Changes for the Better

MITSUBISHI

PROGRAMMABLE CONTROLLERS

MELSEC-F

OPERATION MANUAL

FX-10P-E

FX

This manual explains the programming and monitoring procedures of the micro PC MELSEC-FX series using the FX-10P-E handy programming panel.

Read this manual carefully and thoroughly and make sure the specifications for handy programming panel are correct.

See the corresponding FX series PC manual for details about PC handing and instructions.

CONTENTS

Guidelines for the Safety of the User and Protection of the FX-10P-E HANDY PROGRAMMING PANEL.

This manual provides information for the use of the FX-10P-E HANDY PROGRAMMING PANEL. The manual has been written to be used by trained and competent personnel. The definition of such a person or persons is as follows:

a) Any engineer who is responsible for the planning, design and construction of automatic equipment using the product associated with this manual, should be of a competent nature, trained and qualified to the local and national standards required to fulfill that role. These engineers should be fully aware of all aspects of safety with regards to automated equipment.

b) Any commissioning or service engineer must be of a competent nature, trained and qualified to the local and national standards required to fulfill that job. These engineers should also be trained in the use and maintenance of the completed product. This includes being completely familiar with all associated documentation for said product. All maintenance should be carried out in accordance with established safety practices.

c) All operators of the completed equipment (See Note) should be trained to use this product in a safe manner in compliance to established safety practices. The operators should also be familiar with documentation which is connected with the actual operation of the completed equipment.

Note : The term ‘completed equipment’ refers to a third party constructed device which contains or uses the product associated with this manual.

Note's on the Symbols Used in this Manual

At various times through out this manual certain symbols will be used to highlight points which are intended to ensure the users personal safety and protect the integrity of equipment. Whenever any of the following symbols are encountered its associated note must be read and understood. Each of the symbols used will now be listed with a brief description of its meaning.

Hardware Warnings

1) Indicates that the identified danger **WILL** cause physical and property damage.

2) Indicates that the identified danger could **POSSIBLY** cause physical and property damage.

3) Indicates a point of further interest or further explanation.

Software Warnings

4) Indicates special care must be taken when using this element of software.

5) Indicates a special point which the user of the associate software element should be aware.

6) Indicates a point of interest or further explanation.

- Under no circumstances will Mitsubishi Electric be liable responsible for any consequential damage that may arise as a result of the installation or use of this equipment.
- All examples and diagrams shown in this manual are intended only as an aid to understanding the text, not to guarantee operation. Mitsubishi Electric will accept no responsibility for actual use of the product based on these illustrative examples.
- Please contact a Mitsubishi distributor for more information concerning applications in life critical situations or high reliability.

Caution on operation

Thoroughly read the manual and sufficiently assure safety before executing the operation to forcibly set devices to ON/OFF or the operation to change present values and set values of word devices in the test mode.
Otherwise, the machine may be damaged and accidents may occur by erroneous operations.

Version upgrade to support FX1S, FX1N, FX2N and FX2NC Series Programmable controllers. (ver 4.10 or later)

The FX-10P-E (ver 4.10 or later) fully supports all items and procedures used for the FX1S, FX1N, FX2N and FX2NC PLCs.
Read section 1-3 in addition to the contents of the text when using the FX-10P-E (ver 4.10 or later).

1. New Features

Table 1.1

	New Feature	Supported by Peripheral	
1	Addition of FX2N, FX1N, FX1S as selectable model	○	
2	Addition of basic and applied instructions	○	described in section 2 below
3	Expansion of device range	○	described in section 2 below
4	Addition of 16K-step program memory	○	parameter setting
	Setting of external RUN input	○	parameter setting
5	Writing of EPROM cassette	×	
6	Remote RUN/STOP function	×	
7	Specification of batteryless operation mode	×	
8	Setting communication Parameters	×	
9	Specification of modem initialization	×	

○: Already compatible ×: Not yet compatible

2. List of Additional Devices and Instructions

- For details of each instruction and devices, refer to the FX programming manual.

Table 2.1

Devices		
Auxiliary relays	M0 to M3071	3072 points
Data registers	D0 to D7999	8000 points
Index registers	V0 to V7, Z0 to Z7	16 points

Table 2.2

Basic Instructions	
LDP	ORP
LDF	ORF
ANDP	INV
ANDF	

Table 2.3

Applied Instructions	
FNC No.	Instruction
59	PLSR
110	ECMP
111	EZCP
118	EBCD
119	EBIN
120	EADD
121	ESUB
122	EMUL
123	EDIV

Applied Instructions	
FNC No.	Instruction
127	ESQR
129	INT
130	SIN
131	COS
132	TAN
147	SWAP
155	DABS
156	ZRN
157	PLSV

Applied Instructions	
FNC No.	Instruction
158	DRVI
159	DRVA
160	TCMP
161	TZCP
162	TADD
163	TSUB
166	TRD
167	TWR
169	HOUR

Applied Instructions	
FNC No.	Instruction
170	GRY
171	GBIN
176	RD3A
177	WR3A
180	EXTR
224 to 246	In line Compare

3. New Operations for FX2N, FX2NC, FX1N, FX1S Support

3.1 Basic and Applied instructions are added

Symbol and input operation of additional basic instructions

Table 3.1

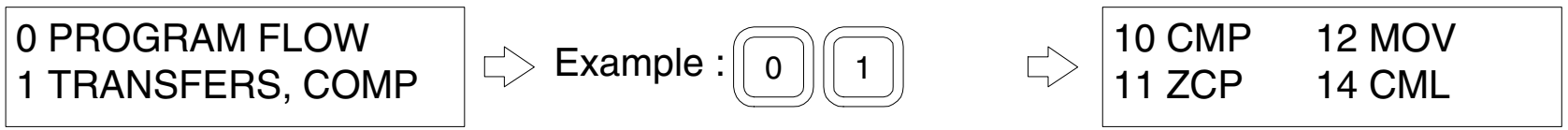
Instruction	Description	Mnemonic	Key Operations
LDP	Initial leading edge pulse contact	LDP	LD P/I Device Number GO
LDF	Initial trailing edge pulse contact	LDF	LD F Device Number GO
ANDP	Serial connection leading edge pulse contact	ANP	AND P/I Device Number GO
ANDF	Serial connection trailing edge pulse contact	ANF	AND F Device Number GO
ORP	Parallel connection leading edge pulse contact	ORP	OR P/I Device Number GO
ORF	Parallel connection trailing edge pulse contact	ORF	OR F Device Number GO
INV	Invert current logical status	INV	NOP P/I GO

Applied instruction input operation

The number of operands vary depending on the instruction.

- It is the responsibility of the operator to ensure that the instruction entered is valid for the current unit.

Application instruction input operation (by using the HELP function)



- When using the HELP function with a model other than FX_{2N} be careful not to select an invalid instruction.

3.2 Expansion of Device Range

- For a list of the new device ranges see section 2.

Handling index registers

"V0" and "Z0" index registers are equivalent to registers "V" and "Z" respectively, so either one can be used for programming.
When entering "V1" to "V7" or "Z1" to "Z7" add a numeral 1 to 7 to the "V" or "Z" before pressing SP or GO .

3.3 Parameter setting

Addition of 16K steps

```
MEMORY SETTING
  2K  4K  8K  16K
```

Setting of external RUN input

```
  RUN  INPUT
▉USE       X002
```

```
  DON'T  USE
```

When a RUN terminal is needed on the PC, select "USE", and specify an input number from X000 to X017 (X000 to X007 only for FX2N-16M☆ and FX2NC-16M☆).

Number of file register blocks

0 to 14 blocks can be specified for file registers. These occupy program space, 1 block = 500 program steps, and are stored with the program. File registers start at D1000.

3.4 Others

Device conversion

Device number conversion is not possible between the following devices because the number of program steps required is different.

- M0 to M1535 ↔ M1536 to M3071 (FX2N and FX2NC only)
- 16-bit counter ↔ 32-bit counter (all FX models)

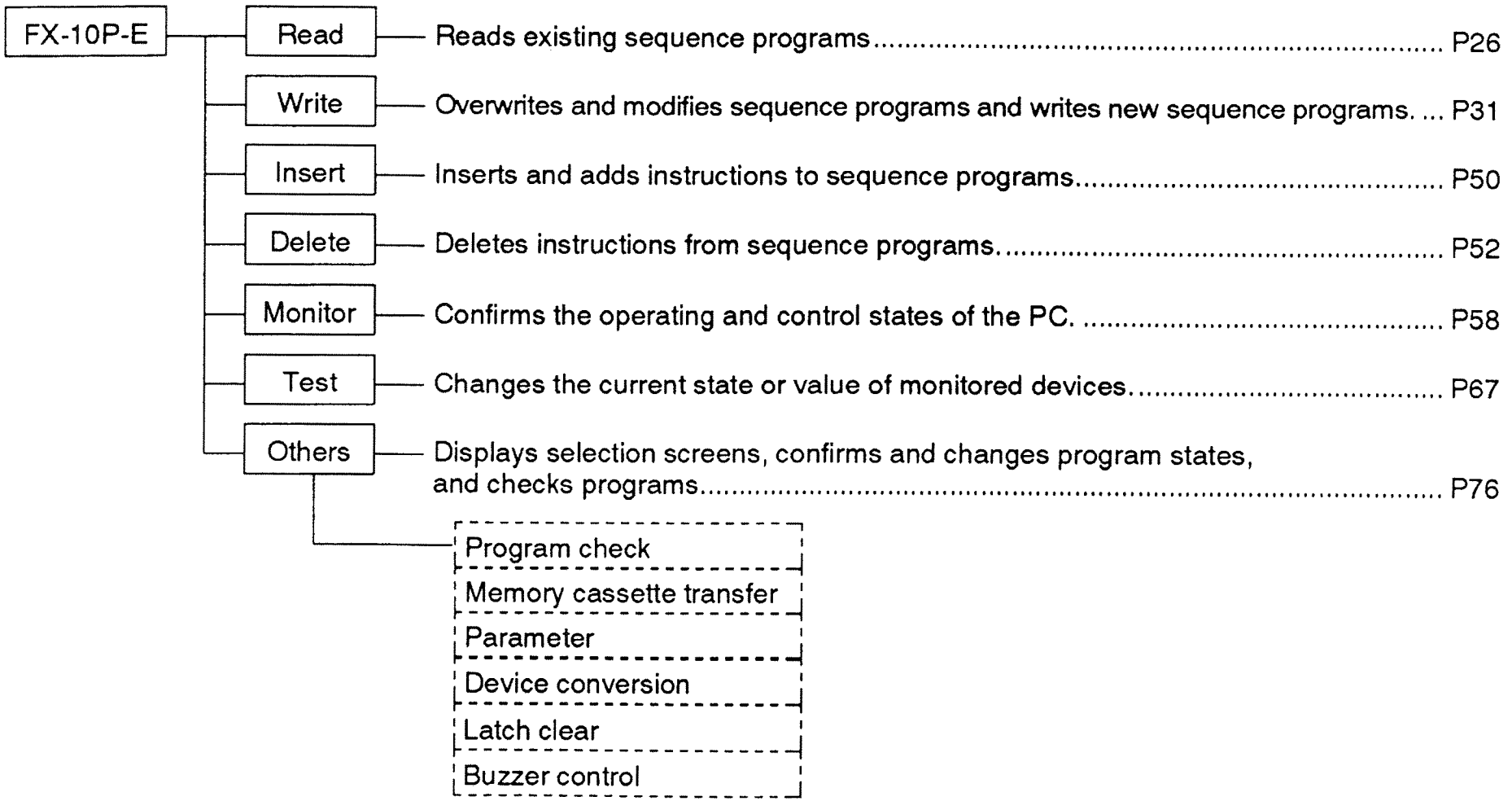
Writing the program to the FX-EEPROM-16

- When an instruction is inserted to a 16K-step program saved in the FX-EEPROM-16 in online mode, it may take 5 to 50 seconds depending on the total program size and the instruction insertion position.

1. INTRODUCTION

List of Functions

The FX-10P-E handy programming panel (hereafter called the HPP) is a hand-held programming and monitoring panel used to write programs (sequence programs and parameters) to a PC and to monitor PC operations when connected to a MELSEC-FX series PC.

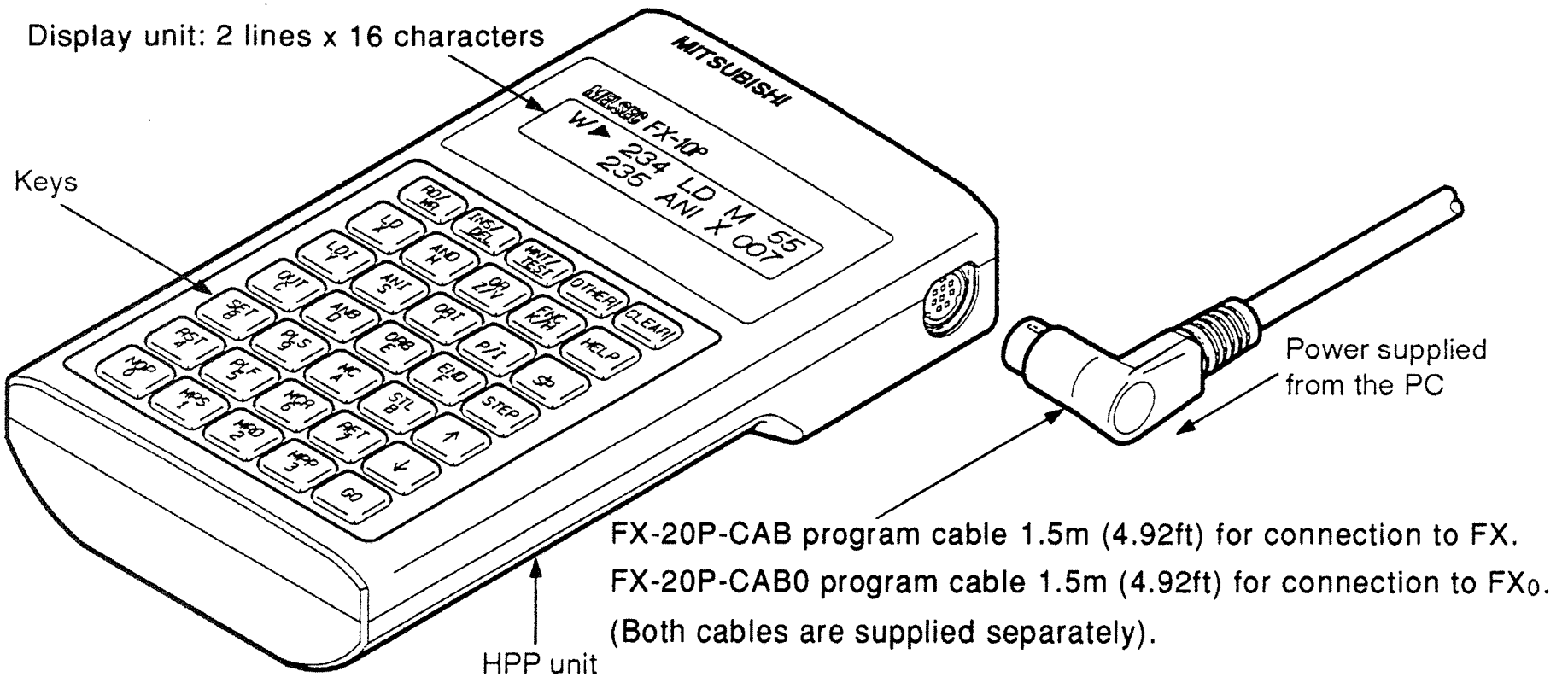


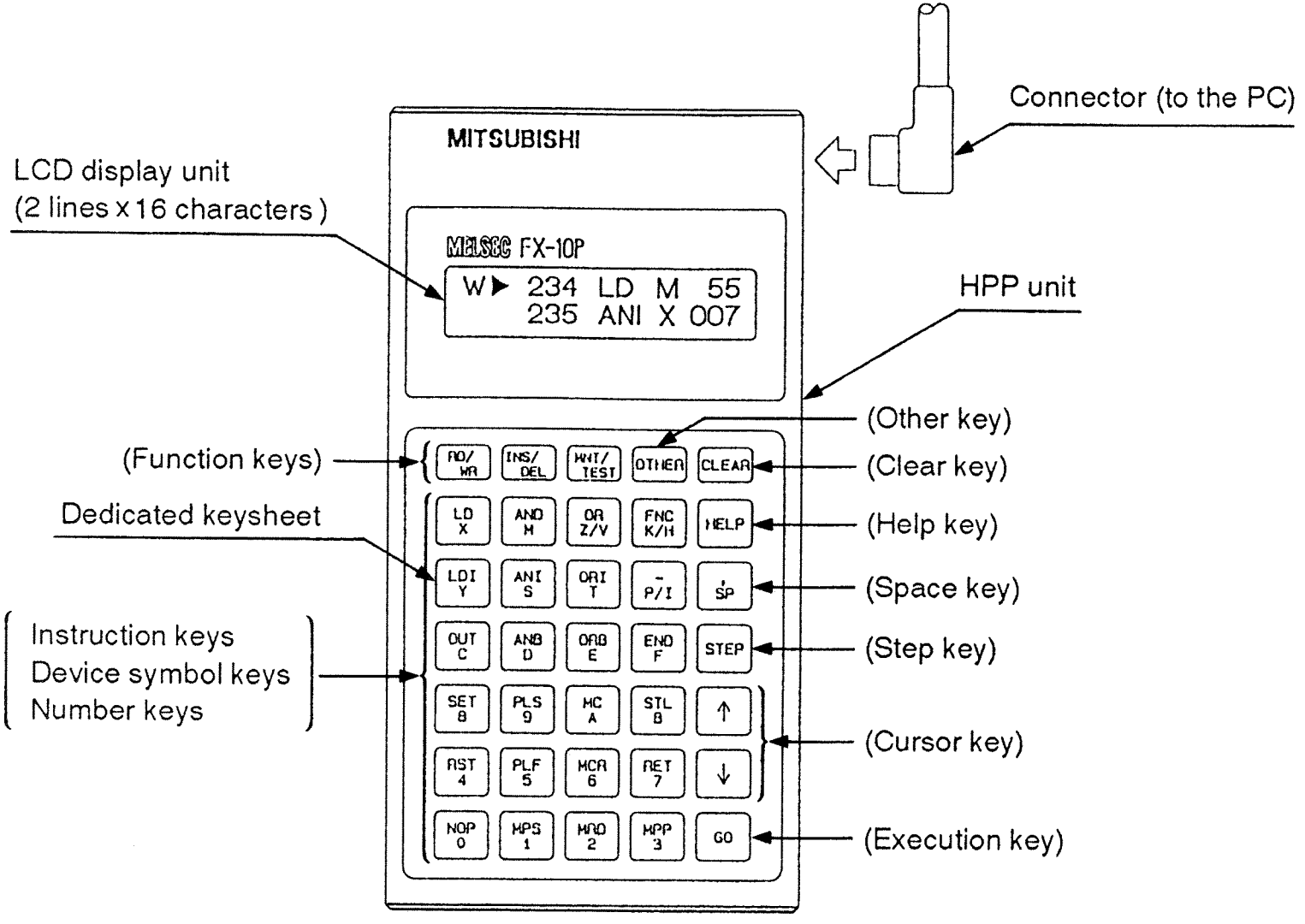
1. INTRODUCTION

This section explains the configuration of the HPP.

The HPP is composed of a liquid crystal display (2 lines x 16 characters) and a dedicated rubber key sheet (function, instruction, device symbol and numeric keys).

The HPP uses an FX-20P-CAB0 program cable (1.5 m) for communications with an FX0 PC. Use an FX-20P-CAB program cable for connections with an FX PC. Always make sure to use the proper cable for the corresponding PC.




Connector (to the PC)
MITSUBISHI
LCD display unit
(2 lines x 16 characters)
MELSEC FX-10P
W▶ 234 LD M 55
235 ANI X 007
HPP unit
(Other key)
(Function keys)
RD/WR
INS/DEL
MNT/TEST
OTHER
CLEAR
(Clear key)
Dedicated keysheet
LD X
AND M
OR Z/V
FNC K/H
HELP
(Help key)
LDI Y
ANI S
ORI T
-P/I
SP
(Space key)
Instruction keys
Device symbol keys
Number keys
OUT C
ANB D
ORB E
END F
STEP
(Step key)
SET B
PLS 9
MC A
STL B
↑
RST 4
PLF 5
MCR 6
RET 7
↓
(Cursor key)
NOP 0
MPS 1
MRD 2
MPP 3
GO
(Execution key)

Key use

This section explains how each key is used.

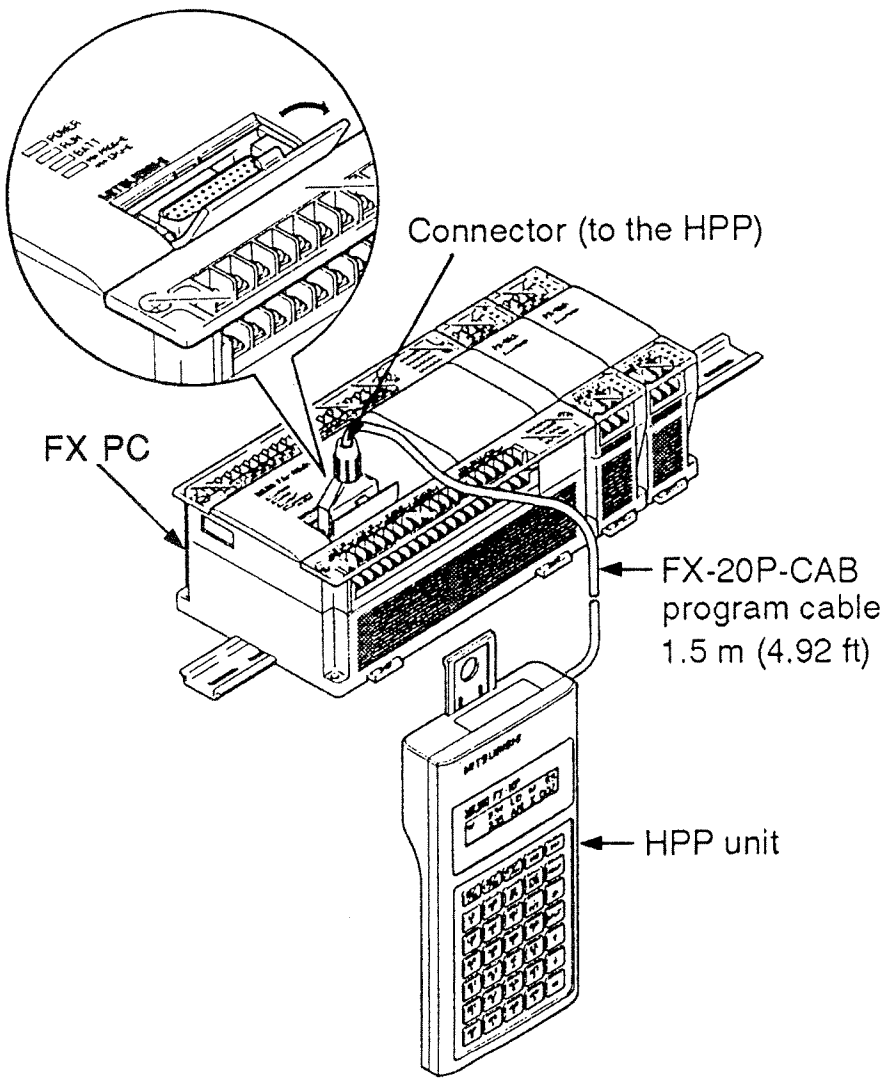
Key	Use
Function keys • Read/write • Insert/delete • Monitor/test	The function keys are toggle keys. Pressing the key once calls the first function. Pressing it again calls the second function.
Other key	Even when other functions are in use, the mode menu selection screen is displayed.
Clear key	Cancels key operations before the [GO] key is pressed, clears error messages or returns to the previous screen.
Help key	Displays the applied instruction menu and switches the display between decimal and hexadecimal when the monitoring function is selected.
Space key	Used whenever advice or a constant is to be input.
Step key	Sets step numbers.
Cursor keys	Move the line cursor and prompt, scroll lines, and designate the numbers of devices preceding and following a designated device.
Execution key	Confirms and executes commands, scrolls information screen and continues a search.
Instruction keys Device symbol keys Number keys	Each of these keys has two functions: instructions (upper part) and device symbols or numbers (lower part). Which key function is effective during each operational step is automatically determined according to the currently executed operation. The following device symbol keys are toggled: [Z/V] key, [K/H] key, and [P/I] key.

1. INTRODUCTION

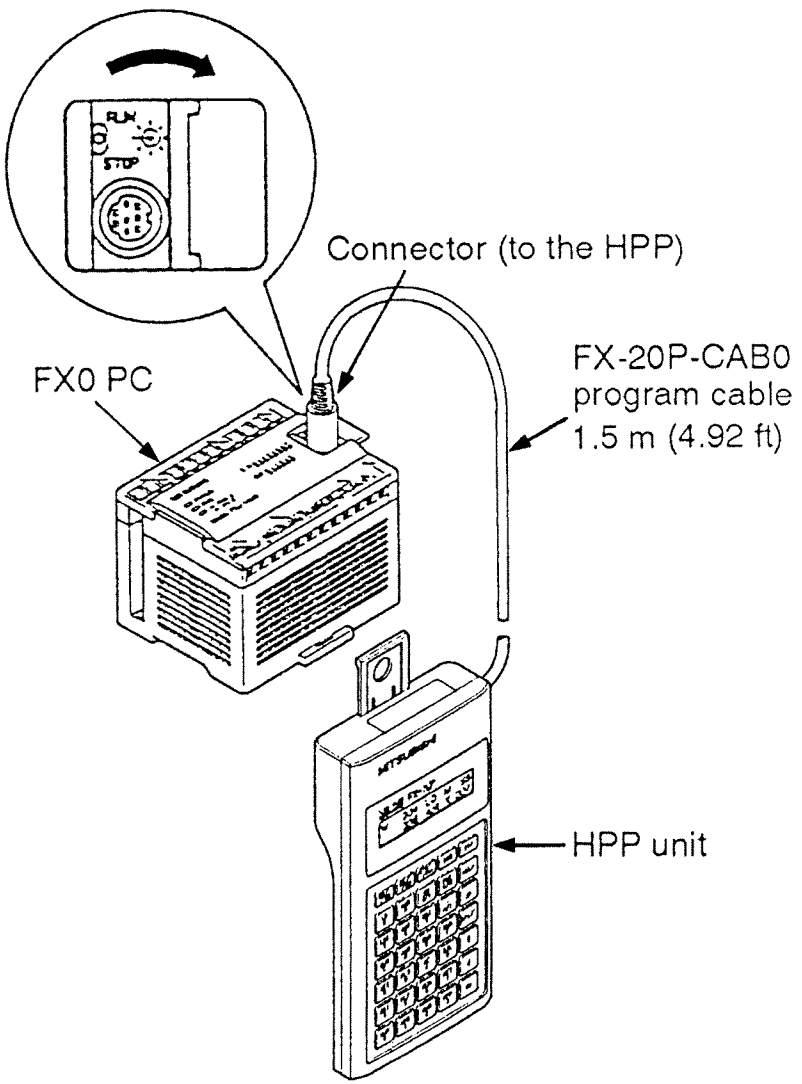
Connection Method

This section explains how to connect the HPP to an FX PC.

〈 When using an FX PC 〉

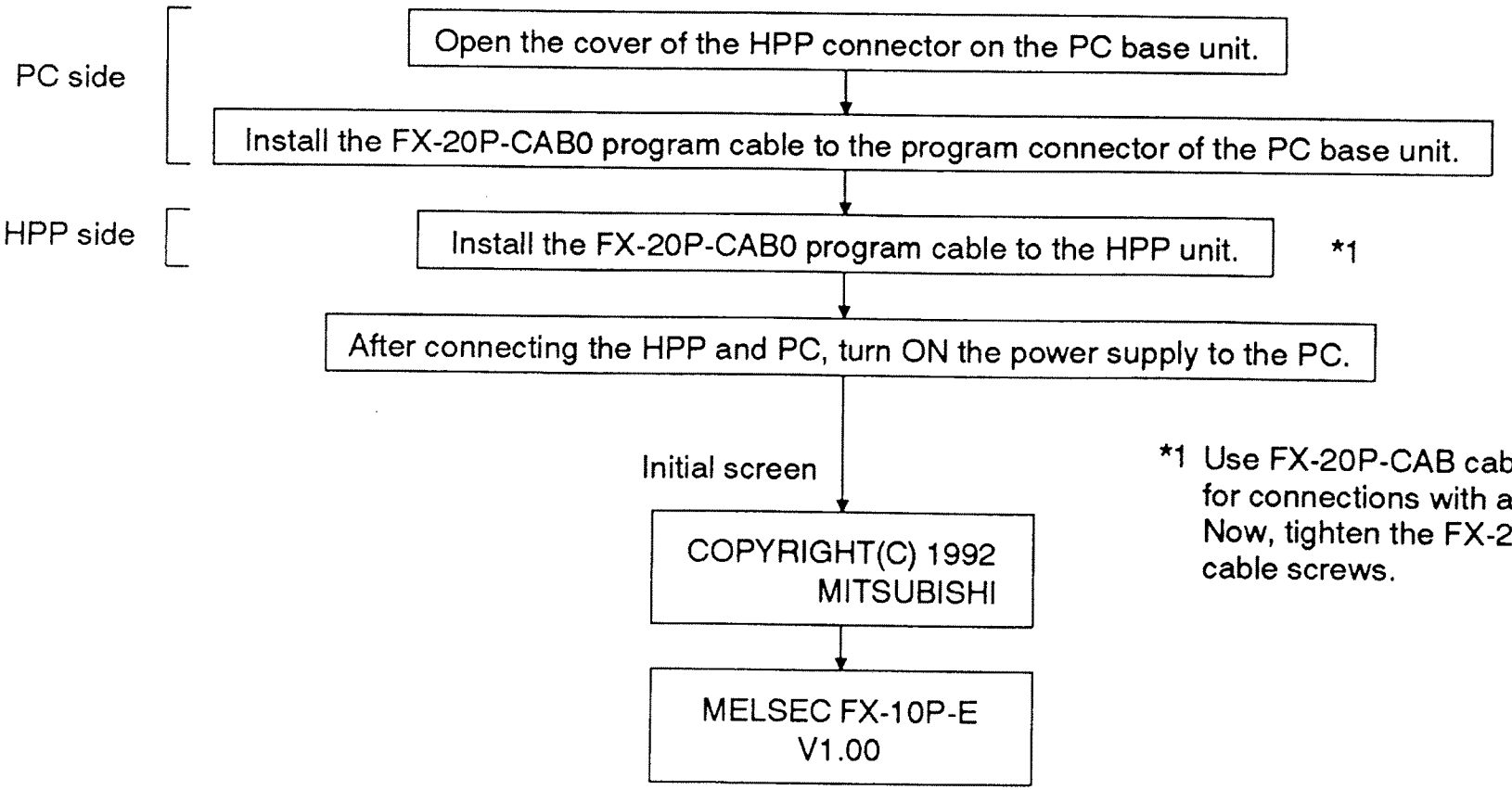


〈 When using an FX_0 PC 〉



1. INTRODUCTION

This section explains the procedure for connecting the HPP and a PC.

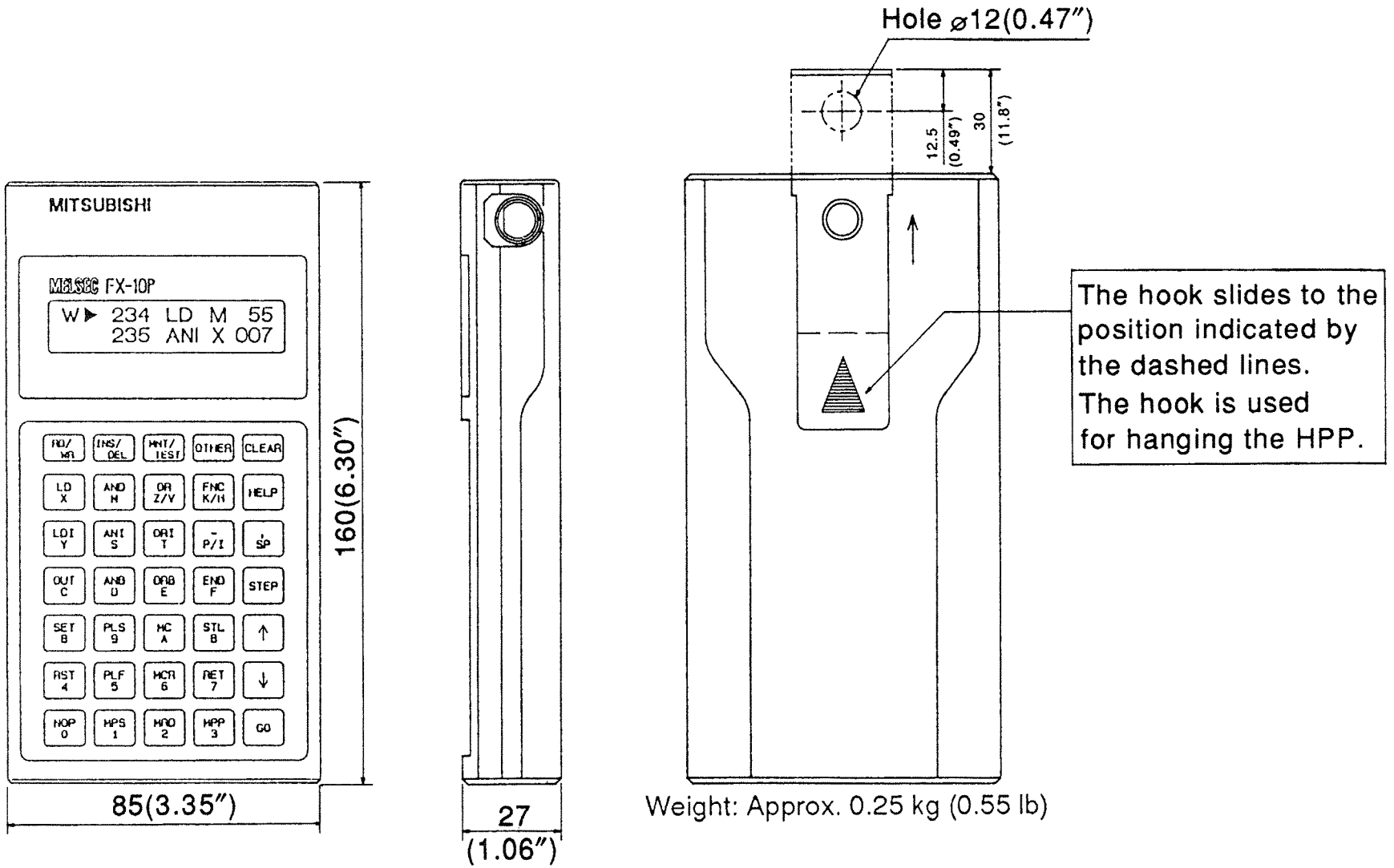


*1 Use FX-20P-CAB cable (optional) for connections with an FX PC. Now, tighten the FX-20P-CAB cable screws.

