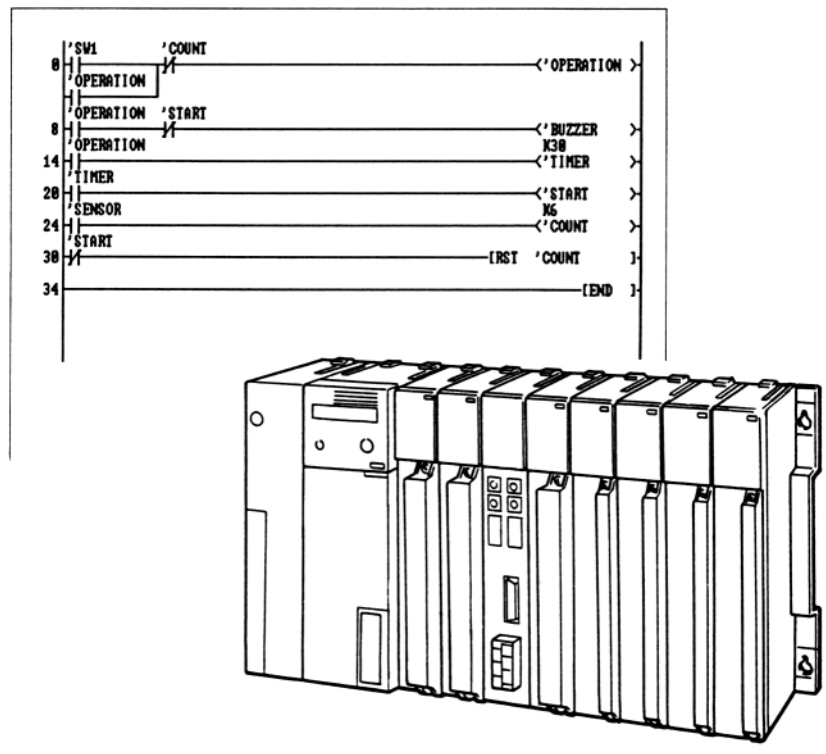
MITSUBISHI

QnA SERIES

QnACPU

GUIDEBOOK



QnA
MELSEC

Mitsubishi Programmable Controller

REVISIONS

*The manual number is given on the bottom left of the back cover.

Print Date	*Manual Number	Revision
Jul., 1996	IB (NA) 66606-A	First edition

SAFETY CAUTIONS

(You must read these cautions before using the product)

In connection with the use of this product, in addition to carefully reading both this manual and the related manuals indicated in this manual, it is also essential to pay due attention to safety and handle the product correctly.
The safety cautions given here apply to this product in isolation. For information on the safety of the PC system as a whole, refer to the CPU module User's Manual.
These **SAFETY CAUTIONS** are classified into two grades: "DANGER" and "CAUTION".

DANGER Safety caution given when incorrect handling could result in hazardous situations involving the possibility of death or serious injury.

CAUTION Safety caution given when incorrect handling could result in hazardous situations involving the possibility of moderate or light injury or damage to property.

Note that, depending on the circumstances, failing to follow a  **CAUTION** may also have very serious consequences.
Both of these classes of safety caution are very important and must be observed.
Store this manual carefully in a place where it is accessible for reference whenever necessary, and forward a copy of the manual to the end user.

[System Design Precautions]

DANGER

- Safety circuits should be installed external to the programmable controller to ensure that the system as a whole will continue to operate safely in the event of an external power supply malfunction or a programmable controller failure. Erroneous outputs and operation could result in an accident.
  1) The following circuitry should be installed outside the programmable controller:
     Interlock circuitry for the emergency stop circuit protective circuit, and for reciprocal operations such as forward/reverse, etc., and interlock circuitry for upper/lower positioning limits, etc., to prevent machine damage.
  2) When the programmable controller detects an abnormal condition, processing is stopped and all outputs are switched OFF. This happens in the following cases:
     - When the power supply module's over-current or over-voltage protection device is activated.
     - When an error (watchdog timer error, etc.) is detected at the PC CPU by the self-diagnosis function.

     Some errors, such as input/output control errors, cannot be detected by the PC CPU, and there may be cases when all outputs are turned ON when such errors occur. In order to ensure that the machine operates safely in such cases, a failsafe circuit or mechanism should be provided outside the programmable controller. Refer to the CPU module user's manual for an example of such a failsafe circuit.
  3) Outputs may become stuck at ON or OFF due to an output module relay or transistor failure. An external circuit should therefore be provided to monitor output signals whose incorrect operation could cause serious accidents.

- A circuit should be installed which permits the external power supply to be switched ON only after the programmable controller power has been switched ON. Accidents caused by erroneous outputs and motion could result if the external power supply is switched ON first.

- When a data link communication error occurs, the status shown below will be established at the faulty station. In order to ensure that the system operates safely at such times, an interlock circuit should be provided in the sequence program (using the communication status information).
  Erroneous outputs and operation could result in an accident.
  1) The data link data which existed prior to the error will be held.
  2) All outputs will be switched OFF at MELSECNET (II, /B, /10) remote I/O stations.
  3) At the MELSECNET/MINI-S3 remote I/O stations, all outputs will be switched OFF or output statuses will be held, depending on the E.C. mode setting.

  For details on procedures for checking faulty stations, and for operation statuses when such errors occur, refer to the appropriate data link manual.

[System Design Precautions]

⚠ CAUTION

- Do not bundle control lines or communication wires together with main circuit or power lines, or lay them close to these lines.
  As a guide, separate the lines by a distance of at least 100 mm, otherwise malfunctions may occur due to noise.
  When file register R that are outside the range are read, e.g. by a MOV instruction, the file register data will become $FFFF_H$ and use of this data will cause malfunctions. Take care not to use file registers that are outside the range when designing programs.
  For details on instructions, refer to the Programming Manual.

[Cautions on Mounting]

⚠ CAUTION

- Use the PC in an environment that conforms to the general specifications in the manual.
  Using the PC in environments outside the ranges stated in the general specifications will cause electric shock, fire, malfunction, or damage to/deterioration of the product.
- Make sure that the module fixing projection on the base of the module is properly engaged in the module fixing hole in the base unit before mounting the module.
  Failure to mount the module properly will result in malfunction or failure, or in the module falling.
- Extension cables should be securely connected to base unit and module connectors. Check for loose connection after installation.
  A poor connection could result in contact problems and erroneous inputs/outputs.
- Plug the memory card firmly into the memory card mounting connector. Check for loose connection after installation.
  A poor connection could result in erroneous operation.

[Cautions on Wiring]

DANGER

- Switch off the external power supply before staring installation and wiring work.
  Failure to do so could result in electrical shocks and equipment damage.
- After installation and wiring is completed, be sure to attach the terminal cover before switching the power ON and starting operation.
  Failure to do so could result in electrical shocks.

CAUTION

- Be sure to ground the FG and LG terminals, carrying out at least class 3 grounding work with a ground exclusive to the PC.
  Otherwise there will be a danger of electric shock and malfunctions.
- Carry out wiring to the PC correctly, checking the rated voltage and terminal arrangement of the product.
  Using a power supply that does not conform to the rated voltage, or carrying out wiring incorrectly, will cause fire or failure.
- Outputs from multiple power supply modules should not be connected in parallel. Failure to do so could cause the power supply module to overheat, resulting in a fire or module failure.
- Tighten the terminal screws to the stipulated torque.
  Loose screws will cause short circuits, fire, or malfunctions.
- Make sure that no foreign matter such as chips or wiring offcuts gets inside the module.
  It will cause fire, failure or malfunction.
- Connectors for external connections should be crimped, pressure welded, or soldered in the correct manner using the correct tools.
  For details regarding crimping and pressure welding tools, refer to the input/output module user's manual.
  A poor connection could cause shorts, fire, and erroneous operation.

[Cautions on Startup and Maintenance]

DANGER

- Do not touch terminals while the power is ON.
  This will cause malfunctions.
- Make sure that the battery is connected properly. Do not attempt to charge or disassemble the battery, do not heat the battery or place it in a flame, and do not short or solder the battery.
  Incorrect handling of the battery can cause battery heat generation and ruptures which could result in fire or injury.
- Switch the power off before cleaning or re-tightening terminal screws.
  Carrying out this work while the power is ON will cause failure or malfunction of the module.

CAUTION

- In order to ensure safe operation, read the manual carefully to acquaint yourself with procedures for program changes, forced outputs, RUN, STOP, and PAUSE operations, etc., while operation is in progress.
  Incorrect operation could result in machine failure and injury.
- Do not disassemble or modify any module.
  This will cause failure, malfunction, injuries, or fire.
- Switch the power OFF before mounting or removing the module.
  Mounting or removing it with the power ON can cause failure or malfunction of the module.
- When replacing fuses, be sure to use the prescribed fuse. A fuse of the wrong capacity could cause a fire.

[Cautions on Disposal]

CAUTION

- Dispose of this product as industrial waste.

INTRODUCTION

Thank you for choosing the Mitsubishi MELSEC-QnA Series of General Purpose Programmable Controllers. Please read this manual carefully so that the equipment is used to its optimum. A copy of this manual should be forwarded to the end User.

CONTENTS

1. USING THIS MANUAL ..... 1 – 1 ~ 1 – 3

1.1 Terms and Abbreviations Used in This Manual ..... 1 – 2
1.2 Basic Key Specifications ..... 1 – 3

2. QnACPU GENERAL DESCRIPTION ..... 2 – 1 ~ 2 – 6

2.1 General Description ..... 2 – 1

3. OPERATING PROCEDURE USING THE QnACPU ..... 3 – 1 ~ 3 – 59

3.1 Applicable System Configuration ..... 3 – 1
3.2 General Procedure for Overall Operation ..... 3 – 2
3.3 Starting Up GPP Functions after Completing Peripheral Device Preparations ..... 3 – 3
3.3.1 Preparing the IBM PC/AT hardware ..... 3 – 3
3.3.2 Installing the GPP function software package ..... 3 – 5
3.3.3 Starting the GPP functions ..... 3 – 9
3.3.4 Corrective action when an error occurs ..... 3 – 10
3.4 Creating Programs ..... 3 – 11
3.4.1 Selecting CPU type used and setting file ..... 3 – 11
3.4.2 Programming in ladder mode ..... 3 – 13
3.4.3 Appending comments to devices ..... 3 – 18
3.4.4 Saving program to hard disk ..... 3 – 20
3.4.5 Convenient functions for programming ..... 3 – 22
3.5 Operating the CPU for Monitoring and Testing ..... 3 – 29
3.5.1 Preparations for operating the CPU ..... 3 – 29
3.5.2 Writing program to CPU ..... 3 – 38
3.5.3 Operating the CPU for monitoring ..... 3 – 40
3.5.4 Modifying program while monitoring CPU operation (online program change (write during RUN)) ..... 3 – 44
3.5.5 Registering a password ..... 3 – 46
3.5.6 Checking responses to forced faults (PC diagnosis) ..... 3 – 49
3.6 Printing the Program ..... 3 – 54
3.6.1 Setting the printer and paper size ..... 3 – 54
3.6.2 Printing out the ladder program ..... 3 – 56
3.6.3 Sample ladder program print-out ..... 3 – 58
3.7 Quitting GPP Functions ..... 3 – 59

4. MAKING THE MOST OF YOUR QnACPU (1) ..... 4 – 1 ~ 4 – 76

4.1 System Configuration ..... 4 – 1
4.2 Programming for Individual Control Processes ..... 4 – 2
4.2.1 File structure ..... 4 – 2
4.2.2 Sample program ..... 4 – 4
4.2.3 Program list monitor ..... 4 – 9
4.3 Easy Programming ..... 4 – 14

4.3.1 Direct access of special-function module buffer memory as devices ..... 4 – 14
4.3.2 Accessing word devices as bit units and using differential contacts ..... 4 – 15
4.4 Convenient Functions When Using QnACPU ..... 4 – 16
4.4.1 Allocation of optimal device points range for the system used ..... 4 – 16
4.4.2 Setting device values required by the system as initial values ..... 4 – 19
4.4.3 Boot operation with a program transferred from memory card to QnACPU internal memory ..... 4 – 28
4.4.4 Remote operation of the QnACPU from a distant location ..... 4 – 35
4.4.5 What are local devices? ..... 4 – 52
4.5 Easy Program Debugging ..... 4 – 56
4.5.1 Simultaneous monitoring by multiple operators ..... 4 – 56
4.5.2 Monitoring devices at any step ..... 4 – 58
4.6 Programming ..... 4 – 62
4.6.1 Creating one program ..... 4 – 62
4.6.2 Creating multiple programs ..... 4 – 67

5. MAKING THE MOST OF YOUR QnACPU (2) ..... 5 – 1 ~ 5 – 28

5.1 Programming Before I/O Module Configuration is Determined ..... 5 – 1
5.1.1 Programming using labels ..... 5 – 1
5.2 Utilizing Program Resources ..... 5 – 16
5.2.1 Program standardization with user-created macros ..... 5 – 16

About Manuals

The manuals related to the QnACPU are listed in the table below.
Please order those you require.

