function code) 150 2 4 ce nt e (Completion code) — (Need not be set) om This function is effective only when bit 2 of parameter 3108 is 1. r.c Entering Data on the Program Check Screen (:Low–Speed response) (Not available for Power Mate–D/F, Series 21–TA) (Data length) 4 6 N = 0 : Spindle tool No. (8 digits) (Data number) N (N = 0, 1) 8 N = 1 : Next tool No. (8 digits) (Data attribute) — (Need not be set) nc 10 w w w .c Data for the spindle tool No. (4 bytes) or data for the next tool No. (4 bytes) 780 Value Unsigned binary APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 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] (Function code) 150 2 (Completion code) ?

(See the explanation of the completion codes.) r.c 4 (Data length) 4 (Input data) 6 N = 0 : Spindle tool No. (8 digits) ce nt e (Data number) N (Input data) 8 N = 1 : Next tool No. (8 digits) (Data attribute) — (Input data) Data for the spindle tool No. (4 bytes) or data for the next tool No. (4 bytes) w w w .c nc 10 om Top address + 0 781 Value Unsigned binary B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.38 Reading Clock Data (Date and Time) (Not available for Power Mate–F) [Description] The current date (year, month, day) and time (hours, minutes, seconds) can be read from the clock built into the CNC. [Input data structure] om Top address + 0 (Function code) 151 2 r.c (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) N = –1: Reads current date and time. ce nt e 6 (Data number) N (N = 0, 1) 8 N = 0: Reads current date.

N = 1: Reads current time. (Data attribute) — (Need not be set) 10 nc (Data area) — (Need not be set) [Completion codes] .c 0 : Data of the clock built into the CNC has been read normally. w w w 3 : A value other than 0, 1, and –1 was specified for the data No. 782 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Output data structure] Top address + 0 (Function code) 151 2 (Completion code) ? (See the explanation of the completion codes.) 4 om (Data length) 6/12 6 (Data number) N (Input data) r.c 8 (Data attribute) — (Input data) 12 14 Unsigned binary Current date (year) or time (hours) ce nt e 10 Value Current date (month) or time (minutes) Current date (day) or time (seconds) When both the current date and current time are specified to be read by entering [–1] for the data No. — (Input data) Value nc 10 12 Unsigned binary Current date (month) Current date (day) .c 14 Current date (year) 16 w 18 w 20 Current time (hours) Current time (minutes) Current time (seconds) w [Example] September 10th, 1990 Data area +2 +4 [Example] 23:59:59 (hours:minutes:seconds) Data area 23 1990 +2 9 +4 10 783 59 59 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.39 Entering Torque Limit Data for the Digital Servo Motor (:Low–Speed response) [Description] Torque limit values for the digital servo motor can be entered. [Input data structure] (Function code) 152 2 r.c (Completion code) — (Need not be set) 4 ce nt e (Data length) 2 6 om Top address + 0 (Data number) — (Need not be set) 8 (Data attribute) M (M: 1 to n) M = 1 to n: Axis No. Value Unsigned binary Torque limit data (1 byte) The high–order byte is always set to 0. Values from 0 to 255 correspond to 0% to 100%. nc 10 w w w .c [Example] To specify a torque limit of 50%, enter 128. 784 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) [Completion codes] 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. [Output data structure] 2 (Completion code) ? (See the explanation of the completion codes.) 4 (Data length) 2 (Input data) ce nt e 6 r.c (Function code) 152 om Top address + 0 (Data number) — (Input data) 8 (Data attribute) M (M: Input data) Value Unsigned binary Torque limit data (1 byte): Input data The high–order byte is always set to 0. Values from 0 to 255 correspond to 0% to 100%. w w w .c nc 10 785 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.40 [Description] Load information of the serial spindle can be read. The equation to normalize the load information is shown below Load (%) + L 32767 Data read from the window λ: 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.) [Input data structure] Top address + 0 ce nt e (Function code) 153 om L: r.c Reading Load Information of the Spindle Motor (Serial Interface) 2 (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) 6 N = –1: Specifies both the first and second axes of the serial spindle. N = 0: Specifies the first axis of the serial spindle.

N = 1: Specifies the second axis of the serial spindle. nc (Data number) N (Need not be set) 8 .c (Data attribute) — (Need not be set) w w w 10 (Data area) — (Need not be set) X X 42 786 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 0 : Load information of the serial spindle has been read normally. [Output data structure] Top address + 0 (Function code) 153 2 4 L = 2: (Data length) L (L = 2, 4) r.c (Data number) N (N: Input data) ce nt e 8 (Data attribute) — Value Load information of the serial spindle (first or second axis) When all axes are specified 8 nc (Data attribute) — 10 First axis in the load information of the serial spindle Second axis in the load information of the serial spindle w w w .c 12 Specifies the first axis. L = 4: Specifies all axes. N = –1: Specifies both the first and second axes of the serial spindle. N = 0: Specifies the first axis of the serial spindle. N = 1: Specifies the second axis of the serial spindle. 6 10 om (Completion code) ? (See the explanation of the completion codes.) 787 Signed binary, 2 bytes long Value Signed binary, 2 bytes long B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.41 [Description] Parameter data in the CNC can be read directly from the CNC via the FANUC bus. This function is basically the same as the function described in Section 3.6 "Reading a Parameter," except that the function code is 154 and some of the completion codes are different. om Reading a Parameter (Not available for Power Mate–D/F, Series 21–TA) [Input data structure] Top address + 0 r.c (Function code) 154 2 ce nt e (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) 6 (Data number) N (N: Parameter No.) M=0 8 (Data attribute) M (M: 0, 1 to n, or –1) M = 1 to n : Specific axis nc 10 M = –1 .c (Data area) — (Need not be set) : All axes When all axes are specified by spindle parameters (parameters 4000 to 4799), only two axes are specified. X w w X : No axis w 42 [Completion codes] 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. 788 B.

WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.42 Reading Set Data (Not available for Power Mate–D/F, Series 21–TA) [Description] Set data stored in the CNC can be read directly from the CNC via the FANUC bus. om This function is basically the same as the function described in Section 3.8 "Reading Set Data," except that the function code is 155 and some of the completion codes are different. [Input data structure] Top address + 0 r.c (Function code) 155 2 (Completion code) — (Need not be set) ce nt e 4 (Data length) — (Need not be set) 6 (Data number) N (N: Setting data No.) M=0 8 (Data attribute) M (M: 0, 1 to n, or –1) M = 1 to n : Specific axis M = –1 nc 10 : No axis : All axes .c (Data area) — (Need not be set) X w w w X 42 [Completion codes] 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. 789 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.43 Reading Diagnosis Data (Not available for Power Mate–D/F, Series 21–TA) [Description] Data displayed on the diagnosis data screen of the CNC can be read directly from the CNC via the FANUC bus. om This function is basically the same as the function described in Section 3.22 "Reading Diagnosis Data," except that the function code is 156 and some of the completion codes are different.

[Input data structure] (Function code) 156 2 ce nt e (Completion code) — (Need not be set) r.c Top address + 0 4 (Data length) — (Need not be set) 6 (Data number) N (N: Diagnosis No.) M=0 8 (Data attribute) M (M: 0, 1 to n, or –1) : No axis nc M = 1 to n : Specific axis M = –1 10 : All axes .c (Data area) — (Need not be set) X w w w X 42 [Completion codes] 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.

790 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.44 [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. om Reading a Character String of the CNC Program Being Executed in the Buffer The maximum number of characters in a character string is fixed to 64. r.c [Input data structure] Top address + 0 (Function code) 157 ce nt e 2 (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) 6 (Data number) — (Need not be set) M=0 nc 8 (Data attribute) M (M: Specified block) : Current block M = 1 to n : Next block M = –1 10 : Next block but one X X w w w .c (Data area) — (Need not be set) 74 NOTE 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. 791 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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] Top address + 0 2 om (Function code) 157 (Completion code) ?

(See the explanation of the completion codes.) r.c 4 (Data length) 64 64 characters 6 ce nt e (Data number) — 8 (Data attribute) M (M: Specified block) 10 nc 11 NC command data for the first character ASCII code If a block consists of less than 64 RC(1) characters, the remaining bytes are NC command data for the second filled with 20H (space). character RC(2) X X NC command data for the 64th character RC(64) w w w .c 73 X X 792 X X B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.45 Reading the Relative Position on a Controlled Axis [Description] The relative coordinates of the machine moving along an axis controlled by the CNC can be read. [Input data structure] om Top address + 0 (Function code) 74 2 r.c (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) ce nt e 6 (Data number) — (Need not be set) 8 M = 1 to n : Reads the relative coordinates of each axis.

n is an axis No. (Data attribute) M (M: 1 to n or –1) 10 M = –1 : Reads the relative coordinates of all axes. nc (Data area) — (Need not be set) X .c X w w w 42 [Completion codes] 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. 793 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Output data structure] Top address + 0 (Function code) 74 4 (Completion code) ? (See the explanation of the completion codes.) (Data length) L (L = 4*n. n is the number of specified axes.) om 2 6 (Data number) — r.c 8 (Data attribute) M (M: Input data) Value Signed binary (A negative value is represented in 2's complement.) Relative coordinates on the specified controlled axis (4 bytes) ce nt e 10 When the number of controlled axes is 4 Value 10 Signed binary (A negative value is represented in 2's complement.) Relative coordinates on the second axis (4 bytes) nc 14 Relative coordinates on the first axis (4 bytes) 18 Relative coordinates on the fourth axis (4 bytes) w w w .c 22 Relative coordinates on the third axis (4 bytes) Output data unit Input system Machining center y system Power Mate–D, F Radius specification Diameter specification Lathe system Radius specification Diameter specification Increment system IS–B Increment system IS–C mm, deg system 0.001 0.0001 inch system 0.0001 0.00001 0.001 0.0001 0.001 0.0001 0.0001 0.00001 0.0001 0.00001 mm, deg g system inch system Double values can be read for a machining center system or when radius specification is used for the relevant axis of a lathe system. 794 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.46 Reading the Remaining Travel [Description] om 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] Top address + 0 r.c (Function code) 75 2 ce nt e (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) 6 (Data number) — (Need not be set) 8 M = 1 to n: Reads the remaining travel along each axis. n is an axis No. 10 nc (Data attribute) M (M: 1 to n or –1) M = –1 : Reads the remaining travel along all axes. .c (Data area) — (Need not be set) X 42 w w w X [Completion codes] 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. 795 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Output data structure] Top address + 0 (Function code) 75 4 (Completion code) ? (See the explanation of the completion codes.) (Data length) L (L = 4*n. n is the number of specified axes.) om 2 6 r.c (Data number) — 8 (Data attribute) M (M: Input data) Value Signed binary (A negative value is represented in 2's complement.) Remaining travel along the specified controlled axis (4 bytes) ce nt e 10 When the number of controlled axes is 4 Value 10 Signed binary (A negative value is represented in 2's complement.) Remaining travel along the second axis (4 bytes) nc 14 Remaining travel along the first axis (4 bytes) 18 Remaining travel along the fourth axis (4 bytes) Output data unit w w w .c 22 Remaining travel along the third axis (4 bytes) Input system Machining center system y Power Mate–D, F Radius specification Diameter specification Lathe specification system Radius specification Diameter specification 796 Increment system IS–B Increment system IS–C mm, deg system 0.001 0.0001 inch system 0.0001 0.00001 0.001 0.0001 0.0005 0.00005 0.0001 0.00001 0.00005 0.000005 g mm, deg system inch system B.

WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 B.4.47 Reading CNC Status Information [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 om (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 Alarm status (7) Status of program edit (Indication) 13 ce nt e (2) STOP (1) EDIT 14 (5) ––EMG–– (3) (4) MTN FIN r.c (6) (6) ALM 16:52:13 00 05 10 15 20 25 30 [Input data structure] nc Top address + 0 (Function code) 76 w .c 2 w 14 (Soft key indication) 15 w (7) READ (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) 6 (Data number) — (Need not be set) 8 (Data attribute) — (Need not be set) 10 X (Data area) — (Need not be set) 42 797 X 15 35 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 0 : CNC status information has been read normally. [Output data structure] Top address + 0 (Function code) 76 (Completion code) ? (See the explanation of the completion codes.) om 2 4 (Data length) 14 6 r.c (Data number) — (Input data) 8 Indication of which mode is currently selected, automatic or manual (2 bytes) nc 10 ce nt e (Data attribute) — (Input data) 12 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 Status of automatic operation (2 bytes) 0 1 2 3 : **** (Reset states) : STOP : HOLD : STaRT 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) 0 : (Releases the emergency stop state) 1 : — —EMerGency — — 2 : — RESET — (The reset signal is on.) 20 Alarm status (2 bytes) 0 : *** (Other states) 1 : ALarM 2 : BATtery low Status of program edit (2 bytes) 0 1 2 3 4 5 6 7 8 9 .c 14 w w w Value 22 798 : ******* (Non editing) : EDIT : SeaRCH : OUTPUT : INPUT : COMPARE : LabelSKip : OFST : WSFT : ReSTaRt APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.48 Reading an Operator Message [Description] An CNC operator message displayed on the CNC screen can be read. An operator message consists of up to 256 characters.

If the specified message is not found, –1 is output for the message No. om [Input data structure] Top address + 0 (Function code) 83 2 r.c (Completion code) — (Need not be set) 4 ce nt e (Data length) — (Need not be set) 6 (Data number) — (Need not be set) 8 (Data attribute) 0 (Always set to 0) 10 nc X (Data area) — (Need not be set) w w w .c 268 799 X B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 0 : The operator message has been read normally. 4 : A value other than 0 was specified for the data attribute. 6 : The option has not been added. [Output data structure] Top address + 0 2 om (Function code) 83 (Completion code) ? (See the explanation of the completion codes.) r.c 4 (Data length) L L : 5 to 257 6 ce nt e (Data number) — (Input data) 8 (Data attribute) — (Input data) 10 Operator message No. 2000 to 2099 or –1 Number of characters of the message 0 to 257 nc 12 14 .c Operator character string Code of the first character: CH(1) 15 w Code of the third character: CH(3) X w w ASCII character string (The null code (¥00) is always appended to the end.) Code of the second character: CH(2) 16 X Code of the Nth character: CH(N) 267 Value Code of the last character: NULL 800 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.49 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. Top address + 0 (Function code) 59 +2 r.c (Completion code) — (Need not be set) om [Input data structure] +4 ce nt e (Data length) — (Need not be set) +6 (Data number) N (P–code macro variable number) + 10 (Data attribute) — (Need not be set) nc + 12 (Data area) — (Need not be set) .c + 18 w w w CAUTION The 'Data number' occupies 4 bytes instead of 2 bytes of usual data structure. 801 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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 ("0.0000001 – "99999999). Top address + 0 (Function code) 59 +2 (Completion code) ? (See the explanation above) +4 +6 ce nt e (Data length) 6 r.c [Output data structure] om 6 : No option, or no Macro ROM module. (Data number) N (Same as input data) + 10 (Data attribute) — (Same as input data) Value of P–code macro variable (4bytes) nc + 12 Signed binary (Minus number is represented by 2's complemental) Figures after decimal point of the value Signed binary (2bytes) (Minus number is represented by 2's complemental) w w w .c + 16 Value 802 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.50 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. (Function code) 60 +2 (Completion code) — (Need not be set) +4 +6 ce nt e (Data length) 6 r.c Top address + 0 om [Input data structure] (Data number) N (P–code macro variable number) + 10 (Data attribute) — (Need not be set) Value of P–code macro variable (4bytes) nc + 12 CAUTION The 'data number' occupies 4 bytes instead of 2 bytes of usual data structure. w w w Signed binary (Minus number is represented by 2's complemental) Figures after decimal point of the value Signed binary (2bytes) (Minus number is represented by 2's complemental) .c + 16 Value 803 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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. [Output data structure] Top address + 0 (Function code) 60 +2 r.c (Completion code) ? (See the explanation above) om 6 : No option, or no Macro ROM module. +4 ce nt e (Data length) 6 (Same as input data) +6 (Data number) N (Same as input data) + 10 (Data attribute) — (Same as input data) + 12 nc Value of P–code macro variable (4bytes) Figures after decimal point of the value (2bytes) w w w .c + 16 804 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.51 [Description] This function gets the Tool life counter type of specified tool group in the Tool life management data. (M series only) [Input data structure] Top address + 0 r.c (Function code) 160 om Reading the Tool Life Management Data (Tool Life Counter Type) (Not available for Power Mate–D/F, Series 21–TA) +2 (Completion code) — (Need not be set) +4 ce nt e (Data length) — (Need not be set) +6 (Data number) N (N = Tool group number) +8 (Data attribute) — (Need not be set) nc + 10 (Data area) — (Need not be set) NOTE 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. w w w .c + 12 805 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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] +2 (Completion code) ? (See the explanation above) +4 (Data length) 2 +6 +8 ce nt e (Data number) N (Same as input data) (Data attribute) — (Same as input data) Tool life counter type (2bytes) w w w .c nc + 10 r.c (Function code) 160 om Top address + 0 806 Value 0 : No counter type 1 : Frequency 2 : Real time (in minutes) APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.52 [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. om Registering the Tool Life Management Data (Tool Group) (:Low–Speed response) (Not available for Power Mate–D/F, Series 21–TA) [Input data structure] Top address + 0 +2 +4 ce nt e (Completion code) — (Need not be set) r.c (Function code) 163 (Data length) 8 +6 (Data number) — (Need not be set) +8 (Data attribute) M (M = Tool number) Tool group number (2bytes) nc + 10 + 12 w w w Unsigned binary 1 to 512 Tool life counter type (2bytes) 1 : Frequency 2 : Real time in minutes Length of Tool life (4bytes) Unsigned binary 1 to 9999 (Frequency) 1 to 4300 (Real time in minutes) .c + 14 Value [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. 807 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Output data structure] Top address + 0 (Function code) 163 +2 (Completion code) ? (See the explanation above) (Data length) 8 (Same as input data) +6 (Data number) — (Same as input data) r.c +8 om +4 (Data attribute) M (Same as input data) + 10 ce nt e Tool group number (2bytes) (Same as input data) + 12 Tool life counter type (2bytes) (Same as input data) w w w .c nc Length of Tool life (4bytes) (Same as input data) 808 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.53 [Description] [Input data structure] Top address + 0 (Function code) 164 +2 +4 ce nt e (Completion code) — (Need not be set) om This function sets the length of Tool life of the specified Tool group in the Tool life management data. r.c Writing the Tool Life Management Data (Tool life) (:Low–Speed response) (Not available for Power Mate–D/F, Series 21–TA) (Data length) 4 +6 (Data number) N (N = Tool group number) +8 nc (Data attribute) — (Need not be set) Length of Tool life (4bytes) w w w .c +10 809 Value Unsigned binary 1–9999 (Frequency) 1–4300 (Real time in minutes) B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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. Top address +0 (Function code) 164 +2 +4 ce nt e (Data length) 4 (Same as input data) r.c (Completion code) ? (See the explanation above) om [Output data structure] +6 (Data number) N (Same as input data) +8 (Data attribute) — (Same as input data) Length of Tool life (4bytes) w w w .c nc +10 810 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.54 [Description] [Input data structure] Top address + 0 (Function code) 165 +2 +4 ce nt e (Completion code) — (Need not be set) om This function sets the Tool life counter in the specified Tool group in the Tool life management data.

r.c Writing the Tool Life Management Data (Tool Life Counter) (:Low–Speed response) (Not available for Power Mate–D/F, Series 21–TA) (Data length) 4 +6 (Data number) N (N = Tool group number) +8 nc (Data attribute) — (Need not be set) Length of Tool life (4bytes) w w w .c +10 811 Value Unsigned binary 1–9999 (Frequency) 1–4300 (Real time in minutes) B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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. Top address + 0 (Function code) 165 +2 r.c (Completion code) ? (See the explanation above) om [Output data structure] +4 ce nt e (Data length) 4 (Same as input data) +6 (Data number) N (Same as input data) +8 (Data attribute) — (Same as input data) Length of Tool life (4bytes) w w w .c nc +10 812 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.55 [Description] [Input data structure] Top address + 0 (Function code) 166 +2 +4 ce nt e (Completion code) — (Need not be set) om This function sets the Tool life counter type of specified Tool group in the Tool life management data. (M series only) r.c Writing the Tool Life Management Data (Tool Life Counter Type) (:Low–Speed Response) (Not available for Power Mate–D/F, Series 21–TA) (Data length) 2 +6 (Data number) N (N = Tool group number) +8 nc (Data attribute) — (Need not be set) Tool life counter type (2bytes) w w w .c +10 813 Value 1 : Frequency 2 : Real time in minutes B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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. [Output data structure] Top address + 0 (Function code) 166 +2 r.c (Completion code) ? (See the explanation above) om 6 : No option for Tool life management. +4 ce nt e (Data length) 2 (Same as input data) +6 (Data number) N (Same as input data) +8 (Data attribute) — (Same as input data) Tool life counter type (2bytes) w w w .c nc +10 814 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.56 [Description] [Input data structure] Top address + 0 (Function code) 167 +2 ce nt e (Completion code) — (Need not be set) +4 om This function sets the Tool length offset number of the specified Tool group in the Tool life management data. (M series only) r.c 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) (Data length) 4 +6 (Data number) N (N = Tool group number) +8 nc (Data attribute) M (M = Tool number) Tool length offset number (4bytes) w w w .c +10 815 Value Unsigned binary 1–255 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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. [Output data structure] Top address + 0 r.c (Function code) 167 om 6 : No option for Tool life management. +2 (Completion code) ? (See the explanation above) ce nt e +4 (Data length) 4 (Same as input data) +6 (Data number) N (Same as input data) +8 (Data attribute) M (Same as input data) nc +10 NOTE The effective value for Tool length offset number depends on Tool compensation number available on NC. w w w .c Tool length offset number (4bytes) 816 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.57 [Description] [Input data structure] Top address + 0 ce nt e (Function code) 168 +2 om 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) r.c 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) (Completion code) — (Need not be set) +4 (Data length) 4 +6 nc (Data number) N (N = Tool group number) +8 (Data attribute) M (M = Tool operation sequence number) Tool length offset number (4bytes) w w w .c +10 817 Value Unsigned binary 1–255 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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. [Output data structure] Top address + 0 (Function code) 168 +2 r.c (Completion code) ?

(See the explanation above) om 6 : No option for Tool life management. +4 ce nt e (Data length) 4 (Same as input data) +6 (Data number) N (Same as input data) +8 (Data attribute) M (Same as input data) Tool length offset number (4bytes) .c nc +10 w w w NOTE The effective value for Tool length offset number depends on Tool compensation number available on NC. 818 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.58 [Description] [Input data structure] Top address + 0 (Function code) 169 +2 ce nt e (Completion code) — (Need not be set) +4 om This function sets the Cutter compensation number of the specified Tool group in the Tool life management data. (M series only) r.c 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) (Data length) 4 +6 (Data number) N (N = Tool group number) +8 nc (Data attribute) M (M = Tool number) Cutter compensation number (4bytes) Unsigned binary 1–255 w w w .c +10 819 Value B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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.

[Output data structure] Top address + 0 r.c (Function code) 169 om 6 : No option for Tool life management. +2 (Completion code) ? (See the explanation above) ce nt e +4 (Data length) 4 (Same as input data) +6 (Data number) N (Same as input data) +8 (Data attribute) M (Same as input data) nc +10 NOTE The effective value for Cutter compensation number depends on Tool compensation number available on NC. w w w .c Cutter compensation number (4bytes) 820 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.59 [Description] om 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) [Input data structure] Top address + 0 +2 ce nt e (Function code) 170 r.c 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) (Completion code) — (Need not be set) +4 (Data length) 4 +6 nc (Data number) N (N = Tool group number) +8 .c (Data attribute) M (M = Tool operation sequence number) Cutter compensation number (4bytes) Unsigned binary 1–255 w w w +10 821 Value B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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. [Output data structure] Top address + 0 (Function code) 170 +2 r.c (Completion code) ? (See the explanation above) om 6 : No option for Tool life management. +4 ce nt e (Data length) 4 (Same as input data) +6 (Data number) N (Same as input data) +8 (Data attribute) M (Same as input data) Cutter compensation number (4bytes) .c nc +10 w w w NOTE The effective value for Cutter compensation number depends on Tool compensation number available on NC. 822 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.60 [Description] This function sets the Tool condition of the specified Tool group in the Tool life management data.(M series only) om Writing the Tool Life Management Data (Tool Condition (1) : Tool Number) (:Low–Speed Response) (Not available for Power Mate–D/F, Series 21–TA) [Input data structure] (Function code) 171 +2 +4 ce nt e (Completion code) — (Need not be set) r.c Top address + 0 (Data length) 2 +6 (Data number) N (N = Tool group number) +8 nc (Data attribute) M (M = Tool number) Tool condition (2bytes) w w w .c +10 823 Value 1 : Tool state clear 2 : Tool state skip B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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. om 6 : No option for Tool life management. [Output data structure] Top address + 0 r.c (Function code) 171 +2 (Completion code) ? (See the explanation above) ce nt e +4 (Data length) 2 (Same as input data) +6 (Data number) N (Same as input data) +8 (Data attribute) M (Same as input data) .c nc +10 Tool condition (2bytes) command before call after call clear skip skip consumed (#) usable (#) in use (*) usable ( ) (@) ( ) SKIP unused in use consumed ( ) (@) (*) (#) (#) (*) w w w This function changes Tool condition as below. 824 skip skip skip APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.61 [Description] [Input data structure] Top address + 0 (Function code) 172 +2 ce nt e (Completion code) — (Need not be set) +4 om This function sets the Tool condition of the Tool of the specified Tool operation sequence number in the Tool life management data. r.c 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) (Data length) 2 +6 (Data number) N (N = Tool group number) +8 nc (Data attribute) M (M = Tool operation sequence number) Tool condition (2bytes) w w w .c +10 825 Value 1 : Tool state clear 2 : Tool state skip B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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.