IMPORTANT

Do not touch the HPP connection ports. Doing so could cause the internal electronic circuits to be destroyed by static electricity.

HPP outside dimensions



Units : mm(inches)

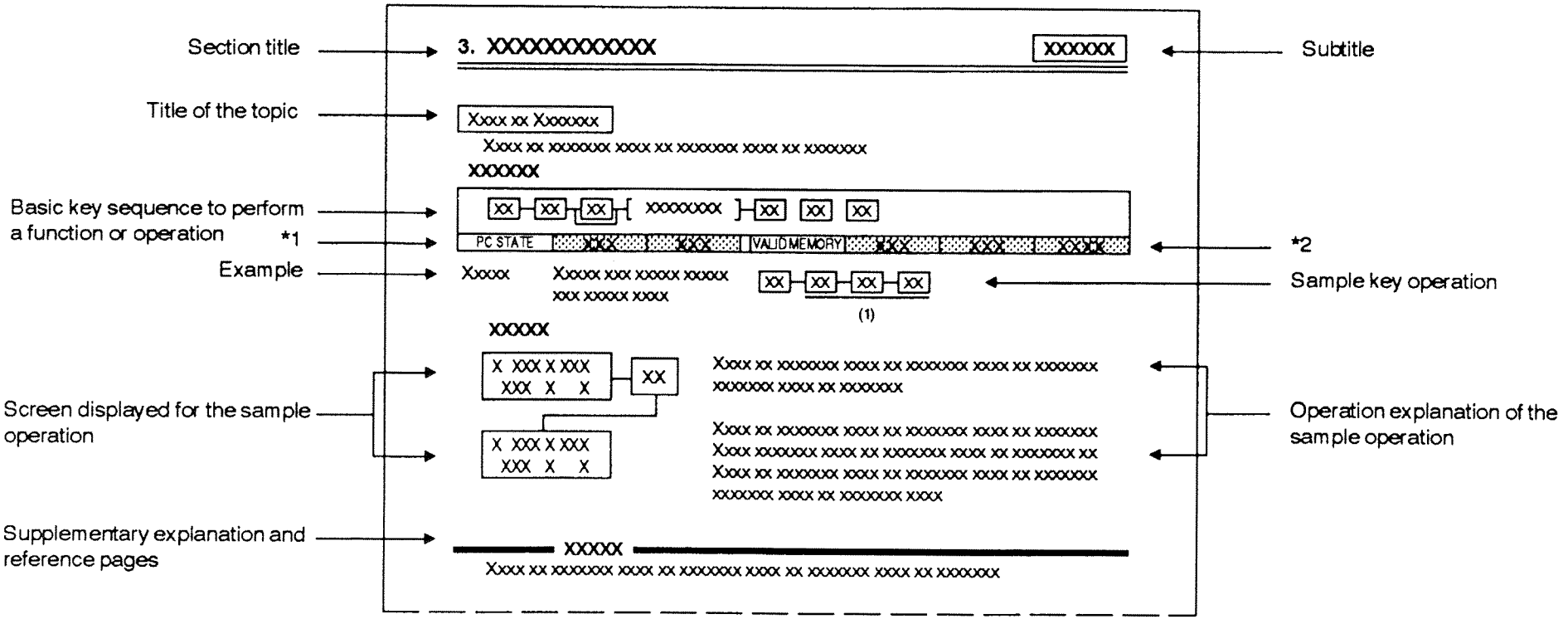
General specifications

Items	General Specifications			
Operating ambient temperature	0 to 40 °C			
Operating ambient humidity	35 to 85 % RH (no condensation)			
Vibration resistance	Confirms to JIS C0911.	Vibration frequency	Acceleration	Amplitude
		10 to 55 Hz	1 G	0.1 mm
		2 hours in 3 axial directions		
Shock resistance	Conforms to JIS C0912. (10G, 3 times in 3 (X, Y, and Z) directions			
Operating ambience	Free from corrosive gases. Dust should be minimal.			

Performance specifications

Items		Performance Specifications
Power supply voltage		5 V DC ± 5 % electric power is supplied from the PC.
Current consumption		120 mA
Display contents	Graphic display	1 character: 8 x 5 = 40 dots; 1 x 5 dots at the bottom is for the prompt.
	Number of displayed characters	16 characters x 2 lines = 32 characters
	Character types	Alphanumeric
Keyboard		35 keys
Built-in interface		Conforms to EIA RS422. FX-20P-CAB cable, FX-20P-CAB0 cable
Outside dimensions		160 x 85 x 27 (mm) (6.30 x 3.35 x 1.06 (in))
Weight		0.25 kg (0.55 lb)

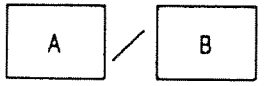
Understanding titles and explanatory displays



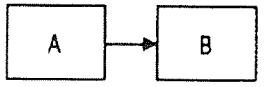
*1 "PC state" shows the valid PC RUN/STOP states for this key sequence (shaded areas are valid).

*2 "Valid memory" shows the PC program memory areas for which this key sequence is valid (shaded areas are valid). However, RAM indicates either PC RAM or a RAM cassette.

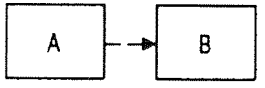
Explanation of the key sequences used in this manual



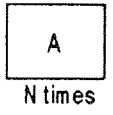
: Press either the [A] or [B] key.



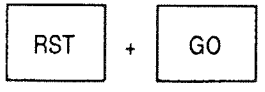
: Press the [A] key, and then press the [B] key.



: Press the [A] key several times, and then press the [B] key.



: Press the [A] key n times.



: Press and hold the [RST] key, then press the [GO] key (for resetting the HPP).

Abbreviations and their meanings

The following abbreviations are used in this manual:

FX-10P-E handy programming panel ..... HPP
FX series PC (Programmable Controller) ..... PC
▶ symbol that is displayed on the left side of the display screen (Execution line) ..... Line cursor
◪ symbol that blinks on the left side of the display screen (Item selection line) ..... Cursor
Underline _ symbol displayed under a character on the display screen (Key input wait place) ..... Prompt

2. SAMPLE PROGRAM OPERATION

This section explains the actual operation of the HPP and how to create programs using a simple training example.

Operations are confirmed by using the monitoring and test functions after creating the program.

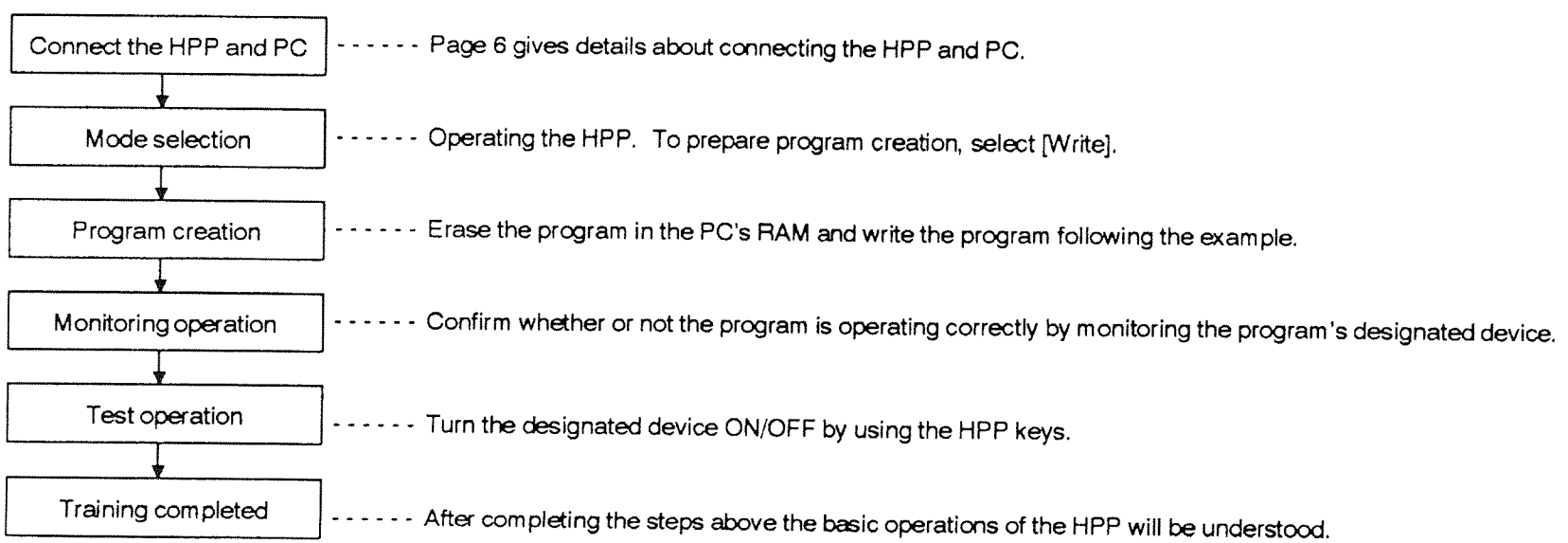
Start by operating the HPP. Since only the basic operation of the HPP is explained in this section, see subsequent sections for details about the different modes and operations.

Prepare the following training devices beforehand.

- PC (PC base unit: FX0-14MR-ES)
- FX-20P-CAB0 programming cable
- HPP (FX-10P-E handy programming panel)
- Simulation switches

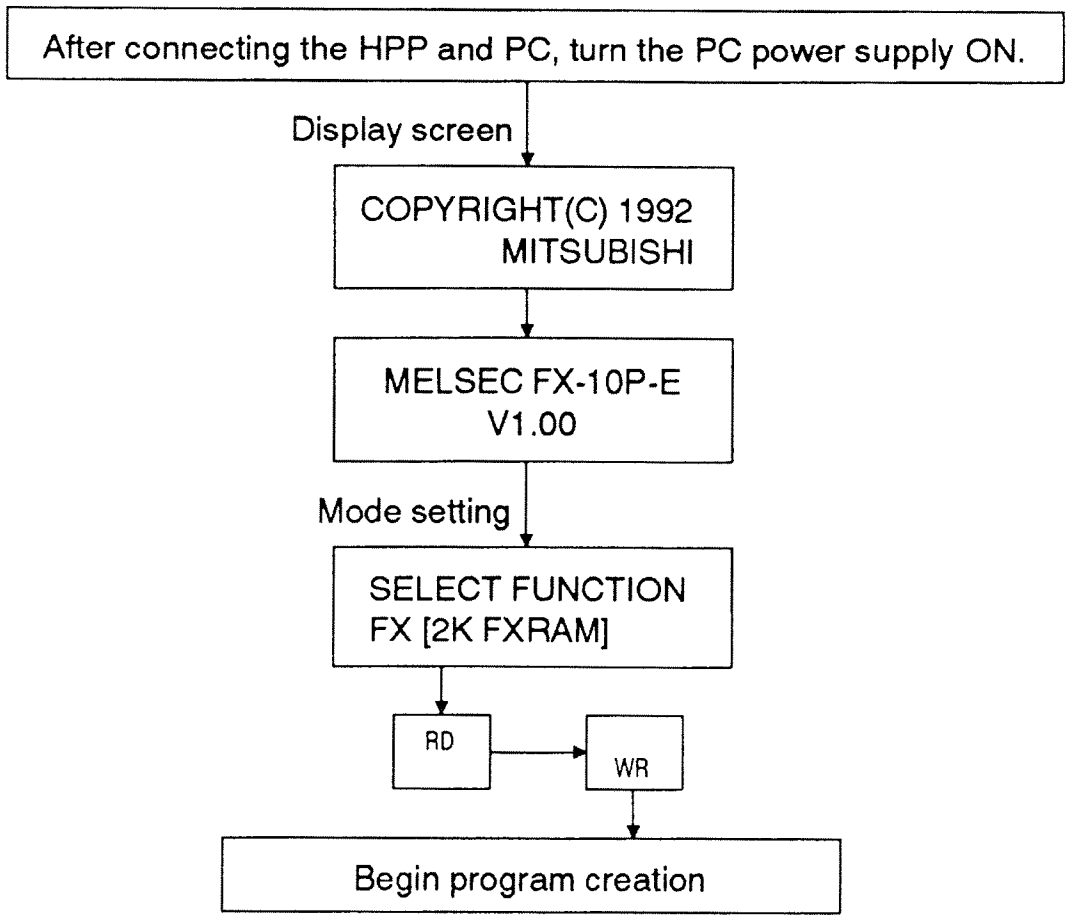
Training procedure

The training procedure is as follows:



2. SAMPLE PROGRAM OPERATION

Mode selection determines which function will be operated by the HPP keys.



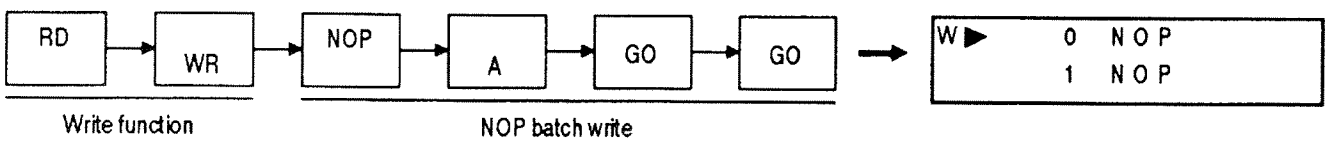
2. SAMPLE PROGRAM OPERATION

Write the sample program, following the examples below, after executing an NOP batch write (all programs erased) to the PC's RAM.

Confirm that RUN/STOP switch of the PC is set to STOP before continuing.

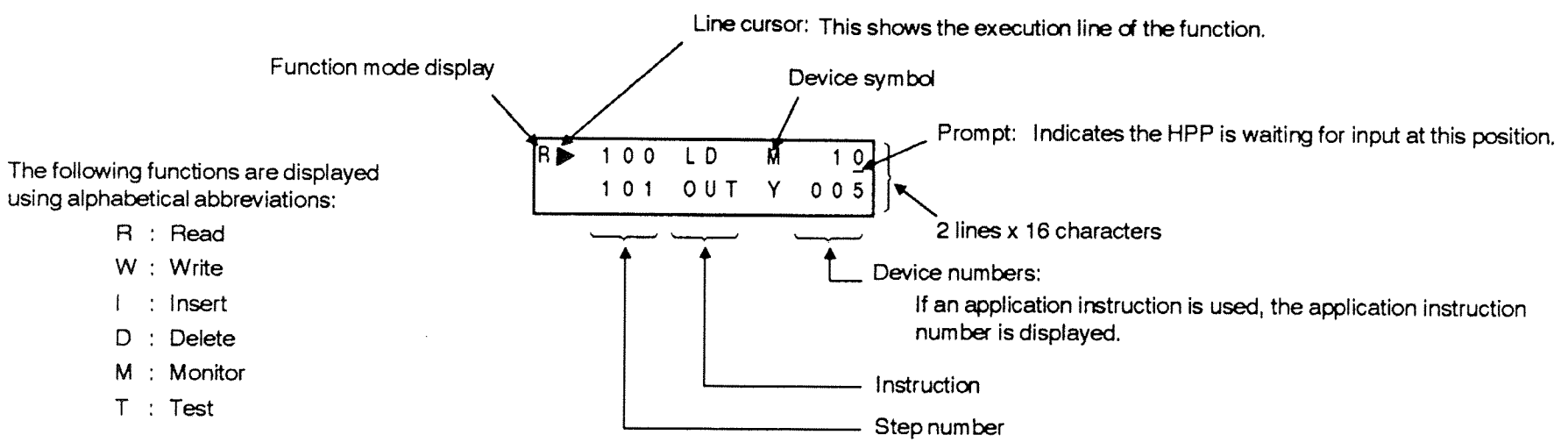
NOP batch write

When writing a new program, clearing the PC RAM (NOP batch write designating "all area") is done by the following operation:



Screen configuration

The screen configuration during HPP program creation is indicated below.

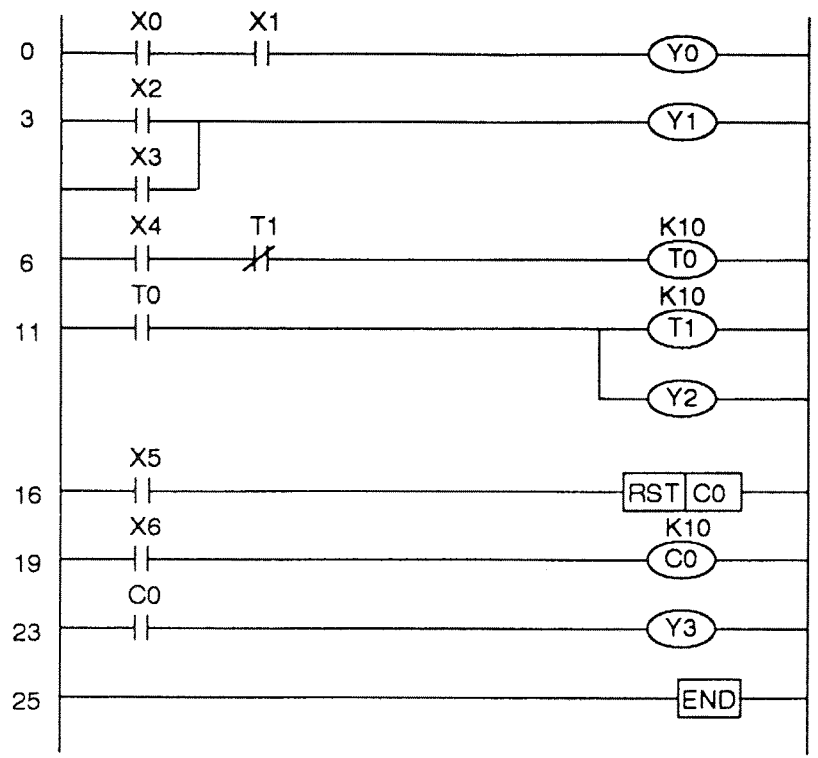


2. SAMPLE PROGRAM OPERATION

Program creation

The sample program used is shown below.

Ladder



List

```
 0  LD    X000
 1  AND   X001
 2  OUT   Y000
 3  LD    X002
 4  OR    X003
 5  OUT   Y001
 6  LD    X004
 7  ANI   T  1
 8  OUT   T  0    K10
11  LD    T  0
12  OUT   T  1    K10
15  OUT   Y002
16  LD    X005
17  RST   C  0
19  LD    X006
20  OUT   C  0    K10
23  LD    C  0
24  OUT   Y003
25  END
```

2. SAMPLE PROGRAM OPERATION

Write the sample program using the following key operations:

Key Operations

(1) LD → X → 0 → GO → AND → X → 1 → GO → OUT → Y → 0 → GO

(2) LD → X → 2 → GO → OR → X → 3 → GO → OUT → Y → 1 → GO

(3) LD → X → 4 → GO → ANI → T → 1 → GO → OUT → T → 0 → SP → K → 1 → 0 → GO

Display Screens

```
W    0  LD   X  000
     1  AND  X    1
```

```
W    1  AND  X  001
▶    2  OUT  Y    0
```

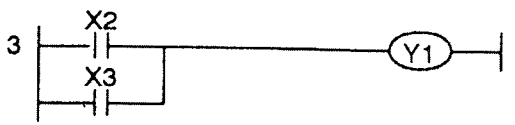
```
W    2  OUT  Y  000
▶    3  LD   X    2
```

```
W    4  OR   X  003
▶    5  OUT  Y    1
```

```
W    6  LD   X  004
▶    7  ANI  T    1
```

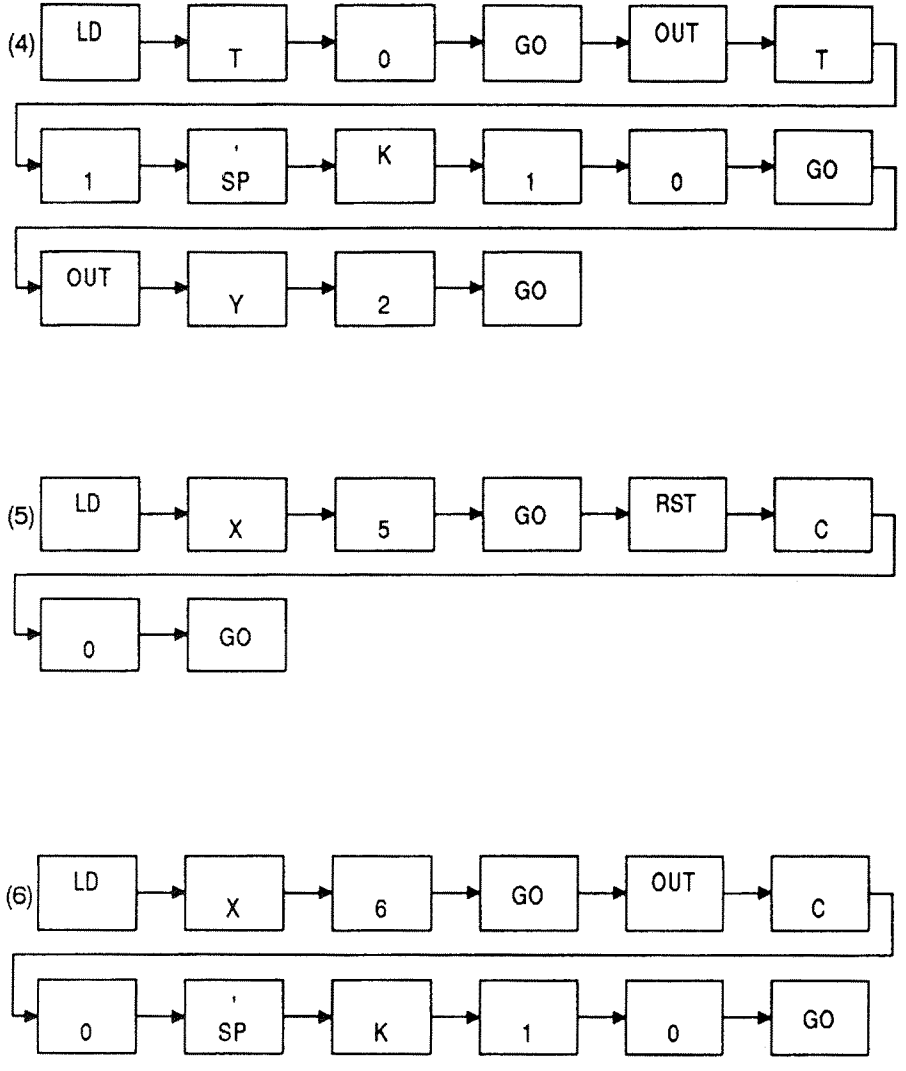
```
W    8  OUT  T    0
▶       K      10
```

Sample Ladders

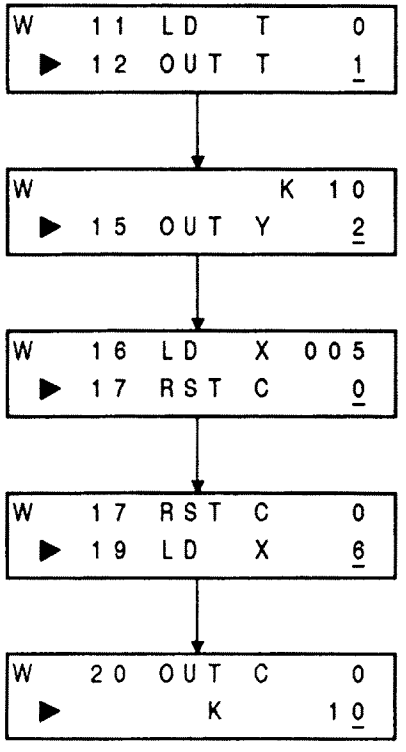


2. SAMPLE PROGRAM OPERATION

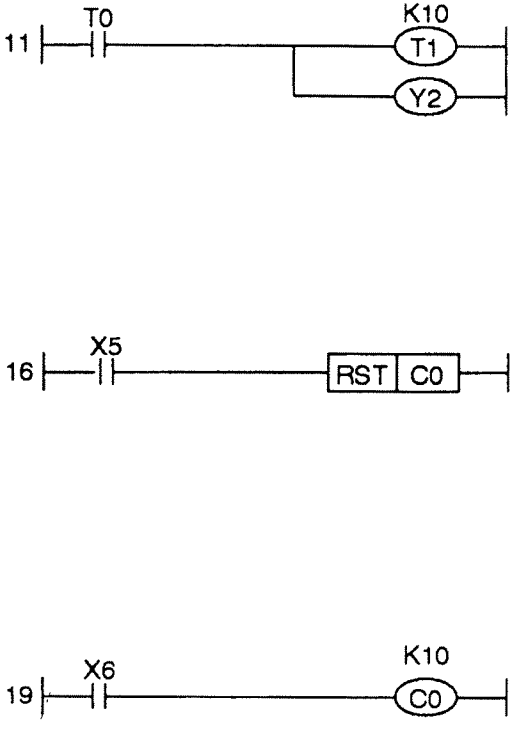
Key Operations



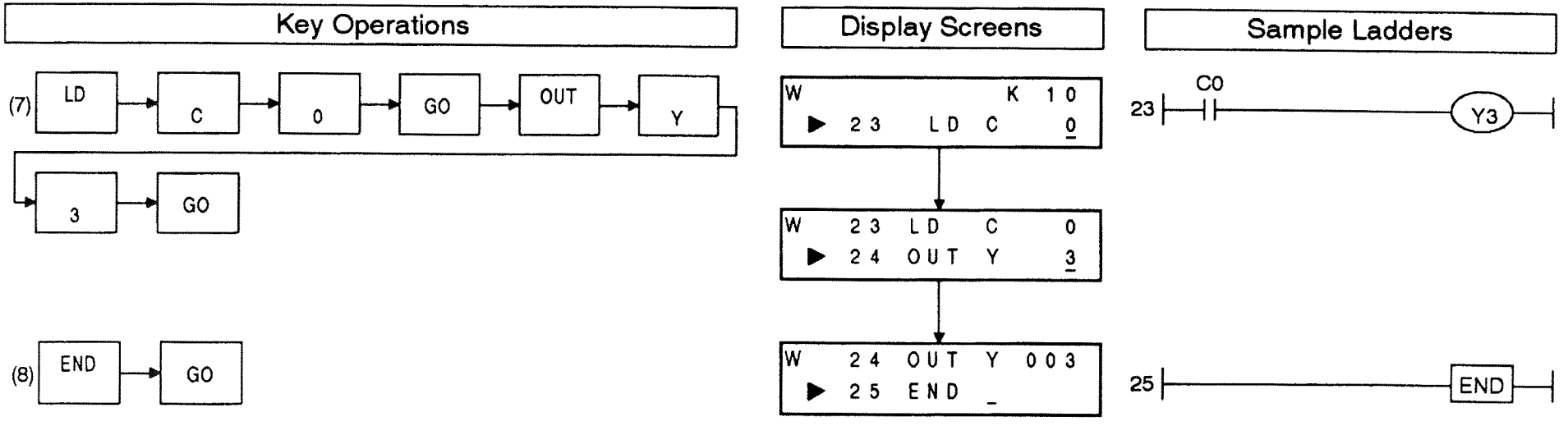
Display Screens



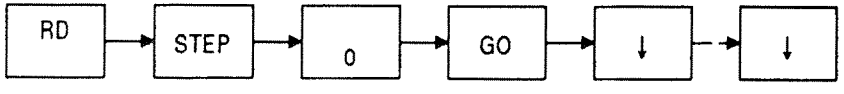
Sample Ladders



2. SAMPLE PROGRAM OPERATION



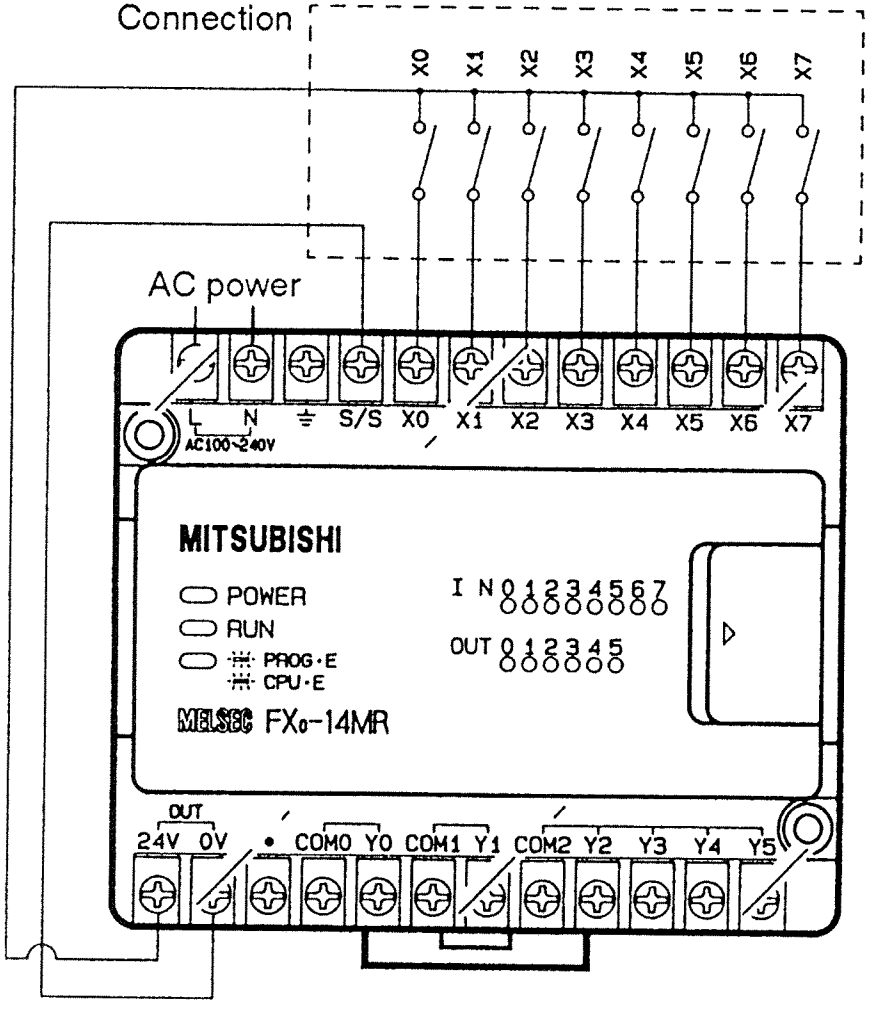
* After writing the program, use the following key operations to read and check the program:



2. SAMPLE PROGRAM OPERATION

How to Connect the Simulation Switches

Confirm the operation of the created program by using simulation switches. Connect the simulation switches and the PC as shown below. First turn OFF the power supply and then make the connection.



Simulation switch
(Simulation input switch : FX_0-14SW-E)

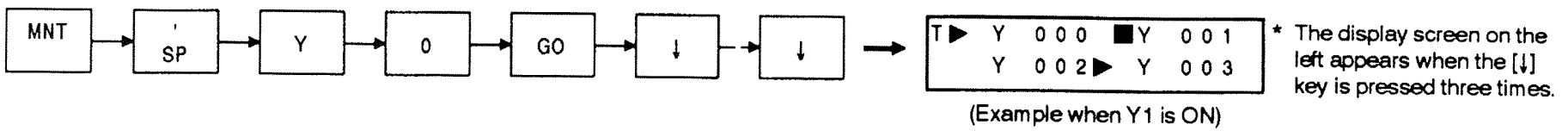
Programmable Controller
(PC base unit : FX_0-14MR-E)

2. SAMPLE PROGRAM OPERATION

Confirm the operation of the sample program by using the monitoring function.

Turn ON the power supply and set the RUN/STOP switch of the PC to RUN.

(1) Read from Y0 to Y3 individually on the display screen of the HPP.



* The display screen on the left appears when the [↓] key is pressed three times.

(Example when Y1 is ON)

(2) Confirm whether Y0 to Y3 operate as follows when simulation switches [X0] to [X6] are operated on the display screen of the HPP. (Begin operations after turning [X0] to [X7] OFF.)

	Simulation Switch Operations	Results
(a)	Turn [X0] and [X1] ON.	[Y0] goes ON.
(b)	Turn either [X2] or [X3] ON.	[Y1] goes ON.
(c)	Turn [X4] ON.	[Y2] flashes.
(d)	Turn [X6] ON then OFF ten times.	[Y3] goes ON.
(e)	After [Y3] goes ON, turn [X5] ON.	[Y3] goes OFF.

If (a) to (e) do not give the expected results the ladder is faulty. Use the read function to read and recheck the program.

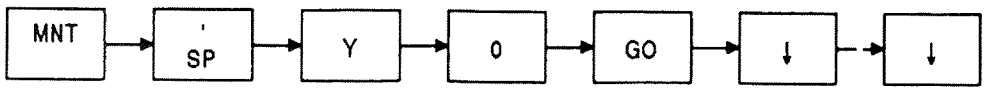
2. SAMPLE PROGRAM OPERATION

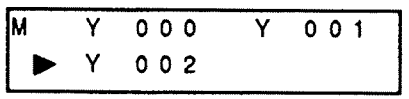
Using the test function to turn designated devices of an already created program ON and OFF with the HPP keys.

Make sure the RUN/STOP switch is set to STOP.

(1) Read Y0 to Y2 individually on the HPP display screen.

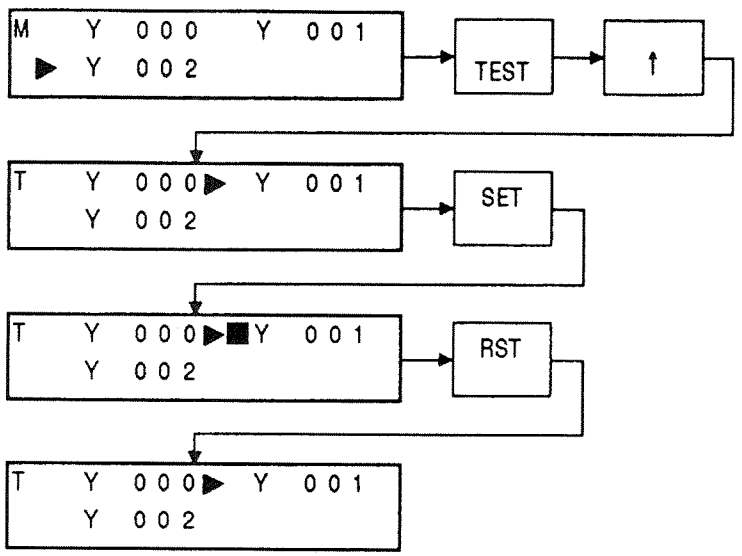
(This procedure is the same as the monitoring function read procedure.)