Related Manuals

Manual Name	Manual Number
Q2A(S1)/Q3A/Q4ACPU User's Manual Describes the performance, functions, and handling of the Q2ACPU(S1), Q3ACPU, and Q4ACPU, and the specifications and handling of memory cards and base units.	IB-66608
QnACPU Programming Manual (Fundamentals) Describes how to create programs, the names of devices, parameters, and types of program. (Purchased separately)	IB-66614
QnACPU Programming Manual (Common Instructions) Describes how to use sequence instructions, basic instructions, and application instructions. (Purchased separately)	IB-66615
QnACPU Programming Manual (Special Function) Describes the dedicated instructions for special function modules available when using the Q2ACPU(S1), Q3ACPU, and Q4ACPU. (Purchased separately)	IB-66616
QnACPU Programming Manual (AD57 Instructions) Describes the dedicated instructions for controlling an AD57(S1) type CRT controller module available when using the Q2ACPU(S1), Q3ACPU, or Q4ACPU. (Purchased separately)	IB-66617
QnACPU Programming Manual (PID Control Instructions) Describes the dedicated instructions for PID control available when using the Q2ACPU(S1), Q3ACPU, or Q4ACPU. (Purchased separately)	IB-66618
Type SW0IVD-GPPQ GPP Function Operating Manual (Offline)	IB-66623
Type SW0IVD-GPPQ GPP Function Operating Manual (Online)	IB-66624

1. USING THIS MANUAL

This manual explains:

- What type of CPU the Q2ACPU(S1), Q3ACPU, and Q4ACPU are;
- The operating procedures when using QnA;
- And examples of use of the QnA features.

It includes descriptions of the operations of the SW□IVD-GPPQ GPP function software package (hereafter abbreviated to "GPPQ").

See the manuals listed in the table of related manuals for more details about information and operations described in this manual. Refer to the related manuals for a deeper understanding of the QnACPU functions and GPPQ functions.

The structure of this manual is as follows:

Chapter 1 USING THIS MANUAL
Describes the terms and abbreviations used in this manual and the basic key operations.

Chapter 2 QnACPU GENERAL DESCRIPTION
Describes the structured files which are a feature of QnACPU and provides a simple explanation of structured programs in multiple files.

Chapter 3 OPERATING PROCEDURE USING THE QnACPU
Describes the QnACPU operation flow using an example system configuration and describes the GPPQ operations for programming.
The operations described include installing the software in the peripheral device, programming, CPU operation, and printing out.

Chapter 4 MAKING THE MOST OF YOUR QnACPU (1)
Gives tips on programming with multiple files, which is a feature of QnACPU.

Chapter 5 MAKING THE MOST OF YOUR QnACPU (2)
Describes programming with labels, which is a feature of QnACPU.
Also describes how to use macro instructions.

1.1 Terms and Abbreviations Used in This Manual

The following abbreviations are used in this manual.

(1) QnACPU ..... An abbreviation for a Q2ACPU, Q2ACPU-S1, Q3ACPU, or Q4ACPU CPU module.

(2) GPPQ (GPP function software) ..... Abbreviation for the SW□IVD-GPPQ GPP function software package.

(3) IBM PC/AT ..... An IBM PC/AT computer, or 100% compatible (*1).

(4) Peripheral device capable of GPP functions ... General term for an IBM PC/AT, or other peripheral device, which operates the GPP function software.

(5) Internal memory ..... QnACPU internal RAM for storing sequence programs, etc.

(6) Memory card ..... Abbreviation for a Q1MEM-□□□ memory card.

(7) ACPU ..... General term for MELSEC-A Series PC CPUs.

*1 IBM is a registered trademark of International Business Machines Corporation.

POINT

The RAM memory built into the QnACPU is referred to as "internal memory" in this manual.
The term "internal RAM" appearing in SW□IVD-GPPQ displays also refers to this internal memory.

1.2 Basic Key Specifications

The key applications with the GPP functions are listed in the table below.

(1) Key Applications

Key Name	Application	Key Name	Application
[Esc]	Closing windows, halting execution, selecting an instruction Opening/closing windows	[F11]	Opens the mode select window
[Tab]	Tab code input, rapid cursor movement	[↑][↓][←][→]	Moves the cursor or scrolls through lines of a ladder or list display ([↑][↓])
[Ctrl] + [Tab]			
[Ctrl]	Used in combination with alphanumeric keys and functions keys	[Back Space]	Deletes the character to the left of the cursor
[Shift]	Selects the character at the Shift position	[Enter]	Inputs the carriage return
[Caps Lock]	Switches between uppercase and lowercase characters	[Ctrl] + [Home]	Moves the cursor to step 0 in the ladder or list mode
[Alt]	Selects the menu	[Ctrl] + [End]	Moves the cursor to the END instruction in the ladder or list mode
[Page Up]	Scrolls page up (– direction) one screen for ladder, list, or help display	[Print Screen]	Copies the screen
[Page Down]	Scrolls page down (+ direction) one screen for ladder, list, or help display	[Scroll Lock]	Disables scrolling up or down
[Insert]	Inserts a space at the cursor position	[Num Lock]	Sets the numeric keypad for numeric input only
[Delete]	Deletes the character at the cursor position (clears entire set contents)	[F12]	Opens the HELP window
[Home]	Moves the cursor to the home position		

2. QnACPU GENERAL DESCRIPTION

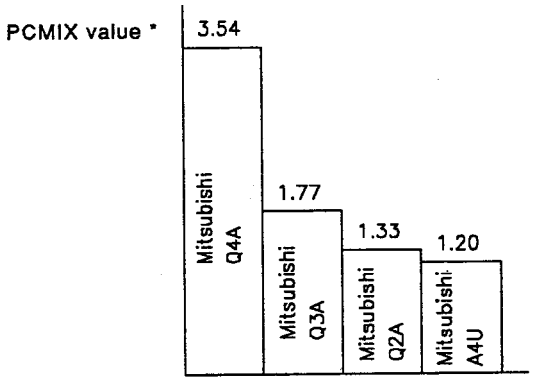
2.1 General Description

The QnACPU is a PC CPU which is both easier to use and more sophisticated than the previous ACPU. In particular, programming efficiency is enhanced by structured programs and instruction definition with macro instructions. QnACPU offers the following features:

- High-speed Processing

(1) Increased processing speed for both basic and application instructions realizes a further dramatic jump in operation speed (approximately 3-times increase for PCMIX value * ratio using Mitsubishi AnU).
This realizes dramatic reductions in tact time and allows high-speed processing of complex data.

	A4UCPU		Q4ACPU
Basic instructions	0.15 μs	→	0.075 μs (2 x)
Transfer instructions	0.90 μs	→	0.225 μs (4 x)

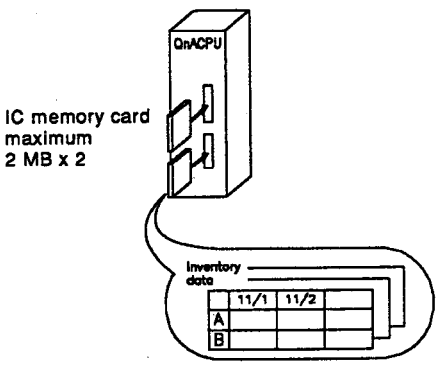


* PCMIX value: The average number of instructions executed in 1 μs.

High Memory Capacity

Serial execution of 124 k steps maximum in internal memory (4-times increase)
Maximum memory card capacity: 2 MB x 2 (4-times increase)

(2) The highly integrated IC memory card allows large volumes of data to be managed (2 MB x 2 max.).

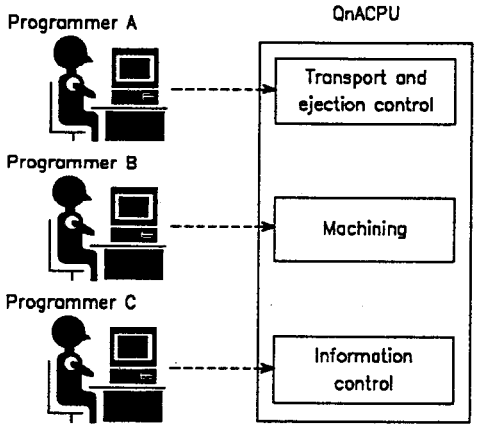


1) The application of highly integrated IC memory cards has significantly increased the capacity of the expansion data memory (approximately 5-times increase using Mitsubishi AnUCPU).
This makes it easier to handle large volumes of data.

2) The cards can also be used to save multiple programs and store comments.

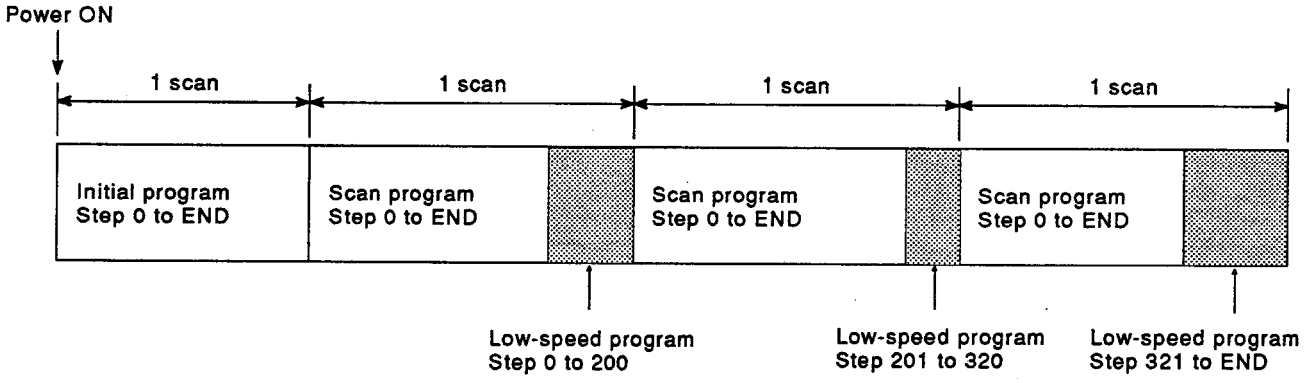
- Program Management
  Decentralized control of programs and data is possible with a maximum of 128 files.
  This promotes structured programming and allows debugging and program maintenance by multiple programmers.

(1) Multiple program management



A program can be stored in the CPU as multiple smaller programs.
Dividing the program by programmer, by function, or by process, allows multiple programmers to simultaneously develop a program and simplifies re-using the program.
When a program is modified, only the relevant file needs to be changed, which means that unwanted effects on other parts of the program can be minimized.
To simplify program maintenance, a comment and time-stamp is appended to each modification.

- Structured Programs
  Programs can be selected from four types: initial execution, scan execution, low-speed execution, and standby.
  These programs can be set to run when required, thereby reducing the scan time of the permanently executing scan program.

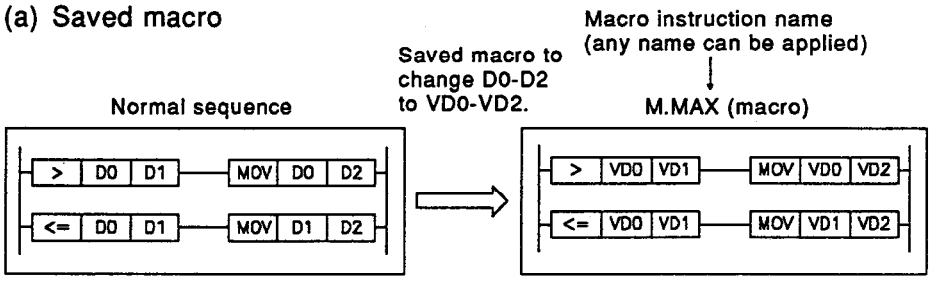


Program Standardization

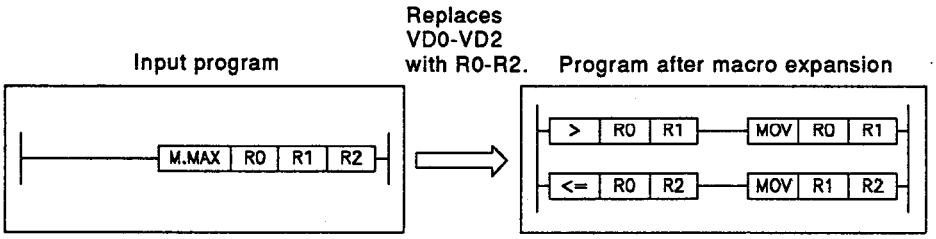
(1) Macro instructions

This function allows user-created instructions to be created by combining sequence instructions.
Using macro instructions allows the device numbers to be easily changed to match the system used, as shown below.