[Output data structure] Top address + 0 (Function code) 172 +2 r.c (Completion code) ? (See the explanation above) om 6 : No option for Tool life management. +4 ce nt e (Data length) 2 (Same as input data) +6 (Data number) N (Same as input data) +8 (Data attribute) M (Same as input data) Tool condition (2bytes) nc +10 w w w .c This function changes Tool condition as shown in B.4.60. 826 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.62 [Description] This function registers a tool to the specified Tool group in the Tool life management data. om Writing the Tool Life Management Data (Tool Number) (:Low–Speed Response) (Not available for Power Mate–D/F, Series 21–TA) [Input data structure] (Function code) 173 +2 +4 ce nt e (Completion code) — (Need not be set) r.c Top address + 0 (Data length) 4 +6 (Data number) N (N = Tool group number) +8 nc (Data attribute) M (M = Tool operation sequence number) Tool number (4bytes) w w w .c +10 827 Value Unsigned binary 1–9999 B.

WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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. [Output data structure] Top address + 0 (Function code) 173 +2 r.c (Completion code) ? (See the explanation above) om 6 : No option for Tool life management.

+4 ce nt e (Data length) 4 (Same as input data) +6 (Data number) N (Same as input data) +8 (Data attribute) M (Same as input data) Tool number (4bytes) w w w .c nc +10 828 APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) B.4.63 Reading the Estimate Disturbance Torque Data Power Mate FS20 FS18 FS16 Δ Δ Δ : The support is decided by CNC series : No support om (1) servo axis [Description] [Input data structure] Top address + 0 +2 ce nt e (Function code) 211 r.c The load torques except a necessary torque for acceleration/ deceleration of the digital servo axis are read. (Completion code) – (Need not to be set) +4 (Data length) – (Need not to be set) +6 nc (Data number) 0 +8 .c (Data attribute) M (M=1 to n or – 1) +10 w (Data area) – (Need not to be set) w w +12 829 M=1 to n : Estimate disturbance torque data for specific axis. "n" is the axis number. M= – 1 : Estimate disturbance torque data for all axes. B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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.

(Function code) 211 +4 +6 (Completion code) ? (See the explanation of the completion codes.) (Data length) L (L=2 n, n is the number of axes specified.) ce nt e +2 r.c Top address + 0 om [Output data structure] (Data number) 0 +8 (Data attribute) M (M : Input data) +10 Estimate disturbance torque data forthe controlled axis specified (2 bytes) nc +12 Value (A negative value is represented in 2's complement.) .c When the number of controlled axes is 4 w +10 w w +12 +14 +16 +18 Estimate disturbance torque data forfirst axis (2 bytes) Estimate disturbance torque data forsecond axis (2 bytes) Estimate disturbance torque data forthird axis (2 bytes) Estimate disturbance torque data forfourth axis (2 bytes) 830 Value Signed binary (A negative value is represented in 2's complement.) APPENDIX B–61863E/10 B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) (2) spindle axis [Description] The load torques except a necessary torque for acceleration/ deceleration of the serial spindle axis are read. [Input data structure] Top address + 0 om (Function code) 211 +2 +4 (Data length) — (Need not to be set) +6 +8 ce nt e (Data number) 1 (Data attribute) M (M=1 to n or – 1) +10 (Data area) – (Need not to be set) w w w .c nc +12 r.c (Completion code) — (Need not to be set) 831 M=1 to n : Estimate disturbance torque data for specific axis. "n" is the axis number. M= – 1 : Estimate disturbance torque data for all axes. B. WINDOW FUNCTION DESCRIPTION (EXCEPT FS 15B PMC–NB/NB2) APPENDIX B–61863E/10 [Completion codes] 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] om Top address + 0 (Function code) 211 +4 (Completion code) ? (See the explanation of the completion codes.) (Data length) L (L=2 n, n is the number of axes specified.) ce nt e +6 r.c +2 (Data number) 1 +8 +10 Value Estimate disturbance torque data forthe controlled axis specified (2 bytes) Signed binary (A negative value is represented in 2's complement.) nc +12 (Data attribute) M (M : Input data) .c When the number of controlled axes is 2 Estimate disturbance torque data for first axis (2 bytes) +12 Estimate disturbance torque data forsecond axis (2 bytes) w w w +10 Value Signed binary (A negative value is represented in 2's complement.) [supporting soft] CNC FS16 : FS18 : B005 SERIES Edition K or later B105 SERIES Edition H or later B205 SERIES Edition H or later BD03 SERIES Edition L or later BE03 SERIES Edition I or later BF03 SERIES Edition I or later SERVO : 9060 SERIES Edition J or later SPINDLE : 9A50 SERIES Edition Q or later 832 APPENDIX B–61863E/10 C C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.1 FUNCTION om This window function is a functional instruction by which the data on the CNC is read or is written. Option (A02B–0162–J917) of NC window is necessary. 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. ce nt e LOW–SPEED RESPONSE AND HIGH–SPEED RESPONSE OF WINDOW FUNCTION r.c C.2 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. nc Moreover, option (A02B–0162–J984) of NC window B is necessary to read (the processing of) the tool offset data, the tool life management data and machining time. w w w .c NOTE 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 833 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 CAUTION There is a version which does not support in the reading or writing of the window data by a new format. ROM VERSION A–E F– 4078 A– It does not support a new format. Please use bit 4 of NC parameter 7401 as 0. It supports a new form.

When the window function of a new format is used, used please set bit 4 of NC parameter 7401 as 1. om 4047 CONTENT Function that is effected by bit 4 of NC parameter 7401. 7401#4 0 The data of tool life management for 128 sets of tools can be read and written. 1 The data of tool life management for 512 sets of tools can be read and written. ce nt e Tool life management data CONTENT r.c FUNCTION Tool offset data according to the specified tool number 0 This function can not be used. 1 The tool offset data can be read and written. w w w .c nc NOTE Functions except the above–mentioned are not related to bit 4 of NC parameter 7401. If there is no option of the corresponding function, window instructions can not be used. 834 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.2.1 Functional Instruction WINDR 0 Function code 2 Completion code 4 CTL1 CTL2 DATA Data length 6 Data number 8 Axis specification 10 * Data length Depends on the function Data area Type of processing X r.c X Type of control data CTL0 CTL1 CTL2 ce nt e Data type om CTL0 part1 Data length (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 (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 w (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 Number 2 byte 2 byte w .c Actual velocity for controlled axes w nc Tool offset data (G function) (other than G function) 835 C.

WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 part2 Data type Type of processing Tool life management data Tool group No. Number of tool groups Number of tools 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) (low) (low) 38 39 40 41 42 160 43 44 45 46 47 48 49 0 Tool group No. Tool group No. Tool group No. Tool group No. Tool group No. Tool group No. Tool group No. Tool group No. Tool group No. Tool group No. Tool No. 0 0 0 0 0 Tool No. Tool order number Tool No. Tool order number Tool No. Tool order number Tool order number 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte Clock data (low) 151 Data format 0 6 byte Relative position of controlled axes (high) 74 0 Remaining travel of controlled axes (high) 75 0 Estimate disturbance torque data of a digital Estimate disturbance torque data of a serial spindle (high) 211 0 (high) 211 1 Axis number 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 Type of control data CTL2 r.c om CTL1 Axis number 4 byte Axis number 4 byte Axis number 2 byte w w w .c nc ce nt e CTL0 836 Data length C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.2.2 Functional Instruction WINDW 0 Function code 2 Completion code 4 CTL1 CTL2 DATA Data length 6 Data number 8 Axis specification 10 Data area * Data length Depends on the function Type of control data CTL0 CTL1 CTL2 ce nt e Type of processing Y r.c Y Data type om CTL0 Data 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 (low) 150 Data type 0 2 byte (low) 150 201 0 2 byte (low) 152 0 Axis number 2 byte Data on the program check screen Spindle tool number Number of the tool to be used next nc Torque limit override (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 group No. Tool group No. Tool group No. Tool group No. Tool group No. Tool group No. Tool group No. Tool group No. Tool group No. 0 0 0 Tool No. Tool order number Tool No. Tool order number Tool order number 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte 4 byte (low) 214 Data format Tool number 4 byte 215 0 0 6 byte 215 Axis specification mode Axis number 8 byte 216 0 0 6 byte w w .c 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. Tool offset data according to the specified tool number w Superposition move command (for three axes) Superposition move command (for four axes) Feedrate (high) (high) 837 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) B–61863E/10 (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 om –10 Meaning 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. ce nt e r.c C.3 FORMAT AND DETAILS OF THE CONTROL DATA OF THE WINDR FUNCTIONAL INSTRUCTION APPENDIX An error occurred. It is a mistake of data form. Though the function supports only a new format, a old format is specified. w w w .c nc 5 838 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.1 The tool offset value (tool

compensation) is read from the CNC. Read data 0 Function code 13 2 2 Completion code — 4 Function code 13 Completion code 4 Data length — 6 Data length 4 6 Offset number Offset number 8 8 Offset format Offset format 10 10 14 w w w .c nc 14 Tool offset value (4 byte) ce nt e Data area (4 byte) r.c Set data 0 [Description] om Reading a Tool Offset (Low–Speed Response) 839 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX (Note 1) Offset format M series (machining center

system) Tool compensation B Geometry compensation Wear compensation 1 Offset number 1 Offset number 1 Offset number +1000 1 Offset number 1 Offset number +1000 2 Offset number 2 Offset number +1000 Data type Tool compensation A Compensation along the X–axis Compensation along the Z–axis Tool–tip radius compensation Compensation along the Y–axis Compensation related to the position of the virtual tool Tool compensation B Geometry compensation Compensation along the X–axis

Compensation along the Z–axis Tool–tip radius compensation Compensation along the Y–axis Wear compensation Compensation along the X–axis Compensation along the Z–axis Tool–tip radius compensation Compensation along the Y–axis Compensation related to the position of the virtual tool Format Offset number (CTL+2, 3) 1 Offset number 2 Offset number 3 Offset number 4 Offset number 5 Offset number Offset number 1 Offset number 2 Offset number 3 Offset number 4 Offset number 1 Offset number

+1000 Offset number +1000 Offset number +1000 Offset number +1000 Offset number w w w .c nc ce nt e Tool compensation C Tool length Geometry compensation Wear compensation Cutter Geometry compensation Wear compensation Offset number (CTL+2, 3) om Tool compensation A Compensation Format T series (lathe system) r.c Data type B–61863E/10 840 2 3 4 5 APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) 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. Set data 0 Read data 0 Function code 15 2 2 Completion code — Completion code 4 Data length — 6 ce nt e Data number 8 Axis number Axis number 10 10 Workpiece origin offset value (4 byte) Data area (4 byte) 14 w w w .c nc 14 Data length 4 6 Data number 0 8 Function code 15 r.c 4 om 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. 841 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.3 Set data 0 [Description] A parameter of the CNC is read. Read data 0 Function code 17 2 2 Completion code — 4 Completion code 4 Data length — 6 Parameter number Parameter number 0 : No axis 8 8 Data attribute Data attribute 14 10 ce nt e 1 to n : A specified axis Data

area (4 byte) Data length 4 r.c 6 10 Function code 17 om Reading a Parameter (Setting Data) (Low–speed response) Parameter data (4 byte) 14 (Note) Format of parameter data If the data is one byte long, it is set in the DATA+0 area. 10 w w w .c nc 11 842 12 13 14 DATA+0 DATA+1 DATA+2 DATA+3 APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.3.4 Reading a Custom Macro Variable (Low–speed response) [Description] A custom macro variable is read from the

CNC. Set data 0 Read data 0 Function code 21 2 2 Completion code — 8 Completion code r.c 6 4 Data length — 6 Custom macro variable number 10 Custom macro variable value (4 byte) 14 Position of decimal point (2 byte) nc 16 Custom macro variable number Data attribute Data area (4 byte) 14 Data length 6 8 Data attribute 0 10 Function code 21 ce nt e 4 om CAUTION The position of the decimal point must be specified beforehand. Position of decimal point (2 byte) 16 w w .c CAUTION 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”. w Examples The relationship between the read value and the stored variable is: (Read value) = (Custom macro variable in the NC) Read value 1 12 123 1234 12340 843 10(Position of decimal point) Custom macro variable in the NC 1. 234 Position of decimal point 0 1 2 3 4 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.5 Reading the CNC

Alarm Status (Low–speed response) If the CNC is in the alarm state, the details of the alarm are read. Read data 0 Function code 23 2 2 Completion code — 8 10 4 Data length — 6 Data number 0 Data number 8 Data attribute 0 Data attribute 10 Data area (2 byte) 12 Data length 2 r.c 6 Completion code ALARM 1 ce nt e 4 Function code 23 om Set data 0 [Description] 11 ALARM 2 12 The following alarm states can be read: ALARM 1 Data format 7 6 5 4 3 2 1 0 ALARM 2 Data format 7 6 5 4 3 2 1 0 w w w .c

nc #0 : Background P/S alarm #1 : Foreground P/S alarm #2 : Overheat #3 : Sub–CPU error #4 : Excessive synchronization error #5 : Parameter enable switch enabled #6 : Overtravel #7 : PC error 844 #0 : External alarm message #1 : Not defined #2 : P/S alarm preventing the processing from continuing (serious alarm) #3 : Not defined #4 : Servo alarm #5 : I/O error #6 : Parameter input requiring the power to be turned off #7 : System error C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–

NB/NB2) APPENDIX B–61863E/10 C.3.6 Reading the Current Program Number (Low–speed response) [Description] Set data 0 Read data 0 Function code 24 2 Function code 24 2 Completion code — Completion code 4 Data length — 6 6 Data number 0 8 Data number 8 Data attribute 0 Data attribute ce nt e 10 Data length 6 r.c 4 om The number of a running machining program is read from the CNC. 10 Data area (6 byte) Program number 14 15 16 FLAG 1 FLAG 2 16 FLAG 2 Data format 7 6 5 4 3 2 1 0 nc

FLAG 1 Data format 7 6 5 4 3 2 1 0 w w w .c #0 to 3 : Number of decimal places #4 : Whether a decimal point is provided (0: Not provided, 1: Provided) #5 : Positive or negative (0: Positive, 1: Negative) #6 : Whether only one digit is provided (0: Not only one, 1: Only one) #7 : Not defined 845 #3 : Whether at least one address in the range of D to the second miscellaneous function is specified after the NC is reset (0: Not specified, 1: Specified) #7 : Whether the read data is specified in the current block of

the part program (0: Not specified, 1: Specified) Other bits are not defined. C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.7 Set data 0 [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. Read data 0 Function code 25 2 Function code 25 2 Completion code — Completion code 4 Data length — 6 6 Data number 0 Data attribute 0 10 Data number 8 Data attribute ce nt e 8 Data length 6 r.c 4 om Reading the Current Sequence Number (Low–speed response) 10 Data area (6 byte) Sequence number (4 byte) 14 15 16 FLAG 1 FLAG 2 16 nc FLAG 1 Data

format 7 6 5 4 3 2 1 0 w w w .c #0 to 3 : Number of decimal places #4 : Whether the decimal point is provided (0: Not provided, 1: Provided) #5 : Positive or negative (0: Positive, 1: Negative) #6 : Whether only one digit is provided (0: Not only one, 1: Only one) #7 : Not defined 846 FLAG 2 Data format 7 6 5 4 3 2 1 0 #3 : Whether at least one address in the range of D to the second miscellaneous function is specified after the NC is reset (0: Not specified, 1: Specified) #7 : Whether the read data is specified

in the current block of the part program (0: Not specified, 1: Specified)Other bits are not defined.

APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.3.8 Set data 0 [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.

Read data 0 Function code 26 2 2 Completion code — Completion code 4 Data length — 6 Data number 8 ce nt e 10 Data attribute 0 Data attribute 10 Data area (4 byte) Feedrate (4 byte) 14 nc 14 Data length 4 6 Data number 0 8 Function code 26 r.c 4 om Reading the Actual Velocity of Controlled Axes (Low–speed response) C.3.9 [Description] The absolute position (absolute coordinates) on a feed axis controlled by the CNC is read. .c Reading the Absolute Position on a Controlled Axis w w w Set data 0 Read

data 0 Function code 27 2 2 Completion code — 4 Completion code 4 Data length — 6 Data length 4 6 Data number 0 8 Function code 27 Data number 8 Axis number 0 Axis number 10 10 Data area (4 byte) Absolute position (4 byte) 14 14 847 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 Data specification Kind of data 1) Indicates the current position in the workpiece coordinate system. The current position is calculated by the following simple expression. The

read current position is stored in the 4–byte area of DATA+0 to DATA+3. Current position = machine coordinate value – workpiece offset value The workpiece offset value can be obtained by summing up the folloing offset. (1) External workpiece origin offset (2) Workpiece origin offset (G54 to G59,G54.1Pp) (3) Workpiece coordinate system (G92) (4) Local coordinate system (G52) om Current position Data specification y r.c Local coordinate system (G52) Workpiece origin offset (G54 or after) ce nt e External

workpiece origin offset Workpiece coordinate system (G92) x Origin of the machine coordinate system w w w .c 3) 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. The present position of a moving axis can be read whenever the function instruction is executed. nc 2) 848 APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.3.10 Set data 0 [Description] The machine position (machine coordinates) on a feed axis controlled by the CNC is read. Read data 0

Function code 28 2 2 Completion code — Completion code 4 Data length — 6 Data number 8 8 Data attribute ce nt e Data attribute 10 Data area (4 byte) 14 Data length 4 6 Data number 0 10 Function code 28 r.c 4 om Reading the Machine Position (Machine Coordinates) of Controlled Axes Feedrate (4 byte) 14 1) The unit of the read value is determined as follows: nc (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. w w w .c (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. 849 C.

WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.11 Set data 0 [Description] The absolute coordinates of the skip position specified in the CNC are read. Read data 0 Function code 29 2 2 Completion code — Completion code 4 Data length — 6 ce nt e Data number 8 Axis number Axis number 10 10 Data area (4 byte) 14 Data length 4 6 Data number 0 8 Function code 29 r.c 4 om Reading a Skip Position (Stop Position of Skip Operation (G31)) of Controlled Axes

(Low–speed response) Skip position (4 byte) 14 nc 1) The unit of the read value is determined as follows: w w w .c (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) 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. 850 APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.3.12 Set data 0

[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. Read data 0 Function code 30 2 2 Completion code — 4 Completion code 4 Data length — 6 Data number 0 Data number 8 8 Data attribute Data attribute ce nt e 10 Data area (4 byte) 14 C.3.13 Servo delay (4 byte) 14 nc Reading the Acceleration/ Deceleration Delay on Controlled Axes Data length 4 r.c 6 10 Function code 30 om Reading the Servo Delay

for Controlled Axes [Description] .c 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. w Set data 0 Read data 0 Function code 31 w w 2 4 Function code 31 2 Completion code — Completion code 4 Data length — 6 Data length 4 6 Data number 0 Data number 8 8 Axis number Axis number 10 10 Data area (4 byte) Acceleration/ deceleration delay (4 byte) 14 14

851 C. WINDOW FUNCTION DESCRIPTION C.

(FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.14 Reading Modal Data (Low–speed response) [Description] The continuous–state data is read from the CNC. om 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 Read data 0 Function code 32 Function code 32 r.c Set data 0 2 2 Completion code — 4 Completion code 4 Data length 2 ce nt e Data length — 6 6 Data type 0 : Previous block 8 Data type 0 – : Each data for G function 8 Specified block Specified block 1 : Current block 10 Data area (2 byte) Modal data 12 w w w .c nc 12 10 2 : Next block 852 (Note) Format of the continuous–state data 7 6 5 4 3 2 1 0 DATA+0 #0 to #6 : Continuous–state data #7 : Specified block 1= Current block 0= Previous block

C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 (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.

Function code 32 2 2 Completion code — 4 Completion code 4 Data length — 6 6 Data type 0 : Previous block 8 Specified block Specified block 1 : Current block 10 10 Modal data (4 byte) ce nt e 2 : Next block 14 15 16 Data length 6 Data type 24 – : All data for G function 8 Data area (6 byte) Function code 32 om Read data 0 r.c Set data 0 FLAG 1 FLAG 2 16 FLAG 1 Data format 7 6 5 4 3 2 1 0 FLAG 2 Data format 7 6 5 4 3 2 1 0 w w w .c nc #0 – 3 : Number of decimal places #4 : Whether the decimal point is

provided (0: Not provided, 1: Provided) #5 : Positive or negative (0: Positive, 1: Negative) #6 : Whether only one digit is provided (0: Not only one, 1: Only one) #7 : Not defined 853 #3 : Whether at least one address in the range of D to the second miscellaneous function is specified after the NC is reset (0: Not specified, 1: Specified) #7 : Whether the read data is specified in the current block of the part program (0: Not specified, 1: Specified) Other bits are not defined. C. WINDOW FUNCTION

DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 (3) Data specification Kind of data Modal data Data specification 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 necessary 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. 00 G00 G01 G02 G03 G33 — — — 01 G17 G18 G19 G code for lathe system (G code system B) G00 G01 G02 G03 G33 G77 G78 G79 Code fetched in DATA + 0 om G code for machining center system r.c Specified number in CTL1 (kinds of data) 0 1 2 3 4 8 9 10 0 1 2 G90 G91 G90 G91 1 0 G22 G23 G22 G23 0 1 G93 G94 G95 — G94 G95 2 0 1 G20

G21 G20 G21 1 0 06 G40 G41 G42 G40 G41 G42 0 1 2 07 G43 G44 G49 — — — 1 2 0 G80 G81 G82 G83 G84 G85 G86 G87 G88 G89 G73 G74 G76 G80 G81 G82 G83 G84 G85 G86 G87 G88 G89 G83.1 G84.1 G86.1 0 1 2 3 4 5 6 7 8 9 10 11 12 09 G98 G99 G98 G99 0 1 10 G50 G51 — — 0 1 02 03 04 .c nc 05 ce nt e G97 G96 — w w w 08 854 APPENDIX B–61863E/10 Kind of data C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) 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 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 15 G69 G68 — r.c 0 1 2 G15 G16 — — 0 1 G50.1 G51.1 G50.1 G51.1 0 1 w w w .c nc 17 1 2 3 0 G17 G18 G19 ce nt e 16 om 11 855 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX Kind of data B–

61863E/10 Data specification Modal data Table — 1 of G code system for a lathe system G code system *1) F nction Function C G00 G01 G02 G03 G04 G07 G09 G10 G10.1 G11 G17 G18 G19 G20 G21 G22 G23 G27 G28 G29 G30 G31 G32 G34 G35 G36 G00 G01 G02 G03 G04 G07 G09 G10 G10.1 G11 G17 G18 G19 G70 G71 G22 G23 G27 G28 G29 G30 G31 G32 G34 G35 G36 G37 G37 G37 G37.1 G37.2 G37.3 G40 G41 G42 G50 G37.1 G37.2 G37.3 G40 G41 G42 G92 G37.1 G37.2 G37.3 G40 G41 G42 G92 G50.1

G51.1 G52 G53 G54 G55 G56 G57 G58 G59 G61 G62 G64 G65 G66 G66.1 G67 G68 G69 G70 G71 G50.1 G51.1 G52 G53 G54 G55 G56 G57 G58 G59 G61 G62 G64 G65 G66 G66.1 G67 G68 G69 G70 G71 G50.1 G51.1 G52 G53 G54 G55 G56 G57 G58 G59 G61 G62 G64 G65 G66 G66.1 G67 G68 G69 G72 G73 nc .c w w w Positioning Linear interpolation Circular interpolation CW Circular interpolation CCW Dwell Hypothetical axis interpolation Exact stop Data setting PC data setting Data setting mode cancel