* The display screen on the left appears when the [↓] key is pressed twice.

(2) After reading Y0 to Y2 on the display screen, try to turn ON and OFF Y1 forcibly as a test.



* This is the state for reading Y0 to Y2 on the screen.

* Press the [TEST] key to switch to the test function. Then, designate the target device number by using the [↑] / [↓] keys.

* The function mode display changes from M to T line cursor points to Y1.

* Press the [SET] key to turn Y1 ON forcibly.

* The ■ symbol is displayed before Y1, and Y1 goes ON.

* Press the [RST] key to turn Y1 OFF forcibly.

* The ■ symbol before Y1 disappears and Y1 goes OFF.

Repeat the above operation for Y0 to Y2, and confirm whether or not they all turn ON and OFF.

3. PROGRAMMING

Programming consists of creating, modifying, and reading the sequence program in the list format. Connect the HPP to the PC and directly access the PC program memory by using the HPP.

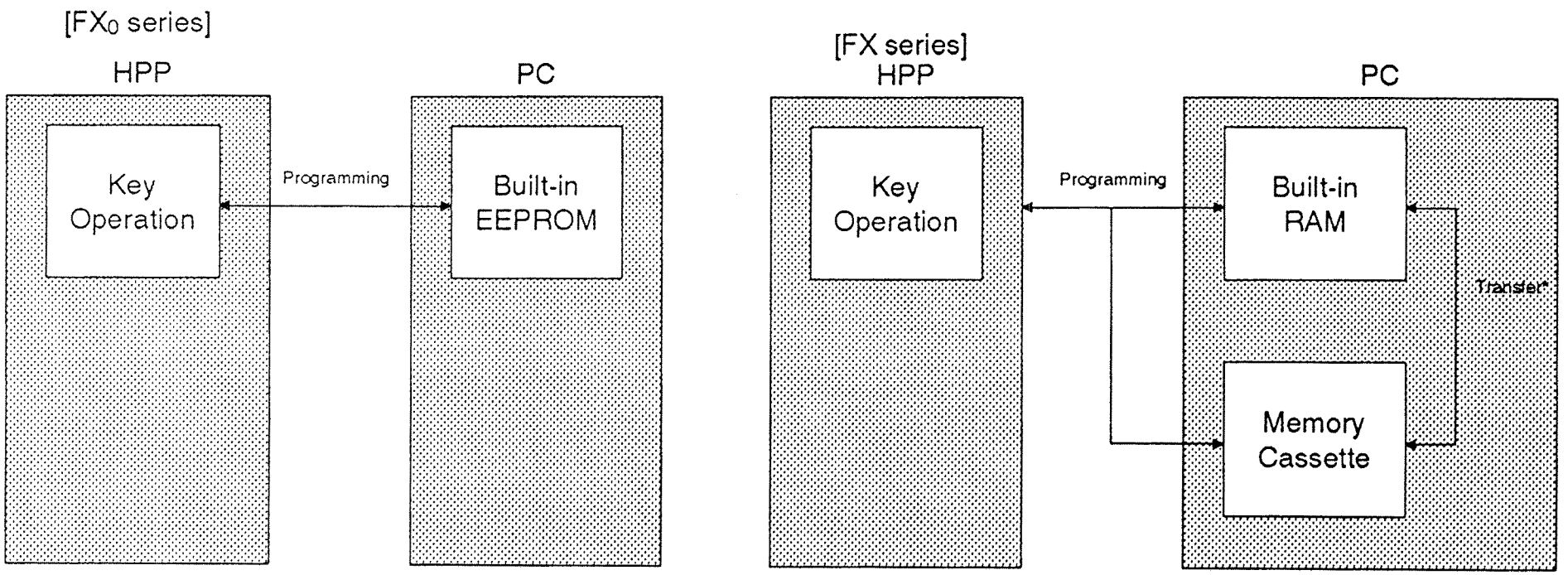
The program memory is composed of parameters and programs.

When the FX_0 series is used, programs are written to the EEPROM built in the PC.

When the FX series is used, if a memory cassette has not been installed in the PC, programs are written to the RAM built in the PC. If a memory cassette has been installed in the PC, programs are written to the memory cassette.

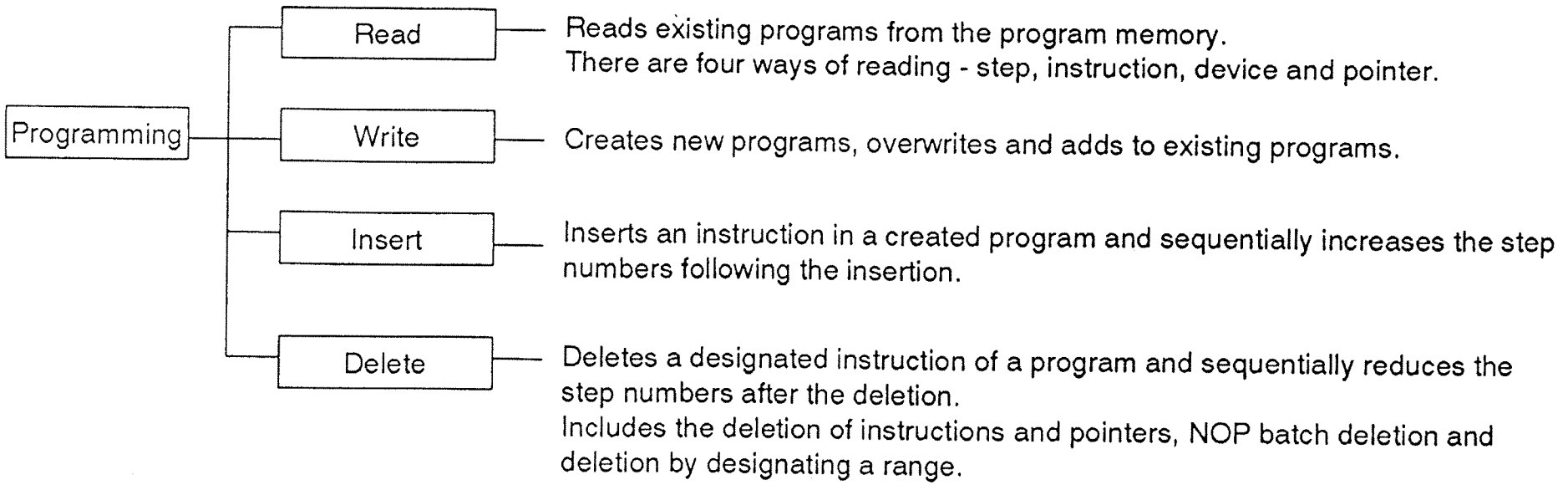
When the FX series is used, transfers between the PC built-in RAM and the memory cassette installed in the PC can be done with the HPP (see page 78).

* Writing and transfer cannot be done if the memory cassette is an EPROM or the memory-protect switch of the EEPROM is set to ON.



General function description

Programming functions:



POINT

(1) Usable program steps display

The usable program steps display can be displayed by pressing the [HELP] key during programming.

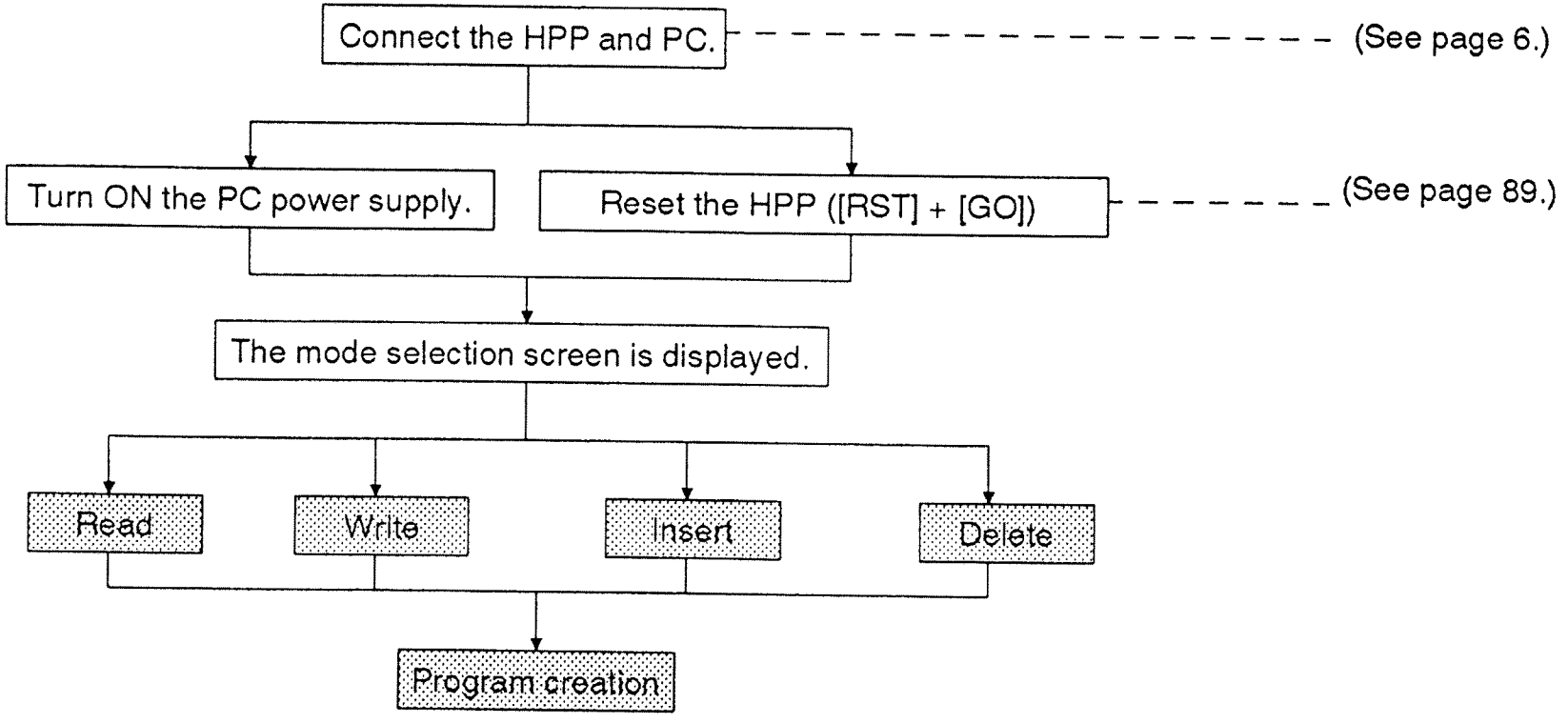
Pressing the [HELP] key again clears this display.

(2) Applied instructions

The applied instruction menus can be displayed and searched by pressing the [FNC] and [HELP] keys.

Programming outline

The programming procedure outline is shown below.



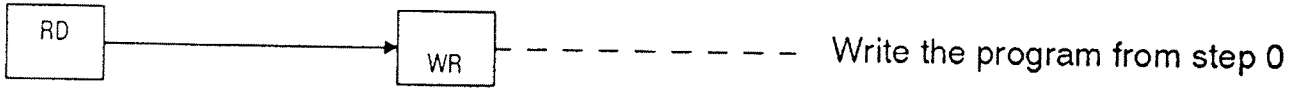
Programming operations

There are two programming methods, writing a new program and editing an existing program.

After starting up the HPP, all functions can be selected from the mode selection screen.

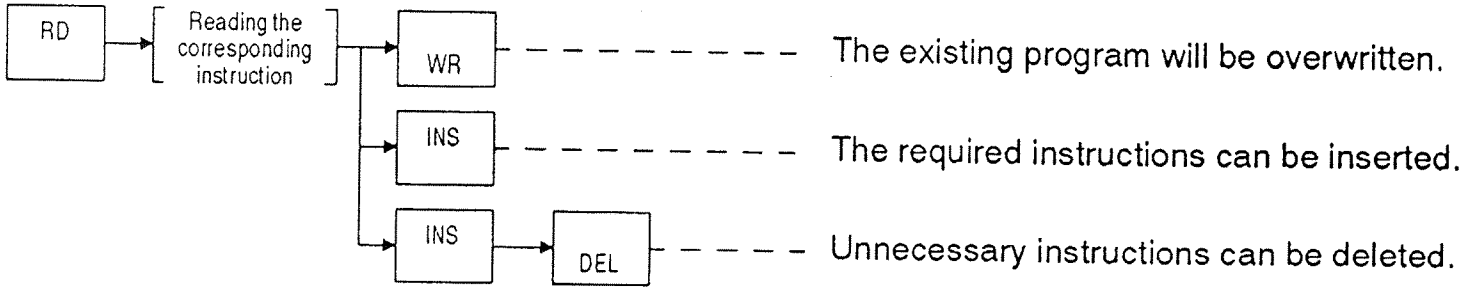
Key sequences and programming operations are shown below.

(1) Writing a new program.



(2) Editing an existing program.

First read the displayed program and then edit it.



3. PROGRAMMING

How to create a program

Creating a program using an instruction list.

Instructions consist of basic and applied instructions (see pages 96 to 98).

The basic operation for creating a program is shown below.

(1) Basic instructions (including step ladder instructions)

Directly input instructions to be set.

(2) Applied instruction

Input [FNC] <applied instruction number>

(The list of applied instructions can be displayed by pressing the [FNC] and [HELP] keys.)

Completion

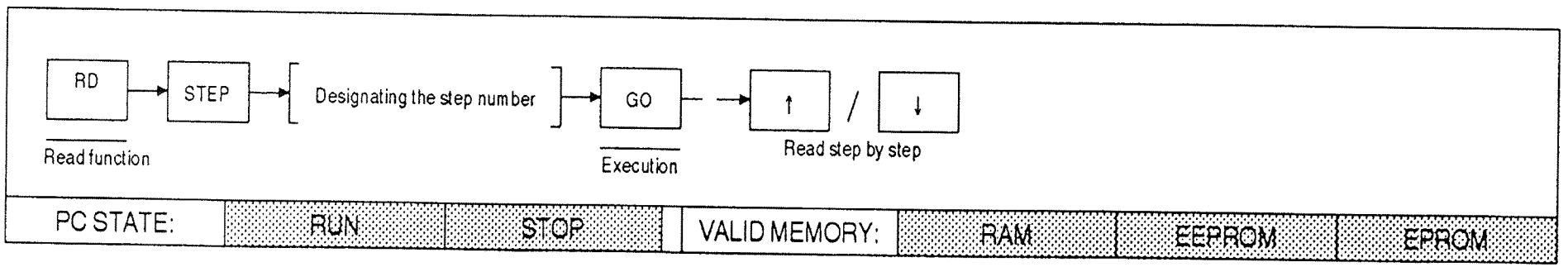
Programs are directly written to the PC program memory.

While programming, switching to other functions and menus is possible by pressing a function key or the [OTHER] key.

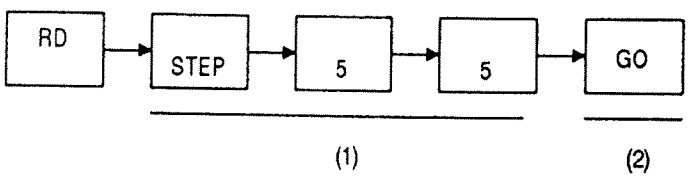
Reading by step number

Designate a step number to read and display a program.

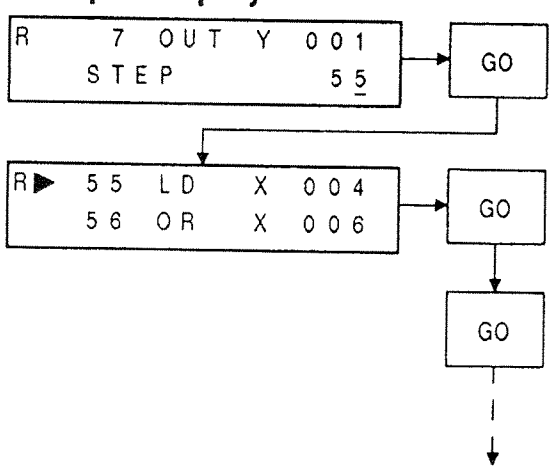
Basic operation



Sample operation 1 To read step 55 of the program



Sample display



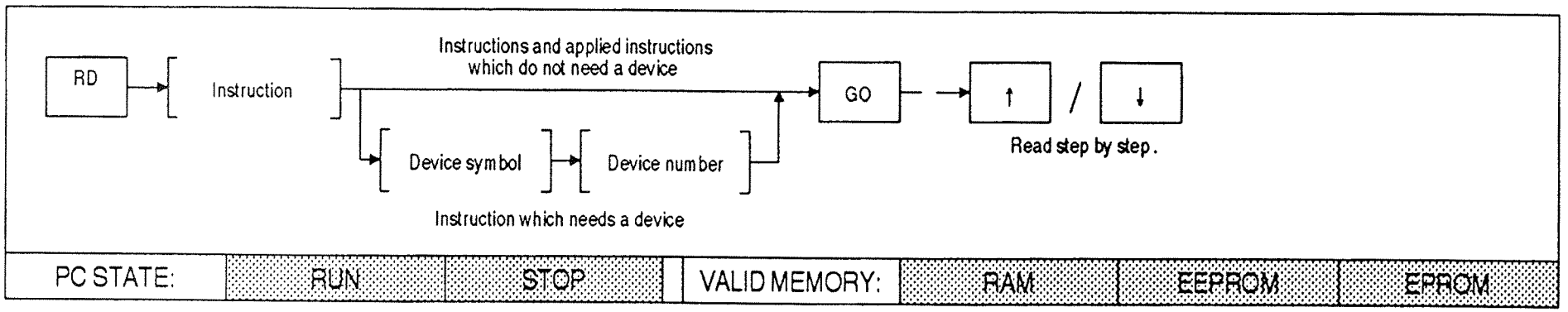
(1) Press the [STEP] key and enter the step number 55.
(2) Press the [GO] key to read the desired program lines.

* Two lines of the program are read and displayed beginning with the designated step number.
* If the designated step number corresponds to an operand, such as a T/C setting value, two lines of the program are read and displayed beginning with the corresponding instruction.
* The following instructions are displayed sequentially by pressing the [GO] key repeatedly.
* Programs can be displayed by scrolling one line at a time using the cursor key.

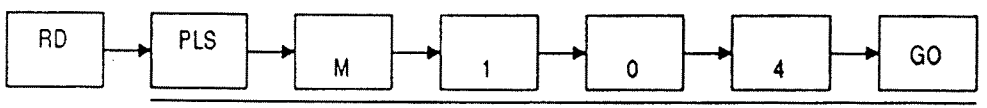
Reading by instruction

Designate an instruction to read and display a program.

Basic operation

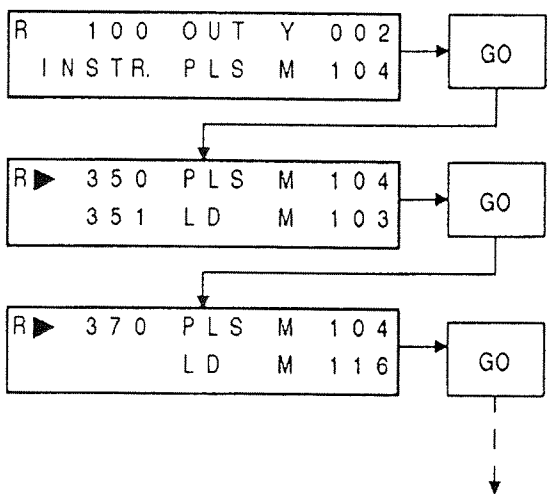


Sample operation 1 To read PLS M104



(1)

Sample display



(1) Press an instruction key and enter the device number, if applicable.
[In the case of an application instruction]
Input [FNC], [D], [1], [2], [GO], or [FNC], [1], [2], [GO]. Both instructions are searched whether or not pulse symbol [P] is pressed.

* The designated instruction is searched for starting from step 0. And then, two lines of the program are displayed on the screen beginning with the first detected instruction.
* The next and following occurrences of this instruction are found by pressing the [GO] key repeatedly.
* When the search of all designated instructions has been completed, or the designated instruction is not found, the "NOT FOUND" message will be displayed (Instructions after the END instruction are not read).
* Pressing the cursor key switches the reading mode to step order. This is the same for a pointer and a device.

Basic sequence instructions and applicable devices

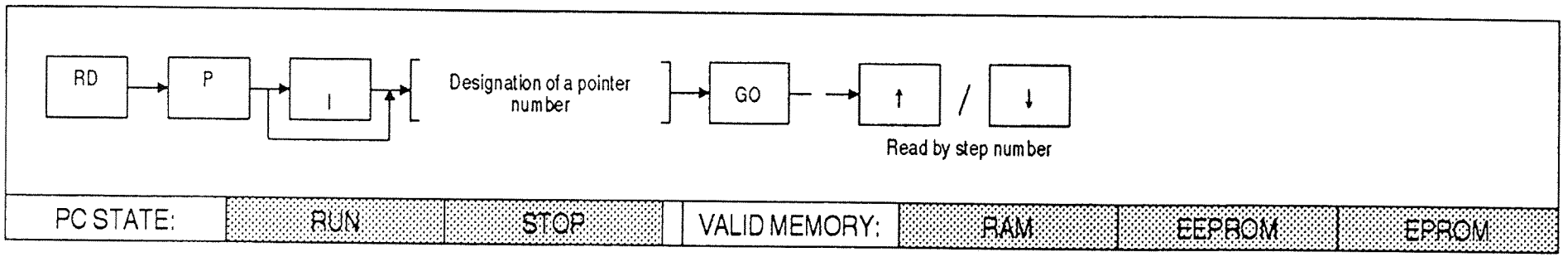
Instructions	Functions	Devices		Instructions	Functions	Devices
LD LOAD	Logical operation start (NO contact)	X, Y, M, S, T, C Special M		LDI LOAD INVERSE	Logical operation start (NC contact)	X, Y, M, S, T, C Special M
AND AND	Logical product (NO contact serial connection)	X, Y, M, S, T, C Special M		ANI AND INVERSE	Negative AND (NC contact serial connection)	X, Y, M, S, T, C Special M
OR OR	Logical OR (NO contact parallel connection)	X, Y, M, S, T, C Special M		ORI OR INVERSE	Negative OR (NC contact serial connection)	X, Y, M, S, T, C Special M
ANB AND BLOCK	Serial connection between blocks	*		ORB OR BLOCK	Parallel connection between blocks	*
OUT OUT	Coil drive	Y, M, S, T, C Special M		NOP NOP	No processing	For erasing programs or a space *
SET SET	Latch a device ON.	Y, M, S Special M		RST RESET	Latch a device OFF.	Y, M, S, T, C, V, Z, D, Special M, Special D
MC MASTER CONTROL	Common serial contact	N, Y, M		MCR MASTER CONTROL RESET	Common serial contact cancellation	N
PLS PULSE	A one-operation cycle pulse is generated at the leading edge of an input signal (turning on).	Y, M		PLF PULSE	A one-operation cycle pulse is generated at the fall of an input signal (turning off).	Y, M
MPS PUSH	Operation memory	*		MRD READ	Memory read	*
MPP POP	Memory read and reset	*		END END	Program completion	The final instruction at the end of a program. The program is returned to step 0.
STL STEP LADDER	Start of the step ladder	S		RET RETURN	Completion of the step ladder	*

- Instructions with an * symbol do not need an applicable device.
- Special M: Special Auxiliary Relay
- Special D: Special Data Register

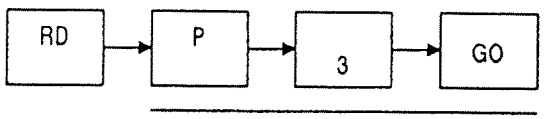
Read by a pointer

Designate a pointer to read and display the program.

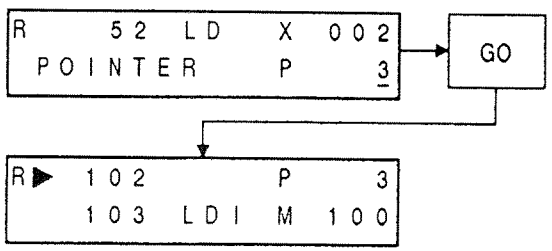
Basic operation



Sample operation 1 To read the label of pointer number 3



Sample display



(1) Press the [P] key and enter the pointer number, then press the [GO] key. The designated pointer number is read.

* Two lines of the program are read and displayed beginning with the designated label.

* If the designated label is not found, the "NOT FOUND" message will be displayed. (Instructions after the END instruction are not read.)

* This pointer search only looks for a pointer as a label. A pointer designated as an operand in an applied instruction will not be searched.

POINT

(1) P (pointer): The device that identifies the label to jump for a branch instruction.

(2) Label: The number that identifies the first step of branch program.

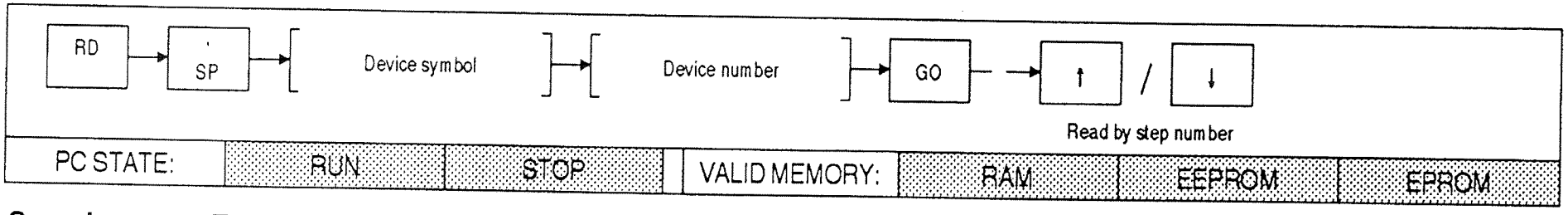
(3) I (interrupt pointer): The label for an interrupt program.

An interrupt program must end with an IRET (interrupt return) instruction.

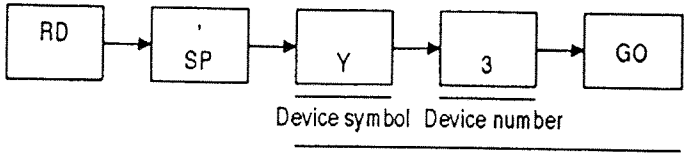
Réad by a device

Designate a device symbol and a device number to read and display a program.

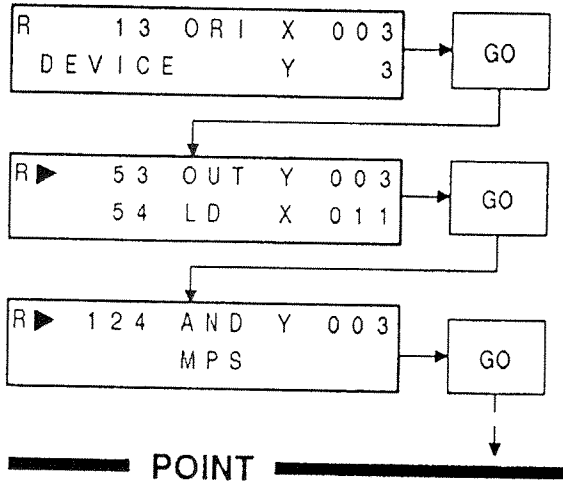
Basic operation



Sample operation 1 To read Y3



Sample display



(1) Press the [SP] key and enter a device symbol and a device number and then press the [GO] key.

* The designated device is searched for beginning with step 0. Two lines of the program are displayed beginning with the first detected instruction that contains the device.
* The next and following instructions containing this device are found by pressing the [GO] key repeatedly.
* When the search of all designated devices has been completed, or the designated device is not found, the "NOT FOUND" message will be displayed (Instructions after the END instruction are not read).

POINT

(1) Precautions when reading using a device

Only the X, Y, M, S, T, C, D, V, and Z basic instruction device can be searched by a device read (V and Z as indices are not searched).

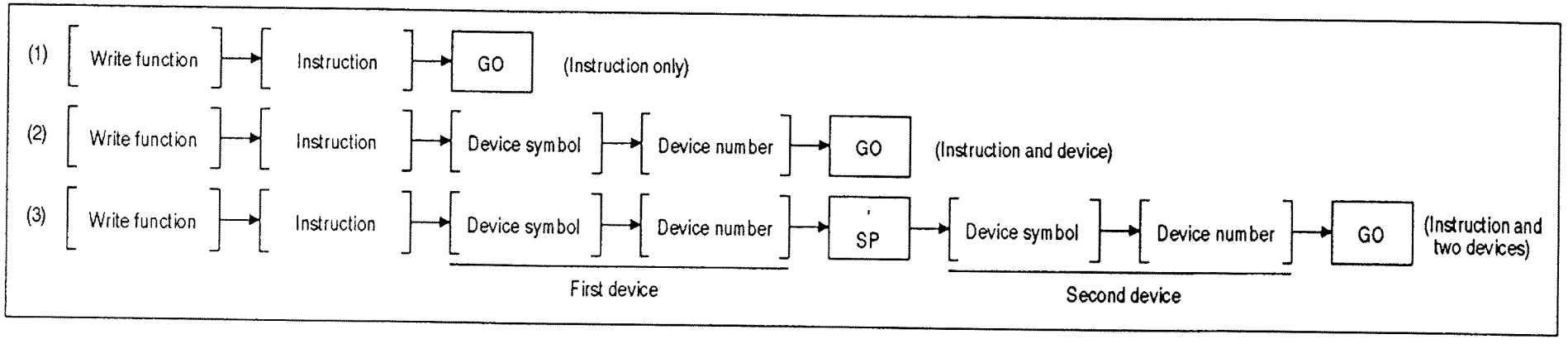
Only a D device used with the OUT instruction of a timer and a counter and the RST D instruction will be searched.

Input of a basic instruction

There are three ways of entering a basic instruction and a step ladder instruction:
instruction only, instruction and device, instruction and two devices.

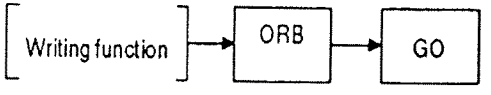
These three input methods are explained below.

Basic operation



(1) Instructions that do not require any devices:
The ANB, ORB, MPS, MRD, MPP, RET, END and NOP instructions are written without designating a device.

Sample operation 1 To input (write) the ORB instruction



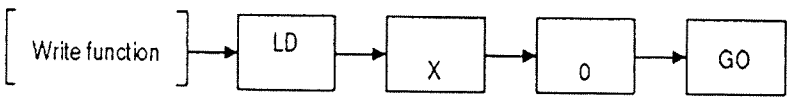
Sample display

```
W      6  O R B
▶      7  N O P
```

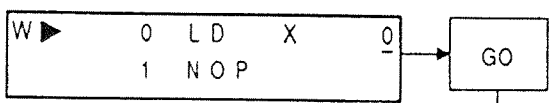
* Press the [ORB] key and then press the [GO] key. Writing has now been completed.

(2) Instructions that are written by an entering an instruction and a device:
The LD, LDI, AND, ANI, OR, ORI, SET, RST, PLS, PLF, MCR, STL and OUT (except T and C) instructions are written by inputting an instruction and a device.

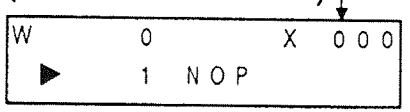
Sample operation 2 To input (write) the LD X0 instruction



Sample display (Before confirmation)



(After confirmation)

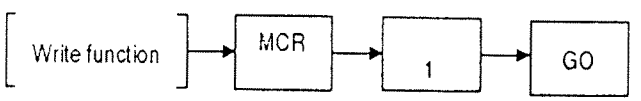


* Press the [LD], [X] and [0] keys.

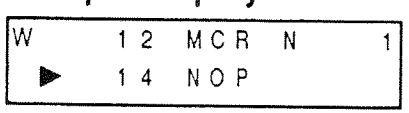
* When the HPP is waiting for a device symbol or a device number key to be input the prompt appears at the end of the line.

* Press the [GO] key. Writing has now been completed.

Sample operation 3 To input (write) the MCR N1 instruction



Sample display

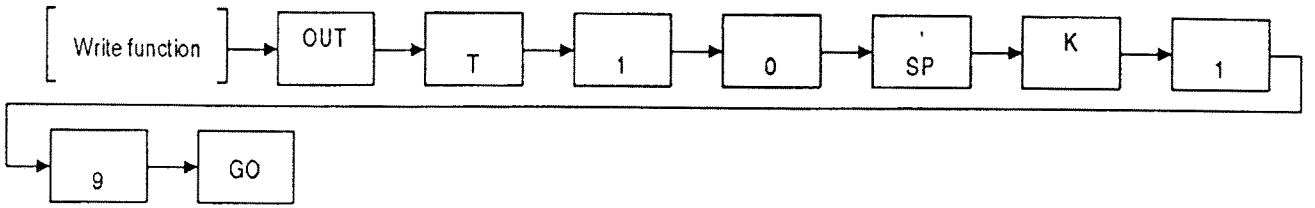


* The nesting level device symbol "N" is automatically displayed when the [MCR] key is pressed. Then press the [1] key and the [GO] key to complete the writing.

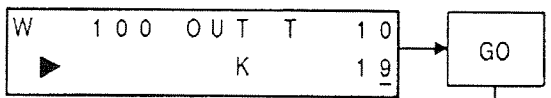
The program step numbers are determined automatically. See page 96 for the step numbers occupied by each instruction.

(3) Instructions that are written by entering an instruction, and two devices: MC and OUT (T,C) instructions.

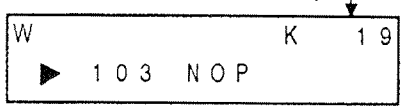
Sample operation 4 To input (write) the OUT T10, K19 instruction



Sample display (Before confirmation)



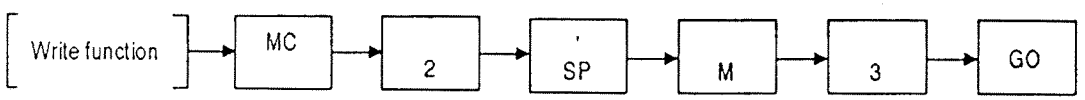
(After confirmation)



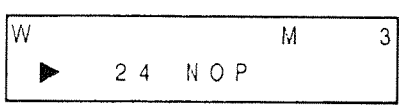
* Press the [OUT], [T], [1], [0], [SP], [K], [1] and [9] keys and then press the [GO] key. Writing has now been completed.