(a) Saved macro

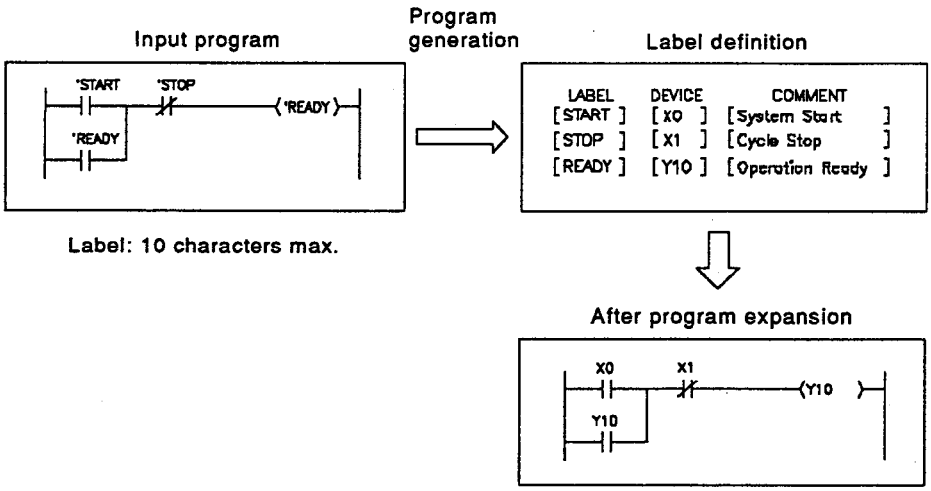


(b) Macro utilization



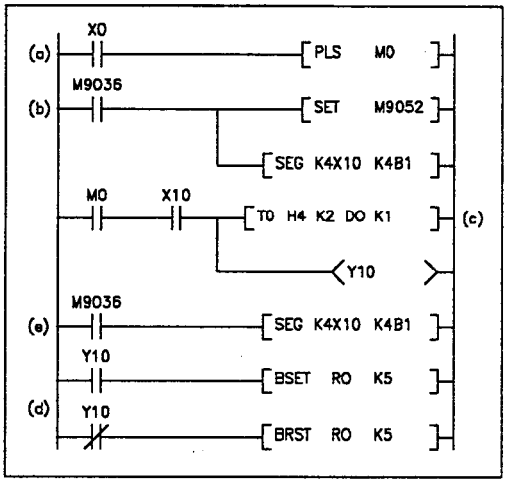
(2) Programming with labels (program standardization)

If the CPU type is set to "Xtype", standard programs can be created using labels, without considering the CPU model name, I/O addresses, or device number allocation. These standardized programs can be used by allocating devices to the labels to match the system used.

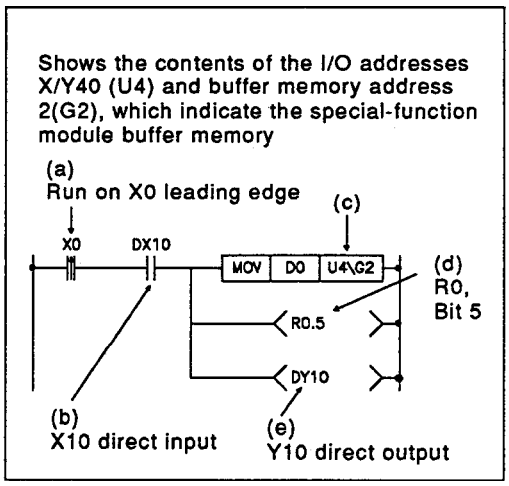


- Reduced Programming Time and Effective Memory Utilization
  A number of convenient instructions are provided to reduce programming time. The number of device points can be allocated to match the system, such that each required device occupies only the required number of points, and unnecessary devices can be removed from the 28 k words of device memory.

MELSEC-A program



MELSEC-QnA program



- Flexible Operation

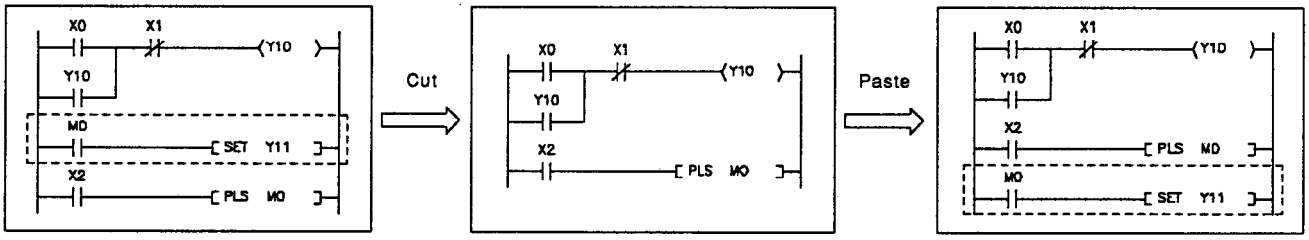
(1) Comprehensive editing functions

(a) Simultaneous, parallel editing of four programs.

(b) One-touch switching of the edited program.

(c) Cutting and pasting between edited programs.

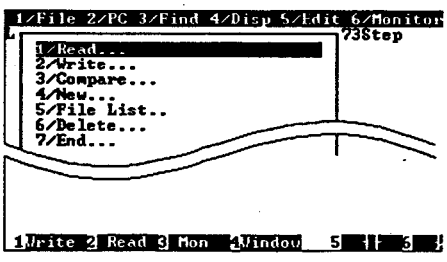
(d) Special features, such as vertical and horizontal line connections and parallel coil insertion, for easy creation of ladder programs.



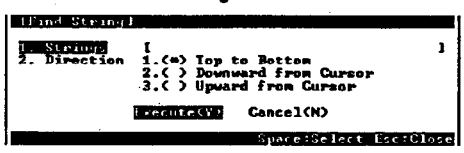
(2) Transparent operation

(a) The pull-down menus and dialog boxes offer ease of operation equivalent to commercial software packages.
Frequently used functions can be allocated to function keys to ensure correct operation every time.

Pull-down menus



Dialog boxes



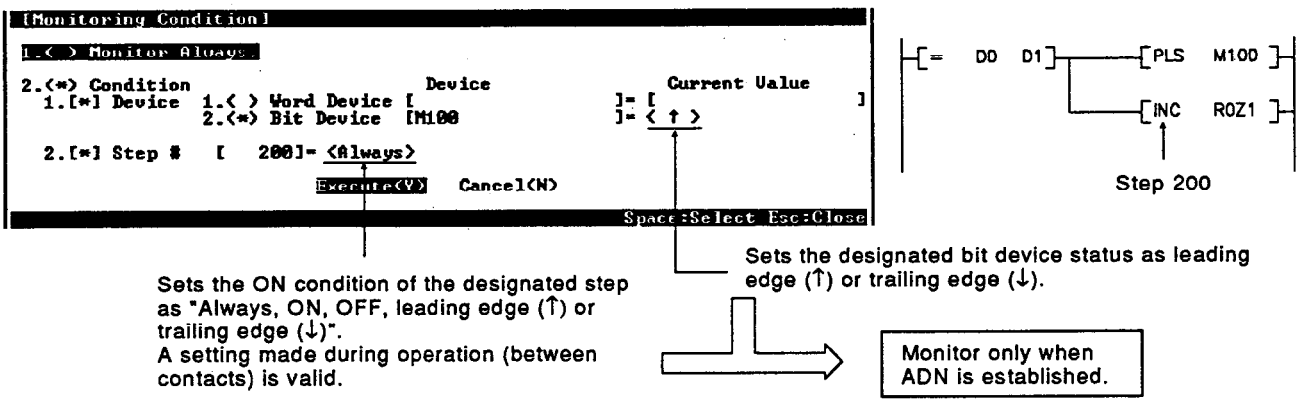
- Comprehensive Debugging Functions

(1) Comprehensive ladder monitor functions

(a) Index-qualified devices and expansion file registers can be monitored.

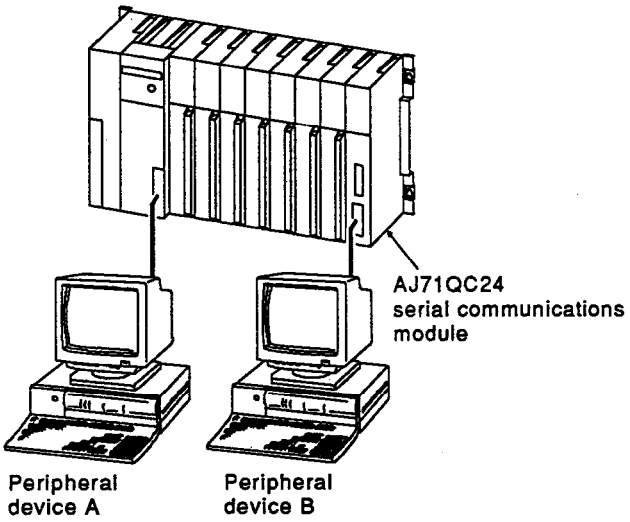
(b) The peripheral device designates the QnACPU monitoring timing, to allow precise monitoring timing.
At the designated timing, the peripheral device reads and displays the monitored information.

<Example>
Monitoring timing set at 200 steps and M100 leading edge.



(2) Debugging by multiple people

Debugging is possible using multiple peripheral devices connected to a single QnACPU.



(a) If each peripheral device debugs a separate file name, monitoring, testing, and online program change (write during RUN) can be conducted freely, without affecting any other peripheral device.
For example, if peripheral device A debugs "Machining program" and peripheral device B debugs "Assembly program."

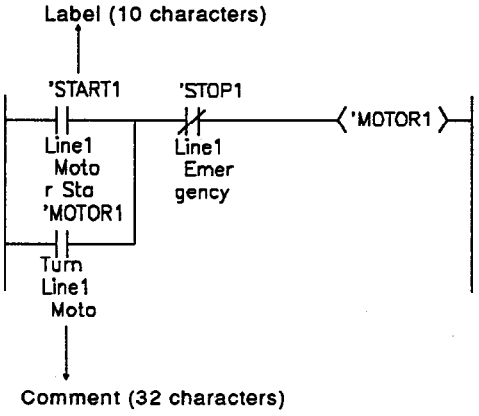
(b) Also, online program change (write during RUN) is possible for the same file from multiple peripheral devices.
A program pointer must be pre-designated for the online program change (write during RUN) programs.

- Simple Programming
  Powerful document-creation support

(1) Integral management of statements and notes with the program.

(2) Comments can be added to all devices.

(3) Comments up to 32 characters and labels up to 10 characters improve program readability.

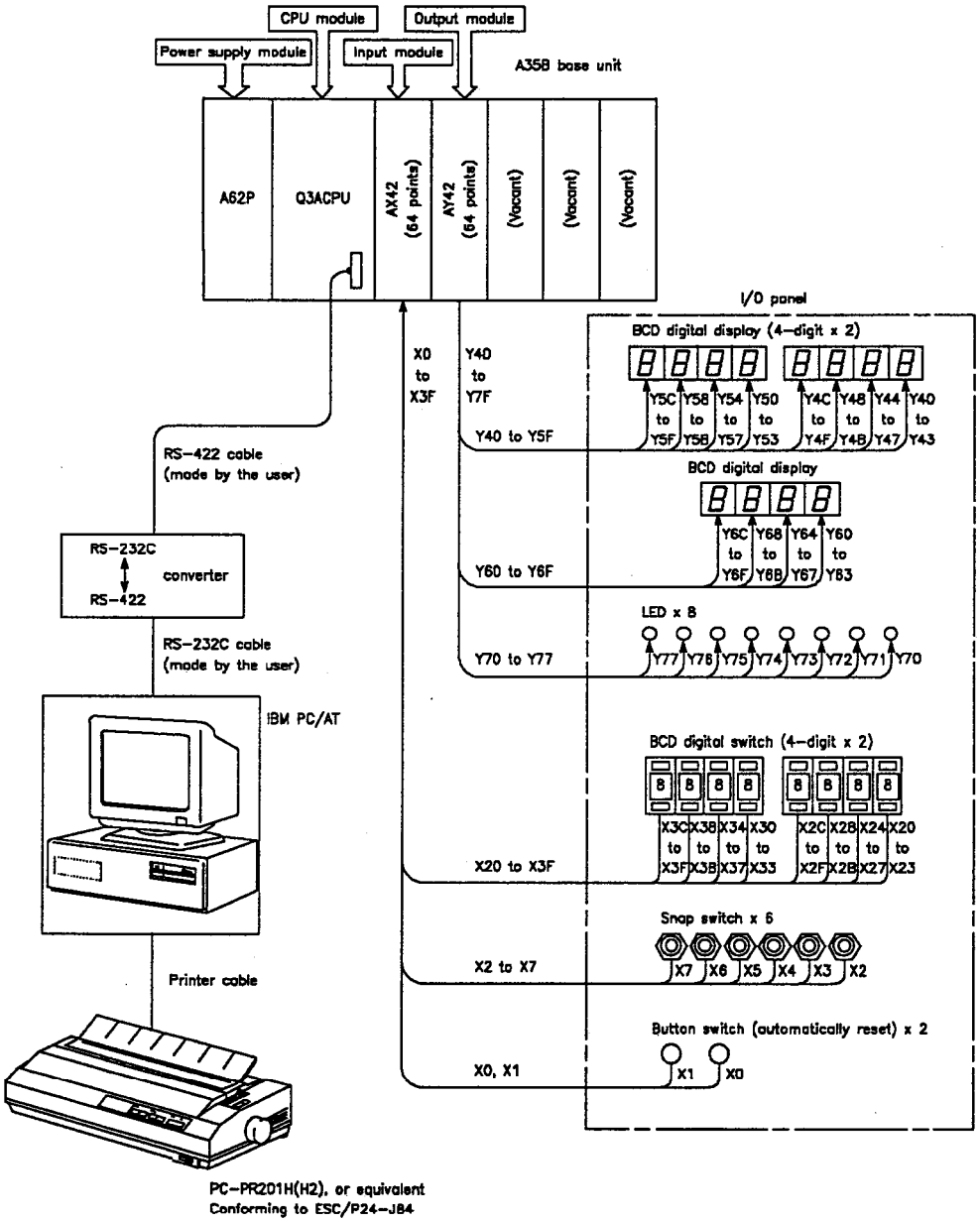


3. OPERATING PROCEDURE USING THE QnACPU

This chapter describes the basic operation of the QnACPU and the preparations and procedures required before operation.