XpYp plane selection Xp: X axis or its parallel axis ZpXp plane selection Yp: Y axix or its parallel axis YpZp plane selection Zp: Z axis or its parallel axis Inch input Metric input Stored stroke check on Stored stroke check off Reference point return check Reference point return Return from reference point Return to 2nd, 3rd, 4th reference point Skip function Thread cutting Variable lead thread cutting Circular thread cutting CW Circular thread cutting CCW or automatic tool compensation (X axis) Automatic

tool compensation #1 or automatic tool compensation (Z axis) Automatic tool compensation #1 Automatic tool compensation #2 Automatic tool compensation #3 Tool nose radius compensation cancel Tool nose radius compensation left Tool nose radius compensation right Work coordinates change/maximum spindle spped setting Programmable mirror image cancel Programmable mirror image Local coordinate system setting Machine coordinate system selection Work coordinate system 1 selection Work

coordinate system 2 selection Work coordinate system 3 selection Work coordinate system 4 selection Work coordinate system 5 selection Work coordinate system 6 selection Exact stop mode Automatic corner override Cutting mode Macro call Macro modal call A Macro modal call B Macro modal call A/B cancel Mirror image for double currets on Mirror image for double currets cancel Finishing cycle Stock removal in turning ce nt e G00 G01 G02 G03 G04 G07 G09 G10 G10.1 G11 G17 G18 G19 G20 G21

G22 G23 G27 G28 G29 G30 G31 G32 G34 G35 G36 om B r.c A 856 APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) Kind of data Data specification Modal data Table — 2 of G code system for a lathe system G code system *1) Stock removal in facing Pattern repeating Peck drilling Z axis Grooving in X axis Threading cycle Canned cycle for drilling cancel Drilling cycle, spot boring Drilling cycle, counter boring Peck drilling cycle Peck drilling cycle Tapping cycle Counter

tapping cycle Boring cycle Boring cycle Fine boring cycle Back boring cycle Boring cycle Boring cycle Outting cycle A Thread cutting cycle Outting cycle B Constant surface speed control Constant surface speed control Feed per minute Feed per revolution Absolute command Incremental command Canned cycle initial level return Canned cycle R point level return 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/B is selectable. Modal data other than the G function .c 2) CTL1 (kinds of data) For machining system D 24 — E 25 24 H 26 25 L 27 26 M 28 27 N 29 28 O 30 29 S 31 30 T 32 31 F 33 32 Second auxiliary function 34 33 w Modal data other than the G function (address in the part program) w w G74 G75 G76 G77 G78 G80 G81 G82 G83 G83.1 G84 G84.1 G85 G86 G86.1 G87 G88 G89 G20 G21 G24 G96 G97 G94 G95 G90 G91 G98 G99 nc *1) G72 G73 G74

G75 G76 G80 G81 G82 G83 G83.1 G84 G84.1 G85 G86 G86.1 G87 G88 G89 G77 G78 G79 G96 G97 G94 G95 G90 G91 G98 G99 ce nt e G72 G73 G74 G75 G76 G80 G81 G82 G83 G83.1 G84 G84.1 G85 G86 G86.1 G87 G88 G89 G90 G92 G94 G96 G97 G98 G99 — — — — F nction Function C om B r.c A 857 For turning system Field from which to fetch data DATA+0 to DATA+5 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.15 Set data 0 [Description] The data on

the diagnostic data screen of the CNC is read. Read data 0 Function code 33 2 2 Completion code — 4 Completion code 4 Data length — 6 6 r.c 8 Data attribute 0 10 Data attribute 0 10 Data area (2 byte) Diagnosis data (2 byte) ce nt e 12 Data length 2 Diagnosis number Diagnosis number 8 Function code 33 om Reading Diagnosis Data (Low–speed response) 12 w w w .c nc NOTE Diagnosis number The diagnosis number must fall in the range of 0 to 103 or 200 to 303. 858 C. WINDOW FUNCTION

DESCRIPTION (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.16 Set data 0 [Description] 1. The load current for an axis controlled by the CNC is converted to adigital value and the digital value is read. 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. Read data 0 Function code 34 2 om Reading A/D Conversion Data for the Feed Motor Function code 34 2 Completion code — 4 Data length — 6 6 Data number Data number 8 8 Axis

number ce nt e Axis number 10 10 A/D conversion data (2 byte) Data area (2 byte) 12 Data length 2 r.c 4 Completion code 12 nc 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.

.c Type of analog voltage input Analog input of a voltage caluculated from the load current for the axis controlled by the NC (AC servo motor only) w w w General–purpose analog input Data number Axis number 0 1 2 3 4 5 6 200 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. 859 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX a) B–61863E/10 In the case of peak current [Ao–p] of load

current is calculated. LOAD CURRENT[Ao–p] + b) (READ DATA) * 128 [Ao–p] (COEFFICIENT) In the case of ratings currents [Arms] of load current are calculated. LOAD CURRENT[Arms] + c) (READ DATA) * 128 [Arms] (COEFFICIENT) Ṅ2 In the case of percent (rate) is calculated. LOAD CURRENT[Ao–p] PEAK CURRENT OF SERVO MOTOR 100[%] om rate of load[%] + COEFFICIENT: It decides depending on the capacity of the amplifier to be used. r.c PEAK CURRENT OF SERVO MOTOR : It dicides with

the servo motor.

Examples When the AC motor model ”30s” is used and the read A/D conversion data is 150, method of calculating each load current.

ce nt e 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.

nc COEFFICIENT + 128 + 128 + 1.6 PEAK CURRENT VALUE OF AMPLIFIER 80 The peak current of the servo motor is calculated. PEAK CURRENT[Ao–p]+ (ratings currents) .c w Ṅ2 + 22.62742 + 16 8 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) w w Ṅ2 Peak current[Ao–p] of load current LOAD CURRENT[Ao–p] + (READ DATA) * 128 + 150 * 128 (COEFFICIENT) 1.6 + 13.75 [Ao–p] b) Ratings currents[Arms] of

load current RATINGS CURRENTS[Arms] + (READ DATA) * 128 + 150 * 128 1.6 Ṅ2 (COEFFICIENT) Ṅ2 + 9.72 [Arms] c) PERCENT(RATE) RATE OF LOAD[%] + LOAD CURRENT[Ao–p] PEAK CURRENT OF SERVO MOTOR + 13.75 23 860 100 + 59.8 [%] 100 APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) [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 om 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. Reading the Tool Life Management Data (Tool Group Number) (Low–speed response) Bit 0 of parameter No. 1811 = 1 (A/D conversion data is output in high–speed mode.) r.c 2 [Description] The number of the tool group in which the tool number is cataloged is read. Read

data 0 Function code 38 nc Set data 0 Bit 6 of parameter No. 1810 = 0 (A/D conversion is performed.) ce nt e C.3.17 1 2 2 Completion code — Data length — .c 4 6 Completion code 4 Data number w 8 8 Tool No. Tool No. 10 w Data length 4 6 Data number 0 w Function code 38 10 Data area (4 byte) Tool group No. (4 byte) 14 14 NOTE The data can be read only when the tool life management data function is provided. 861 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–

61863E/10 C.3.18 Set data 0 [Description] The number of tool groups contained in the tool life management data is read. Read data 0 Function code 39 2 2 Completion code — 4 4 Data length 4 r.c Data length — 6 Data number 0 Data number 8 8 Data attribute 0 Data attribute ce nt e 10 10 Data area (4 byte) 14 Number of tool groups (4 byte) nc 14 [Description] The number of tools cataloged in the specified tool group is read. .c Reading Tool Life Management Data (Number of Tools) (Low–speed response)

w w Set data 0 w Function code 39 Completion code 6 C.3.19 om Reading the Tool Life Management Data (Number of Tool Groups) (Low–speed response) Read data 0 Function code 40 2 2 Completion code — 4 Function code 40 Completion code 4 Data length — 6 Data length 4 6 Tool group No. Tool group No. 8 8 Data attribute 0 Data attribute 10 10 Data area (4 byte) Number of tools (4 byte) 14 14 862 APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.3.20 Set

data 0 [Description] The tool life of the specified tool group is read. Read data 0 Function code 41 2 2 Completion code — 4 om Reading Tool Life Management Data (Tool Life) (Low–speed response) Function code 41 Completion code 4 Data length 4 r.c Data length — 6 6 Tool group No. Tool group No. 8 8 Data attribute 0 ce nt e Data attribute 10 10 Tool life (4 byte) Data area (4 byte) 14 14 nc C.3.21 [Description] The tool life counter of the specified tool group is read. .c Reading Tool Life Management Data

(Tool Life Counter) (Low–speed response) w w w Set data 0 Read data 0 Function code 42 2 2 Completion code — 4 Function code 42 Completion code 4 Data length — 6 Data length 4 6 Tool group No. Tool group No. 8 8 Data attribute 0 Data attribute 10 10 Tool life counter (4 byte) Data area (4 byte) 14 14 863 C.

WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.22 Set data 0 [Description] The tool life counter type of the specified tool group is read. Read data 0 Function code 160 2 2 Completion code — Function code 160 Completion code 4 r.c 4 om Reading Tool Life Management Data (Tool Life Counter Type) (Low–speed response) Data length — 6 Data length — 6 Tool group No. Tool group No. 8 ce nt e 8 Data attribute 0 10 Data attribute 10 Tool life counter type (4 byte)

Data area (4 byte) 14 14 w w w .c nc Tool life counter type 1 : The tool life counter indicates the number of times the tool has been used. 2 : The tool life counter indicates the period of time the tool has been used. 864 APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.3.23 Set data 0 [Description] A tool length compensation number is read according to the specified tool group number and tool number. Read data 0 Function code 43 2 2 Completion code — Completion

code 4 Data length — 6 ce nt e Tool group No. 8 Tool No. Tool No. 10 10 Tool length compensation No. (4 byte) Data area (4 byte) 14 Data length 4 6 Tool group No. 8 Function code 43 r.c 4 om Reading Tool Life Management Data (Tool Length Compensation No.1) (Low–speed response) 14 w w w .c nc If nothing is specified after the H code, the NC transfers 255 (FFH). 865 C. WINDOW FUNCTION DESCRIPTION C.

(FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.24 Set data 0 [Description] A tool length compensation number is read according to the specified tool group number and tool order number. Read data 0 Function code 44 2 2 Completion code — Completion code 4 Data length — 6 ce nt e Tool group No. 8 Tool order number Tool order number 10 10 Tool length compensation No. (4 byte) Data area (4 byte) 14 Data length 4 6 Tool group No. 8 Function code 44 r.c 4 om Reading Tool Life Management Data

(Tool Length Compensation No.2) (Low–speed response) 14 w w w .c nc If nothing is specified after the H code, the NC transfers 255 (FFH). 866 APPENDIX B–61863E/10 C.

WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.3.25 Set data 0 [Description] A cutter compensation number is read according to the specified tool group number and tool number. Read data 0 Function code 45 2 2 Completion code — Completion code 4 Data length — 6 Tool group No. ce nt e 8 Tool No. Tool No. 10 10 Cutter compensation No. (4 byte) Data area (4 byte) 14 Data length 4 6 Tool group No. 8 Function code 45 r.c 4 om Reading Tool Life Management Data (Cutter Compensation

No.1) (Low–speed response) 14 w w w .c nc If nothing is specified after the D code, the NC transfers 255 (FFH).

867 C.

WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.26 Set data 0 [Description] A cutter compensation number is read according to the specified tool group number and tool order number. Read data 0 Function code 46 2 2 Completion code — Completion code 4 Data length — 6 Tool group No. ce nt e 8 Tool order number Tool order number 10 10 Cutter compensation No. (4 byte) Data area (4 byte) 14 Data length 4 6 Tool group No. 8 Function code 46 r.c 4 om

Reading Tool Life Management Data (Cutter Compensation No.2) (Low–speed response) 14 w w w .c nc If nothing is specified after the D code, the NC transfers 255 (FFH). 868 APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.3.27 Set data 0 [Description] The tool information (status) is read according to the specified tool group number and tool number. Read data 0 Function code 47 2 2 Completion code — 4 om Reading Tool Life Management Data (Tool Information

1) (Low–speed response) Function code 47 Completion code 4 Data length 4 r.c Data length — 6 6 Tool group No. Tool group No. 8 8 Tool No. 10 ce nt e Tool No. 10 Tool information (4 byte) Data area (4 byte) 14 14 w w w .c nc Tool information 1 : The tool is cataloged. 2 : The tool life has expired. 3 : The tool was skipped. 869 C. WINDOW FUNCTION DESCRIPTION C.

(FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.28 Set data 0 [Description] The tool information (status) is read according to the specified tool group number and tool order number. Read data 0 Function code 48 2 2 Completion code — 4 Function code 48 om Reading Tool Life Management Data (Tool Information 2) (Low–speed response) Completion code 4 Data length — 6 r.c 6 Data length 4 Tool group No. Tool group No. 8 8 Tool order number Tool order number 10 ce nt e 10 Tool information (4 byte)

Data area (4 byte) 14 14 Tool information See the description in Section 3.27. nc C.3.29 [Description] A tool number is read according to the specified tool group number and tool order number. .c Reading Tool Life Management Data (Tool Number) (Low–speed response) w w w Set data 0 Read data 0 Function code 49 2 2 Completion code — 4 Function code 49 Completion code 4 Data length — 6 Data length 4 6 Tool group No. Tool group No. 8 8 Tool order number Tool order number 10 10 Tool No. (4 byte)

Data area (4 byte) 14 14 870 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.30 The current data (year, month, day) and current time (hours, minutes, seconds) can be read from the clock built into the CNC. Read data 0 Function code 151 2 2 Completion code — 4 Function code 151 Completion code 4 Data length — 0 : Current data 6 6 Data format Data format 1 : Current time 8 8 Data attribute Data attribute 0 10 Data area (6 byte) 16 ce nt e 10 Data length 6

r.c Set data 0 [Description] om Reading Clock Data (Low–speed response) Clock data (6 byte) 16 (Note) Format of clock data The data is binary. Current date DATA+ 0 +4 Years (Example: 1992) Months (Example: 12) Days (Example: 16) +6 Current time DATA+ 0 +2 +4 w .c nc +2 w w +6 871 Hours (Example: 23) Minutes (Example: 59) Seconds (Example: 59) C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.31 Set data 0 [Description] The relative position

(relative coordinates) on a feed axis controlled by the CNC is read. Read data 0 Function code 74 2 2 Completion code — 4 Completion code 4 Data length — 6 Data number 0 Data number 8 8 Axis number Axis number ce nt e 10 Relative position (4 byte) Data area (4 byte) 14 Data length 4 r.c 6 10 Function code 74 om Reading the Relative Position on a Controlled Axis 14 1) The unit of the read value is determined as follows: nc (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. 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. w w w .c (2) When the diameter is specified for the lathe system 872 APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.3.32 Set data 0 [Description]

The remaining traveling distance on a feed axis controlled by the CNC is read. Read data 0 Function code 75 2 2 Completion code — 4 Completion code 4 Data length — 6 6 Data number 0 Data length 4 r.c Data number 8 8 Axis number Axis number 10 10 Remaining travel (4 byte) ce nt e Data area (4 byte) 14 w w w .c nc 14 Function code 75 om Reading the Remaining Travel 873 C.

WINDOW FUNCTION DESCRIPTION C.

(FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.33 Reading an Estimate Disturbance Torque Data 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. Read data 0 Function code 211 2 om Set data 0 [Description] Function code 211 2 Completion code — Completion code 4 Data length — 6 Data number 0 or 1 6 Data

number 8 Axis number ce nt e 8 0: Estimate disturbance torque data of a digital servo 1: Estimate disturbance torque data of a serial spindle Data length 2 r.c 4 Axis number 10 Data area (2 byte) 12 10 Estimate disturbance torque (2 byte) 12 Kind of data Data specification Estimate disturbance torque data Please refer to “FANUC AC SERVO AMPLIFIER AMINTENANCE MANUAL of a digital servo (B–65005E)” for correspondence of the load torque with the value of the read data. nc Estimate disteurbance

torque data Please refer to “FANUC AC SPINDLE SERVO UNIT (SERIAL INTERFACE) MAINTEof a serial spindle 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. w w w .c Load torque + The read data 16384 874 Maximum output torque of spindle APPENDIX B–61863E/10 C.

WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.3.34 Set data 0 [Description] The machining time currently specified for a program is read. Read data 0 Function code 178 2 2 Completion code — 4 Completion code 4 Data length — 6 6 Program number Data length 6 r.c Program number 8 8 Data attribute 1 Data attribute 1 10 10 Machining time (6 byte) ce nt e Data area (6 byte) 16 w w w .c nc 16 Function code 178 om Reading the Machining Time (Low–speed response) 875 C.

WINDOW FUNCTION DESCRIPTION C.

(FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.35 Set data 0 [Description] 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.”) Read data 0 Function code 153 2 2 Completion code — Completion code 4 Data length — 6 Data number 8 8 Axis number ce nt e Axis number 10 A/D conversion data (2 byte) Data area (2 byte) 12 Data length 2 6 Data number 0 10

Function code 153 r.c 4 om Reading the Load Current (A/D Conversion Data) for the Spindle Motor 12 [Hardware] When general analog input data is read, one of the following hardware items is required: Sub–CPU board 2 Analog I/O module on the additional axis board nc 1 w w w .c 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.) 876 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.3.36 Set data 0 [Description] The tool number is spedified and the tool offset data is read. Read data 0 Function

code 213 2 2 Completion code — 4 Completion code 4 Data length — 6 Data format Data format 8 8 Tool number Tool number ce nt e 10 offset data (4 byte) Data area (4 byte) 16 16 Kind of the data to be read .c nc Tool number Pot number Pot number Tool length compensation value Tool length compensation value Cutter compensation value Cutter compensation value The data form CTL1 01 10 11 20 21 30 31 Tool number CTL2 Tool display number Tool No. Tool display number Tool No. Tool display

number Tool No. 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) w w w Data length 4 r.c 6 10 Function code 213 om Reading the Tool Offset Data According to the Specified Tool Number 877 C.

WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.4 (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. (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 –10 om FORMAT AND DETAILS OF THE CONTROL

DATA OF THE WINDW FUNCTIONAL INSTRUCTION Description The window instruction is being processed. Hold ACT until W1 is set to 1. 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. ce nt e r.c 0 An error occurred. it is a mistake of data form. Though the function supports only a new format, a old format is specified. w w w .c nc 5 878 APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.4.1 Writing a Tool Offset Data Set data 0 [Description] The data is directly written

into the tool offset value (tool compensation) area of the CNC. om Function code 14 2 Completion code 4 Data length 4 r.c 6 Offset number 8 Offset format Tool offset value (4 byte) 14 (Note 1) Offset format M system (machining center system) Format Tool compensation A Compensation 1 1 Offset number (CTL+2, 3) Offset number Offset number Offset number +1000 .c Tool compensation B Geometry compensation Wear compensation 1 1 Offset number w Tool compensation C Tool length Geometry

compensation Wear compensation w w Cutter Geometry compensation Wear compensation T system (lathe system) Data type Tool compensation A Compensation along the X–axis Compensation along the Z–axis Tool–tip radius compensation Compensation along the Y–axis Compensation related to the position of the virtual tool 1 Offset number +1000 2 Offset number 2 Offset number +1000 Format Offset number (CTL+2, 3) 1 Offset number 2 Offset number 3 4 Offset number Offset number 5 Offset number

Offset number Tool compensation B Geometry compensation Compensation along the X–axis Compensation along the Z–axis Tool–tip radius compensation Compensation along the Y–axis Wear compensation Compensation along the X–axis 1 2 3 4 Offset number Offset number Offset number Offset number Offset number 1 Compensation along the Z–axis 2 Tool–tip radius compensation 3 Compensation along the Y–axis 4 Compensation related to the position of the virtual tool 5 Offset number +1000 Offset number +1000

Offset number +1000 Offset number +1000 Offset number nc Data type ce nt e 10 879 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.4.2 Writing a Parameter (Setting Data) [Description] The data is written into the parameter area of the CNC. om 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. 2 Function code 18 ce nt e Set data 0 r.c 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. Completion code 4 6 Data length 4 Parameter number 8 0 : No axis Data attribute 10 1 to n : A specified axis nc Parameter data (4

byte) 14 .c (Note 1) Format of parameter data If the data is one byte long, it is set in the DATA+0 area. 11 12 13 DATA+0 DATA+1 DATA+2 DATA+3 14 w w w 10 880 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.4.3 Writing a Custom Macro Variable [Description] The data is written into the custom macro variable area of the CNC. Set data 0 Function code 22 om 2 Completion code 4 Data length 6 6 r.c Custom macro number 8 Data attribute 0 10 14 16 ce nt e

Custom macro variable value (4 byte) Position of decimal point (2 byte) nc (Note 1) In the case of writing a Custom Macro Variable of upper than 100000. Please input ”10” to ”Data attribute”, and input last four digits of variable number to ”Custom macro variable number”. (Note 2) Specification of the position of the decimal point 7 6 5 4 3 2 1 0 w w w Examples .c DATA+4 #0 to #3 : Set number of digit beelow decimal point. #4 to #7 : Set to ”0”. (Value written in the NC) + Value in the NC 1234. 000 123.

400 12. 340 1. 124 0. 1234 881 (value of custom macro variable) 10 (Positon of decimal point) Custom macro variable value 1234 Position of decimal point 0 1 2 3 4 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.4.4 [Description] The data to be displayed on the program check screen of the CNC is rewritten. Set data 0 Function code 150 om Writing a Data on the Program Check Screen 2 Completion code 4 6 Data type 8 Data attribute 0 ce nt e 10 r.c Data length

2 Data on the program check screen (2 byte) 12 Data type Attribute M code which is being executed (1 to 5) 1 to 5 0 Spindle speed range 100 0 Spindle tool number 200 0 Number of the tool to be used next 201 0 w w w .c nc Data type 882 APPENDIX B–61863E/10 C.

WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.4.5 Writing the Torque Limit Override [Description] The torque limit override of the specified feed axis is rewritten. om Function code 152 2 Completion code 4 Data length 2 6 Data type 0 8 Axis number Value 10 Un–signed binary The values from 0 to 255 correspond to 0% to 100%. ce nt e Torque limit override 12 r.c Set data 0 [Example] If the torque limit override is 50%, please set to 128. nc C.4.6 2 w Completion code – w 4 The tool number

and the tool life value are written into the specified tool group. Function code 163 w Set data 0 [Description] .c Writing the Tool Life Management Data (Tool Group Number) Data length 4 6 Data number 0 8 Tool No. Data type Number of tool groups Specification Register the tool group. Set group number, tool life value, and tool life counter for transmission data. 7 6 5 4 3 2 1 0 Group number DATA+0 Life counter type DATA+1 DATA+1 BIT 7 DATA+2 Tool life value DATA+3 Group number 1 to 512 Tool life

value 1 to 9999 (Specified number of time) 1 to 4300 (Specified time) Tool life counter type (DATA+1 BIT 7) 0: Number of time 1: Time (minute) 10 Tool group No. 12 (4 byte) Tool life value 14 883 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.4.7 [Description] Set data 0 Function code 164 2 Completion code — 4 Data length 4 6 Tool group No. 8 Data attribute 0 ce nt e 10 om The data is written into the tool life value area of the specified tool group. r.c Writing

the Tool Life Management Data (Tool Life) Tool life (4 byte) 14 C.4.8 [Description] nc Writing the Tool Life Management Data (Tool Life Counter) The data is written into the tool life counter area of the specified tool group. w w w .c Set data 0 Function code 165 2 Completion code — 4 Data length 4 6 Tool group No. 8 Data attribute 0 10 Tool life counter (4 byte) 14 884 APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.4.9 [Description] The data is written into the tool

life counter type area of the specified tool group. Set data 0 Function code 166 om Writing the Tool Life Management Data (Tool Life Counter Type) 2 Completion code — 4 6 Tool group No. 8 Data attribute 0 ce nt e 10 r.c Data length 4 Tool life counter type (4 byte) 14 w w w .c nc Tool life counter type 1 : The tool life counter indicates the number of times the tool has been used. 2 : The tool life counter indicates the period of time the tool has been used. 885 C. WINDOW FUNCTION DESCRIPTION C. (FS15B

PMC–NB/NB2) APPENDIX B–61863E/10 C.4.10 [Description] The data is written into the tool length compensation number area specified by the tool group number and tool number. Set data 0 om Writing the Tool Life Management Data (Tool Length Compensation Number 1) Function code 167 2 4 Data length 4 6 Tool group No. 8 ce nt e Tool No. r.c Completion code — 10 Tool length compensation No. (4 byte) 14 C.4.11 nc Writing the Tool Life Management Data (Tool Length Compensation Number 2)

[Description] w w w .c The data is written into the tool length compensation number area specified by the tool group number and tool order number. Set data 0 Function code 168 2 Completion code — 4 Data length 4 6 Tool group No. 8 Tool order number 10 Tool length compensation No. (4 byte) 14 886 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.4.12 [Description] The data is written into the cutter compensation number area specified by the tool group

number and tool number. Set data 0 om Writing the Tool Life Management Data (Cutter Compensation Number 1) Function code 169 2 4 Data length 4 6 Tool group No. 8 ce nt e Tool No. r.c Completion code — 10 Cutter compensation No. (4 byte) 14 nc C.4.13 [Description] The data is written into the cutter compensation number area specified by the tool group number and tool order number. w w w .c Writing the Tool Life Management Data (Cutter Compensation Number 2) Set data 0 Function code 170 2

Completion code — 4 Data length 4 6 Tool group No. 8 Tool order number 10 Cutter compensation No. (4 byte) 14 887 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.4.14 [Description] The data is written into the tool information (status) area specified by the tool group number and tool number. Set data 0 Function code 171 om Writing the Tool Life Management Data (Tool Information 1) 2 Completion code — 4 Data length 4 Tool group No. 8 Tool No. ce nt e 10

r.c 6 Tool information (4 byte) 14 nc Tool information 1 : The tool is cataloged. 2 : The tool life has expired. 3 : The tool was skipped. C.4.15 [Description] The data is written into the tool information (status) area specified by the tool group number and tool order number.

w w w .c Writing the Tool Life Management Data (Tool Information 2) Set data 0 Function code 172 2 Completion code — 4 Data length 4 6 Tool group No. 8 Tool order number 10 Tool information (4 byte) 14 888 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.4.16 [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 173 om Writing the Tool Life Management Data (Tool

Number) 2 Completion code — 4 6 Tool group No. 8 r.c Data length 4 Tool order number ce nt e 10 Tool No. (4 byte) w w w .c nc 14 889 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 C.4.17 [Description] The tool number is specified and the tool offset data is written. Set data 0 Function code 214 om Writing the Tool offset Data According to the Specified Tool Number 2 Completion code — 4 6 Data format 8 Tool number ce nt e 12 r.c Data length 4 offset data (4

byte) 16 Kind of the data to be written Tool number CTL2 00 01 10 11 20 21 30 31 40 41 Tool No. Tool display number Tool No. Tool display number Tool No. Tool display number Tool No. Tool display number Tool No. 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) w w w .c nc Change of Tool number Change of Tool number Pot number

Pot number Tool length compensation value Tool length compensation value Cutter compensation value Cutter compensation value Addition of Tool number Addition of Tool number The data form CTL1 890 APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.4.18 (1) For three axes [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. om Writing the Superposition Move Command Set data 0 Function code 215 ce nt e 2 r.c 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) (magnification) 62.5 (pulses/second) The data in parameters 1413 and 1414 of the CNC is valid for this function.