* When the HPP is waiting for a device symbol or a device number key to be input the prompt appears.

Sample operation 5 To input (write) the MC N2 M3 instruction



Sample display

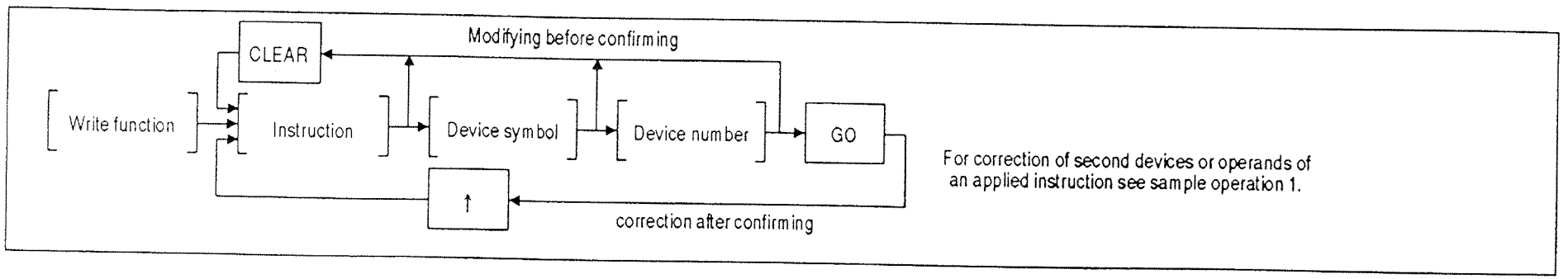


* The nesting level device symbol "N" is automatically displayed when the [MC] key is pressed. Press the [2], [SP], [M] and [3] keys and then press the [GO] key. Writing has now been completed.

Modifying before confirming :

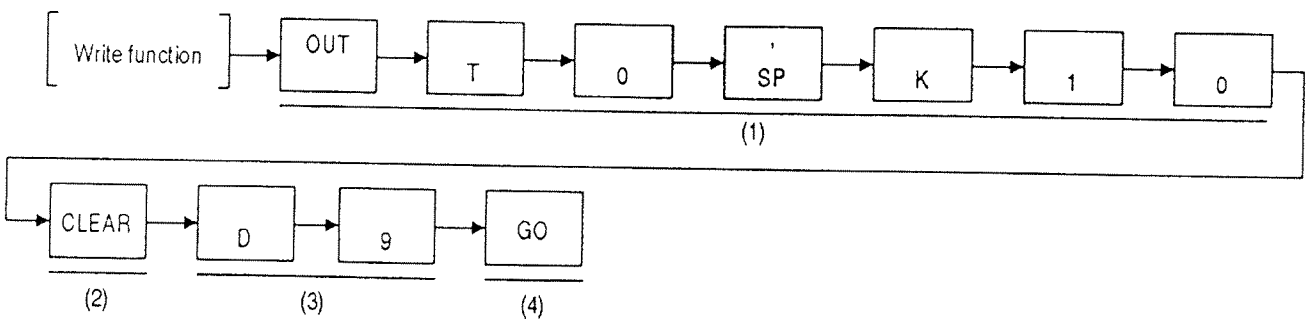
This section explains how to correct keyed-in data while entering an instruction.

Basic operation

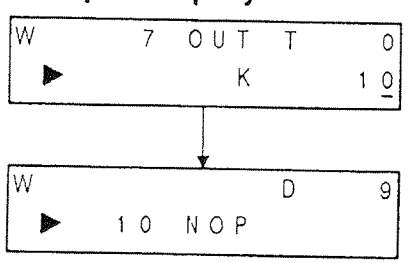


(1) Correcting an instruction with an operand (before confirming)

Sample operation 1 To correct the instruction OUT T0, K10 to be OUT T0, D9 before confirmation with the [GO] key



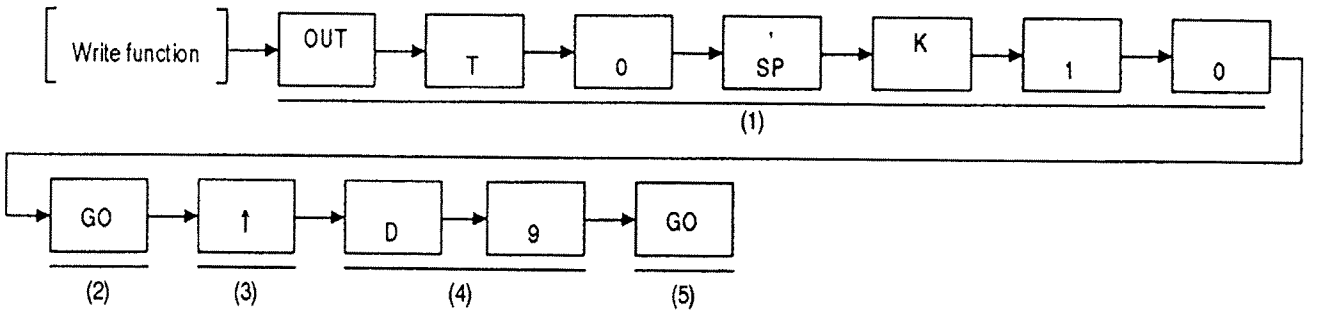
Sample display



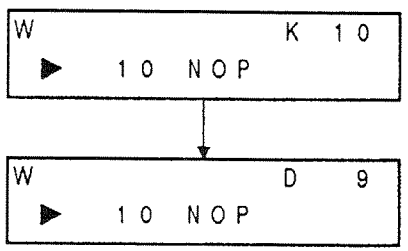
(1) Enter an instruction, the first device and the second device.
(2) To cancel the second device press the [CLEAR] key once.
* If the [CLEAR] key is pressed twice the modification is executed from step 7.
(3) Input the correct second device.
(4) Press the [GO] key. Writing of the instruction has now been completed.

(2) Correcting an instruction with an operand (after confirming)

Sample operation 2 To correct the instruction OUT T0, K10 to be OUT T0, D9 after confirmation with the [GO] key



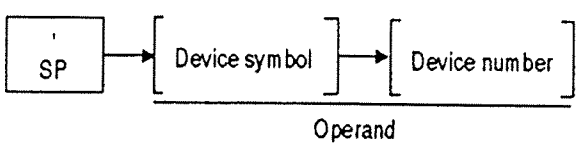
Sample display



(1) Enter an instruction, the first device, and the second device.
(2) When the [GO] key is pressed, and the input (writing) of (1) has been completed, the line cursor moves to the following step.
(3) Move the line cursor to the K10 position.
(4) Enter the correct second device.
(5) Press the [GO] key.
Writing of the instruction has now been completed.

POINT

(1) Operand
An operand is a device used by an instruction.
Always make sure to input [SP] and the operand(s) when inputting an operand.
For example, in the case of the MOV instruction
[MOV D0 D1]
Operand



Inputting an applied instruction

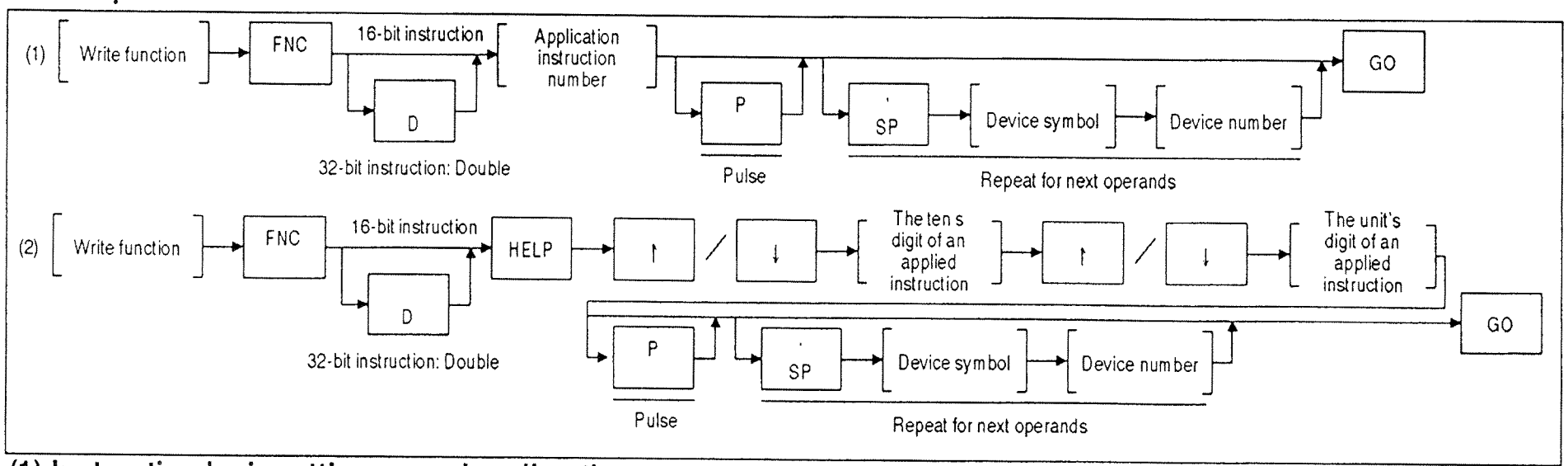
When inputting an applied instruction use the [FNC] key and the applied instruction number.

The applied instructions do not have an instruction symbol key. There are two ways to input instruction numbers:

(1) Inputting the number directly

(2) Searching the applied instruction menu with the HELP function.

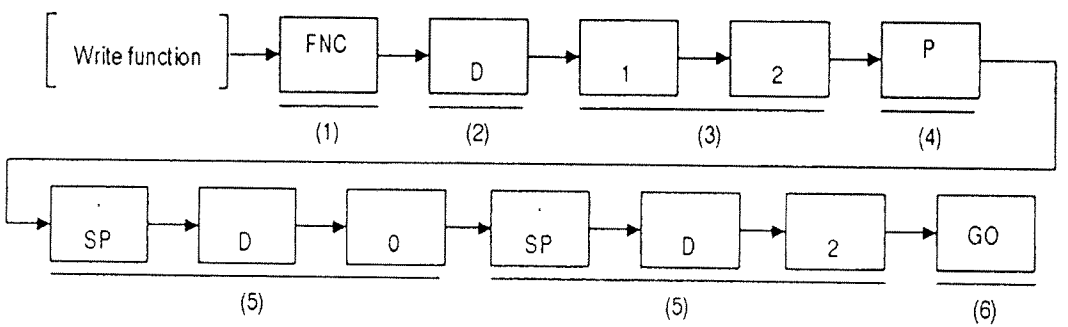
Basic operation



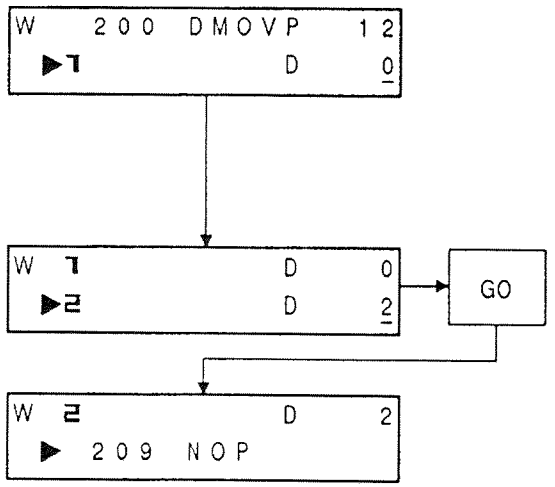
(1) Instruction by inputting a number directly

* When the applied instruction number has been decided press the [FNC] key and then input the number directly (See the instruction lists on pages 97 and 98).

Sample operation 1 To write the DMOVP D0, D2 instruction



Sample display



(1) Press the [FNC] key.
(2) When designating 32-bit (D: double) instructions, before or after the applied instruction number is input, press the [D] key.
(3) Input the applied instruction number.
(4) To designate a P (pulse) instruction press the [P] key after entering the applied instruction number.
(5) To enter a device input [SP], a device symbol and a device number.
* The instruction symbol is also displayed on the screen along with the instruction number.
(6) Press the [GO] key.
Writing of the instruction has now been completed.

POINT

(1) Designation of D and P prefix/suffix:
D and P can be input in the order displayed (Sample operation 1) or input by keying in after the instruction number.
D and P themselves can be input in any order.
If you are not sure whether the instruction in question is effective as a D and/or P instruction, press the [HELP] key to display the list of instructions while keying in the instructions. See the point on Page 39.
FX_0 has D instructions, but not P instructions.
(2) An ASC instruction can be read and displayed in ASCII characters.
However, an ASC instruction cannot be written by the HPP.
(3) If an applied instruction is input which the PC model does not have an error occurs.

(2) Inputting an instruction by referring to the instruction menu displayed with the HELP function

When an applied instruction number is not known, press the [FNC] key and press the [HELP] key.

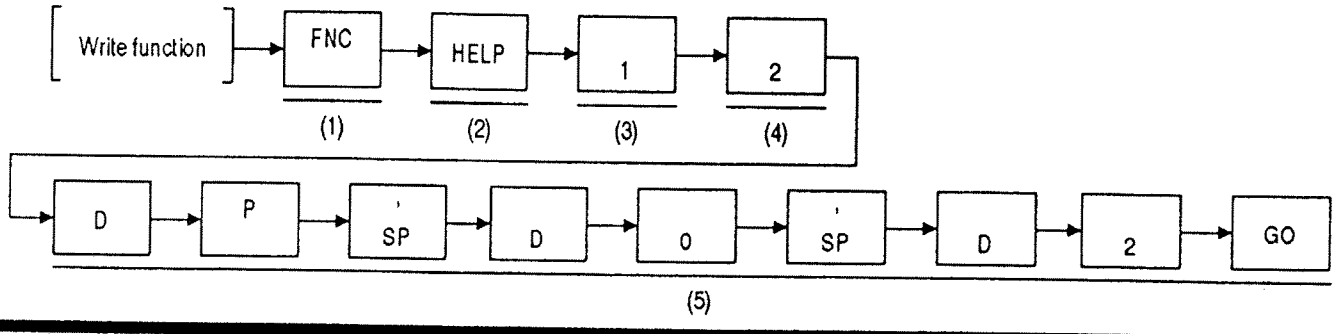
There are 10 applied instruction classifications:

The screen displays the applied instruction classifications and then the instructions symbols menu.

Select the applied instruction number from these screen guides.

0 Program flow
1 Transfer and comparison
2 Four arithmetical operations and logical operation
3 Rotation and shift
4 Data processing
5 High-speed processing
6 Convenient instruction
7 External device FX I/O
8 External device FX SER
9 External device F2

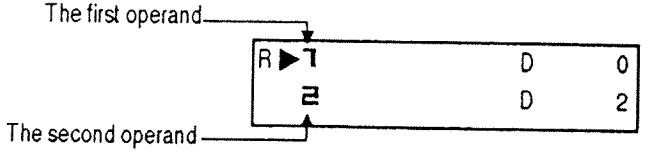
Sample operation 2 To write a DMOVP D0 D2 instruction



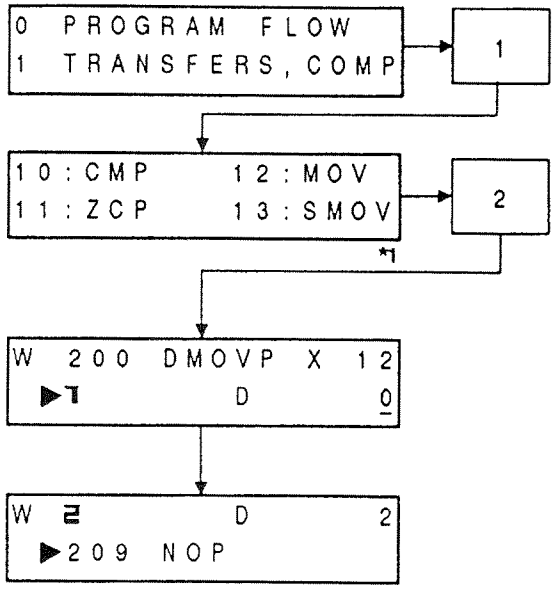
POINT

(1) Applied instructions are often made up of an instruction (FNC number) and several operands.
When the HPP is used, a number (1, 2, 3, 4, 5) is displayed to the left of each operand to indicate which operand is currently displayed on the screen.

Sample display



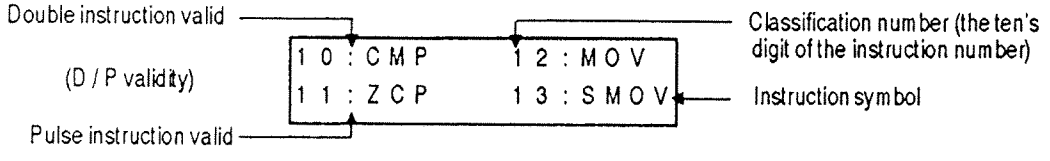
Sample display



(1) Press the [FNC] key.
(2) Press the [HELP] key to display the applied instruction classifications.
Screens can be scrolled by using the cursor key.
(3) Select the required applied instruction group by its number.
The screen changes to display the menu of instruction numbers and instructions corresponding to that selected group (The group number comes from the ten's digit of the instruction number).
The instructions of [ ]0 to [ ]9 are displayed on three screens.
Screens can be scrolled by using the cursor key.
Only valid applied instructions are displayed on the list in accordance with the model selection.
(4) Select the instruction symbol by its unit's digit.
An applied instruction number is designated with the key operations in steps (3) and (4).
(5) Follow steps (4) to (6) of sample operation 1.

POINT

(1) Contents of the applied instruction menu.

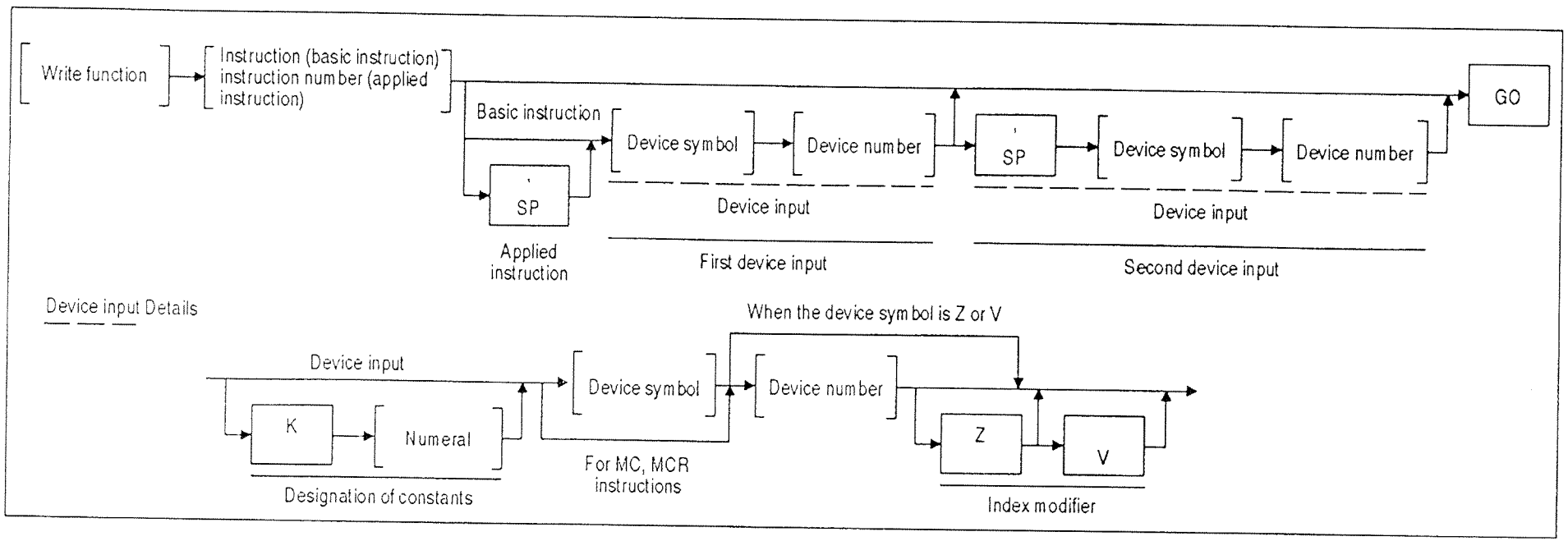


The dot(s) appearing between the instruction number and instruction symbol show the validity of the D/P prefix/suffix. The upper part indicates a valid D (double) instruction and the lower part indicates a valid P (pulse) instruction.

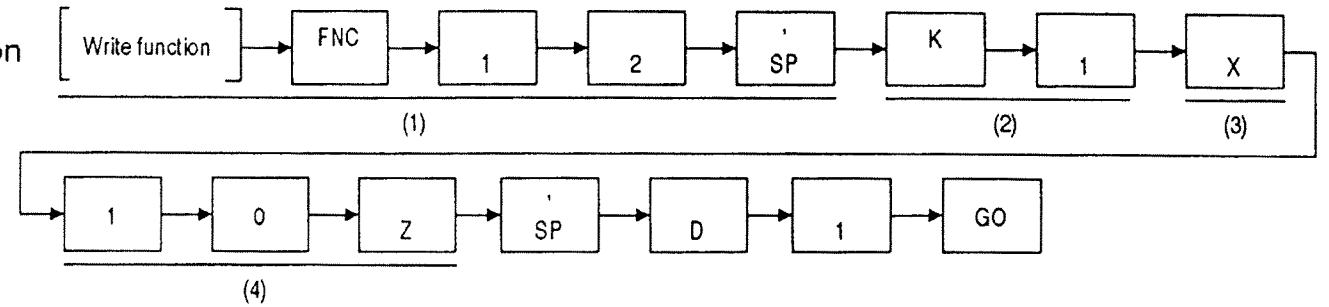
Inputting a device

This section explains how to input a device.

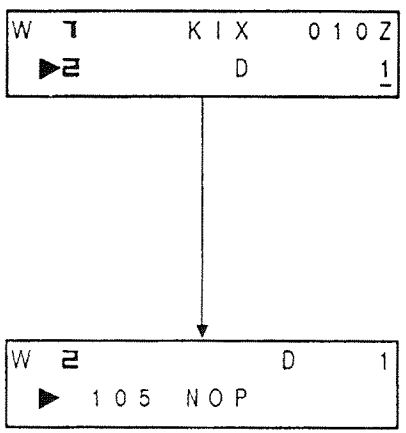
Basic operation



Sample operation 1 To write the MOV K1X010Z, D1 instruction



Sample display



(1) Following the writing procedures, input an instruction and an instruction number.

(2) Designate the necessary number of digits.
K1 to K4 (16-bit instruction) and K1 to K8 (32-bit instruction) are valid for the digit designation. K1 indicates 4 bits.

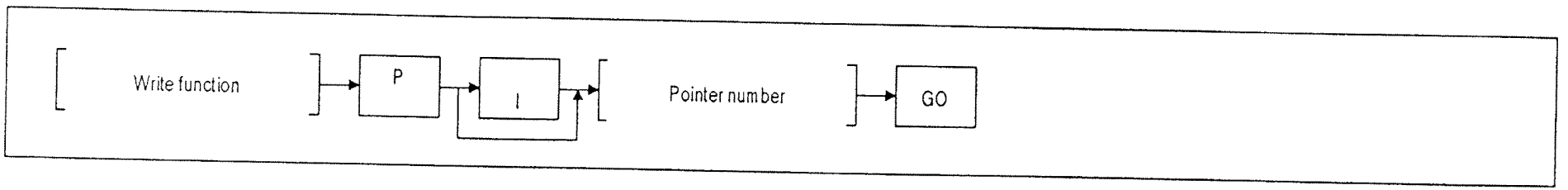
(3) Input the device symbol.
For the MC and MCR instructions the nesting level device symbol "N" is automatically displayed.

(4) Input a device number.
An index modifier Z of V, can be appended to a device number.

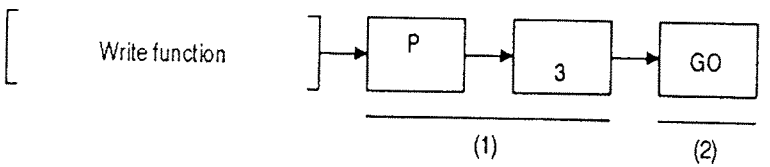
Label inputting (P, I)

When using P (pointer) and I (interrupt pointer) in a sequence program as labels, input them in the same way as an instruction.

Basic operation



Sample operation 1 To write label number 3



Sample display

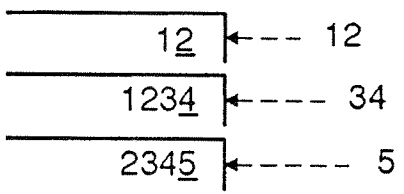
```
W    1 0 0          P    3
▶    1 0 1   N O P
```

(1) Press the [P] key or [P] and [I] keys and then input the label number.
P and I used as labels are handled in the same manner as instructions.
(2) Press the [GO] key to write the keyed in pointer or interrupt pointer.

Inputting numerical values

When writing a program. many different numbers are used (step numbers, device numbers, pointer numbers, and applied instruction numbers). These numbers may contain up to four digits. A keyed-in digit is displayed at the far-right column and the previous input is shifted to the left.

Sample operation 1 To input 1, 2, 3, 4, and 5 to the four-digit display area



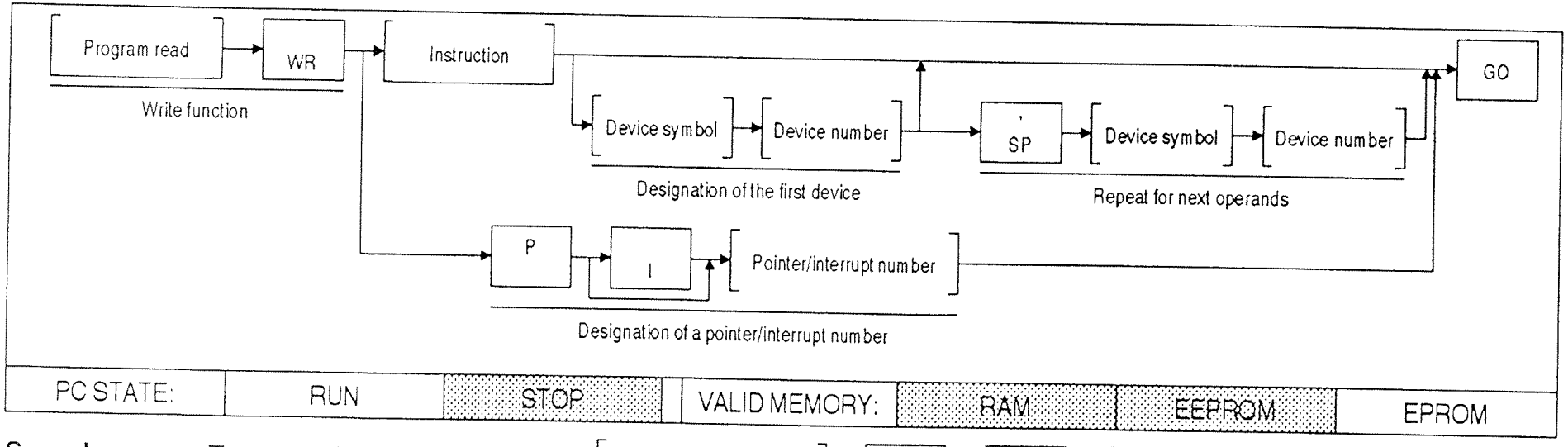
* Each time a number key is pressed, the keyed-in number is displayed in the far-right column and the previously keyed-in numbers are shifted to the left. Therefore, when a number with more than the allowed number of digits is keyed in, the excess digits disappear from the screen beginning with the first keyed-in digit.
  If only three or less digits are keyed-in for X and Y (which accept up to four digits) the other digit places are filled with leading zeros.
* Only the number displayed on the screen is written to memory.

Therefore, pay attention to the screen display when inputting.

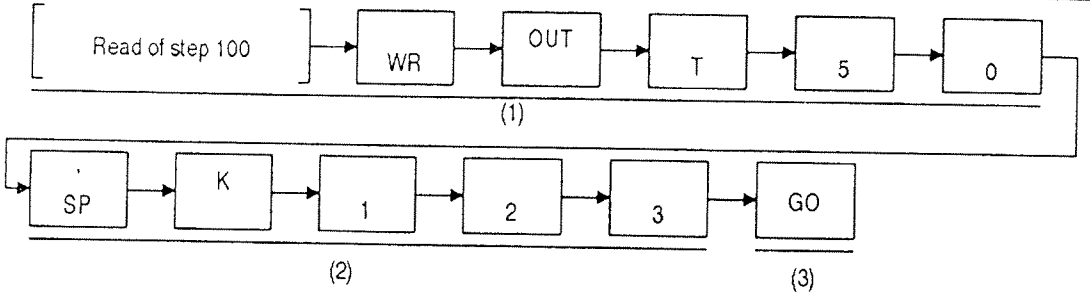
Writing of instructions and pointers

This section explains how to edit a program by overwriting an instruction.

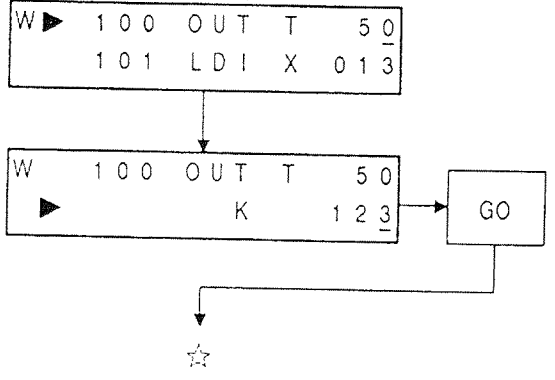
Basic operation



Sample operation 1 To overwrite the OUT instruction T50 K123 to step 100

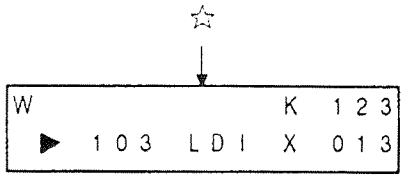


Sample display



(1) Read the program from step number 100 and select [WR] function. Enter the instruction, a device symbol and a device number.
(2) To write the second device press the [SP] key and then enter a device symbol and a device number.

* The set value of timers and counters can be changed using the monitoring function (See page 74).



(3) Press the [GO] key to write the new instruction.

* To overwrite instructions in steps near the currently displayed step number, move the line cursor to the required step number.

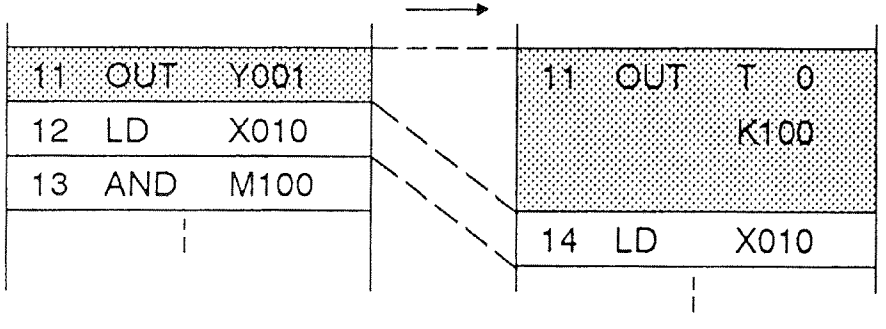
POINT

(1) Processing step numbers when overwriting

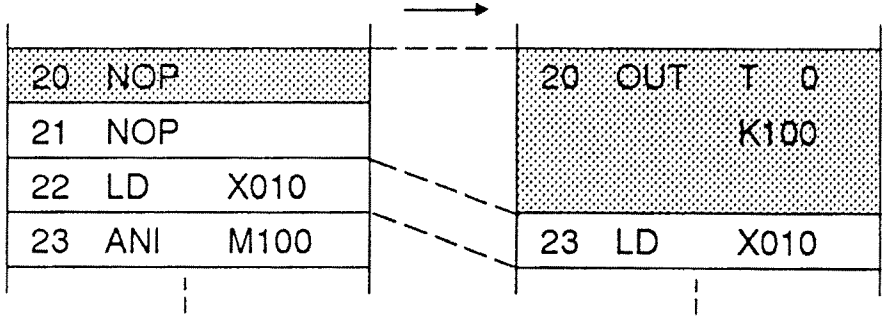
If the number of steps used by the instructions before and after overwriting differ from each other, the ensuring instructions are shifted as illustrated.

When overwriting is made over an NOP instruction, the NOP is deleted and the new instruction is inserted.

When overwriting OUT T0 K100 to step 11



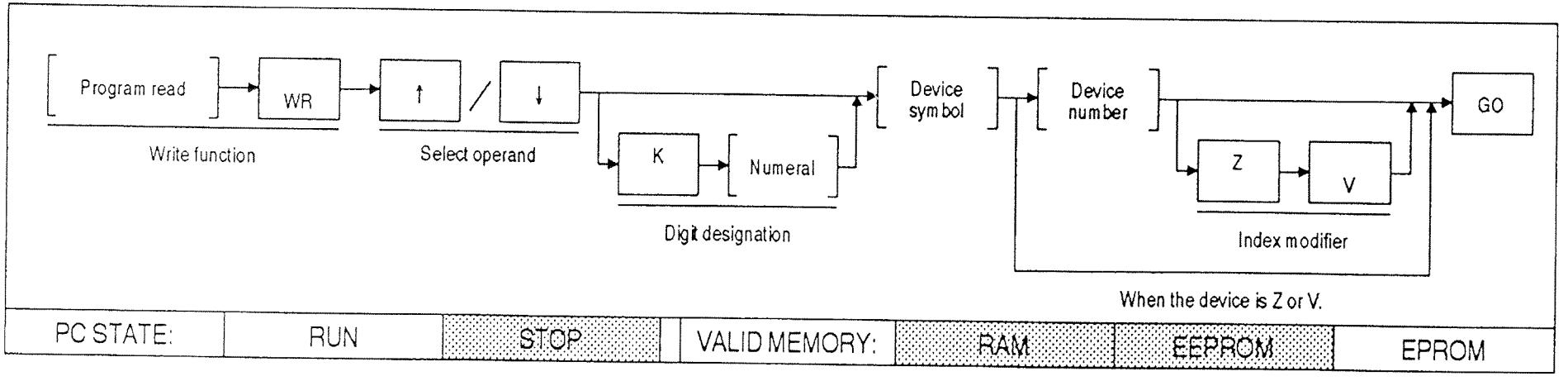
When overwriting OUT T0 K100 to step 20



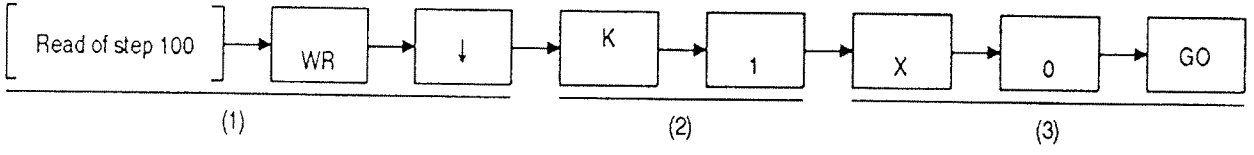
Device writing

This section explains how to edit an instruction operand.