3.1 Applicable System Configuration

The diagram shows the example system used for the following explanations. The peripheral device used is an IBM PC/AT.

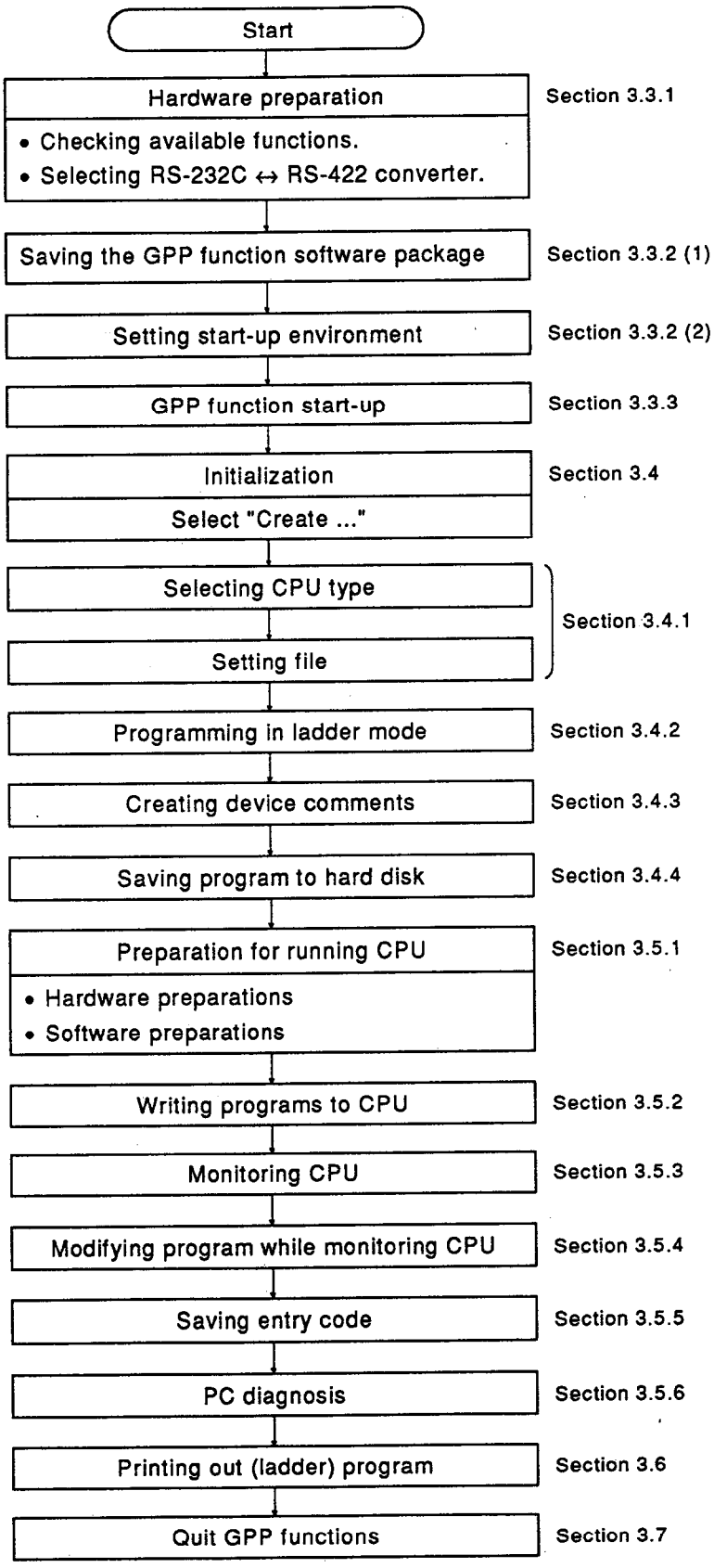


POINT

This guide uses an example of connection to the PC CPU via the built-in RS-232C port.

3.2 General Procedure for Overall Operation

The overall procedure for the basic QnACPU operations described from Section 3.3. The peripheral device is an IBM PC/AT.

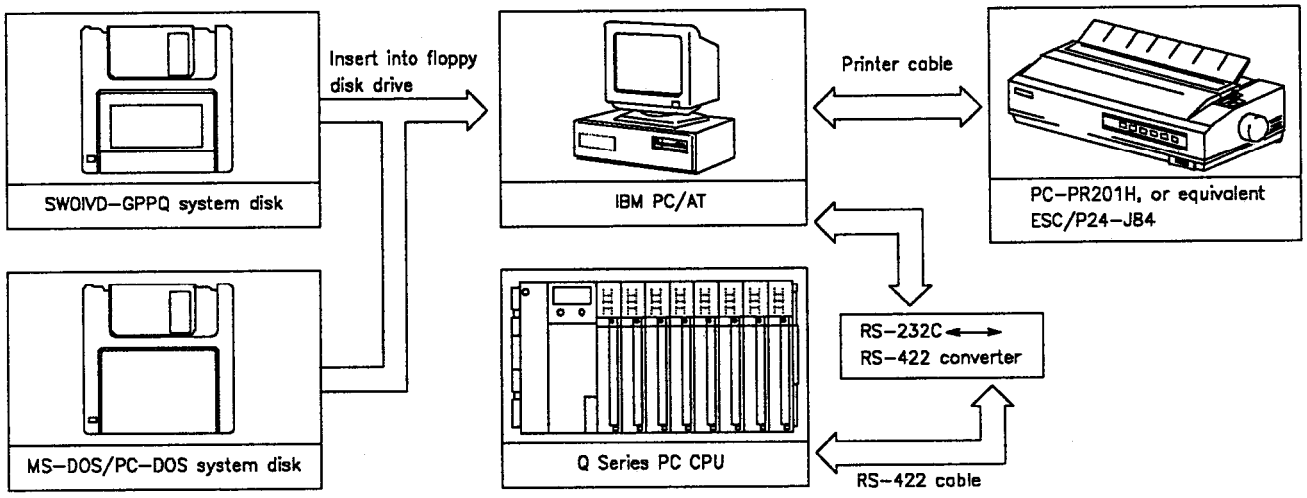


3.3 Starting Up GPP Functions after Completing Peripheral Device Preparations

The peripheral device hardware settings must be made and the GPP function software package must be set before running the GPP functions.

3.3.1 Preparing the IBM PC/AT hardware

The system configuration is shown below.



(1) Applicable IBM PC/AT

The following IBM PC/AT specifications can be used.
CPU : 80486SX(20 MHz), or higher (recommended: 80486DX2 (66 MHz))
EMS : 4 MB min.
Hard disk free space : 20 MB min.
Other : Install HIMEM and SMARTDRV drivers.

(2) MS-DOS/PC-DOS operating system disk

Start the operating system disk before starting the personal computer.
This system uses the MS-DOS or PC-DOS operating system.
Check that the version of the MS-DOS/PC-DOS operating system is version 5.0 or later.

(3) Free space in main memory

At least 540 k bytes free space is required in the main memory of the IBM PC/AT used.

(4) SW0IVD-GPPQ software package

This system disk provides the software package which is installed onto the IBM PC/AT hard disk to run the GPP functions (sequence program environment part) and SFC functions.
The software package is contained in the following disks.

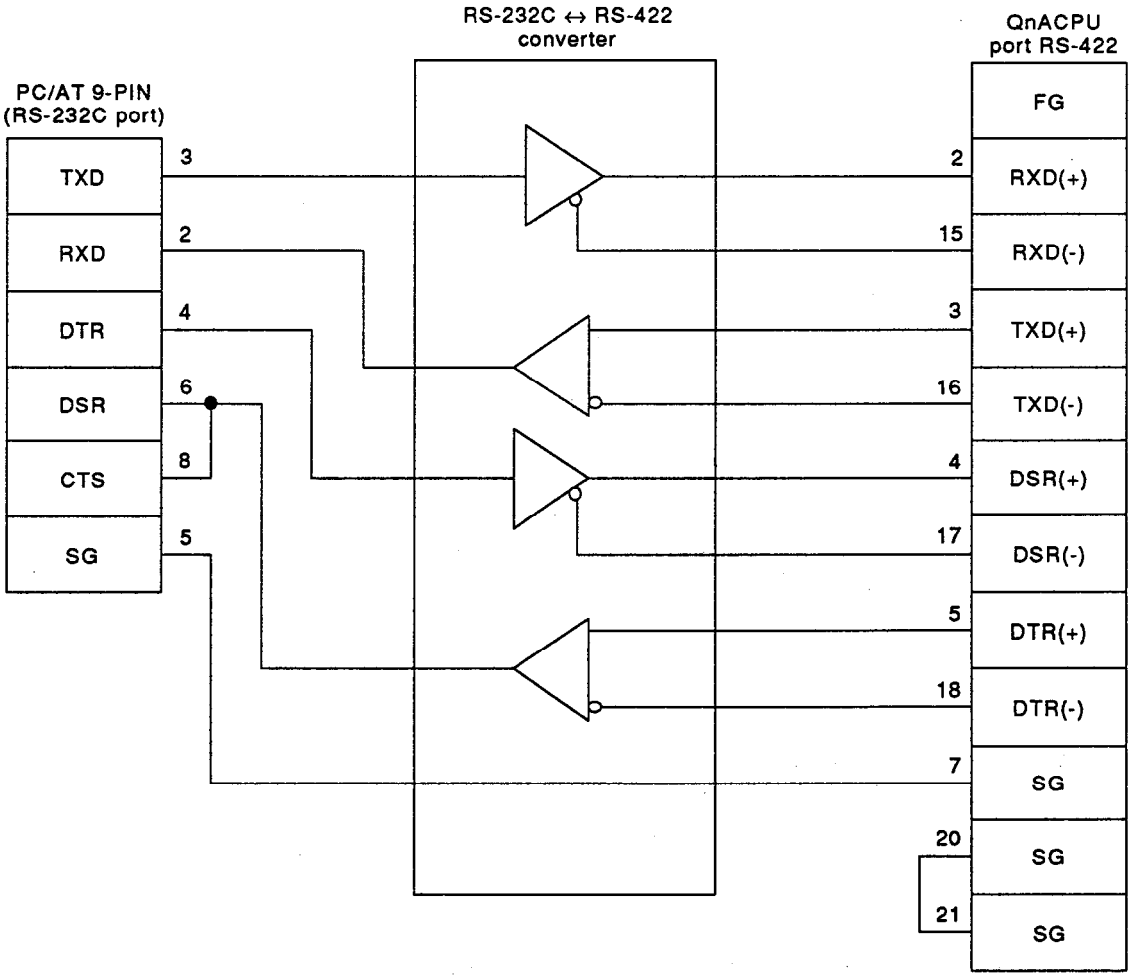
3.5-inch disks

SW0IVD-GPPQ-1	2 floppy disks
SW0IVD-GPPQ-2	1 floppy disk
SW0IVD-GPPQ-3	1 floppy disk

SW0IVD-GPPQ-1 is copy protected.
Installation of the software package requires 8 M byte minimum of free space on the hard disk.

(5) RS-232C ↔ RS-422 converter

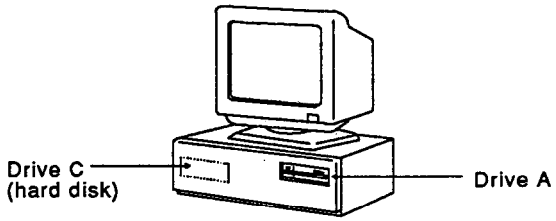
The computer and the QnACPU are connected by means of an RS-232C ↔ RS-422 converter.
Shown below is an example of the connections between the computer and the QnACPU through the RS-232C ↔ RS-422 converter. (Connect the wires to the RS-232C ↔ RS-422 as illustrated in the following figure.)



3.3.2 Installing the GPP function software package

(1) Installation procedure

This section describes the procedure to install SW0IVD-GPPQ on the IBM PC/AT hard disk.
The installation is based on the conditions shown below.



Turn ON or RESET IBM PC/AT power supply.

↓

C:\>_ ····· DOS command prompt status

↓

Insert system disk No.1 into the designated drive. ····· The drive varies according to the system being used.

↓

C:\>A:\INSTHD ····· Input the command shown to the left.

↓

[Enter]

↓

Select the source and destination drives ····· The following screen is displayed.
Select the source and destination drives.

Select Source and Destination Drives.
Source A:
Destination C:

Source : Drive where system disk 1 is inserted
Destination : Drive to install the GPP functions

↓

[Enter]

↓

(1)

(1)

Select language for installation.