Completion code 4 6 8 Data length 6 Data number 0 Data attribute 0 10 nc Superposition move command (6 byte) 16 .c (Note 1) Format of superposition move command DATA+0 DATA+4 First manual pulse generator Second manual pulse generator Third manual pulse generator DATA+6 w w w DATA+2 891 C. WINDOW FUNCTION DESCRIPTION C. (FS15B PMC–NB/NB2) APPENDIX B–61863E/10 (2) For four axes Set data 0 Function code 215 2 Completion code 4 6 om Data length 8 Axis specification

mode 8 Axis number 10 r.c Superposition move command (8 byte) 18 ce nt e Axis specification mode 7 6 5 4 3 2 1 0 #0 : Mode selection (0: The BMI signal is selected. 1: An axis number is selected.) Axis number 7 6 5 4 3 2 1 0 CTL+8 P2 P1 CTL+9 P4 P3 CTL+8, #0 to #3: CTL+8, #4 to #7: CTL+9, #0 to #3: CTL+9, #4 to #7: DATA+0 DATA+2 First manual pulse generator Second manual pulse generator .c DATA+4 nc (Note 1) Format of superposition move command DATA+6 Third manual pulse generator

Fourth manual pulse generator w w w DATA+8 892 Axis number of the first pulse generator (P1) Axis number of the second pulse generator (P2) Axis number of the third pulse generator (P3) Axis number of the fourth pulse generator (P4) APPENDIX B–61863E/10 C. WINDOW FUNCTION DESCRIPTION (FS15B PMC–NB/NB2) C.4.19 Writing the Feedrate [Description] Set data 0 om Feedrate writing can be specified only in the feed–per–minute mode. The velocity command is specified with F 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. Function code 216 2 4 8 Data length 6 Data number 0 ce nt e 6 r.c Completion code Data attribute 0 10 Feedrate (6 byte) 16 w w w .c DATA+6 nc (Note 1) Format of feedrate 7 6 5 4 3 2 1 0 Flag for validating the velocity command DATA+0 : Bit 2 of DATA+0 (0: The NC is valid. DATA+2 Feedrate, F 1:

The PMC is valid.) Feedrate, F : DATA+2 and DATA+3 DATA+4 Exponent Exponent of feedrate, d : DATA+4 and DATA+5 of feedrate, d 893 D. WINDOW FUNCTION DESCRIPTION (FS16–LA) D APPENDIX B–61863E/10 WINDOW FUNCTION DESCRIPTION (FS16–LA) D.1 OUTLINE om 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. r.c (2) Reading a comment

command in a part program.

w w w .c nc ce nt e (3) Reading data commanded to laser oscillator 894 D. WINDOW FUNCTION DESCRIPTION (FS16–LA) APPENDIX B–61863E/10 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) ce nt e r.c om [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] The following End Code is output at reading end. Top address + 0 0 : normal end 3 : The incorrect data is set in Address. 4 : The incorrect data is set in Data attribute.

(Function code) 183 nc 2 (End Code) — (No need to set) 4 .c (Data length) L w 6 L = Set the total byte No. of the data structure for 1 set in non–volatile memory.

This data must be set in case of transferring the data group. (No–need to set in case of transferring the set) N = Set the forward address by the offset from the top address of non–volatile memory. (Address) N w w 8 M = Set the kind of data set or group to transfer (Data attribute) M 1–10 101–103 201–205 1000 1001 1002 10 (Data) : Set of processing data : Set of piercing d data : Set of edge process data : Group of processing data : Group of piercing data : Group of edge process data Structure of data set

Set the correspondence of 1 set of data item in data area and data item in non–volatile memory by relative address for non–volatile memory. 895 D. WINDOW FUNCTION DESCRIPTION (FS16–LA) APPENDIX B–61863E/10 (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. om 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. ce nt e [Structure of input data] r.c And the data of set or group can be transferred from non–volatile memory to data area. The following End Code is output at reading end. Top address + 0 0 : normal end 3 : The incorrect data is set in

Address. 4 : The incorrect data is set in Data attribute. (Function code) 184 2 (End Code) — (No need to set) L = Set the total byte No. of the data structure for 1 set in non–volatile memory.

This data must be set in case of transferring the data group. (No–need to set in case of transferring the set) (Group No.) N N = Set the original address by the offset from the top address of non–volatile memory. .c 6 (Data length) L nc 4 8 M = Set the kind of data set or group to transfer w (Data attribute) M 1–10 101–103 201–205 1000 1001 1002 10 w (Data) : Set of processing data : Set of piercing d data : Set of edge process data : Group of processing data : Group of piercing data : Group of edge process

data w Structure of data set Set the correspondence of 1 set of data item in data area and data item in non–volatile memory by relative address for non–volatile memory. 896 D. WINDOW FUNCTION DESCRIPTION (FS16–LA) APPENDIX B–61863E/10 (3) Data structure of data area (a) Processing data set Address Data item Byte No. Feed–rate 4 4 Peak power 2 6 Pulse frequency 2 8 Pulse duty 2 10 Assist gas pres. 2 12 Assist gas select 14 Assist gas settling time 16 Reference displacement 2 18 Offset amount

4 22 Edge process select 24 Start–up process select r.c ce nt e (b) Piercing data set om 0 Address Data item 2 2 2 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 nc .c w w w 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 2 897 D. WINDOW FUNCTION DESCRIPTION (FS16–LA) APPENDIX B–61863E/10 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. Address The example of data structure for the processing condition file in non–volatile memory Data Address Feed–rate 0 Feed–rate 4 Peak power 4 Peak

power 6 Pulse frequency 6 8 Pulse duty 8 10 Assist gas pres. 10 12 Assist gas select 12 14 Assist gas time 16 Ref. displacement 18 Offset amount 22 Edge select 24 Start–up select Pulse frequency Pulse duty Focus distance Assist gas pres. ce nt e nc .c w w Feed–rate Peak power Pulse frequency Pulse duty Assist gas pres. Assist gas select Assist gas time Ref. displacement Offset amount Edge select Start–up select r.c 0 Data item in data area w Data om Data structure of data area Address Top address +10

Top address +12 Top address +14 Top address +16 Top address +18 Top address +20 Top address +22 Top address +24 Top address +26 Top address +28 Top address +30 898 14 Assist gas select 16 Assist gas time 18 Ref. displacement 20 Offset amount 24 Pulse type 26 Edge select 28 Start–up select Data setting value 0 4 6 8 12 14 16 18 20 26 28 D. WINDOW FUNCTION DESCRIPTION (FS16–LA) APPENDIX B–61863E/10 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. Reading of the Comment Mxxx (* * * * * * *) ; om Less than 24 characters, including alphabet, numeral, decimal–point and +/– [Contents of data] [Structure of input data] r.c Contents of the data can be read for ASCII code. The following End Code is output at writing end. Top address + 0 0 : normal end 1 : The data length of comment data is over 24 bytes ce nt e (Function code) 140 2 (End Code) — (No need to set) 4 L =Data length of the comments read in. (Added 1 in

case of the odd number.) (Data length) L 6 8 nc (Group No.) — .c (Data attribute) — 10 (Data) w w w The read comment data is set. 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. 899 D. WINDOW FUNCTION DESCRIPTION (FS16–LA) APPENDIX B–61863E/10 D.2.3 Reading and Writing the Laser Command Data and Laser Setting Data (1) Reading the laser command data and laser

setting data (:high–speed type) [Contents of the data] om 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] The following End Code is output at writing end. Top address + 0 0 : normal end 2 : The incorrect data is set for Data length. 3 : The incorrect data is set for Data No. r.c (Function code) 186 2 ce nt e (End Code) — (No need to set) 4 L = Set the byte No. assigned for the

group. (Data length) L 6 N = Set the group No. (Group No.) N 8 10 nc (Data attribute) — (No need to set) w w w .c (Data) 900 D. WINDOW FUNCTION DESCRIPTION (FS16–LA) APPENDIX B–61863E/10 (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. om [Structure of input data] The following End Code is

output at writing end. Top address + 0 0: 2: 3: 5: (Function code) 187 r.c 2 normal end The incorrect data is set for Data length. The incorrect data is set for Data No. The data beyond the allowable range is set for data command to write. (End Code) — (No need to set) 4 L = Set the byte No. assigned for the group. ce nt e (Data length) L 6 N = Set the group No. (Group No.) N 8 (Data attribute) — (No need to set) 10 w w w .c nc (Data) 901 D. WINDOW FUNCTION DESCRIPTION (FS16–LA) APPENDIX B–

61863E/10 (3) The data structure of the laser command data and laser setting data Group No. Address Top add. + Byte No. for every 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 Power offset Actual power Actual feedrate 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 30 Assist gas flow–1 2 2 2 6

Processing peak power Processing pulse frequency Processing pulse duty 2 2 2 4 10 Piercing peak power Piercing pulse frequency Piercing pulse duty Piercing time om (Read only) (Read only) (Read only) (Read only) r.c pre–time pre–pres wrk–pres aft–time aft–pres pre–time pre–pres wrk–pres aft–time aft–pres pre–time pre–pres wrk–pres aft–time aft–pres ce nt e Assist gas flow–2 Assist gas flow–3 10 12 14 6 0 12 14 16 7 10 14 16 18 20 22 24 26 28 4 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 2 2 Ref. displacement command w .c nc 5 w w 10 902 E E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 WINDOW FUNCTION DESCRIPTION (FS16–W) READING THE WIRE DIAMETER OFFSET om E.1 [Description] The wire diameter offset value recorded in the CNC can be read.

Top address r.c [Input data structure] (Function) 13 (Completion) — +4 +6 +8 (Data length) L (Number) N (Attribute) M + 10 Offset Corner–R Clearance Condition 0–15 16 17 — 0 0 0 1 nc (Data area) — ce nt e +2 .c [Output data structure] Top address (Function) 13 +2 Offset Corner–R Clearance Condition (Data length) L 4 4 4 8 (Number) N 0–15 16 17 — (Attribute) M 0 0 0 1 (Data area) D Offset value Corner–R value Clearance value Actual offset value w (Completion) ?

w +4 w +6 +8 + 10 + 14 Direction + 16 Offset mode + 18 903 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 [Data number] 0–15 : Reads the Offset value. 0–16 : Reads the Corner–R value. 0–17 : Reads the Clearance value. 0–— : Reads the condition. [Data attribute] om 0 : Reads the Offset value, Corner–R value or Clearance value. 1 : Reads the condition. [Contents of data] Unit of Offset, Corner–R, Clearance and actual offset value r.c a) Metric system input : 10–3 [mm] ce nt e

(In case the incremental system is 1/10, output data unit is 10–4 [mm].) Inch system input : b) 10–5 [inch] 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 nc 0 : Offset mode is 0. 1 : Offset mode is 1. .c [Completion codes] 0 : The data has been read normally. 4 : Invalid data is specified as the data attribute. w w w 3 : Invalid data is specified as the data number. 904 E. WINDOW FUNCTION

DESCRIPTION (FS16–W) APPENDIX B–61863E/10 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] (Function) 14 +2 (Completion) — Offset Corner–R Clearance Condition (Data length) L 4 4 4 8 (Number) N 0–15 16 (Attribute) M 0 0 + 10 (Data area) D + 12 + 14 Offset value Corner–R value [Output data structure] +2 (Function) 14 nc Top address (Completion) ? .c +4 (Data length) L

(Number) N w +6 w +8 17 — 0 1 ce nt e +8 r.c +4 +6 om Top address (Attribute) M + 10 w (Data area) D 905 Clearance value 0, 1 — E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 [Data number] 0–15 : Writes the Offset value. 0–16 : Writes the Corner–R value. 0–17 : Writes the Clearance value. 0–— : Writes the condition. [Data attribute] om 0 : Writes the Offset value, Corner–R value or Clearance value.

1 : Writes the condition. [Contents of data] Unit of Offset, Corner–R, Clearance and actual offset value r.c a) ce nt e NOTE 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. nc [Completion codes] 0 : The data has been written normally. 2 : Invalid data is specified as the data length. .c 3 : Invalid data is specified as the data number. 5 : Invalid data is specified as the data value. w w w 4 : Invalid data is specified as the data attribute.

906 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 E.3 READING THE PARAMETER (:LOW–SPEED RESPONSE) [Description] Parameter data in the CNC can be read. 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. om 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. r.c Specify pitch error compensation data in data Nos. 11000 to 18255. [Input data structure] +2 Top address (Function) 17 +2 (Completion) — +4 +6 (Completion) ? +4 (Data length) — +6 (Number) N +8 +8 (Attribute) M + 10 (Function) 17 ce nt e

Top address [Output data structure] (Data length) L (Number) N (Attribute) M + 10 nc (Data) — (Data) — .c [Data length] w w w L = 1 or 1*n : Reads bit or byte type parameter. 2 or 2*n : Reads word type parameter. 4 or 4*n : Reads 2 words type parameter. (Note: n is the axis number.) [Data number] N = (Parameter number) or (Pitch error data number)+10000 [Data attribute] 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.) 907 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 [Completion code] 0 : Parameter data has been read 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. w w w .c nc ce nt e r.c om 6 : Option is not provided. 908 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 E.4 WRITING THE PARAMETER (:LOW–SPEED RESPONSE) [Description]

Parameter data in the CNC can be written. om 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. r.c 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. ce nt e Specify pitch error compensation data in data Nos. 11000 to 18255. 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 +2 [Output data structure] Top address (Function) 18 +2 (Completion) — (Completion) ? +4 (Data length) L nc +4 +6 +6 (Number) N .c +8 +8 (Attribute) M w w w + 10 (Function) 18 (Data length) L (Number) N (Attribute) M + 10 (Data) Parameter data (Data) D [Data length] L = 1 or 1*n : Reads bit or byte type parameter. 2 or 2*n : Reads word type parameter. 4 or 4*n : Reads 2 words type parameter. (Note: n is the axis number.) [Data number] N = (Parameter number) or (Pitch error data number)+10000 909 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 [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.) [Completion code] om 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.

r.c 4 : Invalid data is specified as the data attribute. w w w .c nc ce nt e 6 : Option is not provided. 910 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 [Types of parameters] In the B908 system, data type of ( ) are used. Length No. Data type No. Data type No. Data type 0000 0001 0002 0003 0004 0005 0006 0007 0008 0009 0010 0011 0012 0013 0014 0015 0016 0017 0018 0019 0020 0021 0022 0023 0024 0025 0026 0027 0028 0029 0030 0031 0032 0033 0034 0035 0036 0037 0038 0039 0040 0041 0042 0043 0044 0045 0046 0047 0048 0049 0050 0051 0052 0053 0054 0055 0056 0057 0058 0059 0060 0061 0062 0063 0064 0065 0066 0067 0068 0069 Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Byte Byte Word Word Word Word Byte Word Word Word Word Word Word Word Word Word Byte(––) Byte(––) Byte Byte Byte Byte Byte Byte Byte Byte Word 2words 0070 0071 0072 0073 0074 0075 0076 0077 0078 0079 0080 0081 0082 0083 0084 0085 0086 0087 0088 0089 0090 0091 0092 0093 0094 0095 0096 0097 0098 0099 0100 0101 0102 0103 0104 0105 0106 0107 0108 0109 0110 0111 0112 0113 0114 0115 0116 0117 0118 0119 0120 0121 0122 0123 0124 0125 0126 0127 0128 0129 0130 0131 0132 0133 0134 0135 0136 0137 0138 0139 Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word 2words Word Word Word Word Word Word Word Word Word 2words Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte 0140 0141 0142 0143 0144 0145 0146 0147 0148 0149 0150 0151 0152 0153 0154 0155 0156 0157 0158 0159 0160 0161 0162 0163 0164 0165 0166 0167 0168 0169 0170 0171 0172 0173 0174 0175 0176 0177 0178 0179 0180 0181 0182 0183 0184 0185 0186 0187 0188 0189 0190 0191 0192 0193 0194 0195 0196 0197 0198 0199 0200 0201 0202 0203 0204 0205 0206 0207 0208 0209 Byte Byte Byte Byte Byte Word Word Word Word Word Word Word Word Byte Word Word Word Word Word Word Word Word Word 2words Word 2words Word Word Word Word Word Word Word Word Word Byte 2W(Byte) 2words Byte(2W) Byte Byte Byte Byte Byte Byte Word Byte Byte 2words 2words 2words 2words 2words Word 2words Byte 2words Byte word — — — — — — — 0210 0211 0212 0213 0214 0215 0216 0217 0218 0219 0220 0221 0222 0223 0224 0225 0226 0227 0228 0229 0230 0231 0232 0233 0234 0235 0236 0237 0238 0239 0240 0241 0242 0243 0244 0245 0246 0247 0248 0249 0250 0251 0252 0253 0254 0255 0256 0257 0258 0259 0260 0261 0262 0263 0264 0265 0266 0267 0268 0269 0270 0271 0272 0273 0274 0275 0276 0277 0278 0279 2W(––) — r.c ce nt e nc .c w w w om No. 911 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 Length No. Data type No. Data type No. Data type 0280 0281 0282 0283 0284 0285 0286 0287 0288 0289 0290 0291 0292 0293 0294 0295 0296 0297 0298 0299 0300 0301 0302 0303 0304 0305 0306 0307 0308 0309 0310 0311 0312 0313 0314 0315 0316 0317 0318 0319 0320 0321 0322 0323 0324 0325 0326 0327 0328 0329 0330 0331 0332 0333 0334 0335 0336 0337 0338 0339 0340 0341 0342 0343 0344 0345 0346 0347 0348 0349 — Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Word — — — Byte Byte Byte Byte Byte Byte Byte Byte Byte Byte Word Word 2words 2words Word Word 2words 2words 2words 2words 2words 2words Byte Word Byte Word Byte 2words 2words 0350 0351 0352 0353 0354 0355 0356 0357 0358 0359 0360 0361 0362 0363 0364 0365 0366 0367 0368 0369 0370 0371 0372 0373 0374 0375 0376 0377 0378 0379 0380 0381 0382 0383 0384 0385 0386 0387 0388 0389 0390 0391 0392 0393 0394 0395 0396 0397 0398 0399 0600 0601 0602 0603 0604 0605 0606 0607 0608 0609 0610 0611 0612 0613 0614 0615 0616 0617 0618 0619 2words Byte Byte Byte Byte Byte Byte Word — — Word 2words — 2words 2words — — — — — — — — — — — Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Bit Byte Byte Byte Byte 0620 0621 0622 0623 0624 0625 0626 0627 0628 0629 0630 0631 0632 0633 0634 0635 0636 0637 0638 0639 0640 0641 0642 0643 0644 0645 0646 0647 0648 0649 0650 0651 0652 0653 0654 0655 0656 0657 0658 0659 0660 0661 0662 0663 0664 0665 0666 0667 0668 0669 0670 0671 0672 0673 0674 0675 0676 0677 0678 0679 0680 0681 0682 0683 0684 0685 0686 0687 0688 0689 Byte Byte Byte Byte Word Word Word Word Word Word Word Word 2words 2words 2words 2words 2words 2words 2words 2words 2words 2words 2words 2words 2words 2words Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word 2words 2words 2words 2words 2words 2words Word Word Word Word Word Word Word Word Word 0690 0691 0692 0693 0694 0695 0696 0697 0698 0699 0700 0701 0702 0703 0704 0705 0706 0707 0708 0709 0710 0711 0712 0713 0714 0715 0716 0717 0718 0719 0720 0721 0722 0723 0724 0725 0726 0727 0728 0729 0730 0731 0732 0733 0734 0735 0736 0737 0738 0739 0740 0741 0742 0743 0744 0745 0746 0747 0748 0749 0750 0751 0752 0753 0754 0755 0756 0757 0758 0759 Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word 2words 2words 2words 2words 2words 2words 2words 2words Word Word Word 2words 2words r.c ce nt e nc .c w w w om No. 912 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 Length No. Data type No. Data type No. Data type 0760 0761 0762 0763 0764 0765 0766 0767 0768 0769 0770 0771 0772 0773 0774 0775 0776 0777 0778 0779 0780 0781 0782 0783 0784 0785 0786 0787 0788 0789 0790 0791 0792 0793 0794 0795 0796 0797 0798 0799 0800 0801 0802 0803 0804 0805 0806 0807 0808 0809 2words 0810 0811 0812 0813 0814 0815 0816 0817 0818 0819 0820 0821 0822 0823 0824 0825 0826 0827 0828 0829 0830 0831 0832 0833 0834 0835 0836 0837 0838 0839 0840 0841 0842 0843 0844 0845 0846 0847 0848 0849 0850 0851 0852 0853 0854 0855 0856 0857 0858 0859 2words Byte Byte Byte Byte Byte Byte Byte Byte Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word — Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word(2W) Word(2W) Word(2W) Word(2W) Word(2W) Word(2W) Word(2W) 0860 0861 0862 0863 0864 0865 0866 0867 0868 0869 0870 0871 0872 0873 0874 0875 0876 0877 0878 0879 0880 0881 0882 0883 0884 0885 0886 0887 0888 0889 0890 0891 0892 0893 0894 0895 0896 0897 0898 0899 0900 0901 0902 0903 0904 0905 0906 0907 0908 0909 Word Bit Bit Byte Byte Byte Byte Byte Byte Byte Byte Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word Word ––(Word) ––(Bit) ––(Bit) ––(Bit) — ––(Byte) ––(Byte) — — — — 0910 0911 0912 0913 0914 0915 0916 0917 0918 0919 0920 0921 0922 0923 0924 0925 0926 0927 0928 0929 0930 0931 0932 0933 0934 0935 0936 0937 0938 0939 0940 0941 0942 0943 0944 0945 0946 0947 0948 0949 — w w w .c nc ce nt e r.c om No. 913 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 E.5 [Data contents] Setting data on the CNC can be read. [Input data structure] Top address [Output data structure] Top address (Function code) 19 +2 +2 (Completion code) – +4 (Function code) 19 om READING SETTING DATA (Completion code) ? +4 (Data length) – +6 +6 (Data number) Input data r.c (Data number) N (Data length) L +8 +8 (Data attribute) – + 10 + 10 (Data area) Setting data ce nt e (Data area) – (Data attribute) – [Data number] See the setting data list. [Completion code] 0 : Read operation was terminated normally. w w w .c nc 3 : An incorrect data number was specified. 914 APPENDIX B–61863E/10 E. WINDOW FUNCTION DESCRIPTION (FS16–W) E.6 WRITING SETTING DATA (LOW–SPEED TYPE) [Data contents] Setting data on the CNC can be written. Top address [Output data structure] Top address (Function code) 20 +2 +2 (Completion code) ? (Completion code) – +4 +4 (Data length) L +6 +6 (Data length) Input data (Data number) Input data r.c (Data number) N +8 +8 (Data attribute) – (Data attribute) – (Data attribute) – + 10 (Data area) Setting data (Data area) Input data ce nt e + 10 (Function code) 20 om [Input data structure] [Data length] See the setting data list. [Data number] See the setting data list. nc [Completion code] 0 : Read operation was terminated normally. 2 : An incorrect data length was specified. w w w .c 3 : An incorrect data number was specified.