Basic operation



Sample operation 1 To change the operand of the MOVP instruction at step 100 from K2X1 to K1X0



Sample display

(1) Read the program from step number 100 and select the [WR] function. And then, move the line cursor to the position of the operand to be changed.
(2) When designating a digit, press the [K] key, and input the numerical value.
(3) Input the device symbol and device number and press the [GO] key. Then the designated device is altered.

* Lines that can be altered in this manner are lines without a step number. (When changing a line with a step number follow the instruction writing operation.)

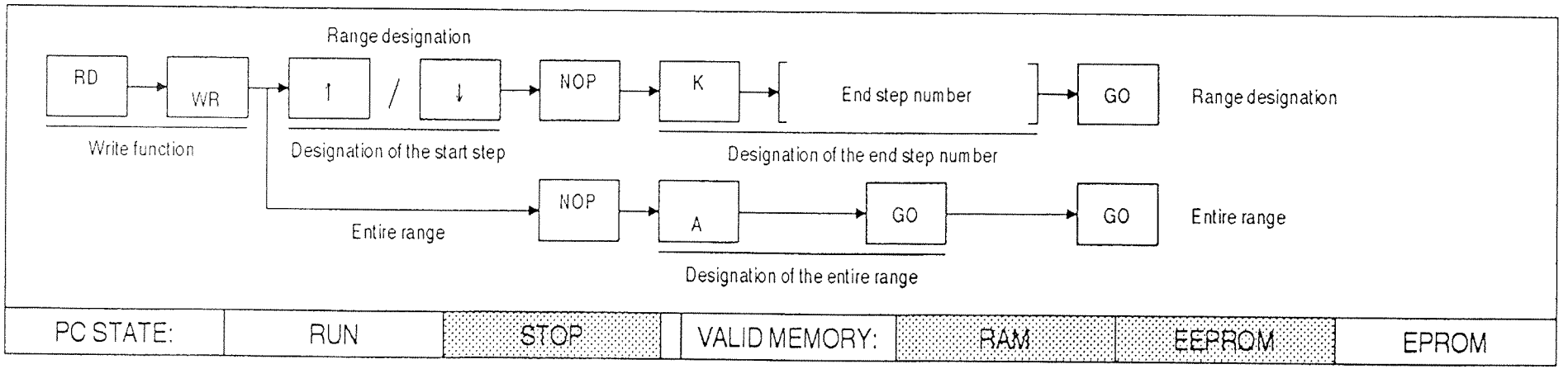
POINT

(1) K-number of grouped bit devices:
K1 to K8 can be designated. Multiples of 4 bits are allocated. Therefore, K2 allocates 8 bits.
(2) Z and V devices:
Z and V devices represent an index register. The index register is used as a modifier to the device number.

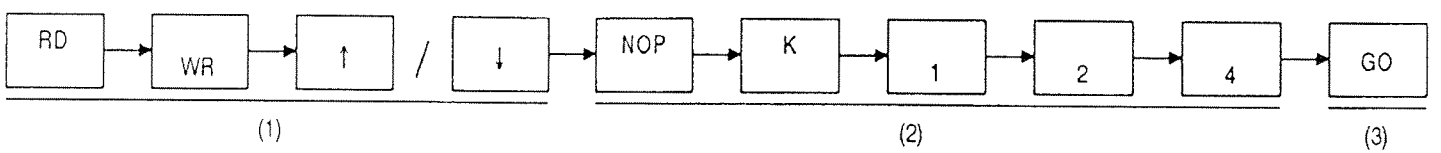
NOP batch writing

This section explains how to perform NOP batch write operation to a specific range.

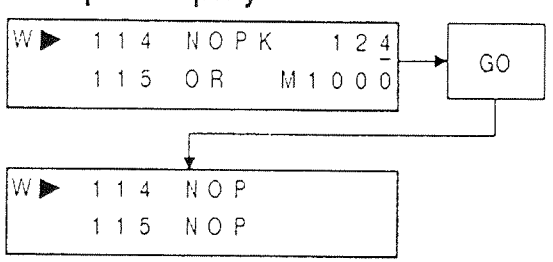
Basic operation



Sample operation 1 To batch write NOP from step 114 to step 124



Sample display

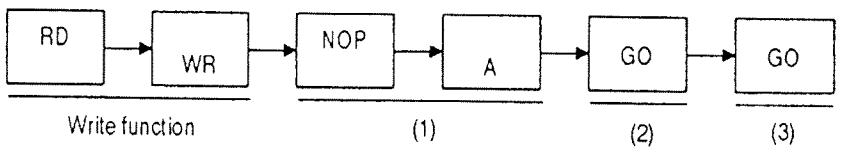


(1) Move the line cursor to the start step where NOP is written after pressing the [RD] and [WR] keys.
Writing cannot begin on lines without step numbers.
(2) Input [NOP], [K] and the end step number.
(3) NOP is batch written to the designated range by pressing the [GO] key.
If the designated end step number is in the middle of an instruction the whole of this instruction is cleared with NOP's.

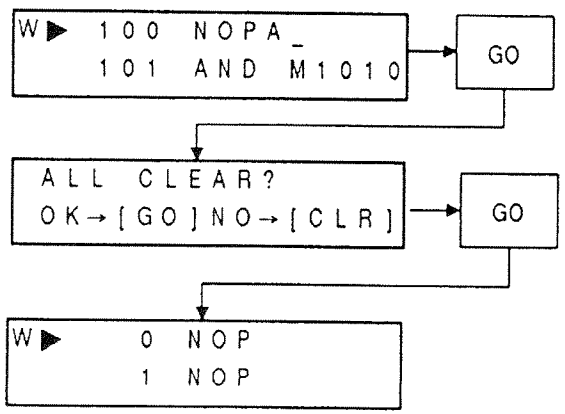
IMPORTANT

Writing of each instruction is valid for RAM/EEPROM. Follow the procedure for transfering to EEROM after writing to the RAM built in the PC. However, when FX_0 is used, each instruction is written directly to the EEPROM built in the PC.

Sample operation 2 To write NOP instructions throughout the entire program memory



Sample display



(1) Press the [RD], [WR], [NOP] and [A] keys.
* In this case the position of the line cursor is unrelated to the writing range.

(2) Press the [GO] key. The "ALL-CLEAR?" confirmation message is displayed.

(3) To respond to the "ALL CLEAR?" message, press the [GO] key. The line cursor moves to step number 0 and the entire program is cleared (i.e., NOP instruction is written to every program step).

IMPORTANT

When a NOP instruction is written to every program step, all the parameters values are reset to the default settings. Latch elements are also reset. Therefore, the comment allocation will be 0 blocks, file register allocation will be 0 blocks, and memory capacity will be 2K steps (in the offline mode or online mode when there is not any memory cassette installed). If a memory cassette is installed, the memory capacity is defaulted to the maximum memory cassette capacity.

Page 109 gives details about parameter default values.

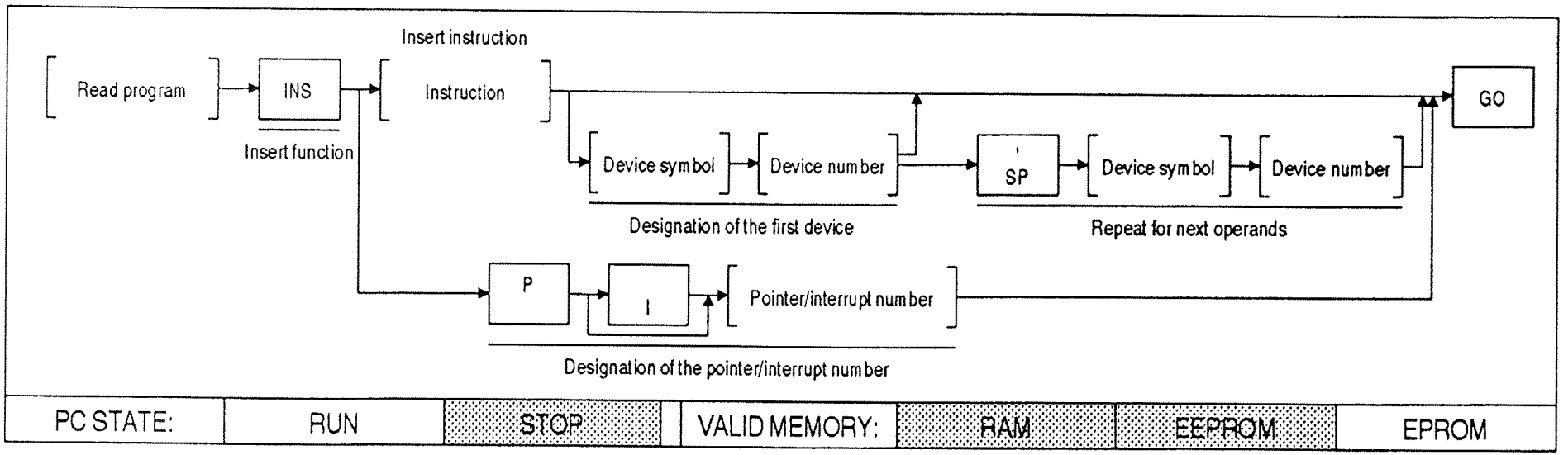
POINT

(1) Devices that can be latched in the FX series (For details, see page 109):

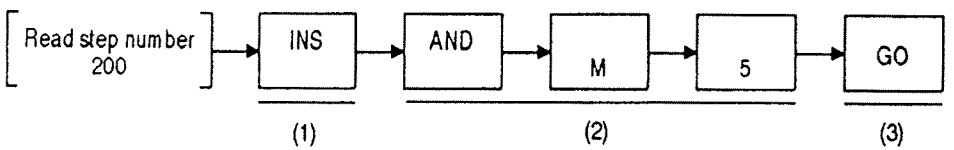
M (internal relay)	C (16-bit counter)	D (data register)
S (state)	C (32-bit counter)	D (file registers)
T (timer)		

This section explains how to insert an instruction at a designated position in a program.

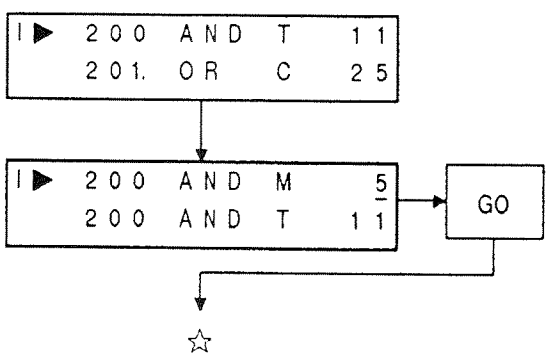
Basic operation



Sample operation 1 To insert the AND M5 instruction before step 200

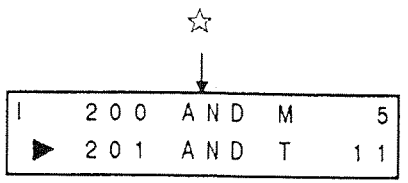


Sample display



(1) Read the program and press the [INS] key. Then insert the required instruction at the step designated by the line cursor.
Lines that do not display a step number cannot be selected for the insert function.

(2) Key in the instruction, a device symbol and a device number or pointer symbol and pointer number.



(3) Press the [GO] key to complete the insertion. Step numbers of the ensuing instructions are reassigned automatically.

* To insert instructions in steps near the currently displayed step number move the line cursor to the required step number.

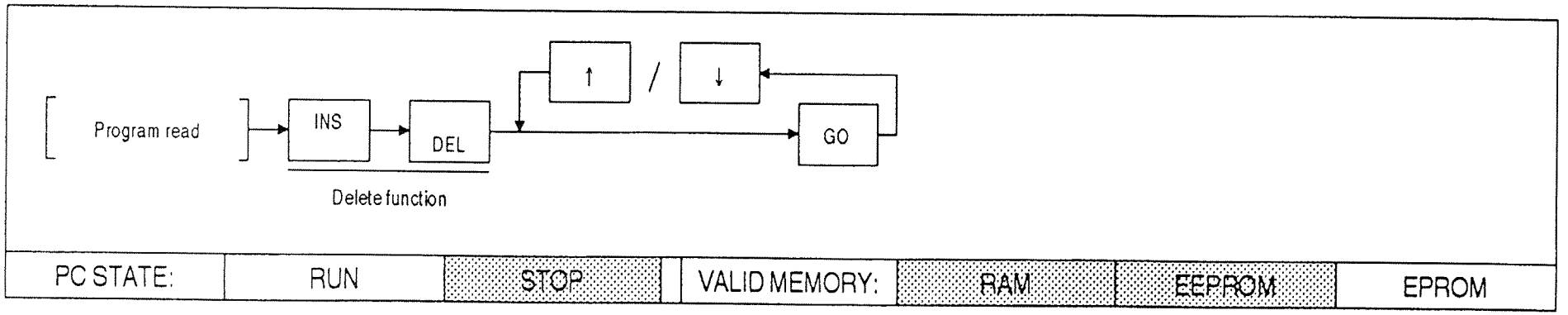
POINT

(1) Insertion is not possible when the current program has filled the whole memory area. If insertion is attempted when the memory area is full the "STEP OVERFLOW" error message will be displayed.

Deleting an instruction and a pointer

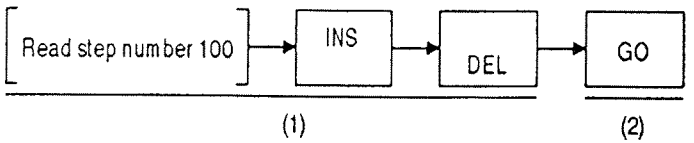
This section explains how to delete an instruction designated by the cursor in one step.

Basic operation

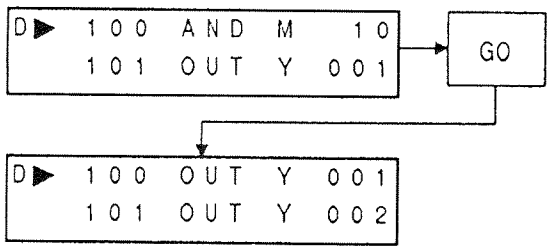


Sample operation 1

To delete the AND M10 instruction at step 100



Sample display



(1) Read the program and then press the [INS] and [DEL] keys.

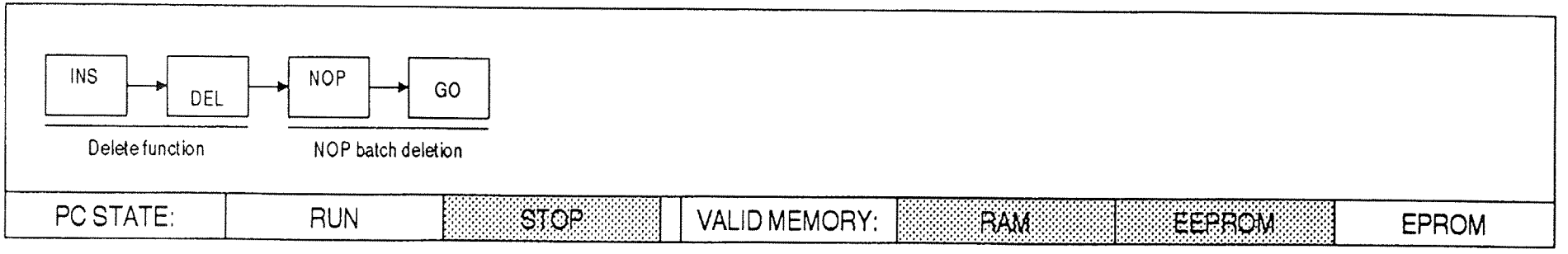
(2) The instruction pointed to by the line cursor is deleted by pressing the [GO] key. The subsequent step numbers are reassigned automatically.

* To delete instructions in steps that are near the currently displayed step number, move the line cursor to the required step number.
* The instruction to be deleted should be designed with the line cursor. However, the delete function cannot be used for a line that does not have a step number displayed.

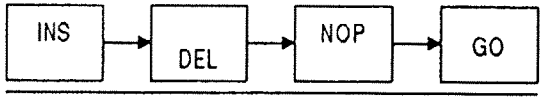
NOP batch deletion

This function allows all the NOP's between program instructions to be removed simultaneously.
This section explains how to batch delete all NOP instructions from a program.

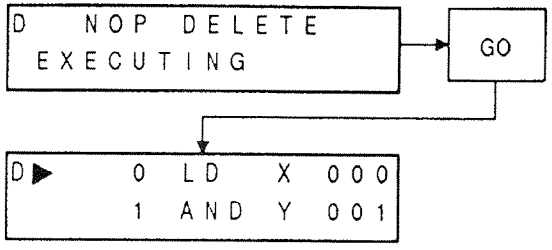
Basic operation



Sample operation 1 To batch delete NOP



(1)



(1) Press the [INS], [DEL] and [NOP] keys and then press the [GO] key. The screen shown on the left is displayed. All NOP instructions are deleted from the program.

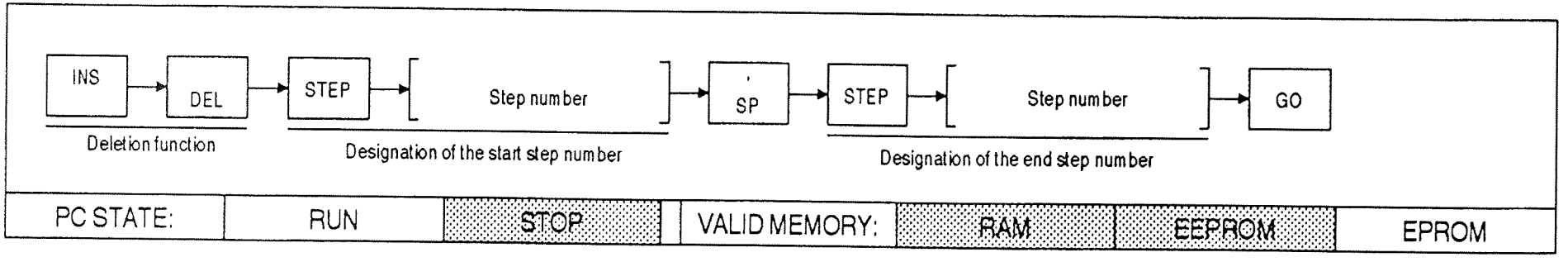
* After all NOP instruction steps have been deleted the step numbers are reassigned automatically.

* After the completion of batch-delete the first line of the program (step number 0) is displayed on the first line of the screen.

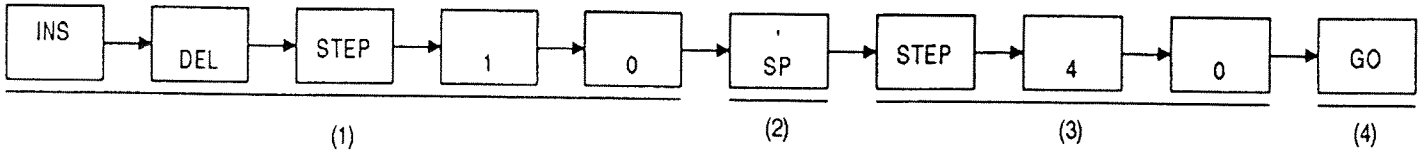
Range deletion

This section explains how to batch-delete program steps within a specified range.

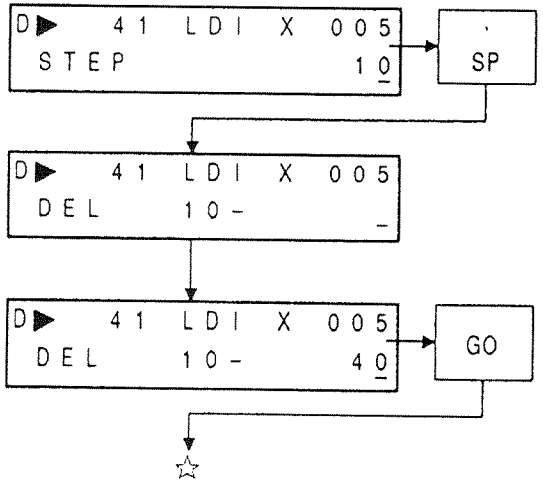
Basic operation



Sample operation 1 To delete program steps from step number 10 to step number 40



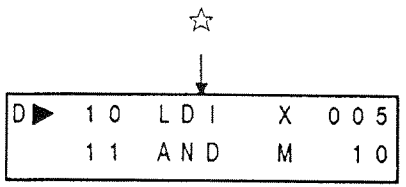
Sample display



(1) Press the [INS], [DEL] and [STEP] keys and then input the start step number of the deletion range.
If the entered step number lies inside an instruction the beginning of the instructions becomes the delete start step.

(2) Press the [SP] key.
The range is displayed on the screen.

(3) Press the [STEP] key and then input the end step number of the deletion range.
If the entered step number lies inside an instruction the last step number of the instructions becomes the final deleted step.



(4) Press the [GO] key to delete the program steps in the designated range.
The step numbers of the ensuing instructions are reassigned automatically.
The program is displayed with the deletion start step number as the first line.

POINT

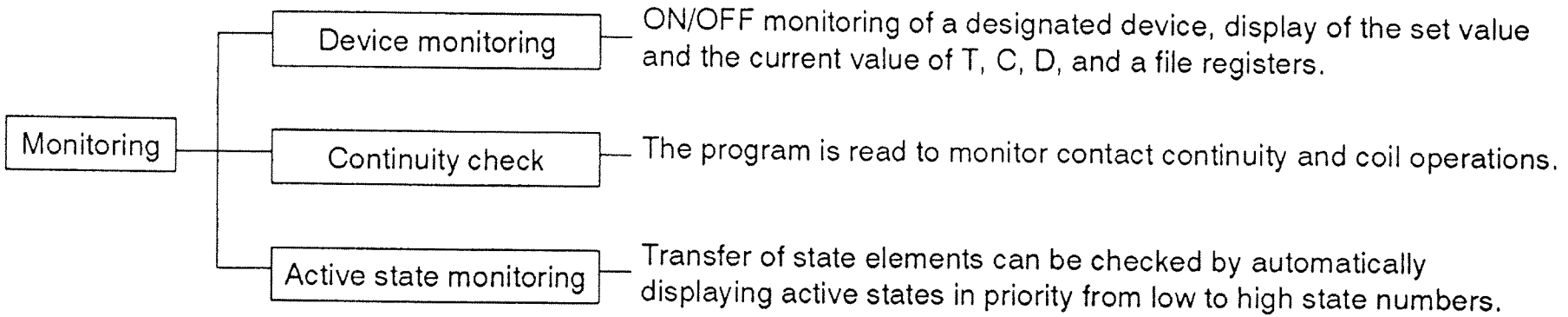
(1) How to cancel this setting
Press the [CLEAR] key before pressing the [GO] key. The first time the [CLEAR] key is pressed, the end step number is cleared. The second time the [CLEAR] key is pressed, the start step number is cleared.

4. MONITORING

The monitoring function is used to display the operating and control states of PC on the HPP screen.

General function description

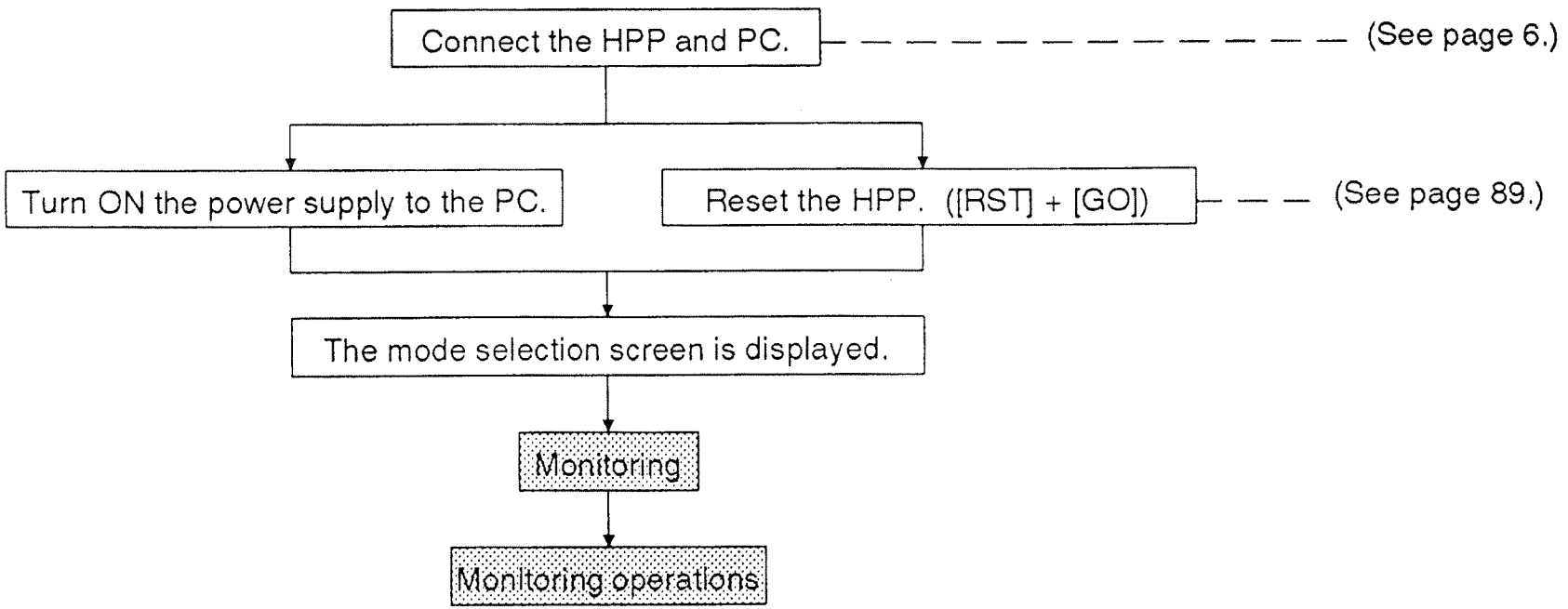
Monitoring consists of the following functions:



4. MONITORING

Outline procedure

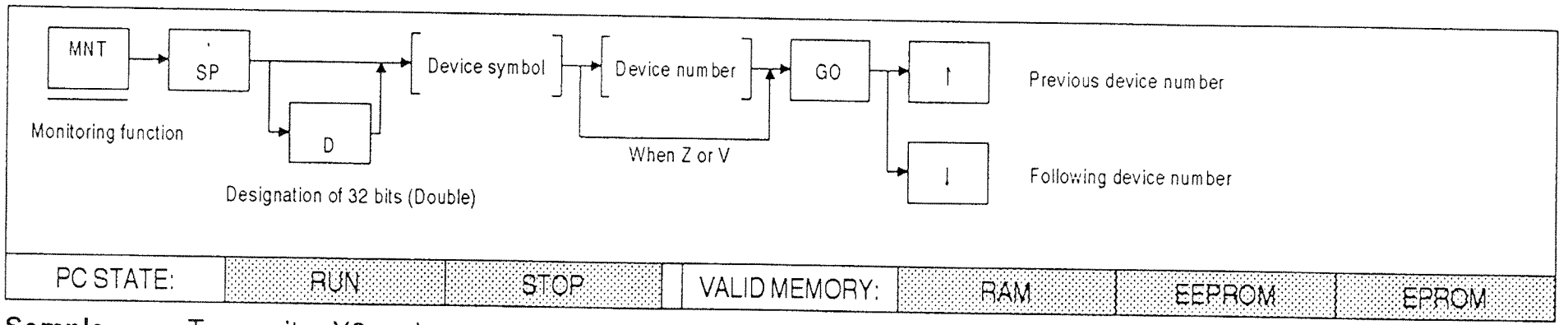
This section shows an outline of the monitoring procedure.



4. MONITORING

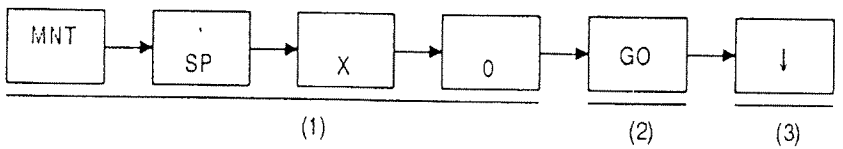
This section explains how to monitor the ON/OFF state and setting and current values for a designated device.

Basic operation

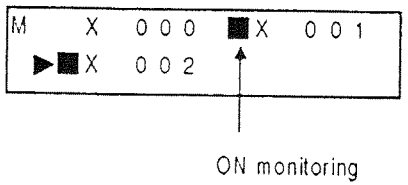


Sample operation 1

To monitor X0 and succeeding devices in order.
[Allowed devices: X, Y, M and S]



Sample display



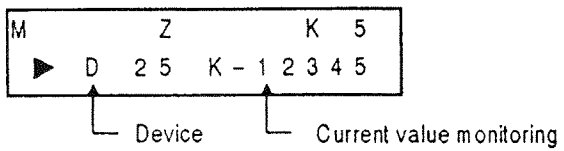
(1) After pressing the [MNT] key, press the [SP] key and enter the device symbol and device number.

(2) Press the [GO] key. The ON/OFF state of the designated device is displayed on the screen using the ■ symbol (the ■ symbol indicates the ON state.)

(3) Use the [↑] and [↓] keys to monitor the ON/OFF state of preceding and succeeding devices.
The screen can display the ON/OFF state for up to 4 devices. Different devices can be monitored on one screen.

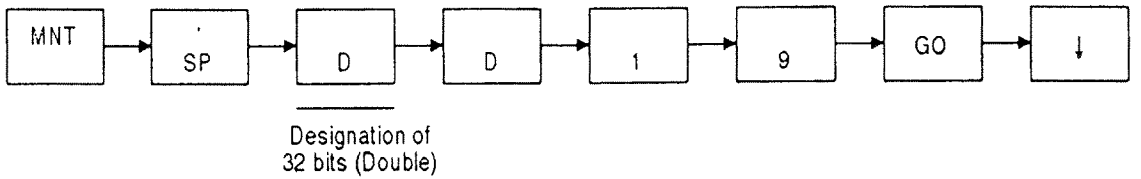
Sample operation 2 To monitor D25 and succeeding devices in order. [Allowed devices: D, Z, and V (16 bits)]

Sample display

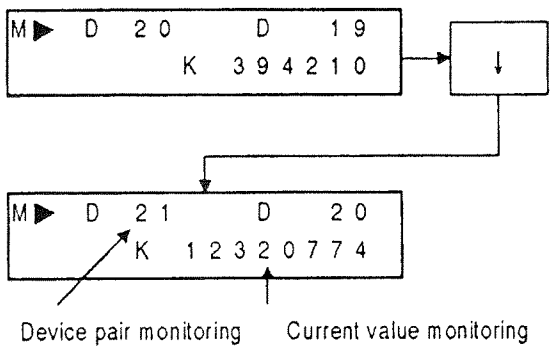


* The key operation is the same as for steps (1) to (3) of sample operation 1. The current value is monitored on the display screen.
A maximum of two devices can be monitored on one screen.

Sample operation 3 To monitor D19 and D20 and succeeding devices as a pair. [Allowed devices: D, Z (32 bits)]

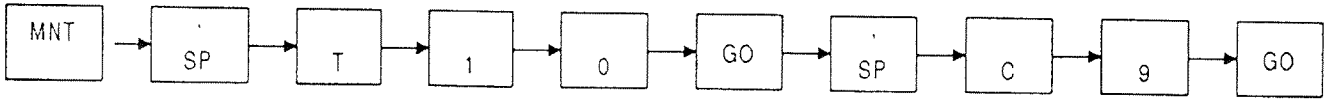


Sample display

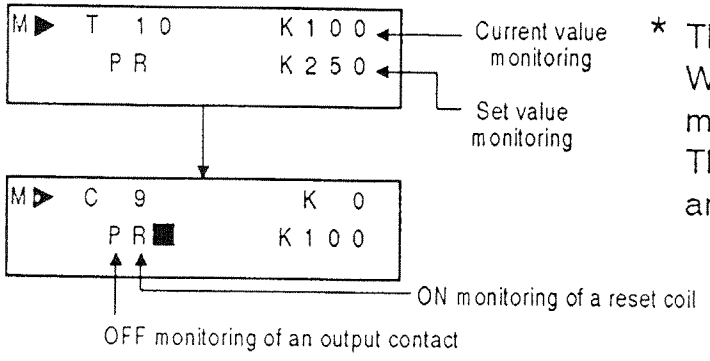


* Before inputting the device press the D key to designate 32 bits monitoring. The other key operations are the same as for (1) to (3) of sample operation 1.
* Because the monitored device is a 32-bit device, the designated device number and the following device number are monitored in pair. For example, D19 is monitored with D20 and D20 is monitored with D21.
* Note: To step on a full 32 bit register press a cursor key twice.

Sample operation 4 To monitor T10 and then C9. [Allowed Devices: T, C (16 bits)]

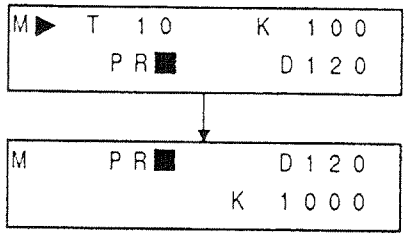


Sample display (Set value = K)



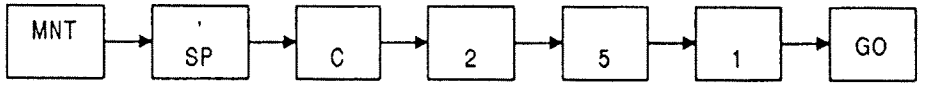
* The key operation is the same as for steps (1) to (3) of sample operation 1. When the devices are T (timer) or C (counter) the current and set values are monitored.
The ON/OFF state of the output contact and the reset coils are also monitored and indicated with the ■ symbol.