..... The screen shown below is displayed. Select the language for installation.

```
1 : JAPANESE / ENGLISH
2 : JAPANESE
3 : ENGLISH
```

[Enter]

If the hard disk's available space is insufficient, the message shown below will be displayed, and the install operation will be aborted.

```
Insufficient disk space.
More than 9MB of free HD space required.
```

The available area required in the HD is as follows:

1: JAPANESE/ENGLISH	12 MB
2: JAPANESE	9 MB
3: ENGLISH	9 MB

The SW0IVD-GPPQ and SW□IVD-GPPA are different software packages and must be installed separately.
SW□IVD-GPPA is not deleted when SW0IVD-GPPQ is installed.

Check system disk 1 is inserted.

..... The following message is displayed.

```
Insert System Disk 1<SW0IUD-GPPQ 1/3>
into Drive A:

Press any key when ready.
```

[Enter]

Change the installation destination directory.

..... Set the directory from the following screen.

```
Change System Directory?
[C:\GPPQ\SYS_                    ]
```

[Enter]

Any directory can be set, but "\GPPQ\SYS" is required after it. (If omitted, "\GPPQ\SYS" is automatically appended.)
Press [Enter] key to proceed without changing the directory.

Select automatic setting of the start-up environment.

(See (2) (a) about the "Y" selection.
See (2) (b) about the "N" selection.)

Select Y or N.

(System disk installation starts.)

Install the system disk data according to the on-screen guidance messages.

..... Follow the instructions displayed on the screen to install system disks 2 and 3.

System disk installation ends.

COMPLETE ..... After CONFIG.SYS is changed, use the GPP functions after resetting and rebooting the computer.

(2) Setting the Start-up environment

Environment settings using CONFIG.SYS are required before starting the GPP functions.

(a) Automatic set-up

The following message is displayed during installation of the software package.

```
Sets Environment Variables.
Modifies CONFIG.SYS.

Perform Modification(Y/N)?
```

Press [Y] then [Enter] to change FILES (number of files) in CONFIG.SYS to 30.
However, if FILES is already greater than 30, no change is made and the original data is used.
The CONFIG.SYS settings before the changes are saved as CONFIG.ORG.
Note that any existing CONFIG.ORG file is overwritten.

(b) User set-up

The user must set the GPP function start-up environment if "N" is selected in response to the message prompting whether to make the automatic settings after the software package is installed.
The following CONFIG.SYS, AUTOEXEC.BAT settings are required to start up the GPP functions.

<Example>PC-DOS Ver. 5.0

```
[CONFIG.SYS]
FILES=30
BUFFERS=20
DOS=HIGH, UMB*1
COUNTRY=081, 932, C:\DOS\CONTRY.SYS
SHELL=C:\DOS\COMMAND.COM /P /E:512
DEVICE=C:\DOS\$FONT.SYS /24=ON
DEVICE=C:\DOS\HIMEM.SYS
DEVICE=C:\DOS\EMM386.EXE 4096 RAM FRAME=E000*2
DEVICE=C:\DOS\DISP.SYS
DEVICE=C:\DOS\$IAS.SYS
DEVICE=C:\DOS\ANSI.SYS /X

[AUTOEXEC.BAT]
@ECHO OFF
C:\DOS\SMARTDRV.EXE /X*3
PATH C:\WINDOWS;C:\DOS;C:\
SET TEMP=C:\TMP
SET DOSDIR=C:\DOS
```

*1: XMS manager (required for EMM386/SMARTDRV)
*2: Reserve 4 MB for EMS
*3: Enable the disk cache.

(c) Batch files

To start up the GPP functions, copy the following GPPQ.BAT batch file to the root directory.
The drive name is replaced by the drive name designated during system installation.

```
[GPPQ.BAT]
C:
CD  C:\GPPQ\SYS
GPPQ.EXE
```

(3) Cautions about environment set-up

(a) A minimum of 540 kbytes free space is required in main memory. Execution speed increases if more free space is available.

(b) The conditions that have to be set in the CONFIG.SYS file to start up the GPP functions are as follows:
FILES: 30 or more
No other specific settings are required.

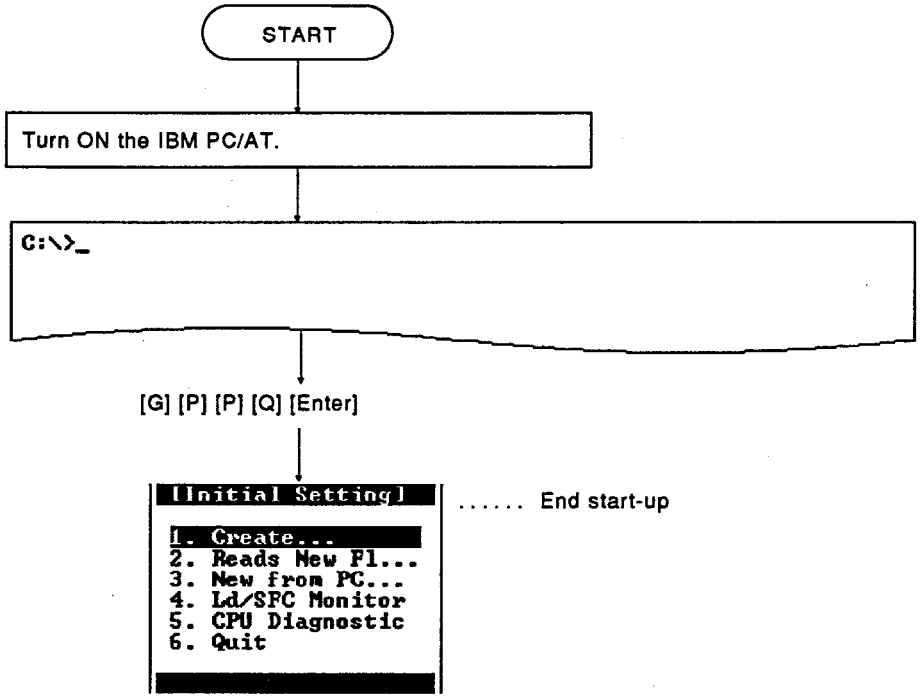
(c) It is not necessary to incorporate PRINT.SYS (printer driver) and RSDRV.SYS (PC communications driver) into CONFIG.SYS. They are included in the GPP functions.

(d) Incorporate PRINT.SYS into CONFIG.SYS to make hard copies.

(e) More than 4 MB of EMS memory is required.
Configure your memory settings according to your system.

3.3.3 Starting the GPP functions

This section describes how to start up the GPP functions installed on the IBM PC/AT hard disk.



3.3.4 Corrective action when an error occurs

(1) Corrective action if the GPP functions cannot be installed.

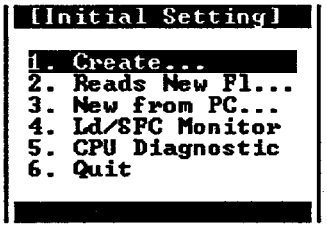
A message during installation indicates that the floppy disk is not a master floppy disk.	
Cause	• Installation from a copied floppy disk.
Corrective action	• The system floppy disk for the GPP functions is copy-protected. Correct installation is not possible from a copied floppy disk. Reinstall from the master floppy disk.
Installation stops and a message indicates that insufficient free space remains on the hard disk.	
Cause	• The hard disk is too full to install the GPP functions.
Corrective action	• Installation of the GPP functions requires at least 8 MB free space on the hard disk. Move user data to floppy disks to ensure sufficient free space on the hard disk.
A message indicates that normal installation was not possible.	
Cause	• The system floppy disk or hard disk is damaged.
Corrective action	• Replace the system floppy disk or hard disk.
A message during installation indicates that the directory could not be created.	
Cause	• Incorrect directory designation.
Corrective action	• Directory designation cannot contain special symbols, such as: "\", ":", or ">". Designate the directory using alphanumeric characters.

(2) Corrective action if the GPP functions will not start up

On start-up, a message indicates that the functions are not correctly installed.	
Cause	• Functions were installed with the copy command instead of the INSTHD command.
Corrective action	• The functions cannot be correctly installed using the copy command. Install the GPP functions again by executing the "INSTHD" command.
A message indicates that the main memory area is insufficient on start-up, or system goes out of control during GPP function operation.	
Cause	• Insufficient free space in main memory (540 k byte min.) • The "BUFFERS" item is set to 20 or less in CONFIG.SYS.
Corrective action	• Use the MEM or CHKDSK command to check the main memory free space. Delete unwanted drivers if the main memory free space is insufficient (see the description of CONFIG.SYS in Section 3.3.2). • Use the TYPE command to check the CONFIG.SYS settings. How to check the CONFIG. SYS settings. C:\TYPE CONFIG.SYS FILES=30 BUFFERS=20 ← BUFFERS size ⋮ If the BUFFERS size is set to 20 or less, use the EDIT, EDLIN, or TE command to modify the CONFIG.SYS contents.

3.4 Creating Programs

Use screen operations after starting GPPQ.



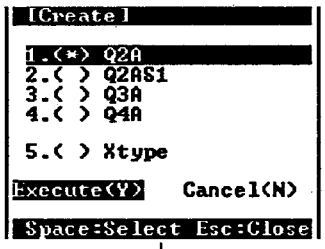
"Initial Setting screen"

[1] Select "Create..."

To Section 3.4.1

3.4.1 Selecting CPU type used and setting file

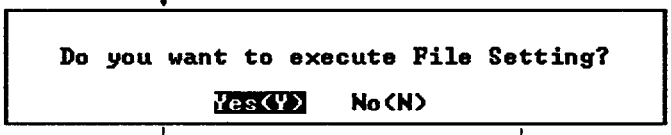
The following screen is displayed after "Create..." is selected above. Press the key corresponding to the CPU type used. In this example, Q3ACPU is selected.



"CPU type select screen"

[3] Select "Q3ACPU"

[Enter]



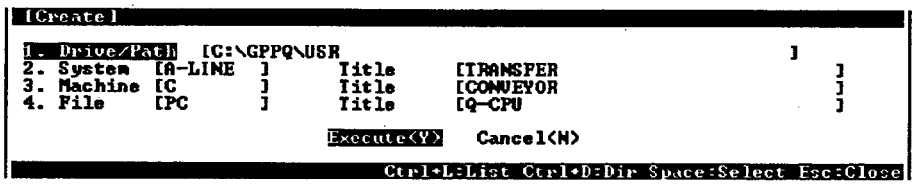
[Y] Select "Yes"

[N] Select "No"

1)

2)

1) 2)



"File setting screen"

The characters and symbols used in the system name, machine name, file name, and titile are unrestricted.

Drive and path are default values.

[↓] [SP] [A] [-] [L] [I] [N] [E] [Tab]

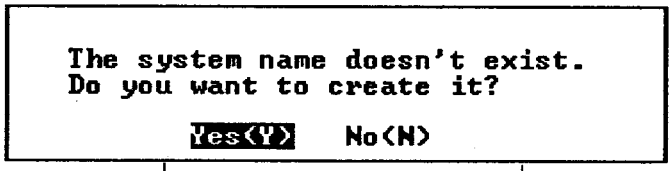
[T] [R] [A] [N] [S] [F] [E] [R] [Enter]

[SP] [C] [Tab]

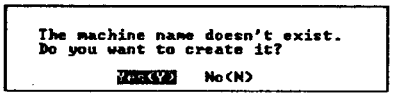
[C] [O] [N] [V] [E] [Y] [O] [R] [Enter]

[SP] [P] [C] [Tab] [Q] [-] [C] [P] [U] [Enter]

[Y] Select "Execute"



The following dialog box is displayed if the system name already exists.

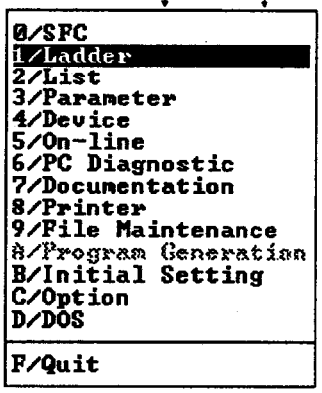


[Y] Select "Yes"

[N] Select "No"

Return to the CPU type select screen.

"Processing" is displayed.