915 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 Setting Data List Screen Data number Data length Bit name X mirror image Y mirror image Axis switching TV check Output code Input unit Parameter writable Input unit multiplication by 10 times Automatic recovery from power failure Automatic power–off M20/M30 Automatic power–off M00/M10 Automatic power backward movement alarm Automatic power disconnection Input/output device Figure magnification ratio Figure rotation angle Handy Handy Handy Handy Handy Handy Handy Handy Handy Handy Handy Handy 1 1 1 1 1 1 1 1 2 3 3 1 1 1 1 1 1 1 1 1 1 1 1 Bit 0 Bit 1 Bit 2 Bit 3 Bit 4 Bit 5 Bit 6 Bit 7 Bit 3 Bit 0 Bit 1 Bit 2 Handy Handy Handy Handy 3 4 5 6 1 1 4 4 Bit 3 – – – Taper machining mode Guide type Program surface position Workpiece thickness Drawing surface position Upper guide position (Lower guide position) (Vertical position U) (Vertical position V) Taper Taper Taper Taper Taper Taper Taper Taper 15 16 17 18 19 20 21 22 23 1 1 4 4 4 4 4 4 4 – Bit 0 – – – – – – – PWB PWB PWB PWB PWB PWB PWB 2 7 8 9 10 11 12 1 4 4 4 4 4 4 – – – – – – – PWB PWB 2 13 1 2 Bit 5 – AWF AWF AWF AWF AWF AWF AWF AWF AWF AWF AWF 2 2 2 2 14 24 26 27 28 40 41 1 1 1 1 1 1 1 4 4 4 1 1 Bit 2 Bit 1 Bit 6 Bit 7 – – – – – – – AWF AWF 42 43 1 1 – – LIFE LIFE LIFE LIFE LIFE LIFE LIFE LIFE 30 31 32 33 34 35 36 37 2 2 2 2 2 2 2 2 – – – – – – – – w w .c nc Enable/disable Disconnection repair Prepared hole of 0.5 Sump machining Portion to be left uncut Power reduction ratio (setting) Connection position U1 Connection position V1 Connection position Z1 Number of retries (setting) Number of allowable disconnections (setting) Number of retries (current) Number of allowable disconnections (current) w WIRE WIRE CUTTER ( ) ( ) ( ) ( ) ( ) ( ) REST REST REST REST REST REST REST REST r.c ce nt e Enable/disable Wire diameter Machining groove width Workpiece thickness Wire deflection Effectiveness of concave Effectiveness of convex Automatic override Enable/disable Differential voltage om Setting data 916 E.

WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 Setting Data List Number or tries Number of retries Number or tries Number of retries Data number Data length Bit name Others Others Others Others Others Others Others Others Others Others 38 38 38 38 38 38 38 38 39 39 1 1 1 1 1 1 1 1 1 1 Bit 0 Bit 1 Bit 2 Bit 3 Bit 4 Bit 5 Bit 6 Bit 7 Bit 0 Bit 1 AWF AWF AWF AWF 40 41 42 43 1 1 1 1 – – – – Graphic 44 2 – w w w .c nc ce nt e r.c Program number (for machining distance calculation) Screen om Setting data OP. BLOCK SKIP /0 OP. BLOCK SKIP /1 OP. BLOCK SKIP /2 OP. BLOCK SKIP /3 OP. BLOCK SKIP /4 OP. BLOCK SKIP /5 OP. BLOCK SKIP /6 OP. BLOCK SKIP /7 OP. BLOCK SKIP /8 OP.

BLOCK SKIP /9 917 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 E.7 READING THE CNC ALARM STATUS [Description] When the CNC is placed in the alarm status, the alarm status data can be read. Top address [Output data structure] Top address (Function) 23 +2 (Function) 23 +2 (Completion) — (Completion) 0 +4 (Data length) — +6 (Data length) 2 +6 (Number) — +8 +8 (Attribute) — (Data area) — + 10 + 11 + 12 (Attribute) — First byte (Data area) Alarm status ce nt e + 10 (Number) — r.c +4 om [Input data structure] Second byte [Contents of data] (1) Alarm status data in first byte. #7 #6 #5 #4 EOR OTM OTS OH #2 #1 #0 SV OTH PS #2 #1 : P/S alarm nc PS #3 w w w .c OTH : Over travel alarm SV : Servo alarm OH : Overheat alarm OTS : First stroke limit alarm OTM : Second stroke limit alarm EOR : Edit alarm (2) Alarm status data in second byte. #7 #6 #5 #4 #3 #0 APCER APCER : Absolute pulco alarm [Completion codes] 0 : This alarm status in the CNC has been read normally.

918 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 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 6 5 4 3 2 1 0 Code in a group : 1 byte 0 : Not specified in the current block 1 : Specified in the current block om 7 Data type Data Identification code Data type Data 0 G04 G19 G28 G30 G92 G31 G70 G71 G72 G73 G74 G75 G76 G77 G78 G79 0 1 5 7 14 15 16 17 18 19 20 21 22 23 24 25 5 G94 G95 G20 G21 G40 G41 G42 G50 G51 G52 G60 G61 G62 G63 G48 G49 G65 G66 G67 0 1 0 1 0 1 2 0 1 2 0 1 2 3 1 0 26 0 1 2 G90 G91 G22 G23 7 ce nt e .c 3 G00 G01 G02 G03 G17 6 8 9 10 0 1 2 3 0 11 nc 1 r.c Identification code w w w 4 0 1 1 0 919 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 (2) Format and types of modal data for other than the G function 6 5 — : 4 bytes Flag : 1 byte 4 3 2 — 1 0 om 7 Data 1 byte r.c Number of input digits 0 : Positive 1 : Negative 0 : Not specified in the current block ce nt e 1 : Specified in the current block Specified address 100 101 102 103 104 105 106 107 108 109 110 111 B D E F H L M S T R P Q [Input data structure] Top address w w .c nc Identification code (Function) 32 +2 w (Completion) — +4 (Data length) — +6 (Number) N +8 (Attribute) M + 10 (Data area) — 920 Meaning of value Offset number Feedrate Tapper data E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 [Output data structure] Top address (Function code) 32 +2 (Completion code) ? +4 +6 (Data number) Input data +8 (Data attribute) Input data + 10 + 12 Modal data of ID code 0 Modal data of ID code 100 r.c Modal data of other than G functions (Data area) Modal data of G functions om (Data length) L Modal data of ID code 1 ce nt e + 14 Flag part of modal data Flag part of data of ID code 100 Modal data of ID code 2 + 16 A A Modal data of ID code 16 Modal data of ID code 111 w w w .c nc + 44 Flag part of data of ID code 111 + 78 [Data length] L= 2 : G function 2* : 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 1 : Current block : Next block 921 A E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX 2 B–61863E/10 : Block after the next block [Completion code] –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. : Modal data read operation terminated normally. 3 : An incorrect data number was specified. 4 : An incorrect data attribute was specified. w w w .c nc ce nt e r.c om 0 922 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 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. Top address [Output data structure] Top address (Function code) 182 +2 +2 (Completion code) ? (Completion code) – +4 (Data length) – +6 +6 (Data number) N +8 +8 (Data number) Input data (Data attribute) – ce nt e (Data attribute) – + 10 (Data length) 4 r.c +4 (Function code) 182 om [Input data structure] + 10 (Data area) Machining distance (Data area) – + 14 [Data number] : Machining distance N= 1 : Whole length obtained with the dry run function nc N= 0 : Whole length obtained by drawing N= 3 : Whole length obtained with the machining distance calculation function .c N= 2 w w w [Data unit] Metric input : 10–3 [mm] (When the increment system is 1/10: 10–4 [mm]) Inch input : 10–5 [inch] [Completion code] –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. 923 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 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. Top address (Function) 185 +2 (Completion) — +6 + 10 Reads measured point (Number) N Point number 0 (Attribute) M 0 1 Reads slit width of hole diameter ce nt e +8 (Length) — r.c +4 om [Input data structure] (Data area) — [Output data structure] Top address (Completion) ? Reads measured point Reads slit width of hole diameter (Length) L 10 4 (Number) N Point number 0 (Attribute) M 0 1 (Data area) Type Slit width of hole diameter nc +2 (Function) 185 +4 .c +6 w w w +8 + 10 + 12 Machine coordinate of X axis + 14 + 16 Machine coordinate of Y axis + 18 + 20 924 APPENDIX B–61863E/10 E. WINDOW FUNCTION DESCRIPTION (FS16–W) [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] [Completion codes] om 0 : The measured point has been read normally. 3 : Invalid data is specified as data number.

w w w .c nc ce nt e r.c 4 : Invalid data is specified as data attribute. 925 E. WINDOW FUNCTION DESCRIPTION (FS16–W) APPENDIX B–61863E/10 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. +4 +6 +8 + 10 + 12 +2 (Completion) — +4 (Length) 10 (Function) 186 (Completion) ? (Length) 10 r.c +2 Top address (Function) 186 +6 (Number) Point number +8 (Attribute) 0 (Number) N (Attribute) 0 ce nt e Top address [Output data structure] om [Input data structure] + 10 (Data area) Type (Data area) + 12 Machine coordinate of X axis + 14 + 16 + 16 Machine coordinate of Y axis nc + 18 + 14 + 20 .c + 20 + 18 [Data unit] w w w 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] [Completion codes] 0 : The measured point has been written normally. 3 : Invalid data is specified as data number. 4 : Invalid data is specified as data attribute. 926 F F. WINDOW FUNCTION DESCRIPTION (FS16–PA) APPENDIX B–61863E/10 WINDOW FUNCTION DESCRIPTION (FS16–PA) F.1 READING OF TOOL SETTING DATA om [Description] Various Tool setting data recorded in the CNC can be read. [Input data structure] r.c Top address + 0 (Function code) 188 2 ce nt e (Completion code) — (Need not be set) 4 (Data length) — (Need not be set) 6 (Data Number) N (See 1.1) nc 8 (Data attribute) M (See 1.1) w w w .c 10 (Data area) — (Need not be set) [Completion codes] 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. 927 F. WINDOW FUNCTION DESCRIPTION (FS16–PA) APPENDIX B–61863E/10 [Output data structure] Top address + 0 (Function code) 188 2 (Completion code) ? (See the explanation of codes) 4 om (Data length) ? (See 1.1) 6 8 r.c (Data Number) N (N : Input data) (Data attribute) M (M : Input data) 10 ce nt e (Data area) ? (See 1.1) X X w w w .c nc 48 928 APPENDIX B–61863E/10 F. WINDOW FUNCTION DESCRIPTION (FS16–PA) F.1.1 Data Number, Data Attribute, Data Length, Data Area 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 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 r.c 0 om Data attribute (M) Tool number of reference point Data number (N) ce nt e Various tool setting data Data number, data attribute, data length and data area of various tool setting data are as follows. 1 4 bytes Binary 1 to 99999999 2 4 bytes Binary ±99999999 3 4 bytes Binary ±99999999 4 4 bytes Binary ±99999999 5 2 bytes Binary 1 to 136 6 4 bytes 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 X Tool position compensation of Y Machine position of tool Number of tool setting tti d data t nc Tool number for tool change 1 to 136 (Not used) w w w .c Tool shape (C) for graphic Tool position compensation of Y 2 4 bytes Binary ±99999999 Tool shape (C) for multi–tool Number of Multi tool setting Multi–tool data +200 3 2 bytes High byte=0 Binary 0 to 4 Tool shape (I) for multi–tool 201 to 264 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 929 F. WINDOW FUNCTION DESCRIPTION (FS16–PA) APPENDIX B–61863E/10 data unit Machine Input of IS–A Input of IS–B mm 0. 01 0. 001 inch 0. 001 0.

0001 Tool position compensation Input of IS–A Input of IS–B mm 0. 01 0. 001 inch 0. 001 0.

0001 deg 0. 01 0. 001 Tool shape and angle for graphic Tool angle for li l multi–tool F.2 WRITING OF TOOL SETTING DATA (LOW–SPEED RESPONSE) r.c [Description] om Input unit The various tool setting data can be directly written into the CNC. ce nt e [Input data structure] Top address + 0 (Function code) 189 2 (Completion code) — (Need not be set) 4 nc (Data length) ? (See 1.1) (Data Number) N (See 1.1) 8 (Data attribute) M (See 1.1) 10 (Data) ? (See 1.1) w w w .c 6 [Completion code] 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. 930 F. WINDOW FUNCTION DESCRIPTION (FS16–PA) APPENDIX B–61863E/10 [Output data structure] Top address + 0 (Function code) 189 2 (Completion code) ? (See the explanation of codes) 4 om (Data length) ? (See F.1) 6 8 r.c (Data Number) N (N : Input data) (Data attribute) M (M : Input data) 10 ce nt e (Data area) ? (See F.1) X X nc 48 w w w .c NOTE See Sec. F. 1 for data unit. 931 F. WINDOW FUNCTION DESCRIPTION (FS16–PA) APPENDIX B–61863E/10 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] (Function code) 141 2 r.c (Completion) – (Need not be set) om Top address + 0 4 (Data length) – (Need not be set) 6 ce nt e (Data number) N (N=Tool number) 10 (Data attribute) M (See F.1) 12 nc (Data area) – (Need not be set) w w w .c 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. 932 F. WINDOW FUNCTION DESCRIPTION (FS16–PA) APPENDIX B–61863E/10 [Completion code] 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. om [Output data structure] Top address + 0 2 r.c (Function code) 141 (Completion code) ? (See the explanation above) 4 ce nt e (Data length) ? (See F.1) 6 (Data number) N (N: Input data) 10 (Data attribute) M (M: Input data) A X (Data area) ? (See F.1) A X 48 w w w .c nc 12 933 F. WINDOW FUNCTION DESCRIPTION (FS16–PA) APPENDIX B–61863E/10 F.4 OTHER WINDOW FUNCTIONS The FS16–PA supports the following window functions, described in this manual.

Function code Number Reading CNC system information 0 2 Reading a tool offset 13 3 Writing a tool offset 4 Reading a workpiece origin offset 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 :low–speed response 22 om 1 :low–speed response 14 ce nt e r.c 15 Writing a custom macro variable Reading the CNC alarm state 13 Reading the current program number 24 14 Reading the current sequence number 25 15 Reading an actual velocity for a controlled axis 26 16 Reading an absolute position on a controlled axis 27 17 Reading a machine position on a controlled axis 28 18 Reading a skip position on a controlled axis 29 19 Reading a servo delay amount on a controlled axis 30 20 Reading an acceleration/deceleration delay amount on a controlled axis 31 21 Reading modal data 32 22 Reading diagnostic data 38 Reading clock data (date and time) 151 41 Reading a parameter 154 42 Reading setting data 155 Reading diagnostic data 156 Reading a character string of the CNC program being executed in the buffer 157 45 Reading the relative position on a controlled axis 74 46 Reading the remaining travel on a controlled axis 75 .c w w 44 w 43 nc 11 12 :low–speed response 23 33 47 Reading CNC status information 76 48 Reading an operator message 83 934 APPENDIX B–61863E/10 G G. SIGNAL ADDRESS CONVERSION (FROM THE PMC–MODEL L/M TO THE PMC–MODEL RB/RC) SIGNAL ADDRESS CONVERSION (FROM THE PMC–MODEL L/M TO THE PMC–MODEL RB/RC) G.1 GENERAL om DI/DO signals used in the PMC–MODEL L/M can be converted to signals for the PMC–MODEL RB/RC using the FAPT LADDER program for the PMC–MODEL RB/RC. G.2 FUNCTION r.c The DI/DO signals used between the NC unit and the PMC correspond to word addresses consisting of addresses and values. Word addresses of bit type are converted. The program is not logically converted.

ce nt e The conversion is performed under the following conditions. (1) A word address of bit type used in a basic instruction is to be converted.

(2) A word address of byte type used in a functional instruction is not converted. (3) Word addresses used in the standard FANUC Series 0–T/M are converted to those used in the standard FANUC Series 16–T/M. If a value in a word address is 1000.0 or more, the address is not converted. .c nc (4) When the same signal name is used in the FANUC Series 0 and 16, and the addresses corresponding to the signal in the Series 0 and 16 have one–to–one relationship, the word address is converted. For details, see the signal conversion table.

G.3 w w w CONVERSION (1) Load the FAPT LADDER program for the PMC–RB/RC. (2) Press the R0 key to display the programmer menu screen. (3) Press the F2 key. Enter 2 and press the key. then the following message appears on the screen. Insert a data floppy for the PMC–RB/RC. Select the name of the file corresponding to the conversion from Table G.3 and enter it. SET FD & KEYIN ‘OK’, ‘KILL’ OR ‘NO’ FD0 = OK FD0 = Table G.3 File Name in the Data Floppy for the PMC–RB/RC File name FS0–T → FS16–T COMV.FS0–T FS0–M → FS16–M COMV.FS0–M 935 G. SIGNAL ADDRESS CONVERSION (FROM THE PMC–MODEL L/M TO THE PMC–MODEL RB/RC) APPENDIX B–61863E/10 om (4) Read a source ladder program created with FAPT LADDER for the PMC–L/M from the floppy in 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 key to return to the programmer menu screen. PART= E (5) Entering 9 and pressing the 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 key to delete the symbol data.

KEYIN ’1, 2, 3, 4, 5 OR 6 OR ‘NO’ CLEAR/KEEP= G.4 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. ce nt e r.c MODIFYING THE CONVERTED SEQUENCE PROGRAM 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. nc G.4.1 (1) When the Series 0 and 16 differ in the number of parameters used in a functional instruction 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 w w w .c Modification Procedure message may not appear.

(XXX.X and XXXX are addresses and values.) 936 APPENDIX B–61863E/10 G. SIGNAL ADDRESS CONVERSION (FROM THE PMC–MODEL L/M TO THE PMC–MODEL RB/RC) (3) Deleting SUB48 (END3) (In the PMC–RB) If SUB48 (END3) is specified in the PMC–RB, an error occurs because the PMC–RB 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. w w w .c nc ce nt e r.c om (4) Address conversion for signals not listed on the signal conversion table Modify the address for a signal by referring to the connection manual. 937 H. CONNECTING THE OPERATOR’S PANEL FOR FS 0 WITH FS16, FS18, FS21, OR Power Mate H APPENDIX B–61863E/10 CONNECTING THE OPERATOR’S PANEL FOR FS 0 WITH FS16, FS18, FS21, OR Power Mate H.1 GENERAL r.c om 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. CNC (Series 16 or 18) Bit image Rk and on ce nt e Therefore, simple bit images of switches and LEDs must only be manipulated with the PMC ladder program. I/O unit I/O card Input Xn and on Coding Output Ym and on Coding Keyboard PMC management software w .c nc Bit image Rl and on w w F* and on Protect key Emergency stop button Override rotary switch etc. Series 0 operator’s manual Input X* and on G* and after (X) CNC software Input X* and on Contact LED PMC ladder program (of the user) Interface with other machines Output Y* and after Fig. H.1 (a) Connection between the CNC and the Operator’s Panel 938 APPENDIX B–61863E/10 H. CONNECTING THE OPERATOR’S PANEL FOR FS 0 WITH FS16, FS18, FS21, OR Power Mate The operator’s panel is made up of the following keys, LEDs, etc. D Key switch (Seat key) D 42 keys (0–TB) D 46 keys (0–MB) D LEDs (red) .... Prepared for all key switches D Override rotary switch ..... 4 bits nc ce nt e r.c D Program protect key .... 1 bit om D Emergency stop button .... 1 bit w w w .c 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 939 APPENDIX B–61863E/10 r.c om H. CONNECTING THE OPERATOR’S PANEL FOR FS 0 WITH FS16, FS18, FS21, OR Power Mate w w .c nc ce nt e Fig. H.1 (d) External view of operator’s panel for 9” CRT/MDI with full–keyboard (0–TC) w Fig. H.1 (e) External view of operator’s panel for 9” CRT/MDI with full–keyboard (0–MC) 940 APPENDIX B–61863E/10 H. CONNECTING THE OPERATOR’S PANEL FOR FS 0 WITH FS16, FS18, FS21, OR Power Mate H.2 CONNECTION H.2.1 Connecting the I/O Unit MAIN CPU Series 0 operator’s panel JD1B DI module CP32 DO module PSU JD1A 24 VDC M2A ce nt e CP6 M1A r.c I/O unit JD1A (I/O Link) om CNC (Series 16 or 18) Another I/O unit DI module : +24 V common, 20 ms Example) AID32A1 DO module : 0 V common Example) AOD32A1 nc H.2.2 Connecting the I/O Card CNC (Series 16 or 18) To another I/O unit w w w .c MAIN CPU JD1A (I/O Link) Interface with other machines Series 0 operator’s panel I/O card DI connector M1A DO connector M2A 941 H. CONNECTING THE OPERATOR’S PANEL FOR FS 0 WITH FS16, FS18, FS21, OR Power Mate APPENDIX B–61863E/10 H.3 SIGNALS FOR CONNECTING THE OPERATOR’S PANEL H.3.1 This signal is used for the fixed address directly monitored by the CNC. om Emergency Stop Signal (*ESP) 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.3.3 Key Switch Signals (Xn, Xn+2) Their key switch contact signals are directly input to the PMC.

Handle them with the PMC ladder program. 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.” ce nt e Override Signals (*OV1 to *OV8) and Program Protect Key Signal (KEY) r.c H.3.2 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. nc 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. w w w .c 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.) 942 H. CONNECTING THE OPERATOR’S PANEL FOR FS 0 WITH FS16, FS18, FS21, OR Power Mate APPENDIX B–61863E/10 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)) om 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. r.c 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 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.) ce nt e Table H.3.4 (a) Key Switch and LED Signal Addresses Xn #7 #6 #5 #4 #3 #2 #1 #0 KD7 KD6 KD5 KD4 KD3 KD2 KD1 KD0 KA3 KA2 KA1 KA0 LD3 LD2 LD1 LD0 Xn+1 KST Ym LD7 nc Xn+2 LD6 LD5 LD4 Table H.3.4 (b) Bit Image Addresses of Key Switch and LED Signals #7 #6 #5 KEY/LED F3 F2 F1 Rk/Rl F4 Rk+1/Rl+1 D4 .c w w w (for the small operator’s panel) Rk+2/Rl+2 Rk+3/Rl+3 F8 Rk+4/Rl+4 D8 D3 C4 F6 F5 #4 C3 #3 #2 #1 #0 D1 C1 B1 A1 D2 C2 B2 A2 B4 B3 A4 A3 D5 C5 B5 A5 D6 C6 B6 A6 A8 A7 C8 B8 Rk+5/Rl+5 F9 D9 C9 B9 A9 Rk+6/Rl+6 F10 D10 C10 B10 A10 943 H. CONNECTING THE OPERATOR’S PANEL FOR FS 0 WITH FS16, FS18, FS21, OR Power Mate APPENDIX B–61863E/10 Table H.3.4 (c) Bit Image Addresses of Key Switch and LED Signals (for the operator’s panel with the full keyboard) #6 #5 #4 #3 #2 #1 #0 KEY/LED E1 C1 A1 E6 D6 C6 B6 A6 Rk/Rl E2 C2 A2 E7 D7 C7 B7 A7 Rk+1/Rl+1 E3 C3 A3 E8 D8 C8 B8 A8 Rk+2/Rl+2 E5 C4 A4 E9 D9 C9 B9 A9 Rk+3/Rl+3 D2 C5 A5 E10 Rk+4/Rl+4 D4 D5 B2 E11 Rk+5/Rl+5 D1 B1 B4 E12 Rk+6/Rl+6 D3 B3 B5 E13 om #7 C10 B10 A10 D11 C11 B11 A11 D12 C12 B12 A12 D13 C13 B13 A13 w w w .c nc ce nt e r.c D10 944 H. CONNECTING THE OPERATOR’S PANEL FOR FS 0 WITH FS16, FS18, FS21, OR Power Mate APPENDIX B–61863E/10 H.4 SPECIFYING ADDRESSES The following section describes how to specify key switch and LED signal addresses and the bit image addresses. H.4.1 Parameter Menu (for PMC–RB) KEY IN ONE OF THE FOLLOWING NO.S WHICH YOU WANT TO SET PARA.S 01 CURRENT PARAMETERS (UNUSED) ; 02 COUNTER DATA TYPE ; BINARY 03 OPERATOR PANEL ; YES KEY/LED ADDRESS ; X0000/Y0000 KEY/LED BIT IMAGE ADRS. ; R0900/R0910 (UNUSED) 06 (UNUSED) 07 (UNUSED) 08 (UNUSED) ; PMC–RB r.c PMC TYPE 05 ; ; ; ; ce nt e 04 om NO.