Sample display (Set value = D)

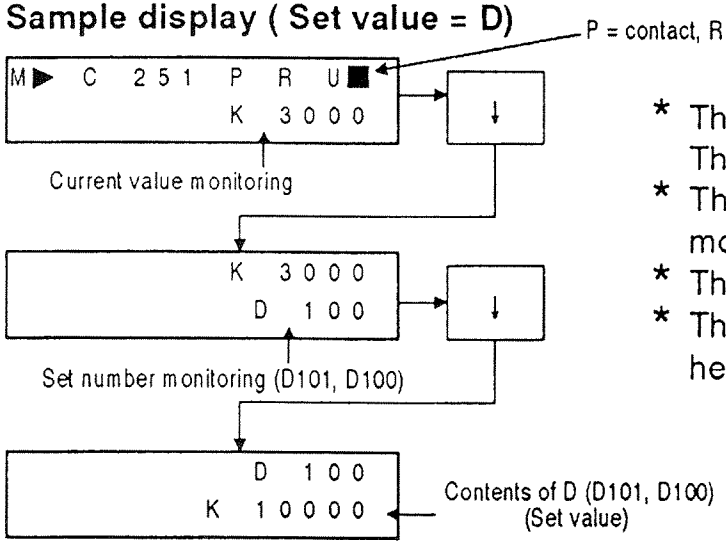


* If the set value of T or C is designated indirectly with a data register (D) the value of the data register can be monitored by using the [↓] key.

Sample operation 5 To monitor C251 (32-bit counter).



Sample display (Set value = D)



* The key operation is the same as for steps (1) to (3) of sample operation 1. The current value and the set value are monitored as 32 bits.
* The display is similar to that in sample operation 4. In addition, the up/down mode of the counter is also monitored (■ indicates up).
* The display screen is scrolled by using the [↑]/[↓] keys.
* The monitored current value can be toggled between decimal display and hexadecimal display pressing the [HELP] key.

POINT

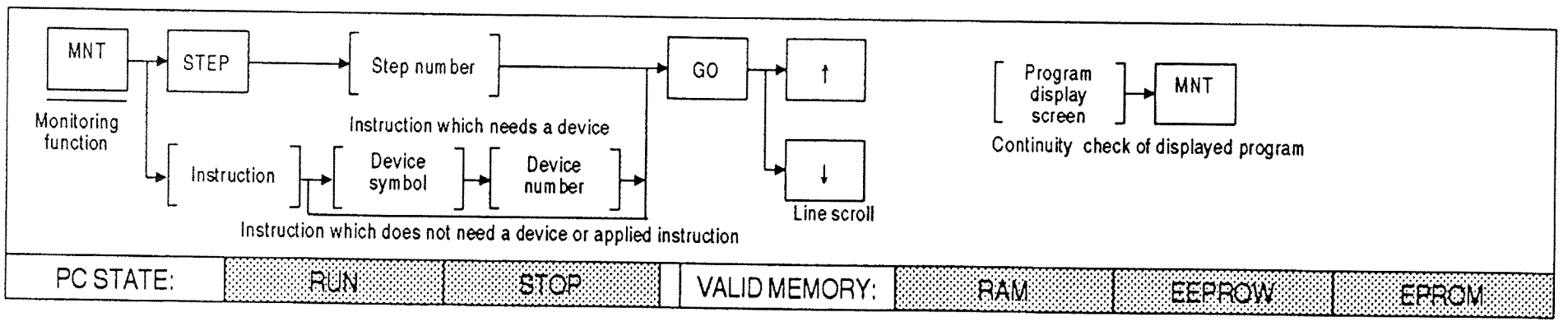
(1) Direct designation.
Numerical values designated by using constant K (decimal) are dealt with as setting values.
Only decimal values for K can be set.
The allowable ranges are as follows.
16 bit: –32768 to 32767 32 bit: –2147483648 to 2147483647

(2) Indirect designation.
The setting values is designated by a data register (D).
For example, if D10 is designated and contents of D10 are 123, it is the same as setting K123.
However, even if the contents of D are latched, if the battery voltage falls below the allowable limit, the data will be lost.

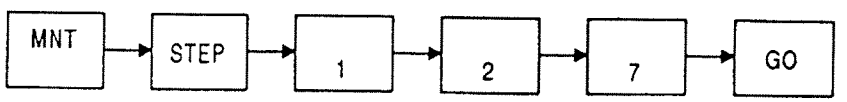
4. MONITORING

This section explains how to read a program by a step number or an instruction and monitor the contact continuity of a device and coil operation.

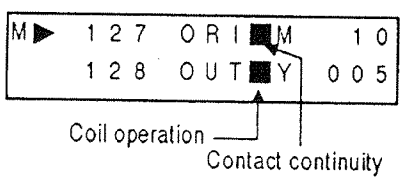
Basic operation



Sample operation 1 To read step 127 and to monitor a contact continuity and coil operation.



Sample display



* Press the [MNT] key.
The following operation is the same as the read operation.

Reading by a step:
Press the [STEP] key and enter the required step number, then press the [GO] key.

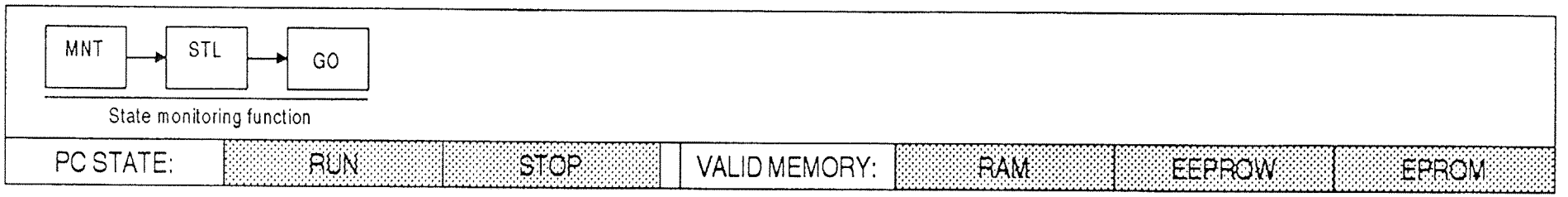
Reading by an instruction:
Press the instruction key to be designated.
With an instruction which needs a device entering, designate the necessary conditions before pressing the [GO] key.

* Two lines of instructions are displayed from the designated step number. The contact continuity and coil operation states are displayed and monitored on the left side of the device by using the ■ symbol.
* Line scrolling can be done in the monitoring state by using the [↑]/[↓] keys.
* If the program is displayed before monitoring, the contact continuity and coil operation are monitored for the current display state by pressing the [MNT] key.

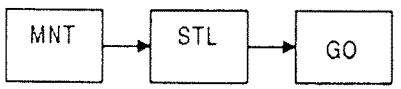
4. MONITORING

This section explains how to monitor up to a maximum of six points of active states beginning with the lowest active state number when using the step ladder instruction.

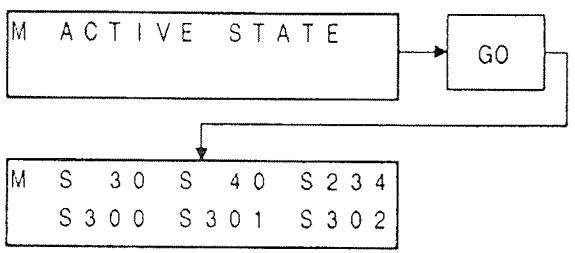
Basic operation



Sample operation 1 To monitor an active state.



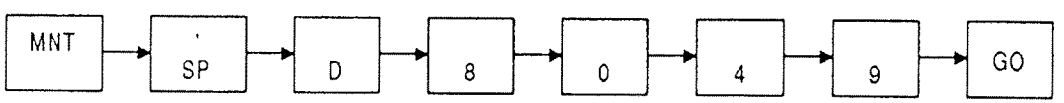
Sample display



* Press the [MNT] and [STL] keys and then press the [GO] key. A maximum of six currently operating sates are monitored, beginning with the lowest number, on the screen.
* When a state element changes, the displayed state numbers change accordingly. And so, the machine's operating processes can be monitored.
* The states in the range from S0 to S899 can be monitored. The state of S900 and after (for the annunciator) will be ignored.
* When seven points or more are turned ON, ">" will be displayed on the lower left side of the screen. The state of the seventh point onwards can be displayed by pressing the [↑]/[↓] keys.
* Active state monitoring is disabled unless special auxiliary relay M8047 is turned ON.

POINT

(1) When monitor M8094 of the annunciator is turned ON, the minimum active state number of S900 to S999 can be monitored by the monitoring operation given above.

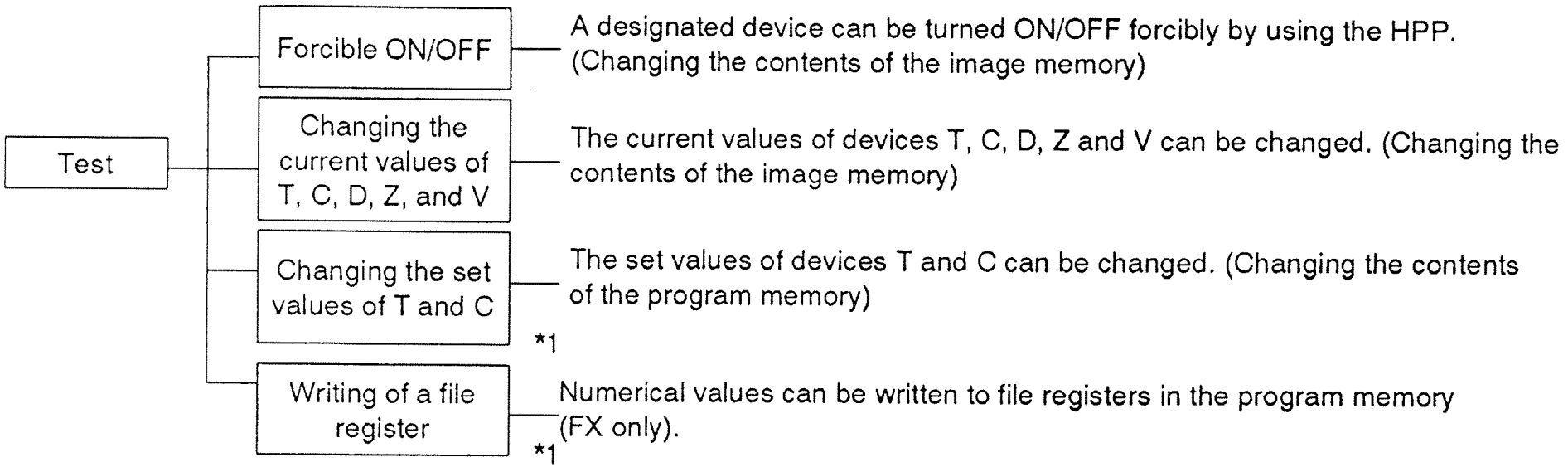


5. TEST

The test function is used to forcibly turn devices ON/OFF and to change the constant value of data devices.

General function description

Test consists of the following functions:



POINT

(1) *1 When the PC is in the RUN state this function is valid for RAM memory.

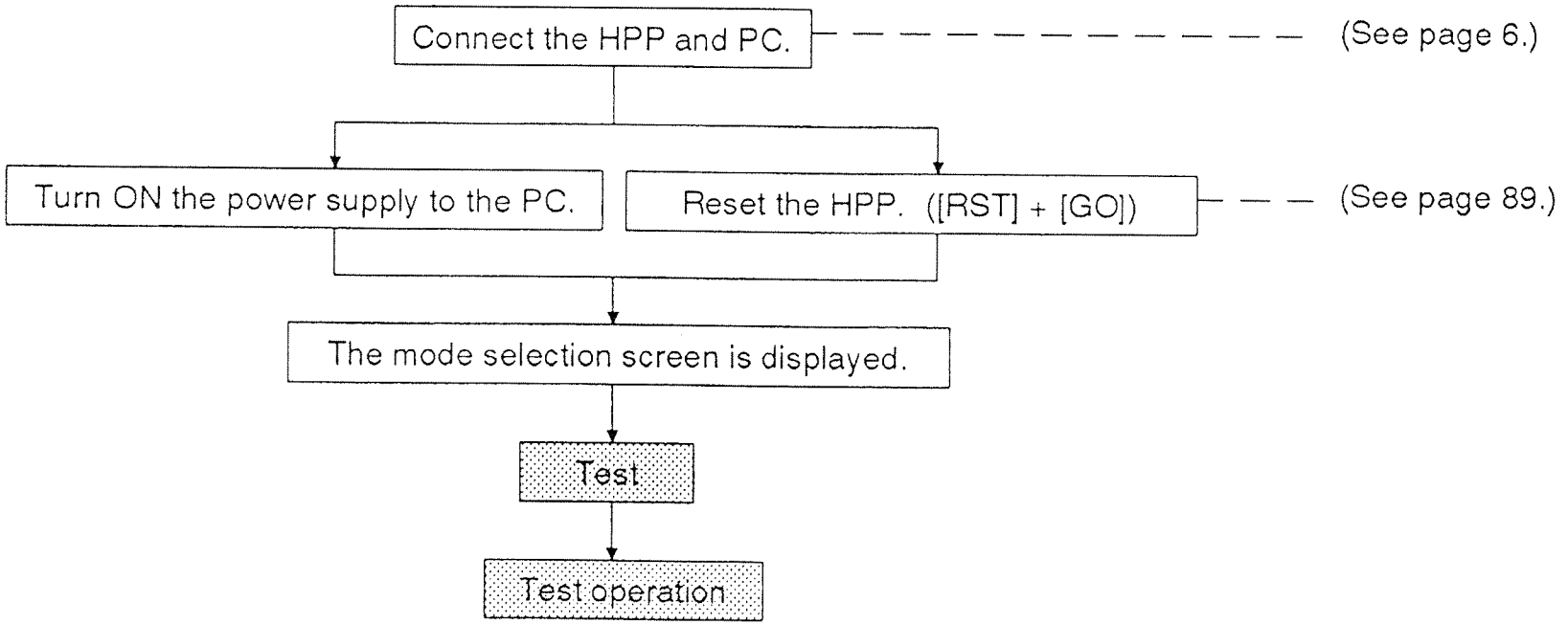
When the PC is in the STOP state this function is valid for RAM and EEPROM (with the memory protect switch is OFF). EPROM is invalid.

However, with FX_0, a change to the set values of T and C is valid when the PC is in the RUN state (program memory: EEPROM).

5. TEST

Outline procedure

This section shows an outline of the test procedure.



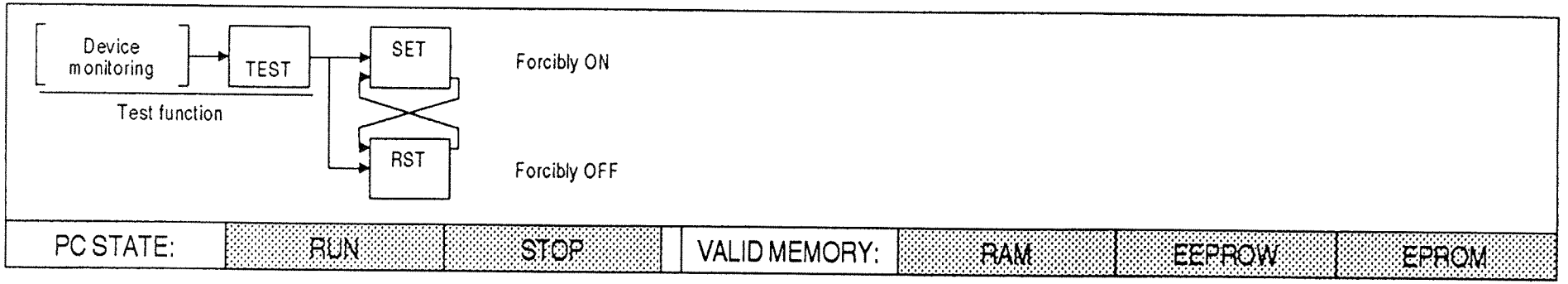
5. TEST

Forcible ON/OFF

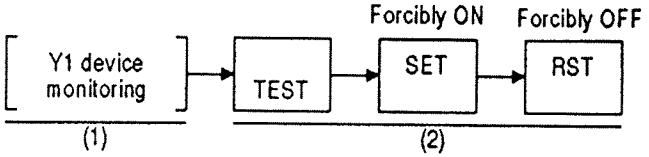
This section explains how to forcibly turn a device ON/OFF. After device monitoring, switch to the test function.

The forced ON/OFF operation is only valid for one operating cycle. Therefore, turning ON/OFF forcibly when the PC is in the RUN state will have an effect on timers, counters, set/reset circuits, latched circuits and auxiliary relays.

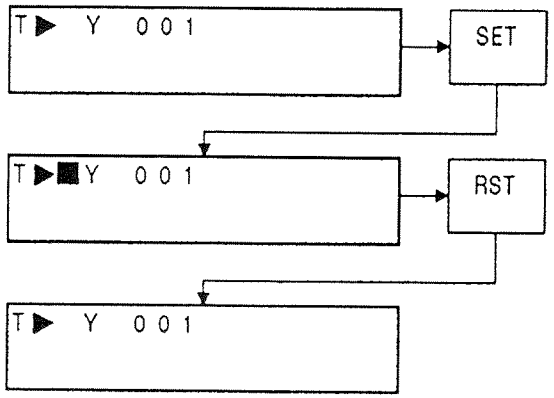
Basic operation



Sample operation 1 To turn device Y1 ON/OFF forcibly.



Sample display

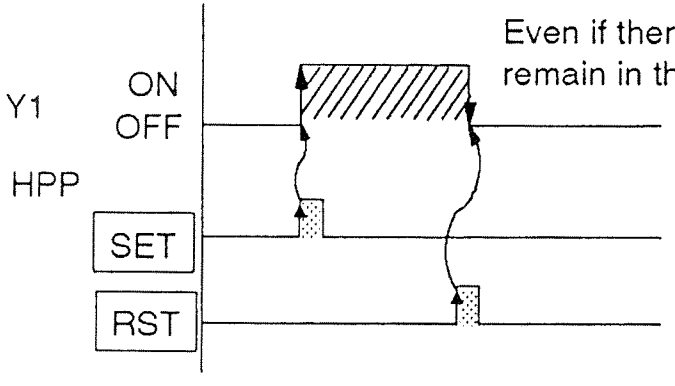


(1) Monitor the device using the monitoring function.

(2) Press the [TEST] key.
If the monitored device is in the OFF state press the [SET] key to turn the device ON forcibly.
If the monitored device is in the ON state press the [RST] key to turn the device OFF forcibly.

POINT

(1)Sample operation when the PC is in the STOP state.

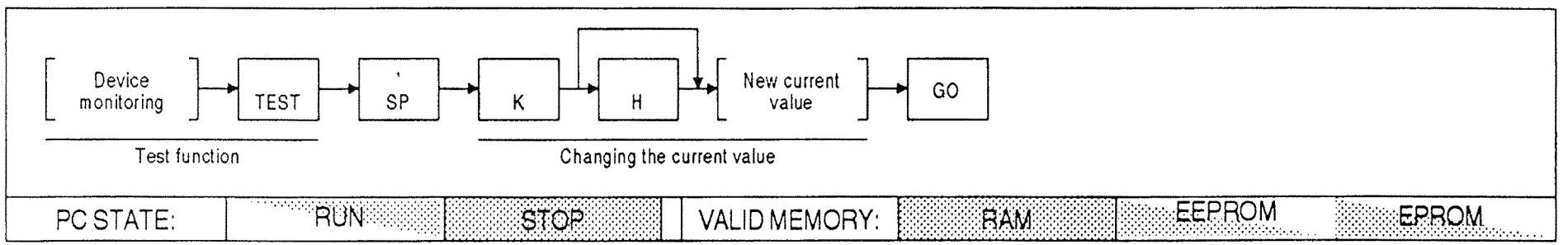


Even if there is not a latch circuit, when the PC is in the STOP state, Y1 will remain in the ON state until the [RST] key is pressed again.

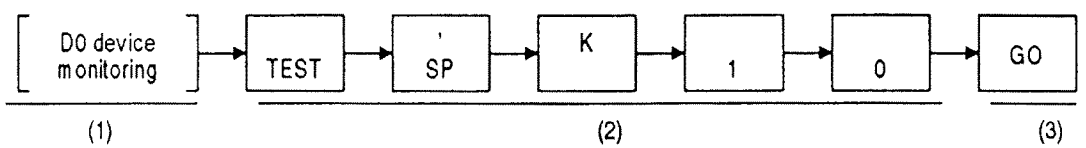
This section explains how to change the current values of devices T, C, D, Z, and V.

Switch to the test function after monitoring the device.

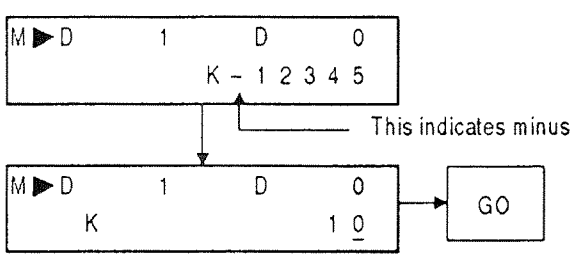
Basic operation



Sample operation 1 To change the current value the 32-bit of counter setting register (D1, D0) from K-12345 to K10.



Sample display



(1) Monitor the device using the monitoring function.

(2) Press the [TEST] and [SP] keys, then press the [K] key or [K] and [H] keys and input the new current value (Constant K is for decimal setting and H is for hexadecimal setting).

(3) Press the [GO] key to change the current value to the new value.

* The display of the current value can be switched between a hexadecimal display and a decimal display by pressing the [HELP] key.

POINT

(1) Writing to a file register (FX only).

Data writing to a file register is done in this mode.

File register changes are possible with RAM memory (PC in the RUN state), or with RAM and EEPROM memory (PC in the STOP state).

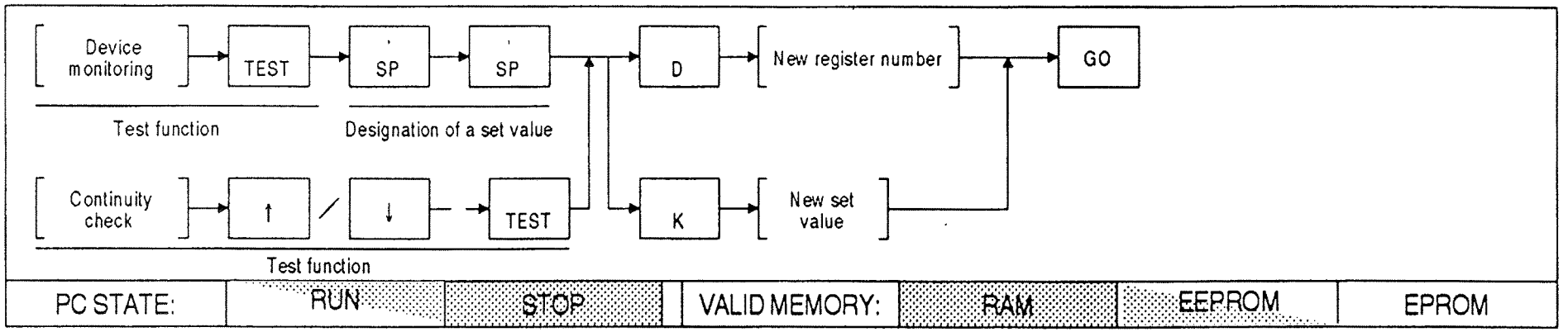
A maximum of 2000 file register points is allocated in the program memory (RAM, EEPROM, EPROM) in 500-point units by the parameter setting. (See page 85).

A file register can be utilized by designating data registers D1000 to 2999.

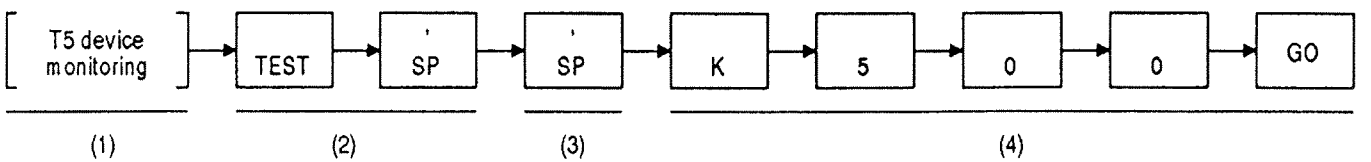
Independently of the PC state (RUN/STOP) and the program memory, the current value of T, C, D, Z and V (other than a file register) can be changed.

This section explains how to change the set value of devices T and C after device monitoring or a continuity check. Changing the set value is possible with RAM memory (PC in the RUN state), or with RAM and EEPROM memory (PC in the STOP state). With FX, it is possible when the PC is in the RUN state.

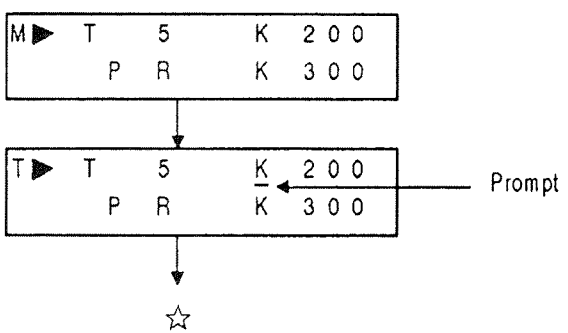
Basic operation



Sample operation 1 To change T5's set value from K300 to K500.

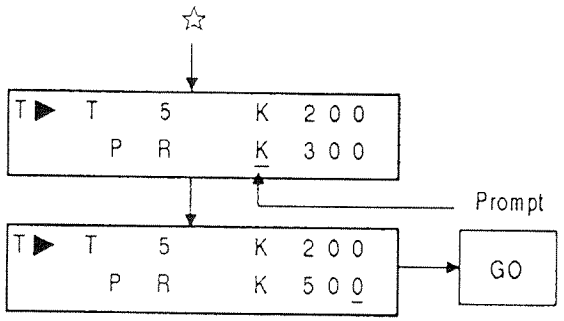


Sample display



(1) Monitor the device using the monitoring function.

(2) Press the [TEST] key and then press the [SP] key once.
The prompt appears at the position where the current value is displayed.



(3) The prompt can be moved to the position where the set value is displayed by pressing the [SP] key again.

(4) Input the new set value and then press the [GO] key. Changing the set value is now complete.

* The changed set value becomes valid for the OUT T or C instruction that appears first in the program.

If there are many out coils for the same device changing the set value for a specific OUT T or C instruction should be made by using the continuity check operation (See page 74).

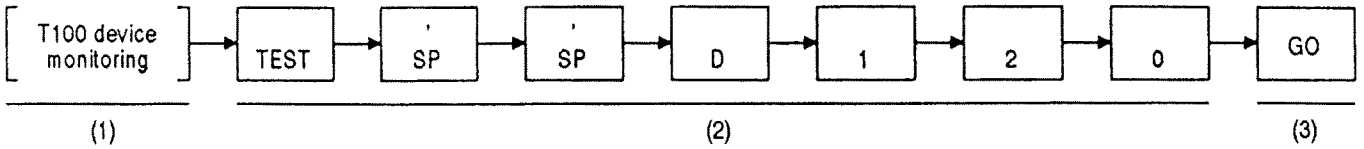
POINT

(1) Changing a constant when the PC is in the RUN state.

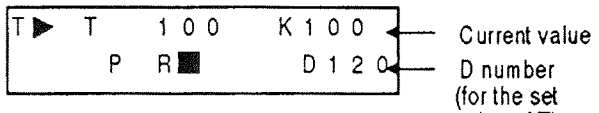
Even if the program memory is EEPROM, including FX_0, a constant (set value of a timer and a counter and the current value of a data register) can be changed when the PC is in the RUN state.

However, when the constant is changed the PC scan time increases 20 to 30 msec and the response to the input interrupts of I00[ ] to I30[ ] is delayed by 20 to 30 msec.

Sample operation 2 To change T100's set value from D123 to D120

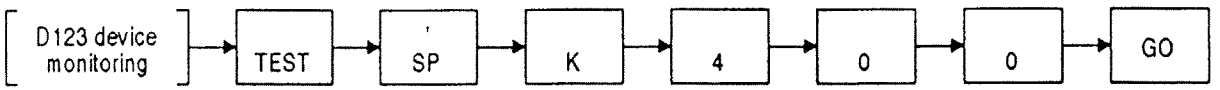


Sample display

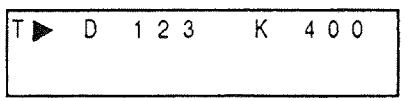


(1) Monitor the device using the monitoring function.
(2) The prompt can be moved to the position for the data register number where the set value is stored by pressing the [SP] key twice.
Input the new data register number.
(3) Press the [GO] key to change the setting value register to the new register number.

To change D123's contents from K300 to K400



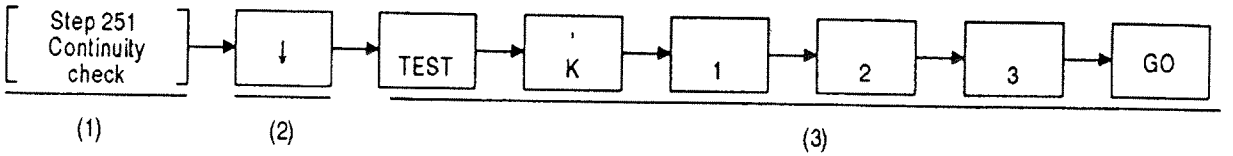
Sample display



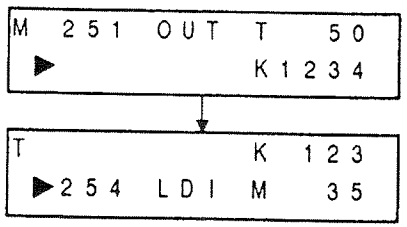
* When changing the contents of D follow the operating procedure for changing the current value of D (See page 69).

Changing the Set Values of T and C

Sample operation 3 To change the set value K1234 of the OUT T50 instruction at step 251 to K123



Sample display



(1) Execute the continuity check for required device using the monitoring function.
(2) Move the line cursor to the line with the set value.
(3) Press the [TEST] key and input a new set value. Then press the [GO] key to change present setting to the new set value.

* With a timer and a counter the change can be made with the monitoring function without needing to switch to the write function.

* The line cursor can be moved after pressing the [TEST] key to switching to the test function.

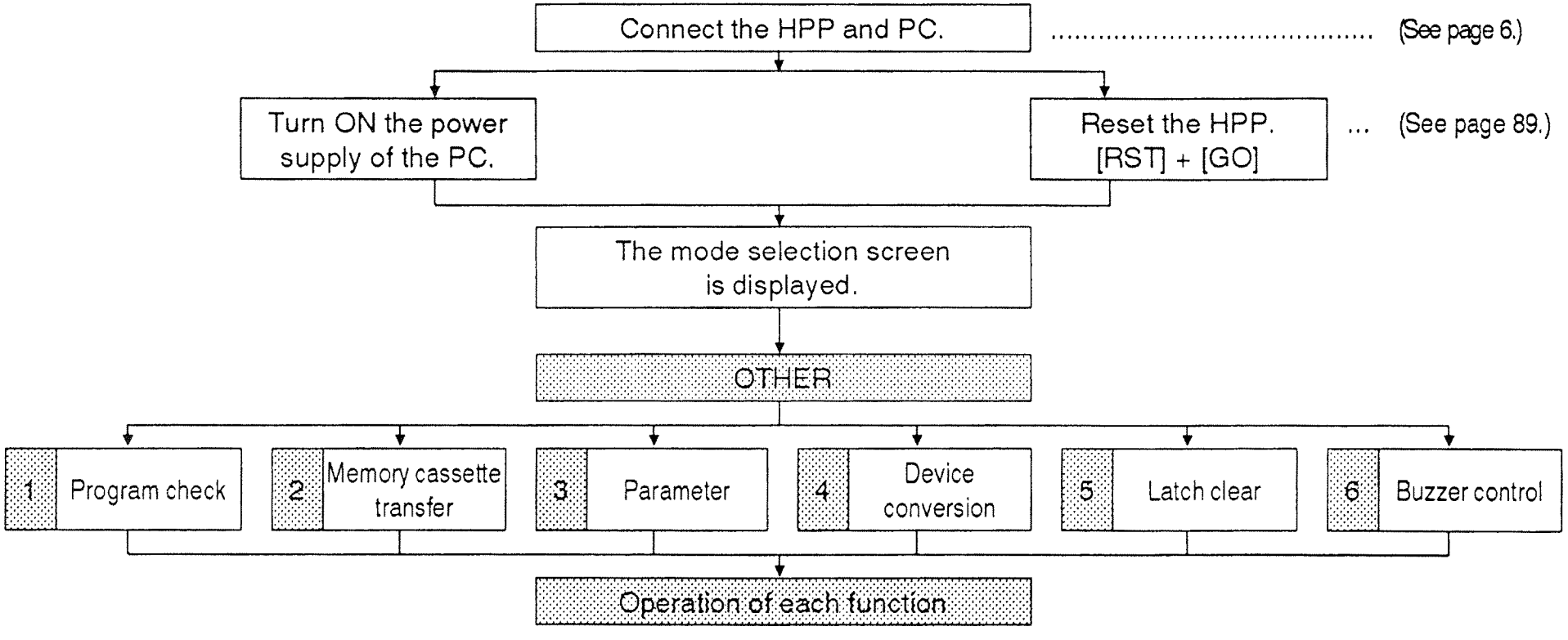
6. OTHER FUNCTIONS

The mode menu can be displayed by pressing the [OTHER] key during programming.

The mode menu consists of a program check, memory cassette transfer, parameter, device conversion, latch clear and buzzer control.

Outline procedure

The outline procedure for the mode menu is shown below.

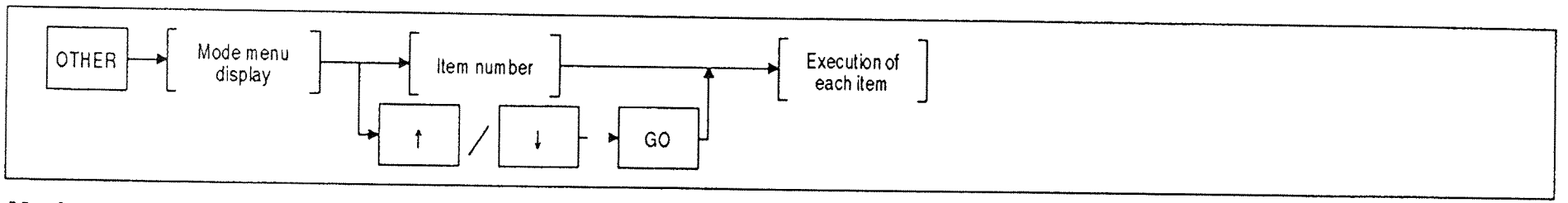


6. OTHER FUNCTIONS

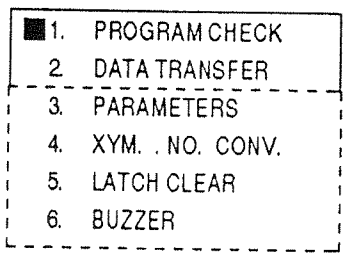
Selection of Each Menu Item

This section explains how to select each menu item from the mode menu.