"Mode Select screen"

3.4.2 Programming in ladder mode

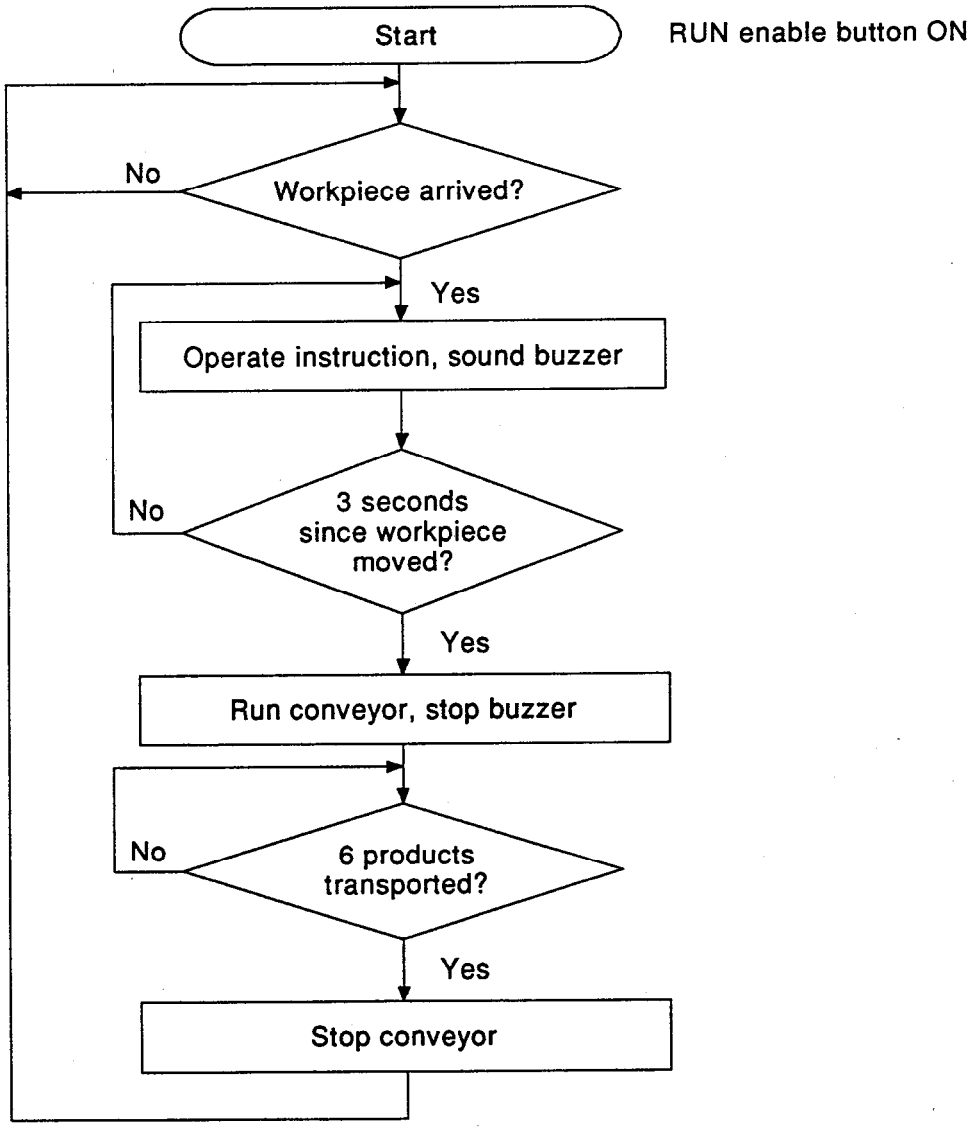
The procedure for programming in the ladder mode is shown below.

(1) About the program

The program created to automatically run the conveyor is shown in Fig. 3.2.

(2) Description of operation

The operation of the program to automatically run the conveyor is outlined below.
When the workpiece (pallet) stops in front of the conveyor, a warning buzzer sounds for 3 seconds and the conveyor automatically starts. The conveyor stops after it has transported 6 products onto the workpiece. The workpiece is automatically transferred to the next process, the products unloaded, and then returns to the position in front of the conveyor.
The flowchart for this procedure appears as follows.



(3) Allocation of devices

- X0.... RUN enable button
- X1.... STOP button
- X2.... Work sensor
- X3.... Product sensor
- Y70... Buzzer output
- Y71... Conveyor Operation
- M0 ..... Operation command
- M1 ..... Operation record
- M10 .... Operating enable
- M11 .... Record operation enable
- T0..... Warning timer
- C0..... Product count

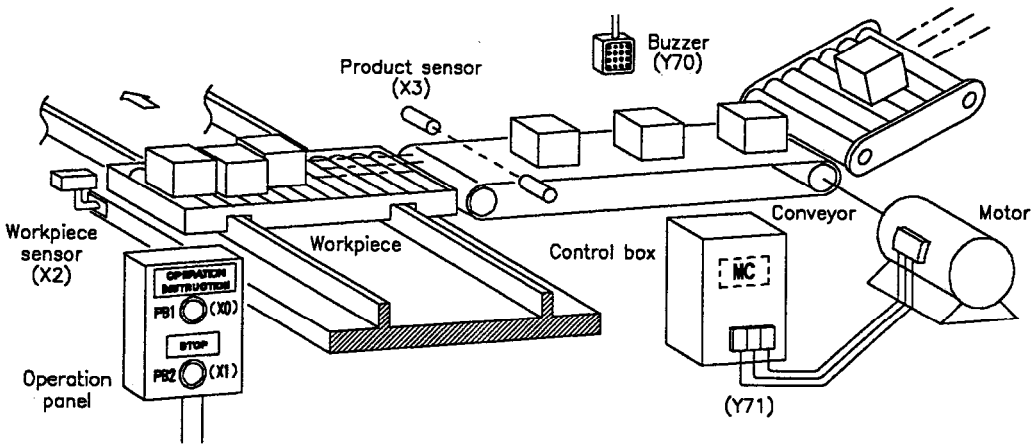


Fig. 3.1 Automatic Transport Equipment

(4) Program for automatic conveyor operation

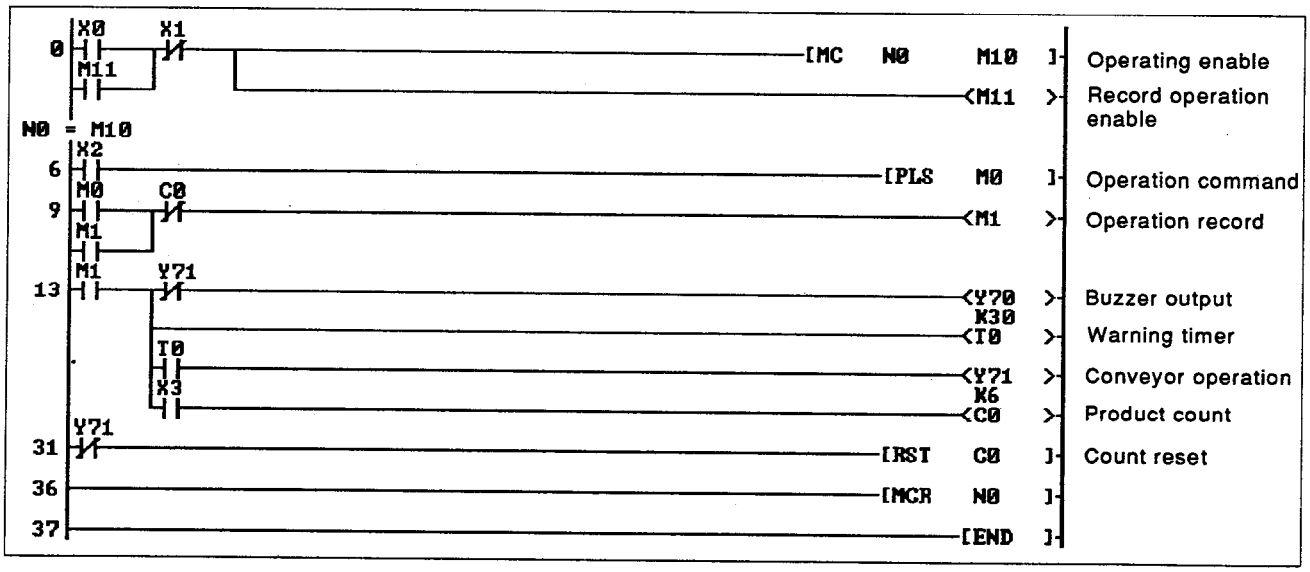
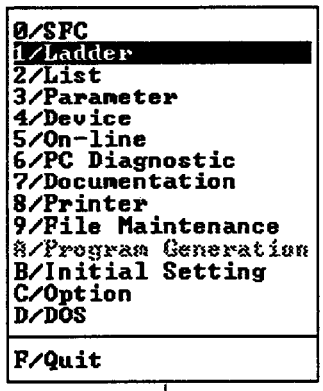


Fig. 3.2 Program for Automatic Conveyor Operation

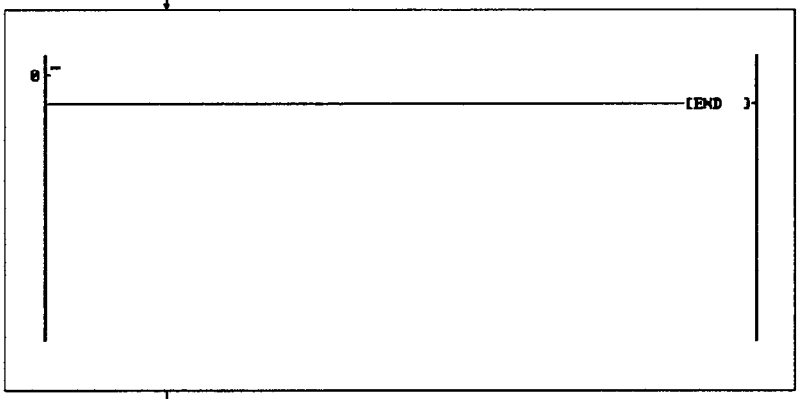
(5) Programming procedure

The program is created using ladder symbols.
Operation starts from the mode select screen.



"Mode Select screen"

[1] Select "Ladder"

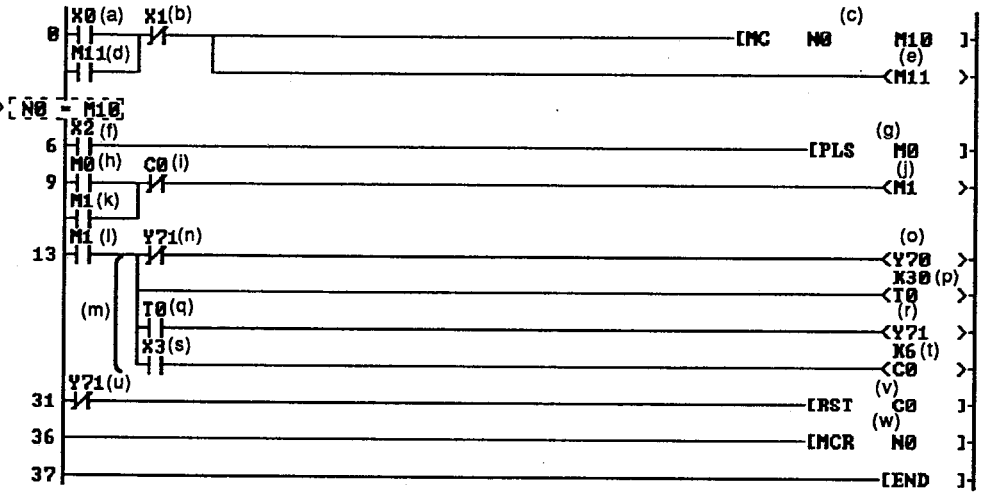


"Ladder Write
(insert mode) screen"

1)

1)

It is unnecessary to write MC contacts in the bus line. They are displayed on the screen after pressing [F4] (Convert) key followed by [F2] key to select the read mode.



[Key input sequence (a) to (w)]

(a) [F5] [X] [0] [Enter]
(⊣⊢)

(b) [F6] [X] [1] [Enter]
(⊣/⊢)

(c) [F8] [M] [C] [SP] [N] [0] [SP] [M] [1] [0] [Enter]
(⊏⊐)

(d) [Shift] + [F5] [M] [1] [1] [Enter]
(⊣⊢)

(e) [→] [↑] [Insert] [Shift] + [F9] [Enter] [↓] [Insert]
(overwrite) (|) (insert)
[F7] [M] [1] [1] [Enter] . . . . . Create vertical line in the overwrite mode
(⊣⊢)

(f) [F5] [X] [2] [Enter]
(⊣⊢)

(g) [F8] [P] [L] [S] [SP] [M] [0] [Enter]
(⊏⊐)

(h) [F5] [M] [0] [Enter]
(⊣⊢)

(i) [F6] [C] [0] [Enter]
(⊣/⊢)

(j) [F7] [M] [1] [Enter]
(⊣⊢)