ITEMS 09 IGNORE DIVIDED CODE ; NO 10 (UNUSED) ; 00 NOTHING TO SET ; ; ROM WRITER = FA WRITER NO. = H.4.2 Procedure Select 3 from the parameter menu. Then, the following message is displayed: nc 1) .c EXAMPLE 0:NO, 1:YES OP.PANEL=_ w w w 2) 3) Select 1(:YES).

Then, the following message is displayed: SET KEY/LED ADDRESS(KEY ADRS., LED ADRS.) ADDR=_ 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=_ 945 H. CONNECTING THE OPERATOR’S PANEL FOR FS 0 WITH FS16, FS18, FS21, OR Power Mate APPENDIX 4) B–61863E/10 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: : 03 : : OPERATOR PANEL KEY/LED ADDRESS ; X0000/Y0000 KEY/LED BIT IMAGE ADRS. ; R0900/R0910 : : om : ; YES ce nt e r.c NOTE 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 → X0000 Rk / Rl →R0900/R0910 Xn+1 → X0001 Rk+1 / Rl+1→R0901/R0911 Xn+2 → X0002 Rk+2 / Rl+2→R0902/R0912 Rk+3 / Rl+3→R0903/R0913 Ym → Y0000 Rk+4 / Rl+4→R0904/R0914 Rk+5 / Rl+5→R0905/R0915 Rk+6 / Rl+6→R0906/R0916 Rk+7 / Rl+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 nc To use X1000, X1001, X1002, and Y1000 for key switches and LEDs, enter the following: w w w .c SET KEY/LED ADDRESS(KEY ADRS., LED ADRS.) ADDR= X1000,Y1000 [NL] 946 APPENDIX B–61863E/10 I I. EDITING FOR Power Mate–MODEL D (PMC–PA1/PA3) EDITING FOR Power Mate–MODEL D (PMC–PA1/PA3) I.1 OUTLINE om 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 RA1/RA2. Following abbreviations are used in this chapter. FANUC Series 18 PMC–PA1 FANUC PMC–MODEL PA3 PMC–PA3 Ladder diagram editing memory card Editing card FANUC PMC–MODEL RA1 PMC–RA1 FANUC PMC–MODEL RA2 PMC–RA2 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. nc COMPATIBILITY WITH CNC BASIC SOFTWARE Abbr. FANUC PMC–MODEL PA1 ce nt e FANUC Power Mate–MODEL D I.2 Product/Card Name r.c CNC Model .c PMC · Version 04(D) or later of PMC–PA1/PA3 control software 4075 Series. I.3 w w w 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. 947 I. EDITING FOR Power Mate–MODEL D I. (PMC–PA1/PA3) APPENDIX B–61863E/10 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 om 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.

r.c When you want to put on and take off, you must turn off the CNC power. ce nt e CAUTION Please do not release the write protect switch of editing card for preventing a mistake deleting. nc WRITE PROTECT w w w .c Editing Card (A02B–0166–K701#4076) Fig. I.3.1.1 948 APPENDIX B–61863E/10 I. EDITING FOR Power Mate–MODEL D (PMC–PA1/PA3) (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) om I.3.1.2 Feed the editing card into connector CNMC of the CNC. Connection of Components w w w .c nc ce nt e r.c When you want to put on and take off, you must turn off the CNC power. (Refer to the fig. I.3.1.2) Fig. I.3.1.2 I.3.1.3 Parameter 949 I. EDITING FOR Power Mate–MODEL D I.

(PMC–PA1/PA3) APPENDIX B–61863E/10 Please set bit 1 in K17 of keep relay area for PMC parameters. #7 #6 #5 #4 #3 #2 K17 #1 #0 PRGRAM PRGRAM 0: The programmer function is disabled. (The programmer menu is not displayed.) om 1: The programmer function is enabled. (The programmer menu is displayed.) I.3.2 FANUC Power Mate–MODEL D can set only COUNTER DATA TYPE. The meaning of this parameter is same as PMC–RA1/RA2. r.c Specification and Display of System Parameters (SYSPRM) BINARY/BCD ce nt e PMC SYSTEM PARAMETER COUNTER DATA TYPE = nc [BINARY] [BCD] [ ] [ ] [ ] .c Fig. I.3.2 PMC–PA1 or PA3 System Parameter Screen w I.3.3 w w Condense 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. 950 APPENDIX B–61863E/10 I. EDITING FOR Power Mate–MODEL D (PMC–PA1/PA3) Message 10.0KB Symbol/Comment 19.1KB Symbol/Comment 18.9KB 20.0KB 19.0KB (Unused area 0.1KB) Ladder 34.0KB Ladder 34.0KB w w .c Ladder 34.0KB Symbol/Comment 18.9KB (Unused area 1.1KB) nc (Unused area 0.9KB) Executed Condense ce nt e Symbol/ Comment 0.2KB Delete Message 10.0KB r.c Message 10.0KB om 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. Fig. I.3.3 (b) w Fig. I.3.3 (a) (Unused area 1.0KB) 951 Fig. I.3.3 (c) I. EDITING FOR Power Mate–MODEL D I. (PMC–PA1/PA3) APPENDIX B–61863E/10 I.4 SYSTEM DIAGRAM OF SOFT KEY PCLAD PCDGN PCPRM STOP EDIT I/O SYSPRM TITLE LADDER SYMBOL MESAGE RETURN om NEXT MODULE CROSS TITLE CLEAR ce nt e Fig. I.4 (a) r.c NEXT LADDER SYMBOL MESAGE FUNCTN nc RETURN DELNET .c INSNET COMAND ADRESS SEARCH NEXT MOVE CHANGE Fig. I.4 (b) w w w COPY INSERT NEXT 952 APPENDIX B–61863E/10 J J. APPLICABLE FAPT LADDER EDITIONS APPLICABLE FAPT LADDER EDITIONS om The following tables list the editions of offline programs required to program each PMC model. J.1 PMC– PA1 PMC– PA3 PMC– RA1 PMC– RA2 PMC– RA3 PMC– RA5 PMC– RB PMC– RB2 PMC– RB3 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 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) – – – A02B–9201–J603#EN A02B–9200–J604#JP – – – – – – 1.0 and later 3.0 and later 4.5 and later 1.0 and later 2.0 and later – 1.0 and later 2.0 and later – – 1.0 and later 1.0 and later Drawing number A02B–9200–J502#JP A02B–9201–J502#EN A02B–9200–J603#JP A02B–9201–J604#EN – A08B–9201–J510 Drawing number – – 2.1 and later – 1.3 and later 2.2 and later – – PMC– RB4 PMC– RB5 PMC– RB6 PMC– RC PMC– RC3 PMC– RC4 PMC– NB PMC– NB2 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 .c Model nc A08B–9201–J503 ce nt e Model r.c FAPT LADDER, FAPT LADDER–II, LADDER EDITING PACKAGES A02B–9201–J502#EN 7.1 and later A02B–9200–J603#JP – – – 1.0 and later 4.5 and later 5.0 and later – – 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 1.1 and later – 1.0 and later 1.0 and later 1.0 and later 1.0 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 w A02B–9200–J502#JP w A02B–9201–J603#EN A02B–9200–J604#JP w A02B–9201–J604#EN Body of FAPT LADDER Body of FAPT LADDER PMC–RA module PMC–RA module PMC–RB/RC module 953 (PC–9801) (IBM PC/AT) (PC–9801) (IBM PC/AT) (PC–9801) A02B–9200–J502#JP A02B–9201–J502#EN A02B–9200–J603#JP A02B–9201–J603#EN A02B–9200–J604#JP J. APPLICABLE FAPT LADDER EDITIONS APPENDIX PMC–RB/RC module PMC–NB module PMC–NB module FAPT LADDER–II Ladder editing package B–61863E/10 (IBM PC/AT) (PC–9801) (IBM PC/AT) (IBM PC/AT) (IBM PC/AT) A02B–9201–J604#EN A02B–9200–J606#JP A02B–9201–J606#EN A08B–9201–J503 A08B–9201–J510 w w w .c nc ce nt e r.c om NOTE When a PMC–RA5 ladder is to be created, set the model to PMC–RA3. 954 APPENDIX B–61863E/10 J. APPLICABLE FAPT LADDER EDITIONS J.2 A08B–0035–J595#E (P–G Mark II): FAPT LADDER PMC–RA1/RA2 A08B–0036–J595#E (P–G Mate): Model Edition PMC– PA1 PMC– PA3 PMC– RA1 PMC– RA2 PMC– RA3 1.1 and later Ę 2.1 and later Ę Ę n Ę Ę n Ę 3.1 and later Ę 4.1 and later : Not supported, Ę om FAPT LADDER (SYSTEM P SERIES) Ę: Supported, n: Restrictedly supported (Note) ce nt e r.c NOTE A sequence program cannot be transferred from the PMC–RA1 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–RA1/RA2/RB/RC/PA1/PA3 data Model Edition PMC– PA1 PMC– PA3 Ę Ę 1.1 and later 2.1 and later PMC– RA2 PMC– RA3 Ę Ę Ę Ę: Supported, n: Restrictedly supported (Note) nc : Not supported, PMC– RA1 .c A08B–0035–J595#E (P–G Mark II): FAPT LADDER PMC–RB/RB2/RC A08B–0036–J595#E (P–G Mate): Model PMC– RB 1.1 and later Ę 4.1 and later Ę : Not supported, w w w Edition 955 PMC– RB2 PMC– RB3 PMC– RC PMC– RC3 Ę Ę n Ę Ę: Supported, n: Restrictedly supported (Note) PMC– NB J. APPLICABLE FAPT LADDER EDITIONS APPENDIX B–61863E/10 NOTE The edition of FAPT LADDER adopted for the PMC–RA2 or PMC–RB2 can be used to program the PMC–RA3 or PMC–RB3 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–RA2 or PMC–RB2). When this edition is used: r.c om (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 ce nt e (2)A sequence program created by the editing function (ladder editing module) contained in the PMC–RA3/RB3 cannot be edited after it is read into the offline programmer. w w w .c nc (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. 956 K K. LEVEL UP OF INPUT/OUTPUT FUNCTION WITH MEMORY CARD APPENDIX B–61863E/10 LEVEL UP OF INPUT/OUTPUT FUNCTION WITH MEMORY CARD K.1 OUTLINE OF LEVELED UP CONTENTS om 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. r.c (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. ce nt e (2) Sequence programs can be inputted from Memory Card by BOOT SYSTEM, by which CNC management software or so can be inputted. (Refer to K.2.3.) 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 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 nc non leveled up · PMC management software non leveled up leveled up 04–05 More than 06 w .c PMC–RA1/RA3 management software (4080) · FAPT LADDER for Personal Computer non leveled up 6.1 PMC–RA1/RA2 MODULE (PMC–RA1/RA2/RA3/PA1/PA3) (A08B–9200–J603#JP (PC–9801)) (A08–9201–J603#EN (IBM PC/AT)) 4.1 w w FAPT LADDER PMC–RA1/RA2/RB/RB2/RC SYSTEM (A08B–9200–J502#JP (PC–9801)) (A08–9201–J502#EN (IBM PC/AT)) 957 6.2 leveled up More than 6.3 More than 4.2 K. LEVEL UP OF INPUT/OUTPUT FUNCTION WITH MEMORY CARD APPENDIX B–61863E/10 K.2 OPERATION K.2.1 CNC ³Offline Programmer (1) Operation of CNC STOP = 1 = M–CARD = WRITE = LADDER = ce nt e CHANNEL DEVICE FUNCTION DATA KIND FILE NO. (#NAME) MONIT r.c PMC I/O PROGRAM om 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]. nc [EXEC][CANCEL][WRITE][READ][COMPAR] [DELETE][LIST][FORMAT][ ][SETUP] Fig. K.2.1(a) PMC I/O Screen w w w .c (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. 4) Select [M–CARD] (Memory Card) from the I/O menu. (See Fig. K.2.1(b)) I/O PMC–RA1 F1 KEY : F2 KEY : PMC WRITER F3 KEY : PMC F4 KEY : Handy File F5 KEY : F10 KEY : FAWRT PMC WRT PMC [A:¥FLADDER¥ Memory Card END FDCAS M CARD Fig. K.2.1(b) I/O Menu Screen 958] FA WRITER END K. LEVEL UP OF INPUT/OUTPUT FUNCTION WITH MEMORY CARD APPENDIX B–61863E/10 5) Select [READ] (PROGRAMMER ← Memory Card). (See Fig. K.2.1(c)) I/O (M_CARD) PMC–RA1 [A:¥FLADDER¥ F1 KEY : WRITE (PROGRAMMER –> Memory Card) F2 KEY : READ (PROGRAMMER END WRITE READ om F10 KEY :] END Fig. K.2.1(c) I/O (I/O M_CARD) Screen r.c 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. ce nt e – – Name of the ROM format file to be created Specify a file name to be given to the converted ROM format data. I/O (FROM MC) PMC–RA1 [A:¥FLADDER¥] READ (PROGRAMMER nc Memory Card FILE NAME : (Specify the MEMORY CARD drive) ROM FORMAT FILE NAME : END 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. w w w .c EXEC 959 K. LEVEL UP OF INPUT/OUTPUT FUNCTION WITH MEMORY CARD APPENDIX B–61863E/10 K.2.2 Offline Programmer ³ CNC (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). om 4) From the I/O menu, select {M–CARD] (memory Card). 5) Select [WRITE] (PROGRAMMER Fig. K.2.1(b)).

6) Specify the following: ³ Memory Card). (See 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.) ce nt e r.c – I/O (TO MC) PMC– RA1 [A:¥FLADDER¥] WRITE(PROGRAMMER –> Memory Card) ROM FORMAT FILE NAME : nc Memory Card FILE NAME : (Specify the MEMORY CARD drive) .c EXEC END Fig. K.2.2(a) I/O (TO MC) Screen w w w (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. 960 APPENDIX B–61863E/10 K. LEVEL UP OF INPUT/OUTPUT FUNCTION WITH MEMORY CARD K.2.3 Note (1) ce nt e r.c om Sequence programs which are output from leveled up CNC or Offline Programmer to Memory Card can not be input to non leveled up CNC or Offline Programmer. (Refer to the table of K.1) (4) (3) .c nc (2) Offline Programmer w CNC w w : Leveled up : Available with no condition : Available with some condition : Non leveled up 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) 961 K. LEVEL UP OF INPUT/OUTPUT FUNCTION WITH MEMORY CARD · APPENDIX B–61863E/10 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 1 M–CARD WRITE LADDER om = = = = = STOP r.c CHANNEL DEVICE FUNCTION DATA KIND FILE NO. (#NAME) MONIT ce nt e [EXEC][CANCEL][WRITE][READ][COMPAR] [DELETE][LIST][FORMAT][PMC SETUP M–CARD][SETUP] MONIT 1 [INPUT][w w w .c nc OUTPUT FORMAT (PROGRAM) = (0:BINARY,1:S–FORMAT) STOP][ ][ ][INIT] Press [INIT] key to reset default value 0.

Input operation 962 : Input sequence programs by selecting F6:“I/O” on main menu screen of FAPT LADDER, then F3:“Handy File & Memory Card”. APPENDIX B–61863E/10 · K. LEVEL UP OF INPUT/OUTPUT FUNCTION WITH 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 sequence programs by selecting F6:“I/O” on main menu screen of FAPT LADDER, then F3:“Handy File & Memory Card”. w w w .c nc ce nt e r.c om Input operation 963 K. LEVEL UP OF INPUT/OUTPUT FUNCTION WITH MEMORY CARD APPENDIX B–61863E/10 K.3 Ladder data can be transferred by using a memory card. NB/NB2 DATA COMPATIBILITY Two data formats are used: – Handy file format – Memory card format r.c om The handy file format defines the S format data used with RS–232C. The memory card format defines the binary format data used for a boot. Data output to a memory card from the I/O screen of an NB of the 4047 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.

Data Transfer Between NB (4047 series) and FAPT LADDER Transfer in handy file format ce nt e K.3.1 FAPT LADDER (personal computer version) Transfer function not provided Transfer function not provided NC boot function NB (4047) nc PMC I/O screen NB (4047) K.3.2 w .c Data Transfer Between NB/NB2 (4048 series) and FAPT LADDER w Transfer in memory card format FAPT LADDER (personal computer version) Transfer in memory card format w Transfer in handy file format PMC I/O screen NB/NB2 (4048) Transfer in memory card format 964 NC boot function NB/NB2 (4048) APPENDIX B–61863E/10 K. LEVEL UP OF INPUT/OUTPUT FUNCTION WITH MEMORY CARD K.3.3 Data Transfer Between NB (4047 series) and NB (4048 series) Transfer in handy file format Transfer in memory card format NC boot function NB (4047) Transfer in memory card format w w w .c nc ce nt e Transfer function not provided om FAPT LADDER (personal computer version) Transfer in handy file format PMC I/O screen NB (4048) r.c PMC I/O screen NB (4047) 965 NC boot function NB (4048) L. ALARM MESSAGE LIST L APPENDIX B–61863E/10 ALARM MESSAGE LIST Alarm messages 1 (alarm screen) Contents and solution om Message Normal status 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 FANUC.

Change the value of MAX LADDER AREA SIZE on the SYSPRM screen, then turn the power off then back on (only with PMC–RC). ce nt e r.c ALARM NOTHING 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 programmer. nc ER03 PROGRAM SIZE ERROR (OPTION) 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–RC only). For a 3–path system, the PMC model must be RB6. (solution) Contact FANUC. .c ER05 PMC MODULE TYPE ERROR There is no step number option of LADDER. ER10 OPTION AREA NOTHING (SERIES–NAME) The PMC–RB 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. w w ER07 NO OPTION (LADDER STEP) w ER12 OPTION AREA ERROR (SERIES–NAME) There is a series mismatch between the basic and option of the PMC–RB management software. (solution) Contact FANUC. 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 There is an edition mismatch between the basic and option of the (SERIES–NAME) PMC–RB management software. (solution) Contact FANUC. ER15 OPTION AREA VERSION ERROR There is an edition mismatch between the basic and option of the PMC C (SERIES–NAME) language board management software. (solution) Contact FANUC. 966 APPENDIX B–61863E/10 Message L. ALARM MESSAGE LIST Contents and solution ER16 RAM CHECK ERROR (PROGRAM RAM) The debugging RAM cannot be read/written normally. (solution) Please exchange the debugging RAM. 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 RAM: Please edit the sequence program once on PMC Still the error occurs, exchange the debugging RAM. F–ROM: (PMC–NB/FS–20) Please edit the sequence program once on PMC and write sequence program to F–ROM again. om ER17 PROGRAM PARITY w w w .c nc ce nt e r.c NOTE 1 The PMC–RB3/RC3 for the Series 16 MODEL–B does not support ER00 and ER06. 2 For the PMC–RB3/RC3 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.

967 L. ALARM MESSAGE LIST APPENDIX B–61863E/10 Alarm messages 2 (alarm screen) Message Contents and solution Transferring the sequence program from offline programmer was interrupted by the power off etc. (solution) Please clear the sequence program and transfer the sequence 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 There is no sequence program. 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. ce nt e r.c om ER18 PROGRAM DATA ERROR BY I/O ER24 LADDER, LANGUAGE AREA OVERLAP The ladder area overlaps the C language area. (solution) Adjust the C program address range. ER32 NO I/O DEVICE Any DI/DO unit of I/O Unit or the connection unit etc.

is not connected. 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 connected with. When I/O Link is used: Please confirm whether the DI/DO units turning on. Or please confirm the connection of the cable. .c ER34 SLC ERROR (xx) The LSI for I/O Link is defective. (solution) Please exchange the module of PMC engine.

nc ER33 SLC ERROR The communication with the DI/DO units of the xx group failed. (solution) Please confirm the connection of the cable connected to the DI/ DO units of the xx group. Please confirm whether the DI/DO units turned on earlier than CNC and PMC.

Or, please exchange the module of PMC engine on the DI/DO units of the xx group. w w ER35 TOO MUCH OUTPUT DATA IN GROUP The number of the output data in the xx group exceeded the max.

The (xx) data, which exceed 32 bytes, become ineffective. (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) w ER36 TOO MUCH INPUT DATA IN GROUP (xx) The number of the input data in the xx group exceeded the max. The data, which exceed 32 bytes, become ineffective. (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) 968 APPENDIX B–61863E/10 Message L.

ALARM MESSAGE LIST Contents and solution ER38 MAX SETTING OUTPUT DATA OVER (xx) The assignment data for a group exceeds 128 bytes. (The assignment data of output side of xx group or later become ineffective.) (solution) Please reduce the assignment data to 128 bytes or less for the number of the output data of each group. The assignment data for a group exceeds 128 bytes. (The assignment data of input side of xx group or later become infective.) (solution) Please reduce the assignment data to 128 bytes or less for the number of the input data of each group. w w w .c nc ce nt e r.c om ER39 MAX SETTING INPUT DATA OVER (xx) 969 L.

ALARM MESSAGE LIST APPENDIX B–61863E/10 Alarm messages 3 (alarm screen) Message Contents and solution 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 confirming whether there is a problem in LADDER or not. WN04 UNAVAIL EDIT MODULE The LADDER editing module cannot be recognized. (PMC–RA1/RA2/RA3/RB/RB2/RB3, except RA1/RA3 for FS–20) (solution) Please confirm the slot position installed. Please confirm the installed module. WN05 PMC TYPE NO CONVERSION A PMC–RA3/RA5 ladder was transferred to PMC–RB5. (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. WN18 ORIGIN ADDRESS ERROR WN19 GDT ERROR (BASE, LIMIT) r.c There is no C language option.

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. 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. 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. nc WN20 COMMON MEM. COUNT OVER ce nt e WN17 NO OPTION (LANGUAGE) om WN01 LADDER MAX SIZE ERROR .c WN21 COMMON MEM. ENTRY ERROR w WN22 LADDER 3 PRIORITY ERROR w WN23 TASK COUNT OVER w WN24 TASK ENTRY ADDR 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. The priority of LADDER LEVEL 3 is out of range. (solution) Please correct the value of LADDER LEVEL 3 in the link control statement within the range of 0 or 10–99 or –1. 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 correct the link control statement, build file and the composition of the files to be linked. 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). 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 control 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 LADDER LEVEL 3). 970 L. ALARM MESSAGE LIST APPENDIX B–61863E/10 Alarm messages 4 (alarm screen) Message Contents and solution 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 WN32 LNK CONTROL DATA ERROR 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 version. 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. A link control statement data version error occurred. (solution) Correct the link control statement in the C program. nc WN33 LNK CONTROL VER. ERROR ce nt e r.c om WN27 CODE SEG TYPE ERROR 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. .c WN34 LOAD MODULE COUNT OVER 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) Clear the language area. [EDIT]³[CLEAR]³[CLRLNG]³[EXEC] 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, replace the RAM. w w w WN36 LANGUAGE SIZE ERROR (OPTION) WN40 PROGRAM DATA ERROR BY I/O Language program read operation was interrupted. (LANG.) (solution) Reenter the language program. WN41 LANGUAGE TYPE UNMATCH WN42 UNDEFINE ADDRESS LANGUAGE There is a C program type mismatch. (solution) Correct the C program. ORIGIN No language origin address is set. (solution) Set a language origin address. 971 L. ALARM MESSAGE LIST APPENDIX B–61863E/10 System alarm messages 1 (PMC–RC) Message STATUS LED Contents and solution PC1nn CPU INTERPT xxxx yyyyyy lL om STATUS LED A CPU error (abnormal interrupt) occurred.

nn : CPU exception handling code It is an exception code of i80386. For details, please refer to the manual of the CPU. 00 Division error such as a divisor is 0 in division instruction. 12 Stack exception such as violations of limit of stack segment. 13 General protection exception such as segment limit over. xxxx : Segment selector where system error occurred. The selector of 0103–02FB is used by C language. yyyyyy: Offset address where system error occurred. PC130 RAM PARITY aa xxxx yyyyyy STATUS LED The parity error occurred on the debugging RAM of PMC. aa : RAM PARITY ERROR information. xxxx : Segment selector where system error occurred. yyyyyy: Offset address where system error occurred. jL The RAM parity error or NMI(Non Maskable Interrupt) generated in module of PMC engine. bb : RAM PARITY ERROR information. 1, 2, 4, 8 Parity error occurred on basic DRAM. 14, 18 Parity error occurred on option DRAM. 20, 60, A0, E0 Parity error occurred on SRAM. xxxx : Segment selector where system error occurred. yyyyyy: Offset address where system error occurred. STATUS LED LJ ce nt e r.c PC140 NMI BOC bb xxxx yyyyyy System alarm messages 2 (PMC–RC) Message STATUS LED Contents and solution The communication error occurred in the I/O Link.

aa, cc : I/O Link error information. This error may occur by the following causes. 1.When I/O Unit–MODEL A is used, base1, 2 or 3 is not connected though allocated. 2.The connection of cable is insufficient. 3.Defects of cable. 4.Defects of DI/DO units (I/O unit, Power Mate etc.) 5.Defects of PMC board (printed circuit board on host side where I/O Link cable is connected.) (solution) Investigate the cause of error. 1.Please confirm the allocation data (by "EDIT"→"MODULE" screen) and compare with the actual connection. 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. Please investigate a defective place by the following methods. 3.Please confirm the specification of the cable referring to "FANUC I/O Unit–MODEL B connecting manuals(B–62163E)". 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.