Basic operation



Mode menu display



* The menu screen is displayed by pressing the [OTHER] key.
  Press the target item number or move the cursor to the target item by using the [↑]/[↓] keys and then press the [GO] key.
  Then, the selected menu item is displayed.

* The screen can be scrolled by using the cursor keys.

POINT

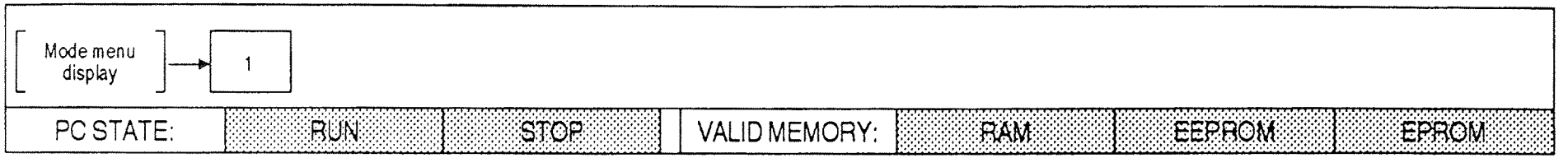
(1) Transferring between Function Screen and Mode Menu Screen
Press the [OTHER] key during a function screen operation and the mode menu screen will be displayed.
Conversely, pressing a function key from a screen accessed by the [OTHER] key permits the entry to that function operation.

6. OTHER FUNCTIONS

Program Check

This section explains how to check the PC's internal program.

Basic operation



Display

```
1. PROGRAM CHECK
   EXECUTING
```

* Display the menu screen and then press the [1] key.
A program check is executed and the check result is displayed.

<When there is not an error>

```
1. PROGRAM CHECK
   NO ERROR
```

* If there are no errors the screen shown on the left is displayed.

<When there is an error>

```
1. PROGRAM CHECK
   ΔΔΔΔ   00000000
   ERROR CODE □□□□
```

Step number display
Error message display
Error code display

* If there is an error the program step number with the error, the error message and error code are displayed. For each operation, only the first error is displayed. The error code is displayed by pressing the [↓] key.

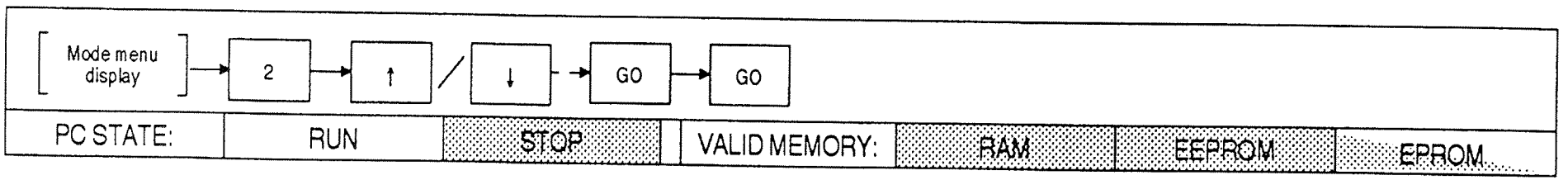
* Press the [CLEAR] key or the [OTHER] key to return to the mode menu.

* Clearing any errors found will automatically reset special auxiliary relay M8068 and special data register D8068. If other errors remain the next error code is stored.

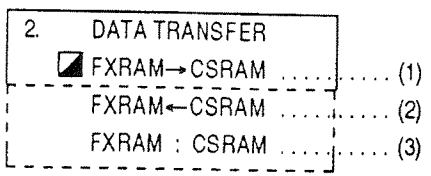
6. OTHER FUNCTIONS

This section explains how to transfer the program and parameters between the RAM built in the PC and a memory cassette installed in the PC.
Menu item displays for transfers between the RAM built in the PC and a memory cassette will differ depending on the type (RAM, EPROM, EEPROM) of memory cassette (invalid for FX0).

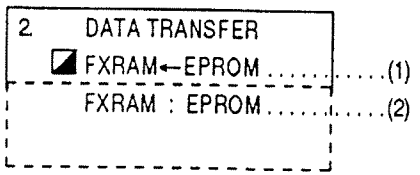
Basic operation



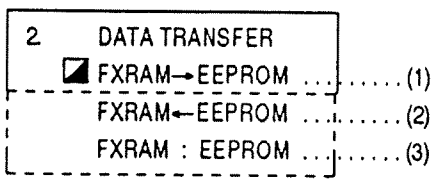
Memory cassette transfer screen



(1) Transfer from the RAM built in the PC to a CSRAM cassette.
(2) Transfer from a CSRAM cassette to the RAM built in the PC.
(3) Verification of the RAM built in the PC and a CSRAM cassette.



(1) Transfer from an EPROM cassette to the RAM built in the PC.
(2) Verification of the RAM built in the PC and a EPROM cassette.



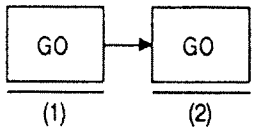
(1) Transfer from the RAM built in the PC to an EEPROM cassette.
(2) Transfer from an EEPROM cassette to the RAM built in the PC.
(3) Verification of the RAM built in the PC and an EEPROM cassette.

POINTS

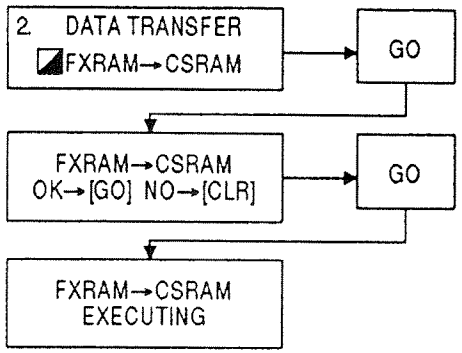
(1) CSRAM
CSRAM is the RAM cassette memory installed in the PC.
This name is used to distinguish it from the RAM (FXRAM) built in the PC.

(2) Precautions when transferring from the FXRAM to an EEPROM
When data is transferred from the FXRAM to an EEPROM cassette, set the EEPROM cassette memory protect switch to OFF.

Sample operation 1 To transfer a program and parameters from the RAM built in the PC to a CSRAM cassette.



Sample display



* Display the menu screen and then press the [2] key.
The memory cassette transfer screen is displayed.

(1) Use the cursor key to select the item to be transferred and then press the [GO] key.

(2) To transfer press the [GO] key.
To cancel the transfer press the [CLEAR] key. The screen returns to the memory cassette transfer screen.

* When the transfer has been completed, the "COMPLETED" message will be displayed.
If the transfer is not performed normally, an error message will be displayed.

POINTS

(1) Program memory capacity.
If a program whose memory capacity parameter is set to 4K or 8K steps is transferred from the memory cassette to the RAM built in the PC, the "PC PARA. ERROR" error message will be displayed and the program will not be transferred.
Set the memory capacity to 2K steps by parameter setting (See page 81).

(2) When a parameter is mismatched.
When verification is selected, if the execution result contains a mismatch, the "VERIFY ERROR" message will be displayed, as well as the mismatched location.

Effective Usage of memory cassettes

<Considerations when using EEPROM>

If an EEPROM cassette is installed in the FX and programming is done, writing can be done to the EEPROM.

Since the writing time to an EEPROM is longer than that to the RAM, it takes much longer to insert and delete programs.

More importantly, you can only write to the EEPROM 10,000 times.

Therefore, if the program is less than 2K steps, it is better to transfer it to an EEPROM cassette after writing it using the RAM built in the PC.

<Masters and copies>

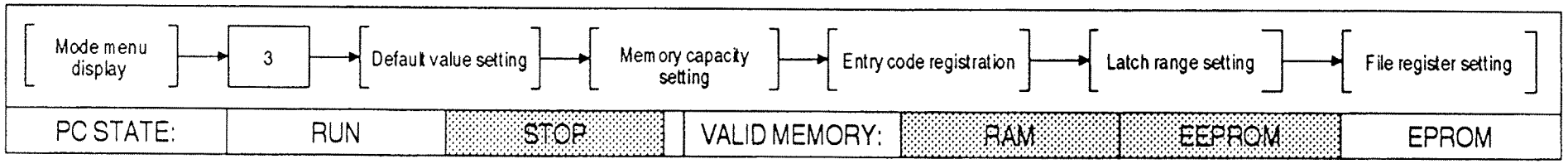
If a program which was transferred to the EEPROM cassette is kept as a master, that program can be (a) transferred to a PC and used as is, or (b) some parts of the program can be changed after by transferring the program from this master to another PC.

6. OTHER FUNCTIONS

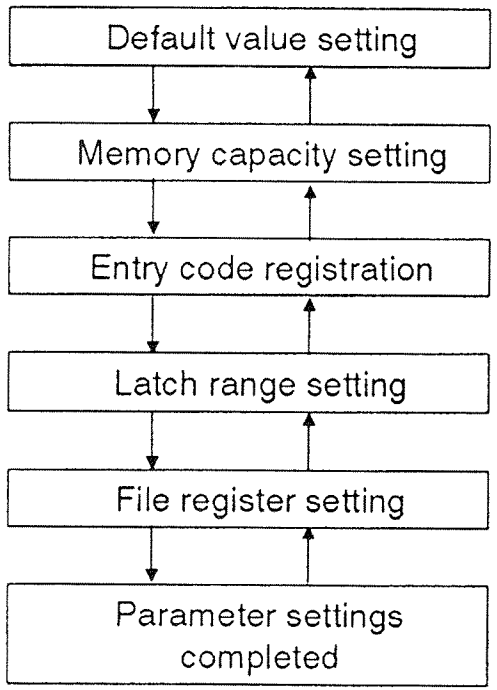
Parameters

Parameters are the values that specify the various functions and ranges of a device.
Parameters are stored independently in the parameter area of the PC's memory.
Setting of default values, memory capacity, entry code registration, latch ranges and file registration of PC program memory are all done by parameter setting.

Basic operation



Operation procedure



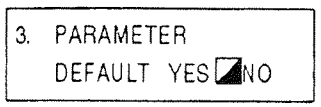
* Select an item and then press the [GO] key to move to the next setting item. When not changing the current set value, press the [GO] key.
* Press the [CLEAR] key to return to the previous setting.
* Press the [OTHER] key to return to the mode menu during parameter setting.

POINT

(1) Changing parameter settings
Parameter settings can be changed during programming.
Programming can begin when the parameters are the default values and then altered later.

Default value setting

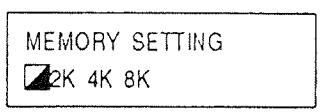
Display



* Display the menu screen and then press the [3] key.
  The parameter setting screen is displayed.
  When setting the default values (initial values), select "YES" by using the cursor key and then press the [GO] key (The default option is "NO").
  When not setting the default values, press the [GO] key.
  Page 109 gives details about parameter default values.

Memory capacity setting

Display



* The memory capacity default value is set to 2K steps.
  When changing the memory capacity, select the steps to be changed by using the cursor key and then press the [GO] key.

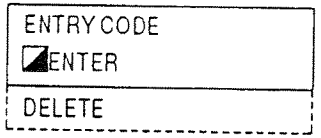
* Types of PCs and memory capacities that can be set

PC \ Memory capacities	2K steps	4K steps	8K steps
FX_0	O	X	X
FX	O	O	O

o : Valid
x : Not valid

Entry code registration

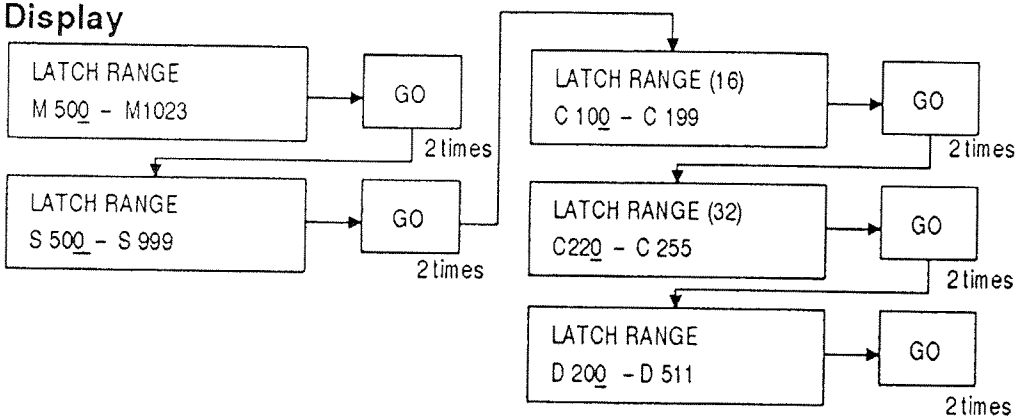
Display



* When registering an entry code, select "ENTER" by using the cursor key, input the entry code and then press the [GO] key (The default is set to "Not Registered").
* When not changing the entry code press the [GO] key.
* When deleting an entry code, select "DELETE" with the cursor key, input that entry code and then press the [GO] key.
* Page 90 gives details about entry code registration.

Latch range setting

Display



* When changing the latch range, input the device number and then press the [GO] key.
  Change the settings in order; the start device then the end device. When not changing a latch range, press the [GO] key twice.
* The latch range end device can be set to a value that is below the default value (See page 109).
* When using FX_0 select the default values.

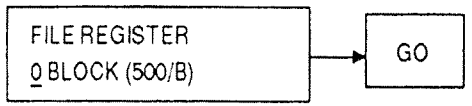
POINT

(1) Latch

Latch is the function which maintains the state of a device when the power supply is turned OFF and it keeps the device from being cleared when the power is turned back ON.
Therefore, the latch function will store the state just before a power failure and restart operations from that state when the power goes ON again.

File register setting (FX)

Display

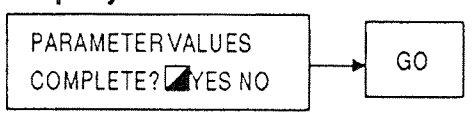


* When setting a file register, input the number of blocks (0 to 4 blocks) of memory to be allocated and then press the [GO] key.
  The maximum range that can be set is four blocks.

* Since the FX_0 does not have the file register function, set 0 blocks.
  Pages 105, 108, and 109 give details.

Parameter setting completed

Display

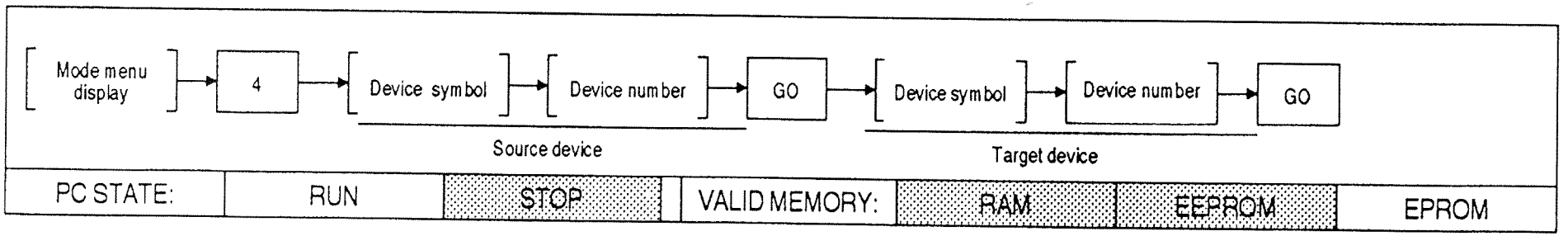


* To end parameter setting press the [GO] key.
  The mode menu will be displayed.
  The default option is "YES".
  To continue parameter setting, select "NO" with the cursor key and then press the [GO] key.
  The screen returns to the initial parameter setting screen.

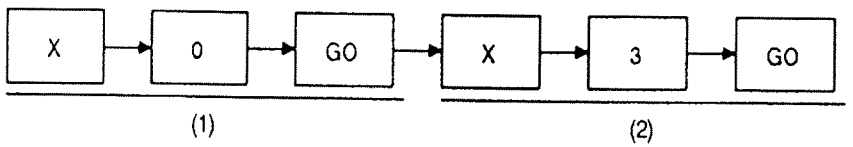
6. OTHER FUNCTIONS

This section explains how to change a device number to a new device number in the entire program memory (even instructions written after the end instruction will be converted).

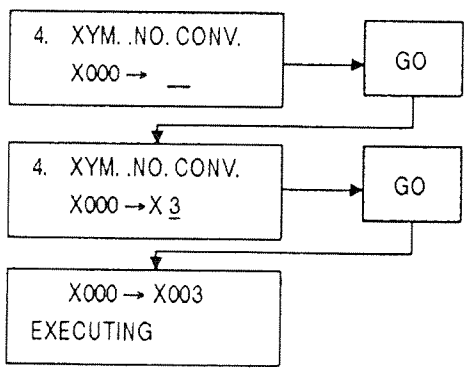
Basic operation



Sample operation 1 To convert device X0 to device X3



Sample display



* Display the menu screen, and then press the [4] key. The device conversion screen is displayed.

(1) Input the source device.
Press the [X] and [0] keys and then press the [GO] key.

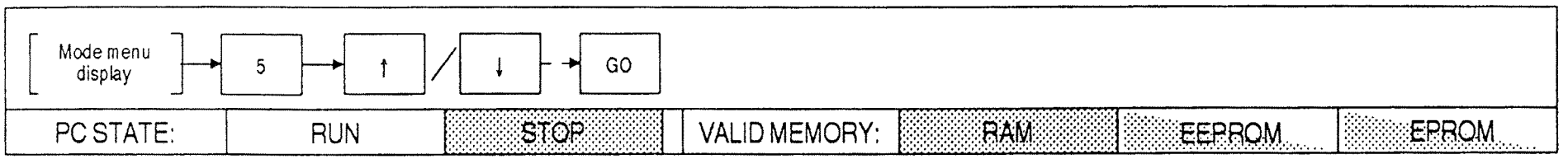
(2) Input the target device.
Press the [X] and [3] keys and then press the [GO] key.

* When canceling the target or source device press the [CLEAR] key. However, do so before confirming the target device (before pressing the [GO] key).

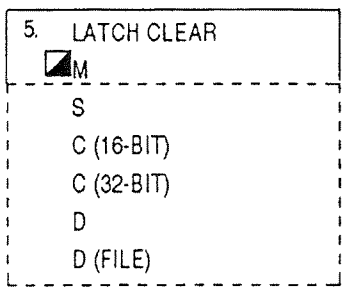
6. OTHER FUNCTIONS

This section explains how to clear the device latch state.

Basic operation



Display



* Display the menu screen and then press the [5] key.
  The latch clear screen is displayed. The devices that can be latched are displayed on the screen.
* Line scrolling can be done with the cursor keys.
* Select the device for which the latched state is to be cleared by using the cursor key and then press the [GO] key. Clearing has now been executed.
  The latch clear operation needs to be done for each device.

* The latch state of all devices except for the file register can be cleared even if the program memory is in RAM, EEPROM, or EPROM.
* If the program memory is in EPROM the file register cannot be cleared.
  If the program memory is in EEPROM set the memory protect switch to OFF to enable clearing of the file register.

POINT

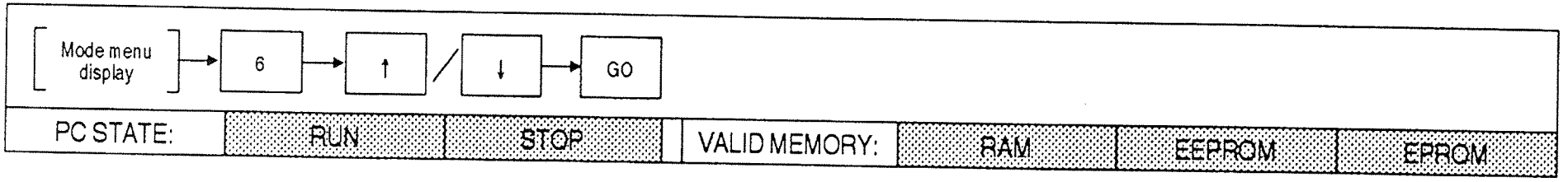
(1) File registers and EPROM
If the program memory is in EPROM, the file registers cannot be cleared because file registers are allocated to normal program memory.

6. OTHER FUNCTIONS

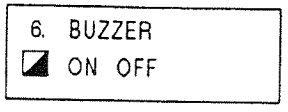
Buzzer Control

This section explains how to control the HPP buzzer.

Basic operations



Display



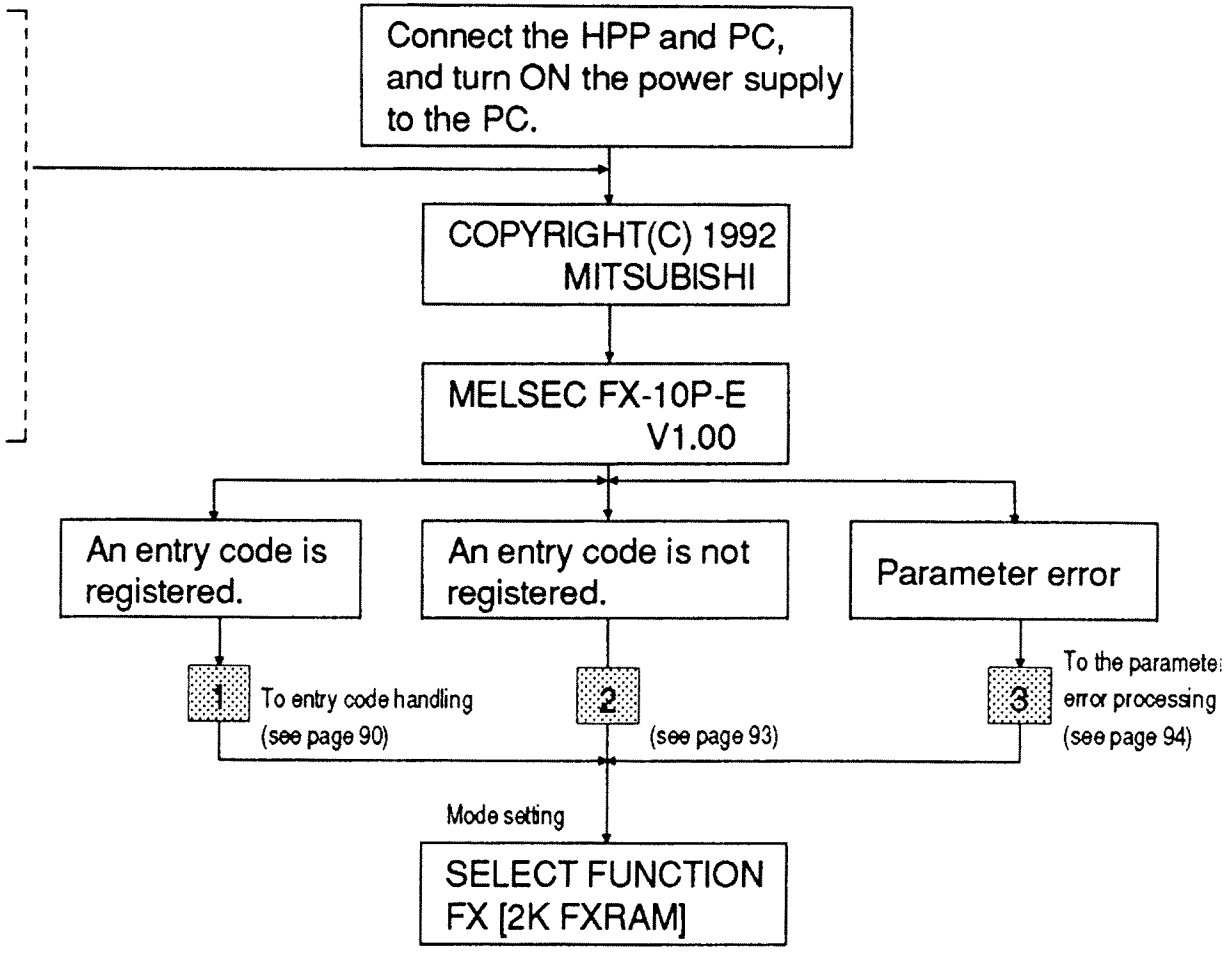
* Display the menu screen and then press the [6] key.
  The buzzer control screen is displayed.
* Select ON or OFF using the cursor keys and then press the [GO] key.
  The display returns to the menu screen and the buzzer status is adjusted.

7. DETAILED PROCEDURE FOR STARTING UP THE SYSTEM

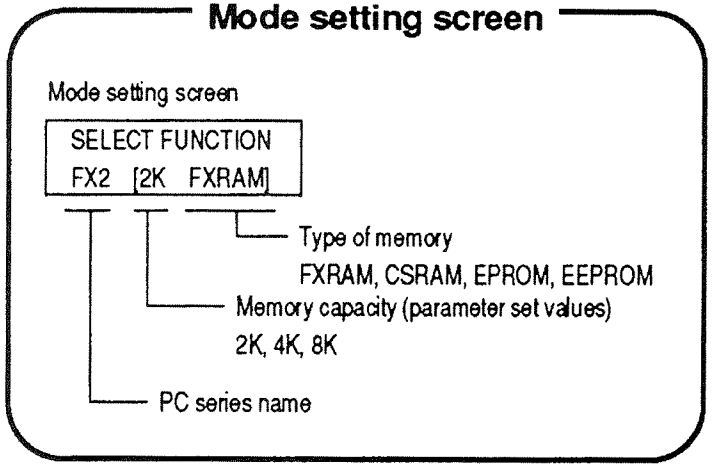
The procedure for starting up the system is shown below.

The PC will be put into the STOP state when the power to the PC is turned on if the [RD/WR] key is held down, even if the PC RUN input is set to the ON position. The PC will be restored to the RUN state when the power is turned ON again.

The HPP is reset by pressing the [RST] + [GO] keys and the screen is switched to the initial power ON screen.



Mode setting screen



7. DETAILED PROCEDURE FOR STARTING UP THE SYSTEM Entry Code Registration

This section explains how to register entry codes.
Input the correct entry code enables all HPP operations. If the input entry code does not match, the functions of the HPP are limited according to the following table.
If the entry code for a PC is unknown operations can be stored by keying in the special entry code (8 [SP] keys). In this case, however, all of the instructions in the PC will be cleared (see page 92).
The entry code is composed of eight hexadecimal digits. The protection level of the program is set by the first of these eight digits.
The program protection level consists of (A) the all-operations prohibition level, (B) the theft prevention level and (C) the miswriting prevention level.

[Valid HPP functions for protection levels]

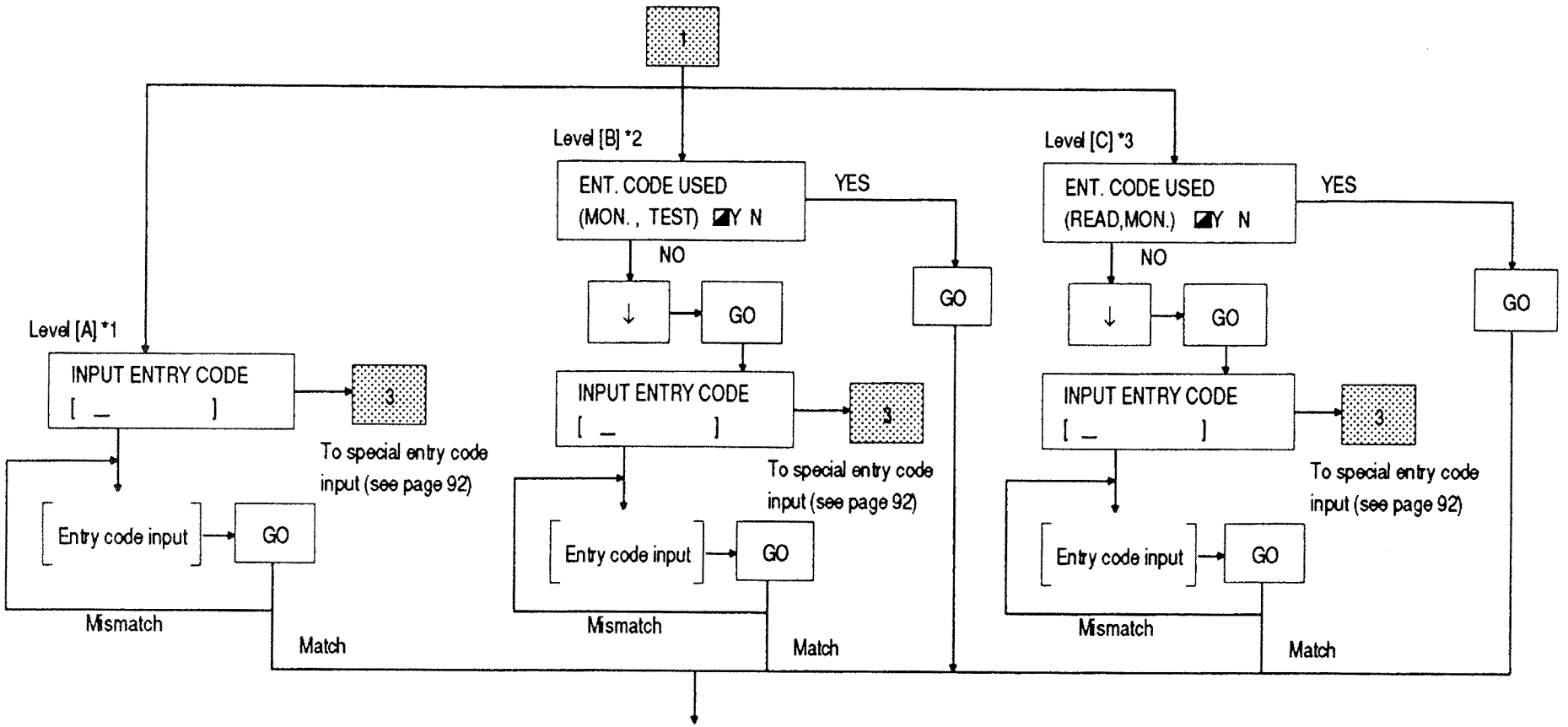
Function \ Protection Levels		All-Operations Prohibition	Theft Prevention	Miswriting Prevention
Program	Read	X	X	O
	Write	X	X	X
	Insert	X	X	X
	Delete	X	X	X
Monitoring	Device monitoring	X	O	O
	Continuity check	X	X	O
	Active state monitoring	X	O	O
Test	Forcible ON/OFF	X	O	O
	Current value change	X	O	O
	Set value change	X	X	X
Mode menu	Parameter	X	X	X
	Program check	X	X	O
	Device conversion	X	X	X
	Transfer	X	X	X
	Latch clear	X	O	O

All-operations prohibition* - - - - - - - - A [ ][ ][ ][ ][ ][ ][ ]
Theft prevention - - - - - - - - - - - - - - B [ ][ ][ ][ ][ ][ ][ ]
Miswriting prevention - - - - - - - - - - - C [ ][ ][ ][ ][ ][ ][ ]

☆ If the most significant digit is not A, B, or C, the all-operations prohibition level is set.

O : Available
X : Unavailable

This section explains how to register an entry code.



To the start-up procedure (See page 89.)

*1 All-operations prohibition level [A]

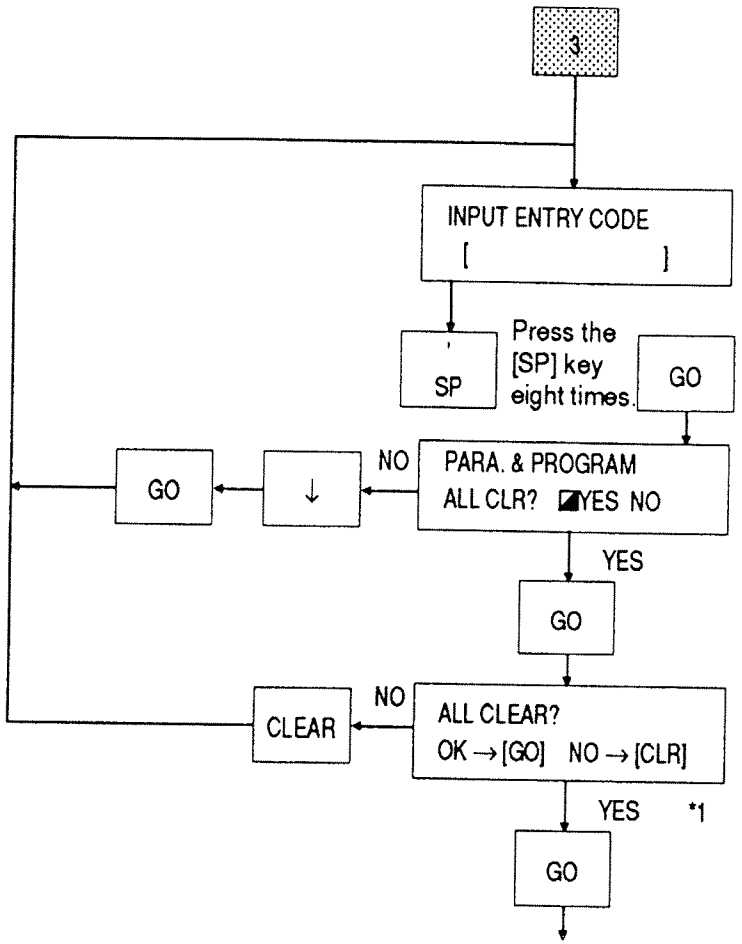
*2 Theft prevention level [B]

*3 Miswriting prevention level [C]

[Special entry code input]

If the entry code is unknown operations can be started by inputting the special entry code.