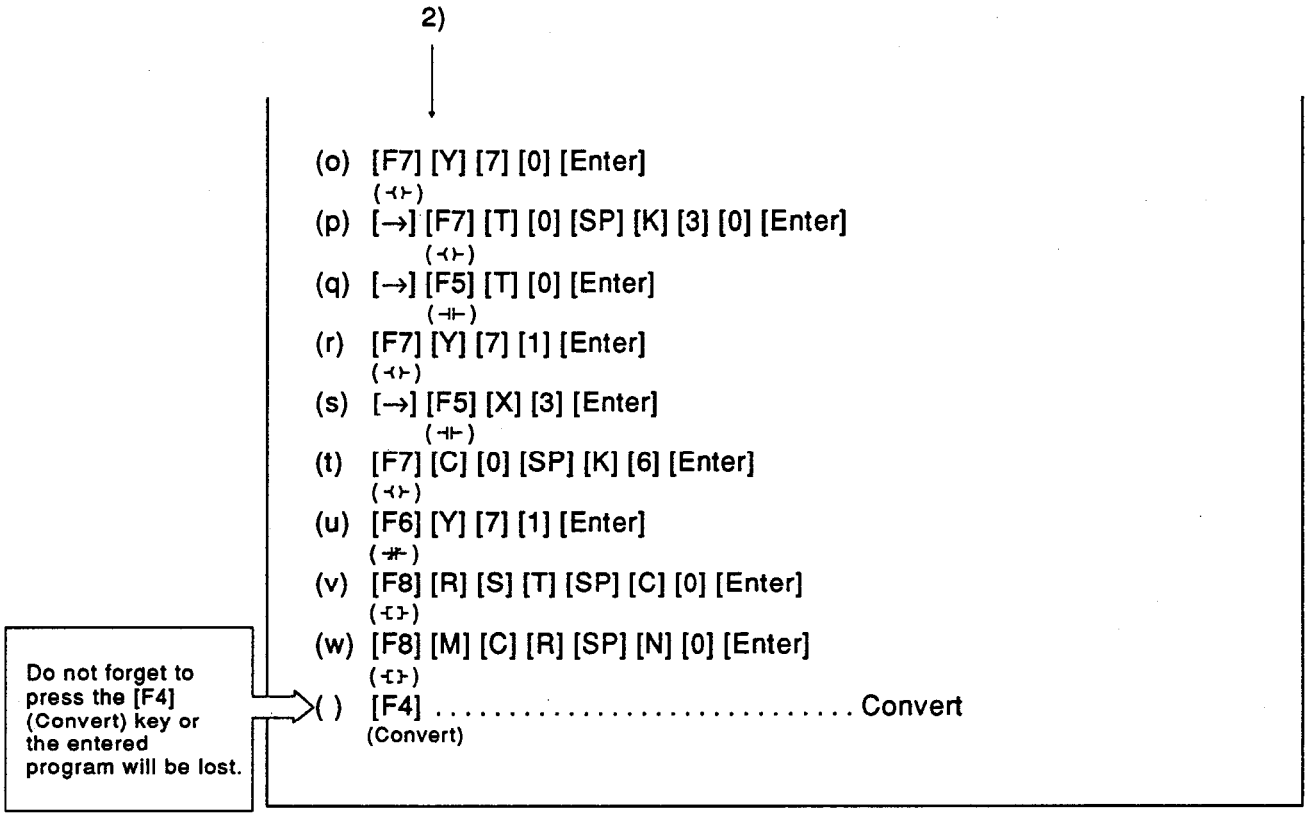
(k) [Shift] + [F5] [M] [1] [Enter]
(⊣⊢)

(l) [←] [Shift] + [F9] [Enter] [↓] [F5] [M] [1] [Enter]
(|) (⊣⊢)

(m) [Shift] + [F9] [3] [Enter] . . . . . Write continuous vertical lines
(|)

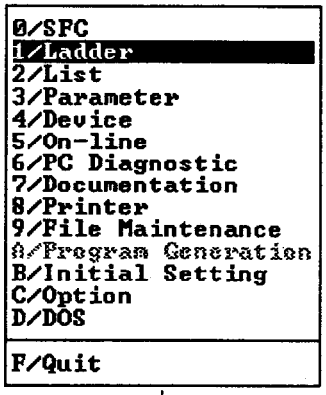
(n) [F6] [Y] [7] [1] [Enter]
(⊣/⊢)

2)


2)
(o) [F7] [Y] [7] [0] [Enter]
(⊣⊢)
(p) [→] [F7] [T] [0] [SP] [K] [3] [0] [Enter]
(⊣⊢)
(q) [→] [F5] [T] [0] [Enter]
(⊣⊢)
(r) [F7] [Y] [7] [1] [Enter]
(⊣⊢)
(s) [→] [F5] [X] [3] [Enter]
(⊣⊢)
(t) [F7] [C] [0] [SP] [K] [6] [Enter]
(⊣⊢)
(u) [F6] [Y] [7] [1] [Enter]
(⊣/⊢)
(v) [F8] [R] [S] [T] [SP] [C] [0] [Enter]
(⊣↓⊢)
(w) [F8] [M] [C] [R] [SP] [N] [0] [Enter]
(⊣↓⊢)
( ) [F4] ..... Convert
(Convert)
Do not forget to press the [F4] (Convert) key or the entered program will be lost.

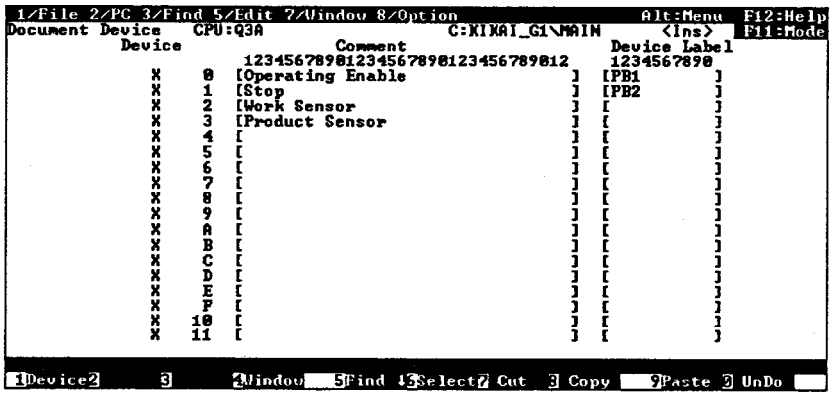
3.4.3 Appending comments to devices

Operation from the mode select screen.
To conduct further operations after creating a sequence program, press [F11] key to display the mode select screen.



"Mode Select screen"

[7] Select "Documentation"



"Documentation screen"

[Input X0 comment]
[O] [p] [e] [r] [a] [t] [i] [n] [g] [E] [n] [a] [b] [l] [e]

[Input X0 device label]
[Tab] [P] [B] [1]

[Input X1 comment]
[Tab] [S] [t] [o] [p]

[Input X1 device label]
[Tab] [P] [B] [2]

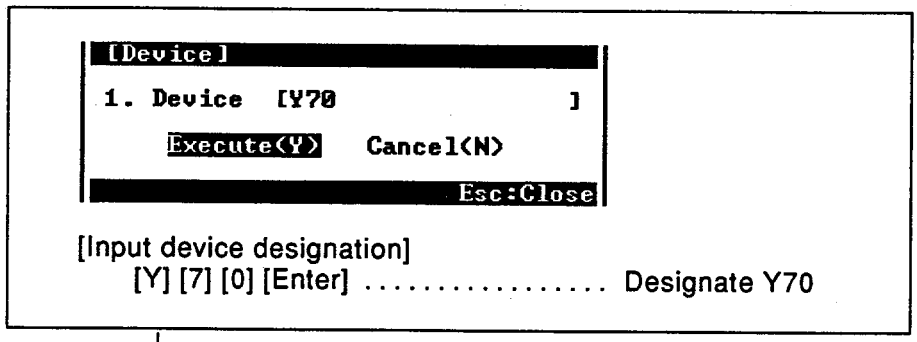
[Input X2 comment]
[Tab] [W] [o] [r] [k] [S] [e] [n] [s] [o] [r] [Enter]

Similarly, create the X3 comment

1)

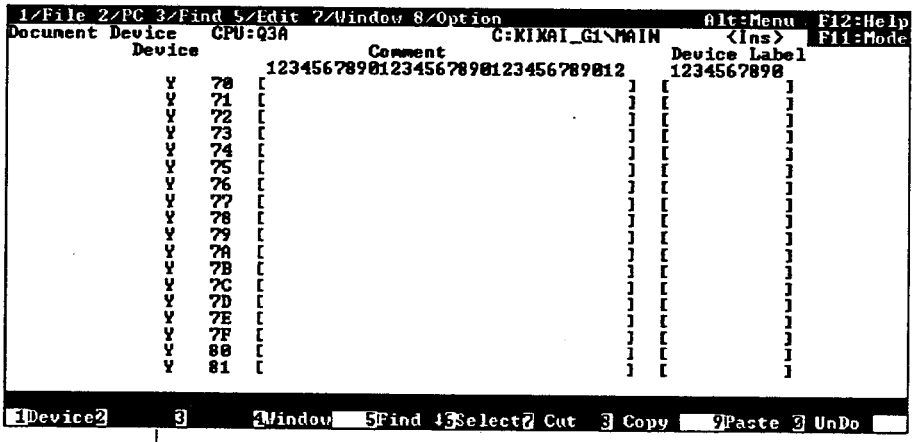
1)

[F1] Device



[Input device designation]
[Y] [7] [0] [Enter] ..... Designate Y70

[Y] Execute



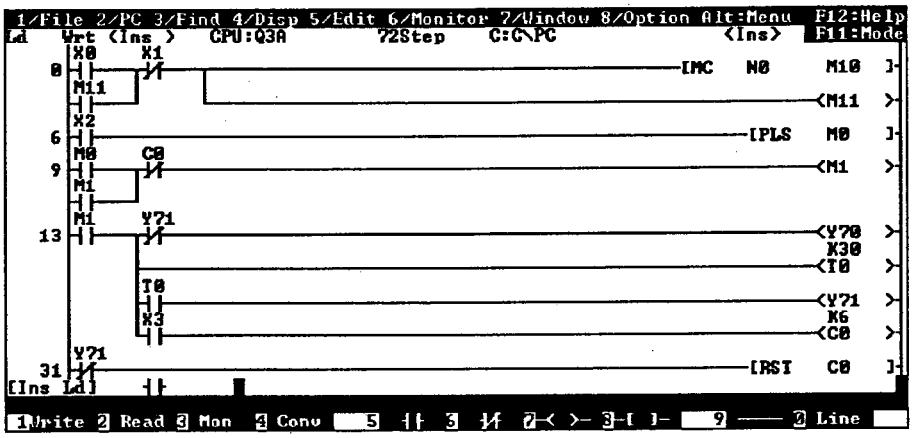
"Documentation screen"

Repeat the procedure on the previous page for X device comments to create the Y device comments.
Then, follow the same procedure to create the comments for M, T, and C devices.

End

3.4.4 Saving program to hard disk

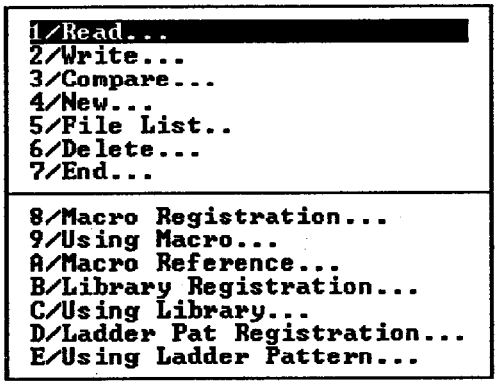
Save the sequence program created in Section 3.4.2 to the hard disk.
This operation starts from the screen displayed after conversion.



"Ladder Write (insert) screen"

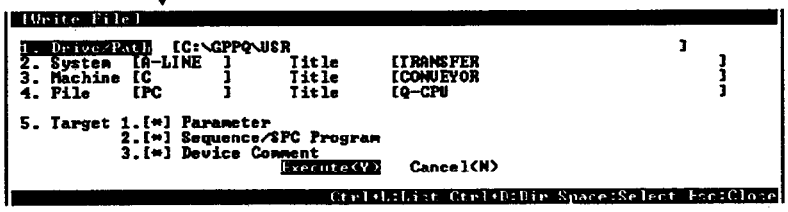
[Alt] Menu

[1] Select "File"



"File access screen"

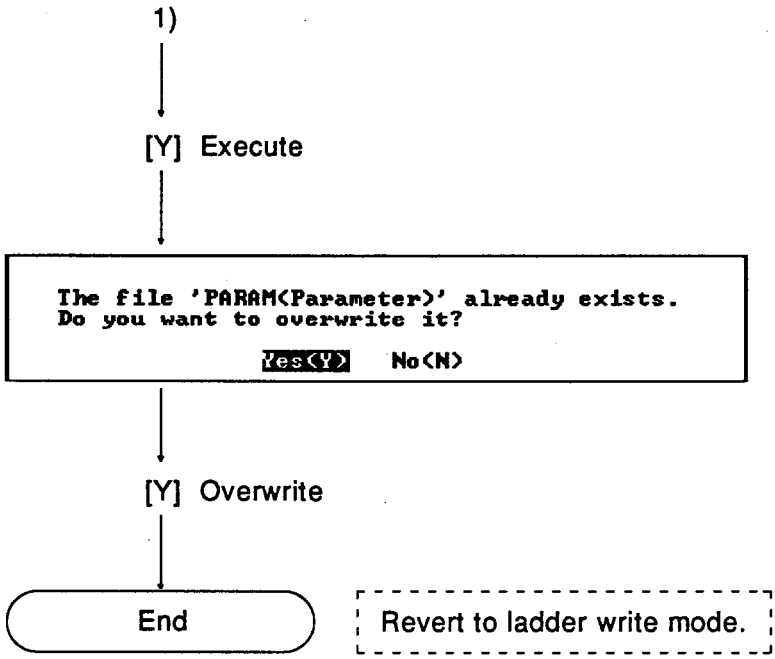
[2] Select "Write"



"Write File screen"

The system name, machine name, and file name set in Section 3.4.1 are automatically displayed.

1)



POINT

The following system name, machine name, and file name defaults are used if no file is set:

- System name [SYSTEM]
- Machine name [KIKAI_G1]
- File name [MAIN]

The default drive and path is "C:\GPPQ\USR".