Please investigate the cause of noise. w w w .c nc PC150 NMI SLC aa cc STATUS LED JL PC160 F–BUS ERROR xxxx:yyyyyyyy PC161 F–BUS ERROR xxxx:yyyyyyyy PC162 F–BUS ERROR xxxx:yyyyyyyy STATUS LED Lj 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 Lj 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 Ll The parity error occur in PMC system ROM. eeeeeeee : ROM parity error information. 972 APPENDIX B–61863E/10 L. ALARM MESSAGE LIST STATUS LED (green) are LED1, LED2 on PMC–RC. CAP–II is LED3 and LED4. j : Off J : On lL : Blinking w w w .c nc ce nt e r.c om NOTE 1 The system error on PMC–RA1,RA2,RA3,RB,RB2 and RB3 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)".) 2 Error information is needed to investigate on FANUC, please take notes of it. 973 L. ALARM MESSAGE LIST APPENDIX B–61863E/10 System alarm messages 3 (PMC–NB/NB2) Message STATUS LED Contents and solution RAM ERROR bbcc xxxx: yyyyyyyy: PC010 STATUS LED LJ or jL 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. The parity error occurs in PMC system ROM. aaaaaaaa : ROM parity information DIVIDE ERROR xxxx: yyyyyyyy: PC040 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. STATUS LED om ROM ERROR aaaaaaaa: PC020 STATUS LED lL lL STATUS LED The BUS error (access on illegal address). xxxx : Segment selector where system error occurred. yyyyyyyy : Offset address where system error occurred. r.c BUS ERROR xxxx: yyyyyyyy: PC040 lL STACK FAULT xxxx: yyyyyyyy: PC040 GENERAL PC040 lL PROTECTION STATUS LED xxxx: lL yyyyyyyy: The general protection exception such as segment limit over was generated. xxxx : Segment selector where system error occurred. yyyyyyyy : Offset address where system error occurred. The communication error occurred in the I/O Link. aa, cc : I/O Link error information. This error may occur by the following causes. 1.When I/O Unit–MODEL A is used, base1, 2 or 3 is not connected though allocated. 2.The connection of cable is insufficient. 3.Defects of cable. 4.Defects of DI/DO units (I/O unit, Power Mate etc.) 5.Defects of PMC board (printed circuit board on host side where I/O Link cable is connected.) (solution) Investigate the cause of error. 1.Please confirm the allocation data (by "EDIT"→"MODULE" screen) and compare with the actual connection. 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. Please investigate a defective place by the following methods. 3.Please confirm the specification of the cable referring to "FANUC I/O Unit–MODEL B connecting manuals(B–62163E)". 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. Please investigate the cause of noise. w w .c nc SLC ERROR aa (cc) : PC050 ce nt e 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. w STATUS LED JL STATUS LED (green) are LED1, LED2 on PMC–NB. j : Off J : On lL : Blinking NOTE Error information is needed to investigate on FANUC, please take notes of it 974 L. ALARM MESSAGE LIST APPENDIX B–61863E/10 System alarm messages (PMC–RB5/RB6) Message Contents and solution PC0nn CPU INTERRUPT xxxxxxxx CPU error nn :Exception code xxxxxxxx :Address at which an error occurred xxxxxxxx:yyyyyyyy xxxxxxxx:yyyyyyyy xxxxxxxx:yyyyyyyy xxxxxxxx:yyyyyyyy A CPU error occurred on the PMC. xxxxxxxx and yyyyyyyy represent an internal error code. This error may be caused by a main board failure. om PC004 CPU ERR PC006 CPU ERR PC009 CPU ERR PC010 CPU ERR (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 configuration, operation, error occurrence timing, error occurrence frequency, and so forth). aa:bb A RAM parity error occurred on the PMC. aa and bb represent an internal error code. r.c PC030 RAM PARITY This error may be caused by a main board failure. PC040 NMI BOC xxxxxxxx ce nt e (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 configuration, operation, error occurrence timing, error occurrence frequency, and so forth) and the internal error code above.

A non–maskable interrupt (NMI) occurred in the PMC control module. xxxxxxxx :Address at which an error occurred PC050 NMI SLC aa:bb A communication error occurred on the I/O Link. aa and bb represent an internal error code. w w w .c nc This error may occur as a result of one of the following: (1) When I/O Unit A is used, an I/O assignment is made for base expansion, but no base is connected. (2) Cables are not connected correctly. (3) Cables are faulty. (4) I/O devices (I/O Unit, Power Mate, and so forth) are faulty. (5) The main board is faulty. PC060 FBUS PC061 FL–R PC062 FL–W (solution) (1) Check if the I/O assignment data matches the actual connections of the I/O devices. (2) Check if the cables are connected correctly. (3) Check the cable specifications by referring to the "FANUC I/O Unit– MODEL A Connection and Maintenance Manual (B–61813EN)" or "FANUC I/O Unit–MODEL B Connection Manual (B–62163EN). (4) Replace the interface module, cables, and main board of the I/O Unit, then check whether this error occurs again. 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. xxxxxxxx:yyyyyyyy xxxxxxxx:yyyyyyyy aa:xxxxxxxx:yyyyyyyy A bus error occurred on the PMC. aa, xxxxxxxx, and yyyyyyyy 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. 975 L. ALARM MESSAGE LIST APPENDIX Message B–61863E/10 Contents and solution PC070 SUB65 CALL (STACK) A stack error occurred with the ladder functional instruction CALL/CALLU. (solution) Check the correspondence between the CALL/CALLU instruction and the SPE instruction. If the cause of the fault cannot be found, contact FANUC with the error status information and the ladder program.

PC080 SYS EMG PC081 FL EMG xxxxxxxx:yyyyyyyy xxxxxxxx:yyyyyyyy A system alarm was issued due to other software. om (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 (LADDER) PC098 PARITY ERR (DRAM) PC099 PARITY ERR (SRAM) A parity error occurred on the PMC system. r.c This error may be caused by a main board failure. w w w .c nc ce nt e (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 occurrence frequency, and so forth). 976 M. EXAMPLE OF STEP SEQUENCE PROGRAMS APPENDIX B–61863E/10 M EXAMPLE OF STEP SEQUENCE PROGRAMS Connection Unit. CNC I/O Unit om The CNC is connected two or three Power Mate units. Y0.0: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 ce nt e r.c X0.0:Safety switch Power Mate #1 X100.0:Ready end#1 Y100.0:Emergency#1 X100.1:Driving#1 Y100.1:Drive start#1 Power Mate #2 Power Mate #3 X120.0:Ready end#3 Y120.0:Emergency#3 X120.1:Driving#3 Y120.1:Drive start#3 w w w .c nc X110.0:Ready end#2 Y110.0:Emergency#2 X110.1:Driving#2 Y110.1:Drive start#2 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) Driving #1 (Power Mate to CNC) 977 1sec 1sec 1sec M. EXAMPLE OF STEP SEQUENCE PROGRAMS APPENDIX B–61863E/10 The following flowchart illustrates the interface with the Power Mate units. CNC Power Mate x When the ready and signal is off, the driving start common is not sent. om The start signal (STx) is set on. STx The start signal (STx) is set off after 1 second. STx DENx r.c Start moving by the start signal and the DEN signal (DENx) is set on for CNC. ce nt e When the moving finishes, the NC program is rewound and the DEN signal (DENx) is set off. DENx Waiting for that the moving finishes. nc Finished w w w .c The interface with the Power Mate units is changed to the Step Sequence program. Step Sequence for the Power Mate Dummy Step Ready end Ready The start signal is set on. finished Wait for 1 second. The start signal is set off. Wait for that the TIMER finishes. The moving status is set the LED. Wait for that the moving finishes. finished 978 M. EXAMPLE OF STEP SEQUENCE PROGRAMS APPENDIX B–61863E/10 Example 1 The Step sequence program for three sequentially driven Power Mate units: Main program L1 The input signal is set the LED. Drive Power Mate#1 Dummy trasition Drive Power Mate#2 Dummy trasition r.c Drive Power Mate#3 om The input signal is 1 (true). Dummy trasition Start the timer. ce nt e Wait for one second. Clear the timer. Wait for that the timer finishes. w w w .c nc L1 979 M. EXAMPLE OF STEP SEQUENCE PROGRAMS Example 2 APPENDIX B–61863E/10 The Step Sequence program for three simultaneously driven Power Mate units: Main program L1 The input signal is set the LED. om The input signal is 1 (true). Dummy trasition Start the timer. Drive Power Mate#2 Wait for one second.

ce nt e Clear the timer. Wait for that the timer finishes.

w w w .c nc L1 980 Drive Power Mate#3 r.c Drive Power Mate#1 APPENDIX B–61863E/10 N N. STEP SEQUENCE CORRESPONDED C LANGUAGE STEP SEQUENCE CORRESPONDED C LANGUAGE N.1 The operation is continued while the condition is true. om WHILE STATEMENT L1 condition r.c Format The condition is false. The condition is true. ce nt e operation Operation finished. w w w .c nc L1 981 N. STEP SEQUENCE CORRESPONDED C LANGUAGE APPENDIX B–61863E/10 Examples L1 S1 (P10) P11 P12 S2 (P20) P21 om L1 P11 R9091.1 COMPB 1 30 R10 ce nt e R9000.0 r.c P10 TRSET P12 R9091.1 COMPB 1 30 R10 R9000.0 nc TRSET w w w .c P20 R9091.0 MULB 1 R10 5 R12 ADDB 1 R10 1 R10 R9091.1 R9091.0 R9091.1 P21 R9091.1 TRSET 982 R0.0 ( ) R0.0 ( ) APPENDIX B–61863E/10 N. STEP SEQUENCE CORRESPONDED C LANGUAGE N.2 DO–WHILE STATEMENT The operation is continued while the condition is true after executing the operation. The difference between do–while and while is that the operation is executed at least one time. L1 operation condition The condition is true. r.c The condition is false. om Format w w w .c nc ce nt e L1 983 N.

STEP SEQUENCE CORRESPONDED C LANGUAGE APPENDIX B–61863E/10 Examples L1 S1 (P20) P21 S2 (P10) P11 P12 om L1 P11 R9091.1 COMPB 1 30 R10 ce nt e R9000.0 r.c P10 TRSET P12 R9091.1 COMPB 1 30 R10 R9000.0 nc TRSET w w w .c P20 R9091.0 MULB 1 R10 5 R12 ADDB 1 R10 1 R10 R9091.1 R9091.0 R9091.1 P21 R9091.1 TRSET 984 R0.0 ( ) R0.0 ( ) APPENDIX B–61863E/10 N.

STEP SEQUENCE CORRESPONDED C LANGUAGE N.3 FOR STATEMENT After the initial data is set, the operation is continued while the condition is true. Format om Set the initial data. L1 operation r.c condition The condition is false. The condition is true. w w w .c nc ce nt e L1 985 N. STEP SEQUENCE CORRESPONDED C LANGUAGE APPENDIX B–61863E/10 Examples S1 (P1) P2 L1 S2 (P20) P21 P11 om S3 (P10) P12 L1 P1 R9091.1 ce nt e P2 R9091.1 1 0 R10 r.c NUMEB TRSET P20 R9091.0 MULB R9091.1 1 R10 5 R12 R0.0 ( ) P21 nc R9091.1 TRSET w w w .c P10 R9091.0 ADDB 1 R10 1 R10 COMPB 1 30 R10 R9091.1 P11 R9091.0 R9000.0 TRSET P21 R9091.0 COMPB R9000.0 TRSET 986 1 30 R10 R0.0 ( ) APPENDIX B–61863E/10 N. STEP SEQUENCE CORRESPONDED C LANGUAGE N.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 false. operation1 operation2 w w w .c nc ce nt e r.c om The condition is true. 987 N.

STEP SEQUENCE CORRESPONDED C LANGUAGE APPENDIX B–61863E/10 Examples S1 (P10) P11 P12 S2 (P20) S3 (P22) P21 P23 R9091.0 COMPB R9000.0 1 30 R10 ce nt e TRSET r.c P1 1 om P1 0 P12 R9091.0 COMPB 1 30 R10 R9000.0 TRSET nc P20 R9091.0 MULB w w w .c R9091.1 1 R10 8 R12 R0.0 ( ) P21 R9091.1 TRSET P22 R9091.0 MULB R9091.1 P23 R9091.1 TRSET 988 1 R10 10 R12 R0.0 ( ) APPENDIX B–61863E/10 N. STEP SEQUENCE CORRESPONDED C LANGUAGE N.5 SWITCH STATEMENT The operation connected to the condition is executed.

Format condition condition2 condition3 operation1 operation2 operation3 w w w .c nc ce nt e r.c om condition1 989 N. STEP SEQUENCE CORRESPONDED C LANGUAGE APPENDIX B–61863E/10 Examples S1 (P10) P11 P12 P13 S2 (P20) S3 (P22) S4 (P24) P21 P23 P25 R9091.1 COMPB R9000.0 1 30 R10 ce nt e TRSET r.c P1 1 om P1 0 P12 R9091.0 COMPB 1 40 R10 R9000.0 TRSET nc P13 w w w .c R9091.1 TRSET P20 R9091.0 MULB R9091.1 1 R10 5 R12 R0.0 ( ) P21 R9091.1 TRSET P22 R9091.0 MULB R9091.1 990 1 R10 10 R12 R0.0 ( ) APPENDIX B–61863E/10 N. STEP SEQUENCE CORRESPONDED C LANGUAGE P23 R9091.1 TRSET MULB R9091.1 P25 R9091.1 R0.0 ( ) w w w .c nc ce nt e r.c TRSET 1 R10 15 R12 om P24 R9091.0 991 0. CHINESE CHARACTER CODE, HIRAGANA CODE, AND SPECIAL CODE LIST APPENDIX * Replace all tables with follows. Pronun ciation Segment and point JIS Shift JIS 0123456789ABCDEF alpha nume 2330 824F 0 l-k 2350 2440 2450 2460 2470 2520 2530 2540 2550 2560 2570 2620 2630 8290 8293 B2AE B2BE BZCE BPDE 62EE 833F 834F 835F 836F 8380 8390 8393 8 9 63AE BSBE 63CE .c 2640 2650 826F 8280 7 r.c 2430 : 6 ce nt e ra ga lla G r e 5 nc hi .