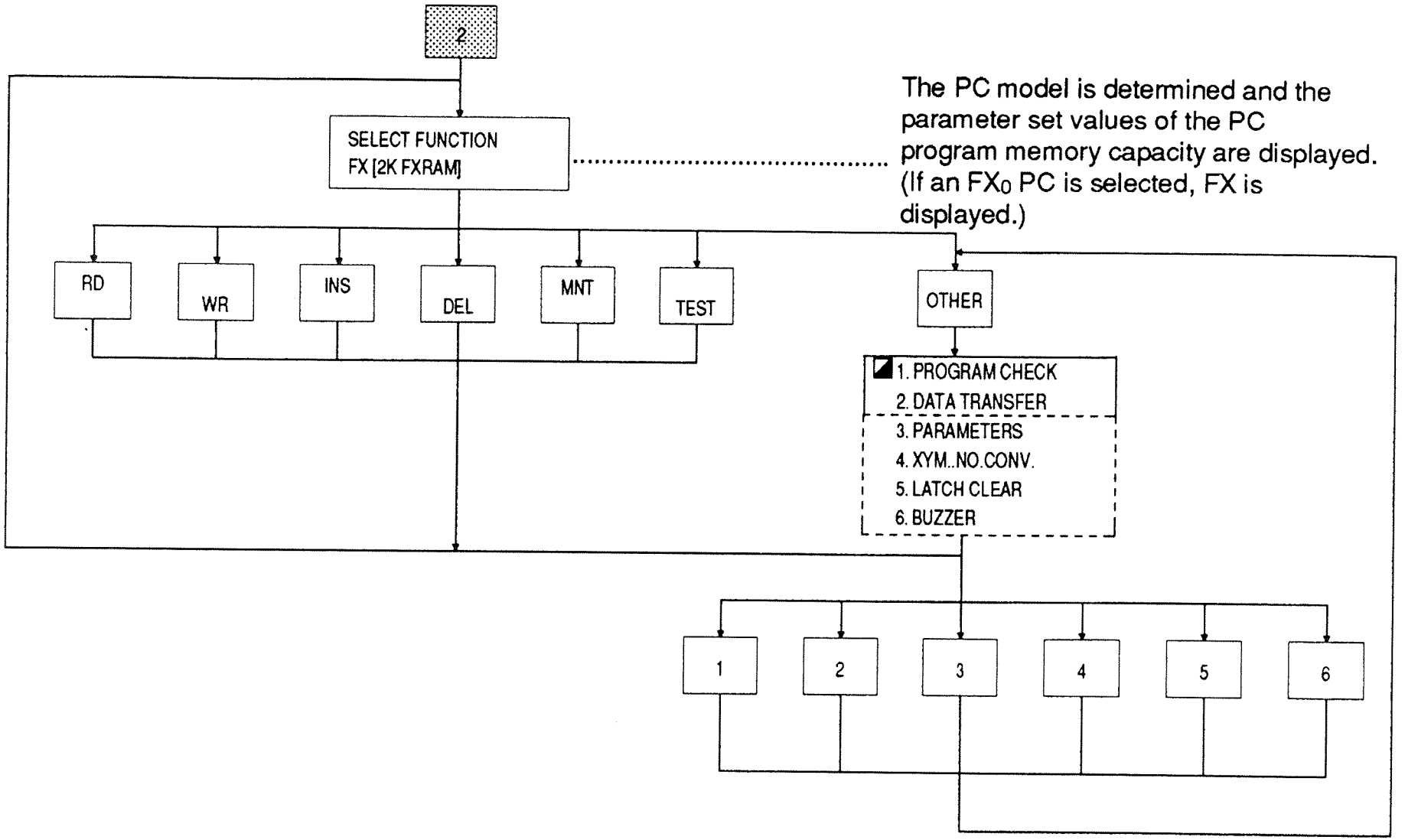
The procedure for inputting the special entry code is shown below.



*1 If "YES" is selected, the program and all parameters (which include the entry code) are cleared.

To the start-up procedure (See page 89).

This section explains how to start up all functions.

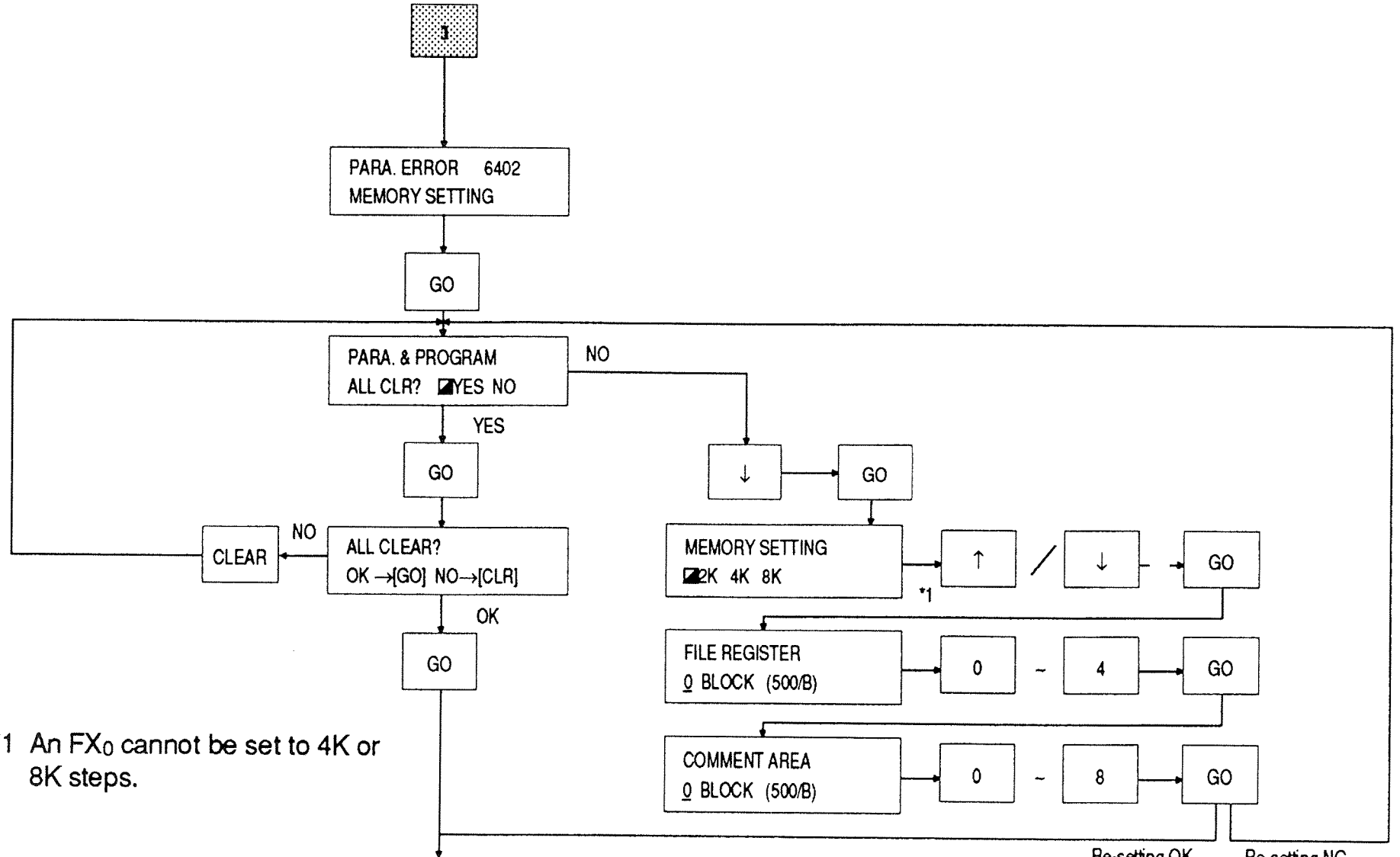


7. DETAILED PROCEDURE FOR STARTING UP THE SYSTEM

Parameter Error

The section explains how to take corrective action if a PC parameter error occurs when the system is started up.

If a PC is used which has been left for a long time with the battery removed, a parameter error occurs.



*1 An FX_0 cannot be set to 4K or 8K steps.

To the start-up procedure (See page 89).

POINTS

(1) File register setting
From 0 to 2000 steps of program memory can be used for file registers by setting 0 to 4 block (One file register point corresponds to one step).
When an FX_0 is used no file registers can be set.

(2) Comment area setting
From 0 to 4000 steps of program memory can be used for the comment area by setting 0 to 8 blocks (The fifteen characters of one comment are equivalent to ten steps).
When an FX_0 is used no comment area can be set (See page 110).

Basic instructions and step ladder instructions

Group	Instructions	Devices	Number of Steps
Contact instructions	LD	X, Y, M, S, T, C, Special M	1
	LDI	X, Y, M, S, T, C, Special M	1
	AND	X, Y, M, S, T, C, Special M	1
	ANI	X, Y, M, S, T, C, Special M	1
	OR	X, Y, M, S, T, C, Special M	1
	ORI	X, Y, M, S, T, C, Special M	1
Joint instructions	ANB	None	1
	ORB	None	1
	MPS	None	1
	MRD	None	1
	MPP	None	1
Other instructions	MC	N, Y, M	3
	MCR	N (nesting)	2
	NOP	None	1
	END	None	1
Step ladder	STL	S	1
	RET	None	1

Group	Instructions	Devices	Number of Steps
Output instructions	OUT	Y, M	1
		S	2
		Special M	2
		T-K, D	3
		C-K, D (16-BIT)	3
		C-K, D (32-BIT)	5
	SET	Y, M	1
		S	2
		Special M	2
	RST	Y, M	1
		S	2
		Special M	2
		T, C	2
		D, V, Z, Special D	3
	PLS	Y, M	2
	PLF	Y, M	2

Group	Instructions	Pointer Number	Number of Steps
Label	P	0 to 63	1
	I	0[ ][ ] to 8[ ][ ]	1

Applied instructions

(The O symbol indicates an available instruction.)

The FX_0 series PCs do not have pulse execution-type application instructions.

Group	FNC Numbers	Instruction Symbols	Instruction Name	FX_0	FX
Program flow	00	CJ	Conditional jump	O	O
	01	CALL	Subroutine call		O
	02	SRET	Subroutine return		O
	03	IRET	Interruption return	O	O
	04	EI	Interruption permission	O	O
	05	DI	Interrupt disable	O	O
	06	FEND	Main program end	O	O
	07	WDT	Watchdog timer	O	O
	08	FOR	Repeat range start	O	O
	09	NEXT	Repeat range completion	O	O
Transfer comparison	10	CMP	Comparison	O	O
	11	ZCP	Zone comparison	O	O
	12	MOV	Move (S)→(D)	O	O
	13	SMOV	Digit move		O
	14	CML	Inverse move $(\overline{S})$→(D)		O
	15	BMOV	Batch move		O
	16	FMOV	Multiple move		O
	17	XCH	Exchange and move (D)↔(D)		O
	18	BCD	BIN-BCD conversion	O	O
	19	BIN	BCD-BIN conversion	O	O

Group	FNC Numbers	Instruction Symbols	Instruction Name	FX_0	FX
Algebraic and logical operations	20	ADD	Addition (binary) (S1)+(S2)→(D)	O	O
	21	SUB	Subtraction (binary) (S1)−(S2)→(D)	O	O
	22	MUL	Multiplication (binary) (S1)x(S2)→(D)(D)	O	O
	23	DIV	Division (binary) (S1)÷(S2)→(D)...(D)	O	O
	24	INC	Increment (binary) (D)+1→(D)	O	O
	25	DEC	Decrement (binary) (D)−1→(D)	O	O
	26	WAND	Logical AND (S1)∧(S2)→(D)	O	O
	27	WOR	Logical OR (S1)∨(S2)→(D)	O	O
	28	WXOR	Exclusive Logical OR (S1)⊻(S2)→(D)	O	O
	29	NEG	Negation $(\overline{D})$+1→(D)		O
Rotation and shift	30	ROR	Rotation (right)		O
	31	ROL	Rotation (left)		O
	32	RCR	Rotation (right) with CY		O
	33	RCL	Rotation (left) with CY		O
	34	SFTR	Right shift bit	O	O
	35	SFTL	Left shift bit	O	O
	36	WSFR	Right shift word		O
	37	WSFL	Left shift word		O
	38	SFWR	Shift register write		O
	39	SFRD	Shift register read		O

* Number of steps of an applied instruction

Instruction.... One step is composed of the sum of [FNC] and the succeeding instruction number.

Operand ..... Sum of the instruction number and the succeeding devices ——┬—16-bit instruction : Two steps per operand
└ 32-bit instruction : Four steps per operand

Applied instruction

(The O symbol indicates an available instruction).

Group	FNC Numbers	Instruction Symbols	Instruction Name	FX_0	FX
Data processing	40	ZRST	Batch reset	O	O
	41	DECO	Decode	O	O
	42	ENCO	Encode	O	O
	43	SUM	Bit check (number of "1" bits)		O
	44	BON	Bit ON/OFF judgment		O
	45	MEAN	Mean		O
	46	ANS	Annunciator set		O
	47	ANR	Annunciator reset		O
	48				
	49				
High-speed processing	50	REF	I/O refresh	O	O
	51	REFF	Filter adjustment		O
	52	MTR	Matrix input		O
	53	HSCS	Comparison set (high-speed counter)	O	O
	54	HSCR	Comparison reset (high-speed counter)	O	O
	55	HSZ	Zone comparison (high-speed counter)		O
	56	SPD	Pulse density		O
	57	PLSY	Pulse output	O	O
	58	PWM	Pulse-width modulation	O	O
	59				
Handy instructions	60	IST	Initial state (step ladder)	O	O
	61				
	62	ABSD	Drum sequence (absolute value)		O
	63	INCD	Drum sequence (incremental value)		O
	64	TIMR	Teaching timer		O
	65	STMR	OFF delay, one shot, and flicker timer		O
	66	ALT	Alternation output	O	O
	67	RAMP	Ramp signal	O	O
	68	ROTC	Rotary table control		O
	69				

Group	FNC Numbers	Instruction Symbols	Instruction Name	FX_0	FX
External device I/O	70	TKY	10-key pad input		O
	71	HKY	16-key pad input		O
	72	DSW	Digital switch time-sharing interruption		O
	73	SEGD	7Segment decode		O
	74	SEGL	7Segment time-sharing display		O
	75	ARWS	Arrow switch control		O
	76	ASC	ASCII conversion		O
	77	PR	ASCII code print		O
	78	FROM	Reading from buffer		O
	79	TO	Writing to buffer		O
External device SER	80				
	81	PRUN	For FX_2-40AP/AW		O
	82				
	83				
	84				
	85	VRRD	FX-8AV read		O
	86	VRSC	FX-8AV graduation read		O
	87				
	88				
	89				O
External devices F2	90	MNET	For F-16NP/NT		O
	91	ANRD	F_2-6A read		O
	92	ANWR	F_2-6A write		O
	93	RMST	F_2-32RM start		O
	94	RMWR	F_2-32RM write		O
	95	RMRD	F_2-32RM read		O
	96	RMMN	F_2-32RM monitor		O
	97	BLK	F_2-30GM block designation		O
	98	MCDE	F_2-30GM M code read		O
	99				

Common to FX series

Basic Units	Input Relay Numbers	Output Relay Numbers
FX_0 - 30M	X0 to X17 16 points	Y0 to Y15 14 points
FX_0 - 20M	X0 to X13 12 points	Y0 to Y7 8 points
FX_0 - 14M	X0 to X7 8 points	Y0 to Y5 6 points
FX - 128M	X0 to X77 64 points	Y0 to Y77 64 points
FX - 80M	X0 to X47 40 points	Y0 to Y47 40 points
FX - 64M	X0 to X37 32 points	Y0 to Y37 32 points
FX - 48M	X0 to X27 24 points	Y0 to Y27 24 points
FX - 32M	X0 to X17 16 points	Y0 to Y17 16 points
FX - 24M	X0 to X13 12 points	Y0 to Y13 12 points
FX - 16M	X0 to X7 8 points	Y0 to Y7 8 points

- The total maximum number of I/O points is 256 points for an FX using an extension units and extension blocks in combination.

FX series device numbers

Items		Specifications		
		FX_0	FX	
Auxiliary relays	General use	496 points (M0 to M495)	500 points (M0 to M499)	Ranges can be changed by parameter.
	Latched	16 points (M496 to M511)★	524 points (M500 to M1023)	
	Special use	56 points (in the range of M8000 to M8254)	256 points (M8000 to M8255)	
States	Initial	10 points (S0 to S9)	10 points (S0 to S9)	
	General	54 points (S10 to S63)	490 points (S10 to S499)	Ranges can be changed by parameter.
	Keep	—	400 points (S500 to S899)	
	Annunciator	—	100 points (S900 to S999)	
Timers	100 msec	56 points (T0 to T55) 0.1 to 3276.7 sec	200 points (T0 to T199)	
	10 msec	24 points (T32 to T55) 0.01 to 327.67 sec *1	46 points (T200 to T245)	
	1 msec additional	—	4 points (T246 to T249) 0.001 to 32.767 sec☆	
	100 msec additional	—	6 points (T250 to T255) 0.1 to 3276.7 sec☆	
	Analog	1 point (indirect designation of special D8013) 0.1 to 25.5 sec	—	
Counters	Up	14 points (C0 to C13) 16-BIT	100 points (C0 to C99) 16-BIT	Ranges can be changed by parameter.
		2 points (C14, C15) 16-BIT★	100 points (C100 to C199) 16-BIT	
	Up/down	—	20 points (C200 to C219) 32-BIT	Ranges can be changed by parameter.
		—	15 points (C220 to C234) 32-BIT	
	High speed	4 points (C235 to C254) 32-BIT★	6 points (C235 to C255) 32-BIT	1-phase
Registers	General-purpose register	30 points (D0 to D29) 16-BIT	200 points (D0 to D199) 16-BIT	Ranges can be changed by parameter.
		2 points (D30, D31) 16-BIT★	312 points (D200 to D511) 16-BIT	
	Special register	27 points (D8000 to D8069) 16-BIT	256 points (D8000 to D8255) 16-BIT	
	Index use	2 points (V, Z) 16-BIT	2 points (V, Z) 16-BIT	
	File use	—	2000 points (D1000 to D2999) 16-BIT	
Pointers	Branch use	64 points (see P0 to P63) JUMP instruction	64 points (see P0 to P63) JUMP-CALL instruction	
	Interrupt use	4 points (input interrupt by I0 to I3)	9 points (for I0 to I8 and timer interrupt)	
Nesting		8 points (N0 to N7) for master control	8 points (N0 to N7) for master control	

★ backed up by built-in EEPROM.

☆ battery-backed up.

*1 T32 to T55 become 10 msec times when the special auxiliary relay, M8028, is turned ON.

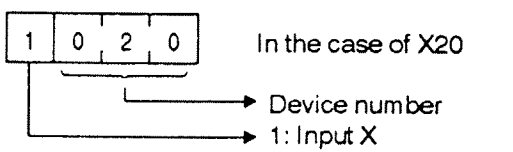
Error messages

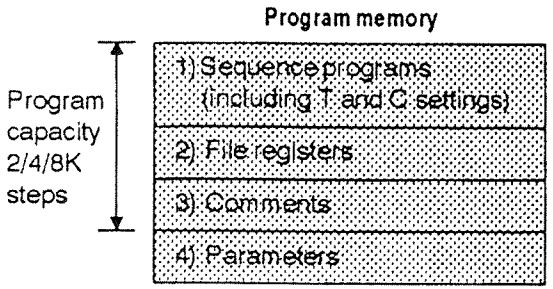
Messages	Causes	Corrective actions
COMMS. ERROR	PC communication error	Check the PC and cable.
WRITE FORBIDDEN	An attempt was made to write data to EPROM.	Change the destination memory.
	The EEPROM cassette memory protect switch is set in the ON position when an attempt was made to write to EEPROM.	Set the memory protect switch to the OFF position before writing data to EEPROM.
NOT FOUND	The designated instruction was not found.	Proceed to next step.
ENTRY CODE ERROR	An operation was attempted that is not allowed with the Keyed-in entry code.	Attempt only the operations that are allowed for the set protection level.
NOT USABLE	The selected function cannot be used under the current conditions.	Select a usable function.
VERIFY ERROR	Mismatched step data was found.	Correct the mismatch.
STEP OVERFLOW	The designated step number is greater than the allowable maximum step number.	Change the step number.
SETTING ERROR	The set value or data is improper.	Key in proper value or data.
PC PARA. ERROR	The set PC parameter is incorrect.	Set a correct PC parameter.
PC RUNNING	A write operation is attempted while the PC is in the RUN state.	Set the PC to the STOP state.
NO PROGRAM SPACE	There is no more program storage area.	Change the parameter settings.
PROGRAM OVERFLOW	No more memory space for inserts.	Delete all NOP instructons from the program. If the program is still larger than the available memory area, revise the program.
COMMAND ERROR	The instruction is incorrect.	Set a correct instruction.
NO MEM. CASSETTE	A memory cassette is not installed to the PC.	Install the memory cassette.
DEVICE ERROR	The designated device or pointer is incorrect.	Input the correct device or pointer.

Error messages (program check)

Error Messages	Error Codes	Error Descriptions
PC H/W ERROR	6101	RAM error
	6102	Operation circuit error
	6103	I/O bus error (M8069 driven)
COMMS. ERR.	6201	Parity error, overrun error, frame error
	6202	Communications character error
	6203	Communications data sum check error
	6204	Data format error
	6205	Command error
LINK ERROR	6301	Parity error, overrun error and frame error
	6302	Communications character error
	6303	Communications data sum check error
	6304	Data format error
	6305	Command error
	6306	Watchdog timer error
PARA. ERROR	6401	Program sum check
	6402	Memory capacity setting error
	6403	Keep area setting error
	6404	Comment area setting error
	6405	File register area setting error
	6409	Other setting error
GRAMMAR ERROR	6501	Instruction, a device symbol, device number combination error
	6502	There is no OUT T or C before the setting value.
	6503	There is no setting value after OUT T and C. Insufficient operands with applied instructions
	6504	Duplicate label number used; Overlapping designation of interrupt input and/or high-speed counter input
	6505	Device number range over
	6509	Other

Error Messages	Error Codes	Error Descriptions
LADDER ERR.	6601	LD or LDI used more than 8 times continuously.
	6602	1) No LD or LDI instruction. No coil. Incorrect relationship of LD/LDI and ANB/ORB 2) One of the following is not connected to the bus line: STL, RET, MCR, P (pointer), I (interrupt), EI, DI, SRET, IRET, FOR, NEXT, FEND, END
	6603	MPS used more than 11 times continuously.
	6604	Incorrect relationship between MPS and MRD/MPP
	6605	1) STL used more than 8 times continuously 2) MC, MCR, I (interrupt), or SRET in STL. 3) RET outside STL. No STL
	6606	1) No P or I 2) No SRET or IRET 3) I, SRET, or IRET designated in the main program 4) STL, RET, MC, and MCR designated in the subroutine or interrupt routine.
	6607	1) Illegal FOR and NEXT designation Nesting level deeper than 5 levels 2) One of the following commands is designated in the FOR-NEXT loop. STL, RET, MC, MCR, IRET, SRET, FEND, END
	6608	1) Illegal MC and MCR designation 2) No MCR N0 3) One of the following commands is designated in the MC-MCR loop. SRET, IRET, I
	6609	Other

Error Messages	Error Codes	Error Descriptions
RUN TIME ERROR	6701	No jump destination of CJ or CALL. A label follows after the END instruction. An independent label is in the FOR-NEXT loop or a subroutine program.
	6702	CALL nesting level deeper than 5 levels
	6703	Interrupt nesting level deeper than 2 levels
	6704	FOR-NEXT loop nesting level deeper than 5 levels
	6705	A device other than a valid is used for an applied instruction operand.
	6706	The device number or data designated as an applied instruction operand is outside the allowable designation range.
	6707	File register operation is accessed without allocating the file register area.
	6709	Other (No IRET or SRET or illegal FOR-NEXT loop)
I/O ERROR	(Example) 1020	The I/O number is not supported by the current hardware. An explanation of the error code. 



The term "program" includes all of the data above [(1) to (4)]. Note that the HPP cannot use comments.

Contents of Program Memory

(1) A sequence program may contain constants for the setting [K] for timers and counters.

(2) It is necessary to allocate a block by a parameter setting for file register.

Block 0: No file register

Block 1: D1000 to D1499 500 points/500 steps

Block 2: D1000 to D1999 1000 points/1000 steps

Block 3: D1000 to D2499 1500 points/1500 steps

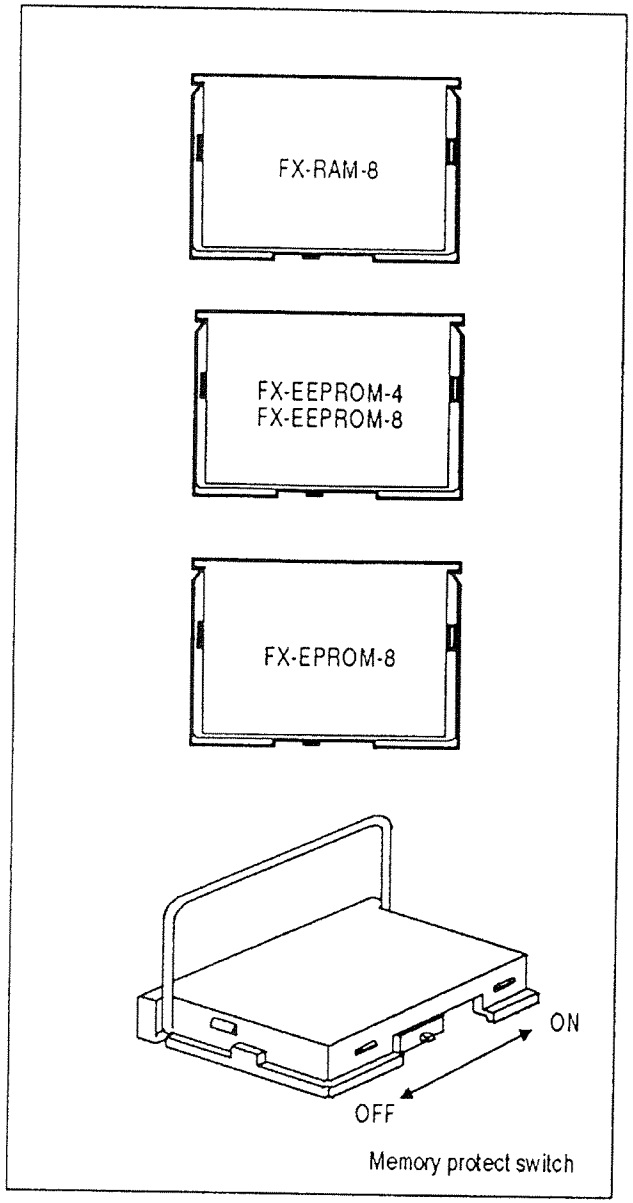
Block 4: D1000 to D2999 2000 points/2000 steps

(3) The HPP cannot read or write comments.

However, if a program containing comments is transferred to the HPP RAM, the comments will be written to the RAM area.

(4) The parameter area sets the program memory capacity, latch memory device number range, file register block number, entry code, etc.

(5) Pages 110 and 111 give details about the FX_0 PC.



Program Memory Types

There are three program memory capacities available:
2K steps, 4K steps and 8K steps.

The FX memory capacity, steps 0 to 1999/3999/7999, depends on the parameter setting and the area available from the RAM, EEPROM, and EPROM cassettes. The program memory capacity of the FX_0 PC is fixed to 800 steps. This is stored on internal EEPROM.

(1) RAM

Battery backup is necessary.
A RAM (2K steps) area is incorporated in the PC and an FX-RAM-8 RAM cassette (8K steps) is also available.
If the RAM cassette is removed from the PC, the program in the cassette will be lost.

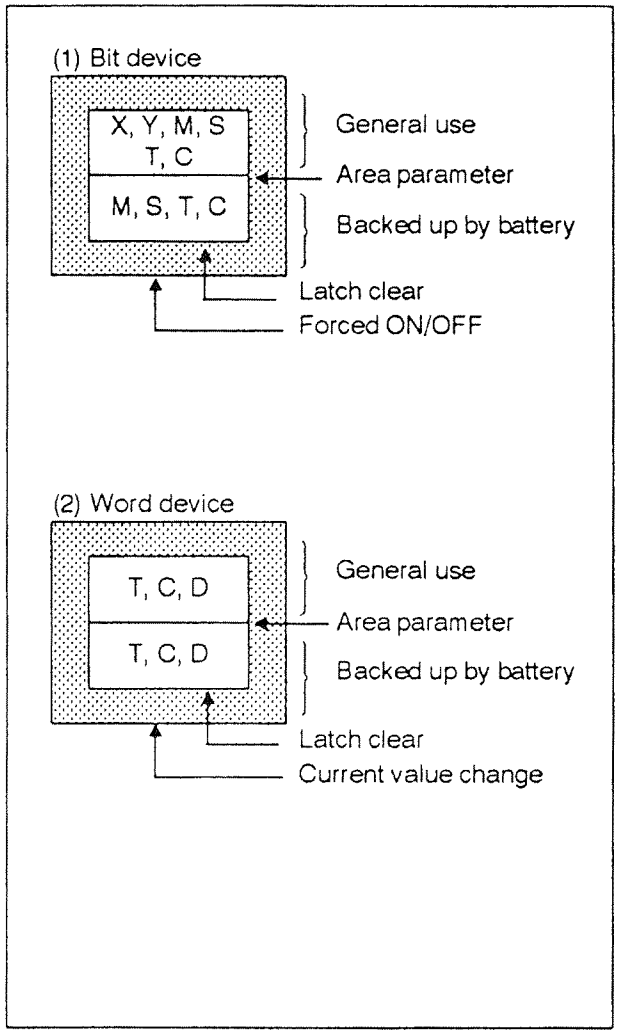
(2) EEPROM

Programs cannot be written unless the memory protect switch is set to the OFF position.
The following EEPROM cassettes are available:
FX-EEPROM-4 (4K steps), FX-EEPROM-8 (8K steps).

(3) EPROM

Use a ROM writer module to write the program to the EPROM cassette.
The FX-EPROM-8 (8K steps) EPROM cassette is available.

- In addition, there are an FX-RAM-8C cassette and the FX-EEPROM-4C EEPROM cassette with a real-time clock (clock function).
  However, these cassettes need battery backup.



(1) Bit device memory

The bit device memory stores the ON/OFF state of a contact or coil operation ie; the status of input relay X, output relay Y, auxiliary relay M, state S, timer T and counter C.
Some M, S, T, and C, device data is backed up by the battery.
For example, M0 to M499 are general use, while M500 to M1023 can be battery backed. The boundary point M499/M500 can be changed by a parameter setting.

(2) Word device memory

The word device memory stores the current value of timers (T), counters (C), data registers (D) and other devices.
The word device memory area is divided into the two parts; the general use area are the battery back up area. The area size can be changed by changing the parameter setting.

(3) Since the FX_0 PC holds backup data in its built-in EEPROM, a battery is unnecessary. The backup range (latch range) is fixed.

Allowable Ranges of Program Memory Capacities and Set Values

(Unit: number of steps)

Parameter \ Types of Program Memory		RAM built in the PC	EX-EEPROM-4	FX-RAM-8 FX-EPROM-8 FX-EEPROM-8
FX	Sequence program	0 to 2K	0 to 4K	0 to 8K
	File register	0 to 2K	0 to 2K	0 to 2K
	Comment	0 to 2K	0 to 4K	0 to 4K
	Total (available memory)	2K	2K, 4K	2K, 4K, 8K

- Each file register or comment block is the equivalent of 500 programming steps.
- The FX_0's built in program capacity is 800 steps. However, file registers and comments cannot be registered.

Items		Default Values	
		FX_0	FX
Memory capacities		2K steps	Same as the memory cassette contents
File registers		None available	0 without allocation
Comment areas		0 without allocation	0 without allocation
Latch ranges	M	496 to 511	500 to 1023
	S	–	500 to 999
	C (16)	14 to 15	100 to 199
	C (32)	1	220 to 255
	D	30 to 31	200 to 511
Entry codes		Unregistered	Unregistered
Comments		Unregistered	Unregistered

- The parameter is rewritten to the default value by batch NOP writing to the entire range, also by the default value setting operation of a parameter.
- Comment areas cannot be allocated and comments cannot be registered by the HPP.
- The default value of the parameter of FX_0 is the same as the 2K steps of FX. The memory capacity is fixed to 800 steps and the latch range for actual operation is fixed to the range indicated by the ★ symbol in the table on page 100. Although file registers and comment areas can only be set to 0, the entry codes and titles can be registered.

8. APPENDICES

This appendix explains the precautions to take when handling peripheral devices while using an FX_0 series PC.

1. Parameter settings: Latch ranges, memory capacities, comment capacities and file register capacities must be default values.

(1) Memory capacity settings

(a) Program capacity: Select 2K step. Do not set a value other than 2K step.

If a value more than 2K step is set:

If writing is executed to an FX_0 series PC, the "PC MEMORY TOO SMALL" message is displayed and writing cannot be executed.

Writing to an FX_0 series PC when 2K step is set:

Step 800 and after are ignored.
If a program with more than 800 steps is written to an FX_0 series PC, step 800 and after will not be written to the PC.

Reading from an FX_0 series PC when 2K step is set:

NOP instructions will be written after the end of the program or step 800, which ever is lower.

(b) Comment capacity: Do not set the comment capacity.

Comment capacity set less than 2 blocks:

If writing is executed to an FX_0 series PC, the "WRITE ERROR" message is displayed.
(No comment is written.)

Comment capacity set more than 3 blocks:

If writing is executed to an FX_0 series PC, the "WRITE ERROR" message is displayed and program writing cannot be executed.

(c) File register: Do not set a file register.

Writing to an FX_0 series PC when a file register is set:

If writing is executed to an FX_0 series PC, the "WRITE ERROR" message is displayed and program writing cannot be executed.

(2) Latch range setting: Latch ranges cannot be changed when an FX_0 series PC is used.

When writing is executed after setting (changing) a latch range, the "COMPLETED" message is displayed and writing is executed. However, the FX_0 series PC ignores the new latch range.

(3) Program title registration: Program titles can be registered when an FX_0 series PC is used.

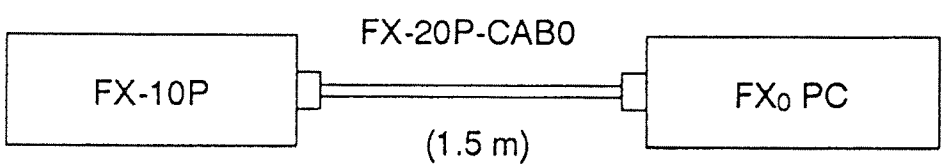
(4) Entry code registration: Entry codes can be registered when an FX_0 series PC is used.

2. Changing constants during RUN

When an FX_0 PC is used, constants (timer and counter set values, present value of data register) can be changed even though an EEPROM is used as the program memory.
However, the PC scan time becomes longer (20 to 30 msec) and a response delay (20 to 30 msec) is caused by input interrupt I00[ ] to I30[ ].

3. Program cable

The FX-20P-CAB type program cable (1.5 m, supplied seperately) should be used to connect the FX-10P-E to an FX PC. However, when connecting to an FX_0, the FX-20P-CAB0 type program cable (1.5 m) should be used (supplied separately).



MEMO

OPERATION MANUAL

FX-10P-E

MITSUBISHI ELECTRIC CORPORATION
HEAD OFFICE: MITSUBISHI DENKI BLDG MARUNOUCHI TOKYO 100-8310
HIMEJI WORKS: 840, CHIYODA CHO, HIMEJI, JAPAN

MODEL	FX-10P-O-E
MODEL CODE	09R913

JY992D33401B
(SEN)

Effective Mar. 2003
Specifications are subuject tochange without notice.