4 825F 2360 2370 2420 % 3 12 om CHINESE CHARACTER CODE, HIRAGANA CODE, AND SPECIAL w w w X The characters with mark % cannot be displayed on FANUC Series 16-MODEL A. - 992 - 4 0. CHINESE APPENDIX 6--61863E/lO 7t n 2 3 4 5 6 w w 4 1 3560 9 7 8 9 A B C D E F om L 0 r.c 3 Segment and point ce nt e 4 1020 I030 1040 FzjI050 1060 1070 $120 Kz $130 3140 3140 3150 3160 3170 3170 3220 3230 3230 3240 3250 3260 3270 3320 3330 3340 3350 3360 3370 3420 3430 3440 3450 3460 3460 3470 3520 3530 3540 3550 Shift JIS nc 7 JIS .c Vonun :iation 3570 3620 3630 3640 3650 3660 3660 3670 3720 w 1AL CHARACTER CODE, HIRAGANA CODE, AND SPECIAL CODE LIST - 993 - 0. CHINESE CHARACTER CODE, HIRAGANA CODE, AND SPECIAL CODE LIST 8-61863Em ronun iation Shin JIS 'V x w w w .c nc 3870 3920 3930 3940 3950 3960 3970 3A20 3A30 3A30 3A40 3A50 3A60 3A70 3B20 3B30 3B40 3B40 3B50 3B60 3B70 3C20 3C30 3C40 3c50 3c60 3c70 3D20 3D30 3D40 3D50 3D60 4 om 3840 8C4F 8C4F 8C5F 8C6F 8C80 8C90 8C9E SCAE 8CBE 8CBE 8CCE 8CDE 8CEE 8D3F 8D4F 8D5F 8D6F 8D80 8D90 8D9E 8DAE 8DAE 8DBE 8DCE 8DDE 8DEF 8E3F 8E4F 8E5F 8E5F 8E6F 8E80 8E90 8E9E 8EAE 8EBE 8ECE 8EDE 8EEE 8F3F 8F4F 8F5F 8F6F 8F80 8F90 8F9E 8FAE 8FBE 8FCE 8FDE r.c JIS ce nt e Pronun ciation APPENDIX 3D70 3E20 3E30 3E40 3E50 3E60 ;; -i .. .. .: " .,...(_...~ /.. '.'!I .:. .;;,. I:;,. shows the character which is impossible to display. 0. CHINESE APPENDIX &61863010 prDnun JIS Shift JIS ciation Segment and point 0123456789A~cn~.~ r.c ce nt e nc .c 909E 9OAE 90BE 90CE 90DE 9oEE 913F 914F m 915F 916F 9180 9190 919E 91AE 9lAE 91BE 91CE 91DE 91EE 923F 924F 925F 915F 926F 9280 9290 9293 92AE 92BE 92CE 92DE 92DE 92EE 933F 934F 9353 9353 936F 9380 om 8FEE 903F 904F 905F 906F 906F 9080 9090 909E w w 3E70 3F20 3F30 3F40 3F50 3F50 3F60 3F70 4020 4020 4030 4040 4050 4060 4070 4120 4130 4130 4140 4150 4160 4170 4220 4230 4230 4240 4250 4260 4270 4320 4330 4340 4340 4350 4360 4370 4420 4430 4450 -izz 4470 4520 4530 4540 4540 4550 4560 w f --r CHARACTER CODE, HIRAGANA CODE, AND SPECIAL CODE LIST ;..::. F';".' ?I..':.: :j:;:;:;;,. .,.: -'.:;;:;,. :'+:: shows the character which is impossible to display. - 995 - 0. CHINESE CHARACTER CODE, HIRAGANA CODE, AND SPECIAL CODE LIST Pronun ciation APPENDIX ~1863Vlcl Pronun ciation JIS r.c ce nt e nc 4850 zz 4860 4870 4920 4930 4940 4950 4950 4960 4970 MO MO lm a40 &A50 4A50 4A60 4A70 4B20 4B30 4B40 4B50 ZiY 4B70 4C20 .c / % om 4570 4620 4630 4640 4650 ziz 4670 4670 4720 4720 4720 4730 4730 4740 4740 4750 4760 4770 4820 4830 w 4 d- w w ,\ 7 zzii 4c30 4 4c30 _:~~~.~~~~~~~ail~~ ... shows the character which is impossible to display. -!W6- . 0. CHINESE CHARACTER CODE, HIRAGANA CODE, AND SPECIAL CODE LIST w w w .c nc ce nt e r.c om APPENDIX - 997 - Index B–61863E/10 [A] ADD (Addition), 191 ADDB (Binary Addition), 193 Additions to Sequence Programs, 665 Address, 42 Address Change of Sequence Program, 453 Address of Counter (C), 73 om Address of Data Table (D), 77 Address of Keep Relay and Nonvolatile Memory Control (K), 75 Addresses, 304 Addresses Between PMC and CNC (PMCNC), 52 r.c Addresses Between PMC and Machine Tool (PMCMT), 53 Addresses Between PMC and Machine Tool for PMCRB/RC, 53 ce nt e Addresses for Message Selection Displayed on CRT (A), 71 Addresses, Signal Names, Comments, and Line Numbers, 304 Alarm Message List, 966 Alarm Screen (Alarm), 356 Alter, 637 Alteration of Sequence Programs, 445 AND, 90 AND. STK, 93 nc AND. NOT, 90 Applicable Fapt Ladder Editions, 953 Applications, 312 .c Arbitrary Functional Instructions, 243 Area Managed by the System Program, 68 w Assignment of I/O Module Addresses, 55 Automatic Operation when the Power is Turned on, 340 w Automatic Tracing Function at Power on, 359 Axctl (Axis Control by PMC), 285 w CALLU (Unconditional Subprogram Call), 279 Cautions for Subroutines, 319 Chinese Character Code, Hiragana Code, and Special Code List, 992 Clearing the PMC Parameter, 466 Clearing the Sequence Program, 340, 465 Clearing the Sequence Program and Condensation of the Sequence Program, 465 CNC → Offline Programmer, 958 COD (Code Conversion), 148 CODB (Binary Code Conversion), 151 Coding (Step 5), 22 COIN (Coincidence Check), 177 Collation of Program, 657 Collation of Source Programs, 657 COM (Common Line Control), 156 COME (Common Line Control End), 161 Comments, 305 COMP (Comparison), 173 Compatibility of Ladder Diagram, 603 Compatibility with CNC Basic Software, 947 COMPB (Comparison Between Binary Data), 175 Component Units, 423, 611, 948 Component Units and Connections, 422, 610, 948 Compress the Sequence Program, 466 Compressed Input by [COMAND] Key, 674 Condense, 950 Condense Command – Release of Deleted Area, 686 Configuration and Operation of Step- Sequence Programs, 562 Configuration of Command, 683 Confirming the Ladder Mnemonics, 526 Connecting Component Units, 428 Connecting the I/O Card, 941 Connecting the I/O Unit, 941 Connecting the Operator's Panel for FS 0 with FS16, FS18, FS21, or Power Mate, 938 Connection, 941 Connection of Components, 949 Connections of Units, 612 Convergence of Selective Sequence, 567 Convergence of Simultaneous Sequence, 569 Conversion, 935 Copy a Ladder Program [CPYLAD], 495 Copy I/O Module Data [CPYMDL], 496 Copy Message Data [CPYMSG], 495 Copy Symbol Data and Comment Data [CPYSYM], 495 Copy the Sequence Programs [CPYALL], 496 Copy Title Data [CPYTTL], 495 Copying a Sequence Program, 671 Copying Message Data (COPY), 464 Copying the Sequence Program, 451 [B] Block Step, 571 [C] C Input/Output, 477 CALL (Conditional Subprogram Call), 278 i–1 Index B–61863E/10 Corresponding Function, 602 Display of Ladder Diagram, 643 Counter Screen (COUNTR), 390 Display of Signal Status (Status), 355 Creating a Program, 317 Displayed Items, 503 Creation of Interface Specifications (Steps 1 to 3), 21 Displaying and Setting the Configuration Status of I/O Devices(IOCHK), 370 Creation of Ladder Diagram (Step 4), 21 Displaying Input Code (DSPMOD), 464 Cross Reference Display, 467 Displaying of Sequence Program, 587 CRT/MDI Operation, 586 Displaying the Contents of Memory, 360 CTR (Counter), 133 Displaying the GDT (Global Descriptor Table), 498 om CTRC (Counter), 139 Displaying the Memory Allocation Information of a User Program CODED IN C., 501 Displaying the Running State of a User Task (USRDGN), 368 [D] Displaying Title Data, 353 Data Input to and Output from other Devices, 489 DIVB (Binary Division), 205 Data Keys and Screen Scroll Key, 618 Divergence of Selective Sequence, 567 Data Number, Data Attribute, Data Length, Data Area, 929 Divergence of Simultaneous Sequence, 568 r.c Data Display and Setting (Title, Symbol, Ladder Program, Comment, Message, I/O Module), 630 ce nt e DIV (Division), 203 Dividing Display of Ladder Diagram, 415 Data Table (DATA), 396 Do–while Statement, 983 Data Transfer Between NB (4047 series) and FAPT LADDER, 964 DSCH (Data Search), 181 DSCHB (Binary Data Search), 184 Data Transfer Between NB (4047 series) and NB (4048 series), 965 Dump Display on Ladder Diagram, 407 Data Transfer Between NB/NB2 (4048 series) and FAPT LADDER, 964 DCNV (Data Conversion), 169 [E] Debugging, 504 DEC (Decode), 129 nc DCNVB (Extended Data Conversion), 171 Editing a Character String in Message Data, 463 Editing Character Strings of Symbol Data and Comment Data, 460 Editing Character Strings of Title Data, 438 Delete, 641 Delete of Sequence Program, 448 Editing for Power Mate–MODEL D (PMC–PA1/PA3), 947 Delete of Symbol data and Comment, 459 Editing Function of Ladd ER Diagram, 597 Deleting a Sequence Program, 668 Editing Ladder Mnemonics, 525 Deleting Title Data, 437 Editing of Sequence Program (EDIT), 436 Deletion of Programs, 658 Editing Symbol Data and Comment at Once, 453 Descriptions of Displayed Items, 500 Editing the Sequence Program (EDIT), 524 w w .c DECB (Binary Decoding), 131 Emergency Stop Signal (*ESP), 942 DIFD (Falling Edge Detection), 254 Enabling Automatic Debugging at Power-on, 508 Difference of Status of Signals Between 1st Level and 2nd Level, 37 END (End of a Ladder Program), 277 DIFU (Rising Edge Detection), 253 END1 (1st Level Sequence Program End), 120 Direct Editing by Ladder Diagram, 660 END2 (2nd Level Sequence Program End), 121 DISP(Message Display) (PMC–RB/RB2/RB3/ RB4/RB5/RB6/RC/ RC3/RC4 Only), 209 End3 (End of 3rd Level Sequence) (PMC-RC/ RC3/RC4/ NB/NB2 Only), 122 DISPB, 223 Ending Edit of a Sequence Program, 675 w Details of Basic Instructions, 84 End of Block Step, 572 i–2 Index B–61863E/10 Ending Ladder Mnemonics Editing, 528 Format and Details of the Control Data of the Windr Functional Instruction, 838 Entering Data on the Program Check Screen (*Low– Speed response) (Not available for Power Mate– D/F, Series 21–TA), 780 Format And Details of the Control Data of the Windw Functional Instruction, 878 Entering Title Data, 437 Formats and Details of Control Data, 698 Entering Torque Limit Data for the Digital Servo Motor (*Low–Speed response), 784 Function, 315, 694, 833, 895, 935 Function for Displaying Signal Waveforms (Analys), 362 Error Codes List (For FAPT Ladder P–G), 691 Function for Storing Data in Memory, 361 Error Details, 533 Function of Copying Symbol and Comment Data, 461 Error List, 537 Functional Instruction Trset, 574 Error Messages (For EDIT), 517 Functional Instruction WINDR, 835 Error Messages (For I/O), 519 Functional Instruction WINDW, 837 Error Messages (For Ladder Mnemonics Editing), 531 Functional Instructions, 98 Example of Step Sequence Programs, 977 Functions for Displaying Memory Areas and Debugging the Program (MONIT), 497 r.c om EOR (Exclusive OR), 255 Examples of Structured Programming, 311 Functions of Processing, 608 Except Power Mate–D and –F, 721 ce nt e Examples of Using the Counter, 136 Exclusive Control for Functional Instructions, 583 Execution Method, 316 Execution of a Sequence Program, 472 Execution of Step Sequence, 558 [G] General Rules, 577 Graphical Symbols, 545 Execution Procedure of Sequence Program, 26 EXIN (External Data Input), 233 [H] Help Screen, 644 nc Extended Ladder Instructions, 573 [F] .c F Keys (F1 to F0), 616 FAPT Ladder (System P Series), 955 [I] I/O Error Messages, 493 I/O Link Connecting Check Screen, 371 FAPT LADDER System Floppy Loading, 625 I/O Link Connection Unit Assignment, 61 I/O Link–II Parameter Setting Screen, 372 Fdlist Command – File Attribute Display, 684 I/O Signals to CNC, 36 File Editing Function, 679 I/O Unit Address Setting (MODULE), 455 w w FAPT Ladder, FAPT Ladder–II, Ladder Editing Packages, 953 I/O Unit Model B Assignment, 63 For MDI Units other than Standard MDI Units (for FS20 PMC-RA1 and RA3), 340 w If Else Statement, 987 For Power Mate–D and –F, 724 Implementation Techniques, 311 For Statement, 985 Infinite Number of Relay Contacts, 308 For the FS15 (PMC-NB), 417 Initial Block Step, 572 For the FS16 (PMC-RC or PMC-RC3), 417 Initial Step, 565 Forced Input/Output Function, 375 Input of Program, 645 Forcibly Stopping the Sequence Program, 474 Input PMC Parameters from MDI Panel, 388 Foreign Language Display, 230 Input Signal Processing, 35 i–3 Index B–61863E/10 Input with a Katakana Identification Code, 464 List of Window Functions, 696 Input/Output Ladder/PMC–parameter by MDI/DPL, 534 LOAD Key (System Program Loading Key), 616 Loading of Floppy, 624 Input/Output Method to FANUC FLOPPY CASSETE (Fixed 4800bit/sec.), 534 Loading the Standard Ladder (for Power Mate –D/F PMC–PA1 AND PA3), 341 Input/Output Method to Office Programmer (P–g Mate/Mark II) (Fixed 9600bit/Sec.), 534 Location Search, 641 Input/Output of Ladder Program with P–G and Floppy Cassette/FA Card, 676 Logical AND, 257 Inputting a Multi-byte Character (D.CHAR), 464 Low–Speed Response and High–Speed Response of Window Function, 833 om Logical OR, 259 Insert, 640 Low–speed Response and High–speed Response of Window Function, 694 Insert of Sequence Program, 445 Interlocking, 38 Internal Relay Addresses (R), 66 r.c [M] Message Data Input, 463 [J] Message Data Setting (MESSAGE), 462 JMP (Jump), 162 ce nt e Miscellaneous Item, 309 JMP Instructions with Label Specification, 321 MMC3 R (MMC-III Window Data Read), 263 JMPB (Label Jump), 282 MMC3W (MMC-III Window Data Write), 266 JMPC (Label Jump), 283 MMCWR (Reading MMC Window Data) (Other than PMC–PA1/PA3), 246 JMPE (Jump End), 166 MMCWW (Writing MMC Window Data) (Other Than PMC–PA1/PA3), 248 Jump, 570 Modification Procedure, 936 Modifying the Converted Sequence Program, 936 nc [K] Keep Relay (KEEPRL), 390 Modifying the Ladder Mnemonics, 527 Modifying the Values of Signals by Forced Input/Output, 382 Keyboard of System P Series, 615 Monitor Time Screen, 595 .c Key Switch Signals (Xn, Xn+2), 942 Monitoring Elapsed Time, 594 MOVB (Transfer of 1 Byte), 250 [L] Moving a Sequence Program, 672 Label Addresses (JMPB, JMPC, LBL) (L), 80 Moving the Sequence Program, 452 Ladder Debugging Function, 509 Ladder Diagram Display, 406 MOVN (Transfer of an Arbitrary Number of Bytes), 252 Ladder Diagram Format, 303, 307 MOVOR (Data Transfer After Logical Sum), 155 Ladder Password Function, 341 MOVW (Transfer of 2 Bytes), 251 Ladder Screen, 590, 600 MUL (Multiplication), 199 LBL (Label), 284 MULB (Binary Multiplication), 201 Level Up of Input/Output Function with Memory Card, 957 Multiple Data Input, 388 w Label, 570 w w MOVE (Logical Product Transfer), 153 Limitations in SYSTEM P Mate, 660 [N] Limitations with the SYSTEM P Mate, 624 NB/NB2 Data Compatibility, 964 Line Numbers, 305 i–4 Index B–61863E/10 Nonvolatile Memory, 295 PMC Basic Instructions, 82 NOT (Logical NOT), 261 PMC Data Table, 298 Note, 404, 961 PMC I/O Signal Display and Internal Relay Display (PMCDGN), 352 Notes, 508 PMC Ladder Diagram Display (PMCLAD), 405 Notes on Using an MDI Keyboard without Cursor Keys (when using the FS20 PMC–MODEL RA1/RA3), 492 PMC Menu Selection Procedure by Softkey, 346 PMC Operation for Loader Contorol Function, 345 PMC Parameters Setting and Display (PMCPRM), 386 NUME (Definition of Constant), 207 PMC Programmer (CRT/MDI OR PDP/MDI) [Ladder Editing Function], 947 om Notes when this Functional Instruction is Used in Subroutine, 239 NUMEB (Definition of Binary Constants), 208 PMC Programmer (DPL/MDI) (Only for the Power Mate–D/F/H), 520 Power Mate–model D/H Assignment, 64 [O] r.c Preparation before Operation, 624 Offline Programmer → CNC, 960 Priority of Execution (1st Level, 2nd Level and 3rd Level), 28 On–line Debugging Function (Only for Power Mate–H), 535 Procedure, 945 On–line Edit, 416 ce nt e Processing I/O Signals, 34 Online Function, 514 Program Collation, 678 Online Setting Screen, 514 Program Configuration List (Main Screen), 587, 597 Operation, 357, 360, 362, 467, 498, 501, 504, 621, 958 Program Editing, 630 Program Input, 676 Operations, 480 Program Output, 677 OR, 90 Programmer Menu Screen, 626 OR. NOT, 90 Programming, 546 Other Setting Screens, 401 nc OR. STK, 93 Programming from Keyboard, 636 Other Window Functions, 934 PSGN2 (Position Signal Output 2), 293 Outline of Leveled Up Contents, 957 PSGNL (Position Signal Output), 290 .c Output of Program, 650 Output Signal Processing, 35 [R] R Keys (R0 to R3), 617 Overview, 375 RD, 86 w w Override Signals (*OV1 to *OV8) and Program Protect Key Signal (KEY), 942 RD . NOT, 87 RD. NOT. STK, 92 [P] RD. STK, 91 Reading a Character String of the CNC Program Being Executed in the Buffer , 791 w Paper Command, 653 Parameter, 949 Reading a Custom Macro Variable (*Low–Speed Response), 717 Parameter Display on Ladder Diagram, 408 Parameter Menu (for PMC–RB), 945 Parameter Screen, 363, 505 Reading a Custom Macro Variable (Low–speed response), 843 Parameter Setting and Display, 627 Reading a Parameter (*Low–Speed Response), 709 Parameter Setting Screen, 357, 468 Reading a Parameter (Not available for Power Mate– D/F, Series 21–TA), 788 PARI (Parity Check), 167 Reading a Parameter (Setting Data) (Low–speed response), 842 PMC Address (S ADDRESS), 574 i–5 Index Reading a Skip Position (Stop Position of Skip Operation (G31)) of Controlled Axes, 737 Reading a Skip Position (Stop Position of Skip Operation (G31)) of Controlled Axes (Low–speed response), 850 Reading a Tool Offset, 701 Reading a Tool Offset (Low–Speed Response), 839 Reading a Workpiece Origin Offset Value, 841 Reading A/D Conversion Data, 750 Reading A/D Conversion Data for the Feed Motor, 859 Reading an Estimate Disturbance Torque Data, 874 r.c Reading an Operator Message, 799 Reading and Writing of Nonvolatile Memory data, 297 ce nt e Reading and Writing the Laser Command Data and Laser Setting Data, 900 Reading Clock Data (Date and Time) (Not available for Power Mate–F), 782 Reading Clock Data (Low–speed response), 871 Reading CNC Status Information, 797 Reading CNC System Information, 699 Reading Diagnosis Data (*Low–Speed Response), 748 Reading Diagnosis Data (Low–speed response), 858 nc Reading Diagnosis Data (Not available for Power Mate–D/F, Series 21–TA), 790 Reading Load Information of the Spindle Motor (Serial Interface), 786 Reading Machining Distances, 923 .c Reading Modal Data, 743 Reading the Actual Velocity of Controlled Axes, 731 Reading the Actual Velocity of Controlled Axes (Low–speed response), 847 Reading the CNC Alarm Status, 721, 918 Reading the CNC Alarm Status (Low–speed response), 844 Reading the Current Program Number, 727 Reading the Current Program Number (Low–speed response), 845 Reading the Current Sequence Number, 729 Reading the Current Sequence Number (Low–speed response, 846 Reading the Estimate Disturbance Torque Data, 829 Reading the Load Current (A/D Conversion Data) for the Spindle Motor, 876 Reading the Machine Position (Machine Coordinates) of Controlled Axes, 735, 849 Reading the Machining Time (Low–speed response), 875 Reading the Measured Point, 924 Reading the Parameter (*Low–speed Response), 907 Reading the Relative Position on a Controlled Axis, 793, 872 Reading the Remaining Travel, 795, 873 Reading the Servo Delay for Controlled Axes, 739, 851 Reading the Tool Life Management Data (Number of Tool Groups) (Low–speed response), 862 Reading the Tool Life Management Data (Tool Life Counter Type) (Not available for Power Mate–D/F, Series 21–TA), 805 Reading the Tool Life Management Data(Tool Group Number)(Low–speed response), 861 Reading the Tool Offset Data According to the Specified Tool Number, 877 Reading the Wire Diameter Offset, 903 Reading Tool Life Management Data (Cutter Compensation No. (1): Tool No.) (Not available for Power Mate–D/F, Series 21–TA), 768 Reading Tool Life Management Data (Cutter Compensation No. (2): Tool Order No.) (Not available for Power Mate–D/F, Series 21–TA), 770 Reading Tool Life Management Data (Cutter Compensation No.1) (Low–speed response), 867 Reading Tool Life Management Data (Cutter Compensation No.2) (Low–speed response), 868 Reading Tool Life Management Data (Number of Tool Groups) (Not available for Power Mate–D/F, Series 21–TA), 756 Reading Tool Life Management Data (Number of Tools) (Low–speed response), 862 Reading Tool Life Management Data (Number of Tools) (Not available for Power Mate–D/F, Series 21–TA), 758 om Reading a Workpiece Origin Offset Value (Not Supported by the Power Mate–D or –F), 705 B–61863E/10 Reading Modal Data (Low–speed response), 852 w Reading Model Data, 919 Reading of the Comment, 899 Reading of Tool Setting Data, 927 w Reading Set Data (Not available for Power Mate–D/F, Series 21–TA), 789 w Reading Setting Data, 914 Reading Setting Data (*Low–Speed Response), 713 Reading Signals Automatically at Power on, 367 Reading the Absolute Position (Absolute Coordinates) of Controlled Axes, 733 Reading the Absolute Position on a Controlled Axis, 847 Reading the Acceleration/Deceleration Delay on Controlled Axes, 741, 851 Reading the Actual Spindle Speed, 778 i–6 Index B–61863E/10 [S] Reading Tool Life Management Data (Tool Group No.) (Not available for Power Mate–D/F, Series 21–TA), 754 Scratch Command – Deletion of Files, 686 Reading Tool Life Management Data (Tool Information (1) : Tool No.) (Not available for Power Mate– D/F, Series 21–TA), 772 Screen Display, 379 Reading Tool Life Management Data (Tool Information (2): Tool Order No.) (Not available for Power Mate–D/F, Series 21–TA), 774 Screen of Ladder Debugging Function, 509 Screen for Displaying Traced Data, 508 Search of Sequence Program, 449 Search of Specified Relay Coil Points in Ladder Diagram, 412 Reading Tool Life Management Data (Tool Information 1) (Low–speed response), 869 om Searching a Sequence Program, 669 Reading Tool Life Management Data (Tool Information 2) (Low–speed response), 870 Searching for an Address (SRCH), 463 Selecting the PMC Programmer Menu, 522 Reading Tool Life Management Data (Tool Length Compensation No. (1): Tool No.) (Not available for Power Mate–D/F, Series 21–TA), 764 Selection of Program Menu by Soft Keys, 660 Selection of Programmer Menus by Softkeys, 429 r.c Reading Tool Life Management Data (Tool Length Compensation No. (2): Tool Order No.) (Not available for Power Mate–D/F, Series 21–TA), 766 Sequence Program, 25 Sequence Program Check and Write into ROM (Steps 8 to 11), 23 Reading Tool Life Management Data (Tool Length Compensation No.1) (Low–speed response), 865 ce nt e Sequence Program Copy Function, 495 Reading Tool Life Management Data (Tool Life Counter Type) (Low–speed response), 864 Sequence Program Creating Procedure, 3 Reading Tool Life Management Data (Tool Life Counter) (Low–speed response), 863 Sequence Program Generation (LADDER), 440 Sequence Program Entry (Steps 6, 7), 23 Sequence Program Input, 441, 662 Reading Tool Life Management Data (Tool Life Counter) (Not available for Power Mate–D/F, Series 21–TA), 762 Sequence Program Memory Capacity, 41 Sequence Program Processing Time, 39 Reading Tool Life Management Data (Tool Life) (Low–speed response), 863 Sequence Program Structuring, 32, 310 SET, 94 nc Reading Tool Life Management Data (Tool Life) (Not available for Power Mate–D/F, Series 21–TA), 760 Reading Tool Life Management Data (Tool No.) (Not available for Power Mate–D/F, Series 21–TA), 776 .c Reading Tool Life Management Data (Tool Number) (Low–speed response), 870 Set Items, 478 Setting and Display Screen, 389 Setting and Displaying System Parameters (System Param), 523 Setting I/O Commands, 676 Reading Tool Setting Data by Specifying Tool Number, 932 Setting Method, 515 Reading Value of the P–code Macro Variable (*Low– speed response), 801 Setting on the NC Parameter Screen, 515 w Setting of I/O Device, 619 Setting Screen, 398 Registering the Tool Life Management Data (Tool Group) (*Low–Speed response) (Not available for Power Mate–D/F, Series 21–TA), 807 w Setting the Transfer Speed ([SPEED] Soft Key), 490 Setting/Clearing OVERRIDE, 383 Remove Command – File Copy, 687 Setting/Operation for Enabling Forced Input/Output, 378 Repetitive Operation, 27 SFT (Shift Register), 179 Restrictions, 325, 496 ROM Format Program, 647, 654, 657 Signal Address Conversion (from the PMC–MODEL L/M to the PMC–MODEL RB/RC), 935 ROT (Rotation Control), 142 Signal Diagnosis Screen, 366 w Rename Command – File Attribute Change, 685 Signal Names, 305 ROTB (Binary Rotation Control), 145 Signals for Connecting the Operator's Panel, 942 RST, 96 Soft key menu of Ladder Debugging Function, 510 i–7 Index B–61863E/10 Source Program, 645, 650 Symbol and Comment Display, 411 SP (Subprogram), 280 Symbol Data and Comment Input, 459 SPCNT (Spindle Control), 269 Symbol Data Display, 673 Spe (End of a Subprogram), 281 Symbol Data Search (SRCH), 459 Special Uses of the R3 Key, 659 Symbol Data Setting (SYMBOL), 458 Specification, 576 Symbols Used in the Ladder Diagram, 306 Specification and Display of System Parameters (SYSPRM), 950 System Diagram of Soft Key, 952 Specification of Step Sequence, 575 om System Floppy, 624 Specification of PMCs, 5 [T] Specifications, 314, 321, 362, 504 Terminology, 549 Specifying and Displaying System Parameters (SYSPRM), 431 The Value of Functional Instruction Parameter, 408 r.c Specifying Addresses, 945 Time Screen, 593 Specifying and Displaying Title Data (TITLE), 437 Timer Addresses (T), 78 Start and Stop of a Sequence Program, 473 Timer Screen, 593 Starting and Stopping the On–line Debugging Function, 535 ce nt e Timer Screen (TIMER), 389 Timer Set Time Address, 128 Starting and Stopping the Sequence Program (RUN/ STOP), 530 Starting Ladder Mnemonics Editing, 525 Timer, Counter, Keep Relay, Nonvolatile Memory Control, Data Table, 295 Starting or Stopping the Trace Function, 358 TMR (Timer), 123 Starting the Sequence Program, 474 TMRB (Fixed Timer), 125 TMRC (Timer), 127 Step, 563 Step Operation [STEP], 511 Trace Function (Trace), 357 Trace Screen, 359 nc Step Sequence Basics, 548 Step Sequence Corresponded C Language, 981 Step Sequence Method, 542 Step Sequence Screen, 588, 598 .c Stop Function of Break with Condition [BRKCTL], 512 Transfer Between Data Area and Non–Volatile Memory, 895 Transfer to and from a FANUC FD Cassette, 480 Transfer to and from a FAPT LADDER, 480 Transfer to and from a ROM WRITER, 491 Transition, 566 w Stop of Ladder Diagram Display by Trigger of Signal, 413 Storage and Control of Sequence Program (Steps 12 to 14), 24 [U] w Storage to a Memory Card, 484 User PMC Screen (PCMDI), 417 Storage to Flash ROM, 482 w Storing the Sequence Program into Flash Eeprom (I/O) (Only for the Power Mate–H), 532 [W] SUB (Subtraction), 195 SUBB (Binary Subtraction), 197 What is a Sequence Program?, 19 Subprogram Numbers (CALL, CALLU, SP) (P), 81 While Statement, 981 Subprogramming and Nesting, 315 WINDOW (Writing CNC Window Data), 240 Substitution of Sequence Programs, 665 Window Function Description (Except FS 15B PMC– NB/NB2), 694 Summary of Specification of Ladder Program, 15 Window Function Description (FS15B PMC–NB/ NB2) , 833 Switch Statement, 989 i–8 Index B–61863E/10 Window Function Description (FS16–LA), 894 Writing the Tool Life Management Data (Tool Length Compensation Number 2), 886 Window Function Description (FS16–PA), 927 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), 815 Window Function Description (FS16–W), 903 WINDR (Reading CNC Window Data), 237 Writing a Custom Macro Variable, 881 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), 817 Writing a Custom Macro Variable (*Low–Speed Response), 719 Writing a Data on the Program Check Screen, 882 Writing the Tool Life Management Data (Tool Life Counter Type), 885 om Writing a Parameter (*Low–Speed Response), 711 Writing a Parameter (Setting Data), 880 Writing the Tool Life Management Data (Tool Life Counter Type) (*Low–Speed Response) (Not available for Power Mate–D/F, Series 21–TA), 813 Writing a Tool Offset (*Low–speed response), 703 Writing a Tool Offset Data, 879 Writing the Tool Life Management Data (Tool Life Counter), 884 r.c Writing a Workpiece Origin Offset Value (*Low– Speed Response) (Not Supported by the Power Mate–D or –F), 707 Writing the Tool Life Management Data (Tool Life Counter) (*Low–Speed response) (Not available for Power Mate–D/F, Series 21–TA), 811 Writing of Tool Setting Data (Low–speed Response), 930 Writing the Tool Life Management Data (Tool Life), 884 ce nt e Writing Setting Data (*Low–Speed Response), 715 Writing Setting Data (Low–speed Type), 915 Writing the Feedrate, 893 Writing the Tool Life Management Data (Tool life) (*Low–Speed response) (Not available for Power Mate–D/F, Series 21–TA), 809 Writing the Measured Point (*Low–speed Response), 926 Writing the Tool Life Management Data (Tool Number), 889 Writing the Parameter (*Low–speed Response), 909 Writing the Tool Life Management Data (Tool Number) (*Low–Speed Response) (Not available for Power Mate–D/F, Series 21–TA), 827 Writing the Superposition Move Command, 891 nc 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), 819 .c 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), 821 Writing the Tool Life Management Data (Cutter Compensation Number 1), 887 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), 825 Writing the Tool offset Data According to the Specified Tool Number, 890 Writing the Torque Limit Override, 883 Writing the Wire Diameter Offset (*Low–speed Response), 905 Writing the Tool Life Management Data (Cutter Compensation Number 2), 887 w Writing Value of the P–code Macro Variable (*Low– speed response), 803 Writing, Reading, and Verifying the Sequence Program and PMC Parameter Data, 475 Writing the Tool Life Management Data (Tool Group Number), 883 WRT. NOT, 89 w Writing the Tool Life Management Data (Tool Condition (1) : Tool Number) (*Low–Speed Response) (Not available for Power Mate–D/F, Series 21–TA), 823 w WRT, 88 Writing the Tool Life Management Data (Tool Information 1), 888 [X] Writing the Tool Life Management Data (Tool Information 2), 888 XMOV (Indexed Data Transfer), 186 XMOVB (Binary Index Modifier Data Transfer), 189 Writing the Tool Life Management Data (Tool Length Compensation Number 1), 886 i–9 May,'94 Aug.,'93 Mar., '92 Aug., '91 Oct., '90 Date 05 04 03 02 01 Edition w w Revision Record Contents 07 All pages are revised. PMC–MODEL RC is added. Edition 06 08 09 Addition of PMC–MODEL RA1/RA2/RB2 Addition of PMC–MODEL PA1/PA3/RA3/RB3/RC3/NB. Dec., '97 Date Nov.,'94 Apr.,'95 Oct.,'95 Mar.,'96 Corresponds to 18–B Total revision Contents om Addition of PMC–RB5/RB6 Addition PMC–NB2 Corresponds to 16i/18i/21i–MODEL A r.c ce nt e nc .c Addition of PMC–MODEL RB4/RC4 Addition of the following Appendix. S Window function description (FS16–LA) S Window function description (FS16–W) S Window function description (FS16PA) S PMC MODEL RA1/RA3 Supplementary Explanation of Programming 10 LADDER LANGUAGE PROGRAMMING MANUAL (B–61863E) FANUC PMC–MODEL PA1/PA3/RA1/RA2/RA3/RA5/RB/RB2/RB3/RB4/RB5/RB6/RC/RC3/RC4/NB/NB2 w om r.c All specifications and designs are subject to change without notice. w w w .c nc · No part of this manual may be reproduced in any form. ce nt e ·