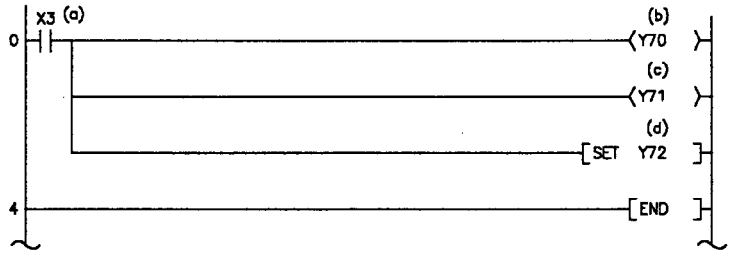
3.4.5 Convenient functions for programming

The parallel coil, line, and cut and paste editing are convenient functions for programming in the ladder mode.

(1) Parallel coils

When coil-equivalent instructions (⊣⊢, ⊣↕⊢) are input, the vertical lines and parallel coils are created automatically to create a ladder block. The ladder block below was created using parallel coils.

[Operations in the ladder write (insert) mode.]



[Key input sequence (a) to (d)]

(a) [F5] [X] [3] [Enter]
(⊣⊢)

(b) [F7] [Y] [7] [0] [Enter]
(⊣⊢)

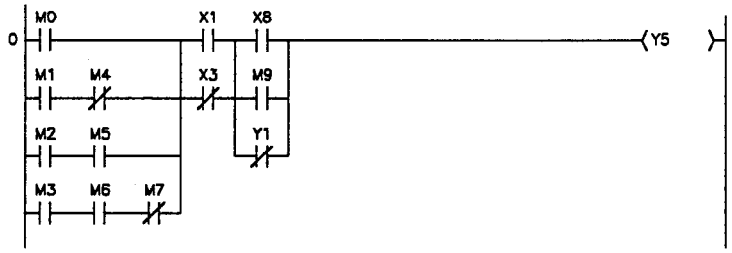
(c) [→] [↑] [F7] [Y] [7] [1] [Enter]..... Write parallel coils
(⊣⊢)

(d) [→] [↑] [F8] [S] [E] [T] [SP] [Y] [7] [2] [Enter]
.....(⊣↕⊢)..... Write parallel coils

(e) [F4] ..... Convert
(Convert)

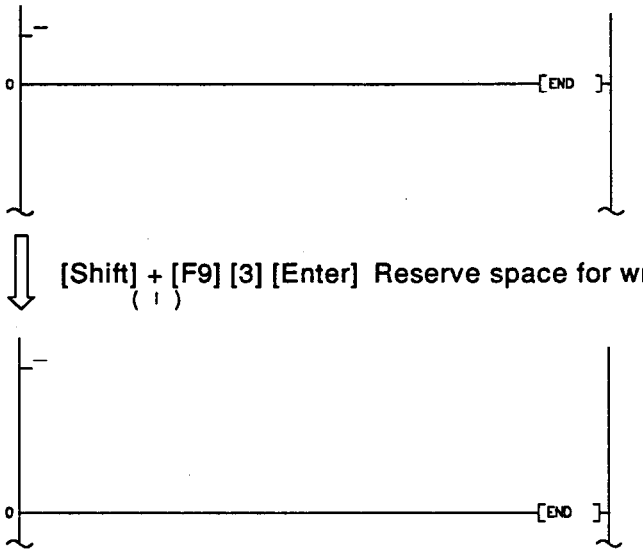
(2) Lines

A ladder block can be created using lines.
The ladder block below was created using lines.

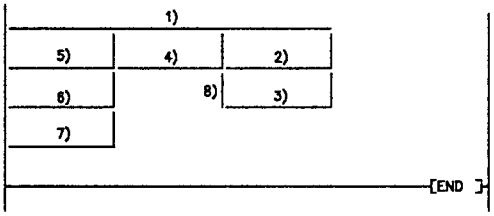


(a) Provide space before the END instruction for the ladder.

Operations in the ladder write (insert) mode.



(b) Draw lines as shown in the diagram below.



[Key input sequence 1) to 8)]

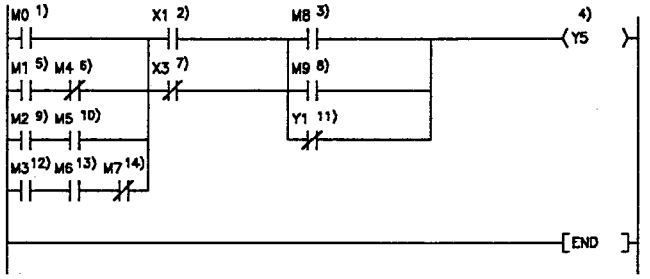
1) [F10] [Enter] [→] ..... [→] [Enter]
(Line) (Start) (Designate — line, repeat 9 timese) (End)
2) [Esc] [Esc] [Enter] [↓] [←] [←] [←] [Enter]
(Continue) (Start) (Designate ⌟ line) (End)
3) [→] [→] [→] [Enter]
(Designate copy destination)
4) [↑] [←] [←] [←] [Enter]
(Designate copy destination)
5) [←] [←] [←] [Enter]
(Designate copy destination)
6) [↓] [Enter]
(Designate copy destination)
7) [↓] [Enter]
(Designate copy destination)

Copy line (2)

8) [Esc] [Esc] [→] [→] [→] [Enter] [↑] [Enter]
(Continue) (Create position) (Start) (End)

(c) Write contacts and coils on lines

Operations in the ladder write (overwrite) mode.
To continue after operation (b), press the [Esc] key three times to cancel the line mode and press the [Insert] key to select the ladder write (overwrite) mode.



[Key input sequence 1) to 14)]

1) [F5] [M] [0] [Enter]
   (⊣⊢)
2) [→] [→] [F5] [X] [1] [Enter]
   (⊣⊢)
3) [→] [→] [F5] [M] [8] [Enter]
   (⊣⊢)
4) [→] [→] [F7] [Y] [5] [Enter]
   (⊣/⊢)
5) [F5] [M] [1] [Enter]
   (⊣⊢)
6) [F6] [M] [4] [Enter]
   (⊣/⊢)

11) [→] . . . . . [→] [F6] [Y] [1] [Enter]
    (Input 5 times)
12) [↓] [←] . . . . . [←] [F5] [M] [3] [Enter]
    (Input 7 times)
13) [F5] [M] [6] [Enter]
    (⊣⊢)
14) [F6] [M] [7] [Enter]
    (⊣/⊢)
) [F4] . . . . . Conversion
  (Convert)

COMMENT

It is also possible to first draw the contact and coil instruction symbols and subsequently draw the lines. In this case, lines cannot overwrite instruction symbols.

POINTS

- Horizontal, vertical, └── line, and ──┘ lines can be drawn by a single operation. Rectangles, ┐ lines, and ┌ lines cannot be drawn using a single operation.
- Lines can be copied to any position by designating the destination.
- Lines and contact positions can be drawn as required. Line lengths and contact positions are corrected during conversion.

(3) Cutting and pasting

Parts of ladder programs can be cut and pasted.
The cutting and pasting operations can be conducted in the following units:
- ladder lines
- ladder blocks
- ladder program sections

(a) Cutting and pasting ladder lines

These operations are possible by selecting Edit from the menu or by selecting the ladder edit (insert/overwrite) mode with the functions keys.

1) Operations by selecting Edit from the menu

Select the ladder write (insert/overwrite) mode. (The operations are also possible in the ladder edit (insert/overwrite) mode.)

- Cut
  [Alt] → [5] → [1] → [↑]/[↓] → [Enter]
  (Menu) (Edit) (Cut) (Designate start)
  → [↓]/[↑] → [Enter]
  (Designate range) (Cut)
- Paste
  [Alt] → [5] → [3] → [↑]/[↓]
  (Menu) (Edit) (Paste) (Designate paste destination)
  → [Enter]
  (Paste)
- Conversion
  [F4]
  (Convert)

2) Operations by selecting the ladder edit (insert/overwrite) mode with the functions keys

- Cut
  [Shift] + [F1] → [↑]/[↓] → [Shift] + [F6]
  (Edit) (Designate start) (Select line)
  → [↑]/[↓] → [F7]
  (Designate range) (Cut)
- Paste
  [↓]/[↑] → [F9]
  (Designate paste destination) (Paste)
- Conversion
  [F1] → [F4]
  (Write) (Convert)

(b) Cutting and pasting ladder blocks

Operation by selecting Edit from the menu.

Select the ladder read mode.

- Cut
  [Alt] → [5] → [1] → [↑]/[↓] → [Enter]
  (Menu) (Edit) (Cut) (Designate start)
  → [↓]/[↑] → [Enter]
  (Designate range) (Cut)
- Paste
  [Alt] → [5] → [3] → [↑]/[↓]
  (Menu) (Edit) (Paste) (Designate paste destination)
  → [Enter]
  (Paste)
- Conversion
  Not required (Cut and paste changes are directly reflected in memory when [Enter] key is pressed.)

REMARK

The "Close up" setting affects the program after cut and paste, as follows:

"Close up" set..... The part where a program is cut is closed up.
(The number of program steps remains unchanged after the editing.)
The part where a program is cut is not closed up but filled with NOP instructions. (The number of program steps after the editing increases by the number of steps cut and pasted.)

Method for closing up ..... [Alt] → [8] → [4] → [1] → [1] Close up the program after conversion.
[2] Do not close up the program after conversion.

The default is "Do not close up".

(c) Cutting and pasting ladder program sections

Operations by selecting the ladder edit (insert/overwrite) mode with the function keys

- Cut: [Shift] + [F1] → [↑]/[↓]/[←]/[→] → [F6]
  (Edit) (Designate cut position) (Select)
  → [↑]/[↓]/[←]/[→] → [F7]
  (Designate cut range) (Cut)
- Paste: [↑]/[↓]/[←]/[→] → [F9]
  (Designate cut position) (Paste)
- Conversion: [F1] → [F4]
  (Write) (Convert)

After cutting a pasting a program section, write the vertical and horizontal lines in the program section before conversion. Otherwise, a message will be displayed indicating that the ladder program is incomplete.

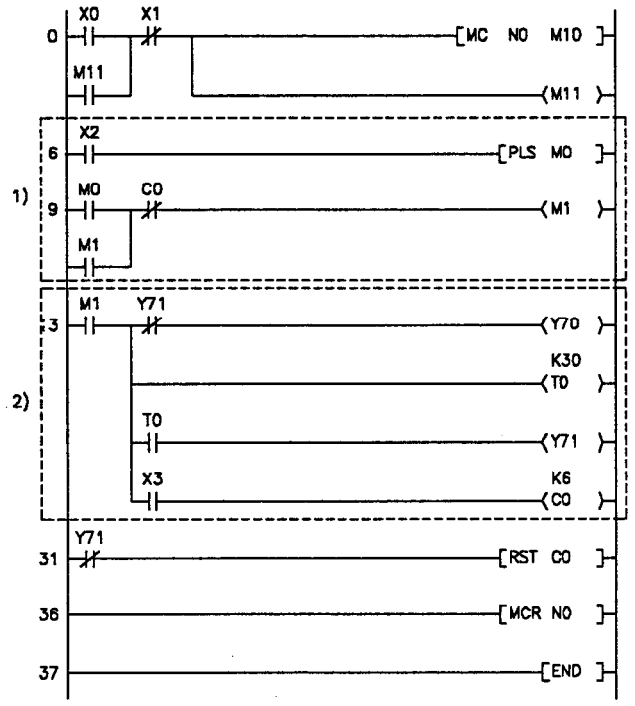
COMMENT

- A cut operation alone is equivalent to deleting ladder blocks, except that the cut ladder blocks are stored in buffer memory.
- A paste operation after a cut operation is equivalent to copying the cut ladder blocks from the buffer memory to the designated position.
- After a cut operation, pasting can be carried out any number of times. This allows the cut ladder blocks to be inserted into any number of programs.
- Cutting and pasting are also possible for instruction lists.

(d) Examples of cutting and pasting

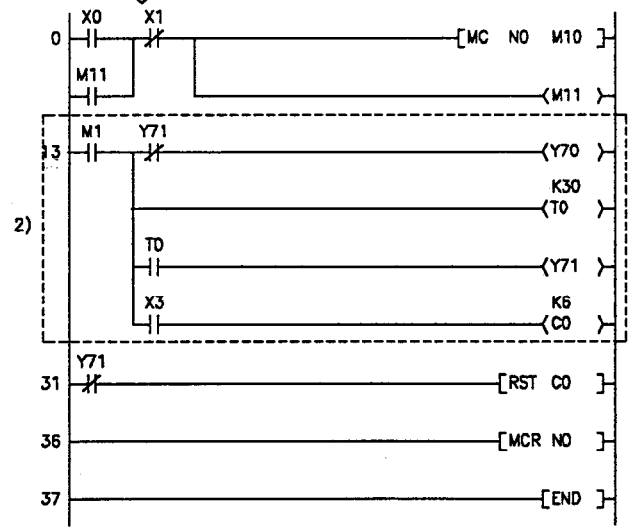
As an example, cutting and pasting will be shown for the ladder program below. All operations act on units of ladder lines.
The three lines at 1) are cut and pasted after ladder block 2).
The operations are carried out in this example by selecting Edit from the menu.

Operations in the ladder write (insert/overwrite) mode.



"Ladder write mode screen"

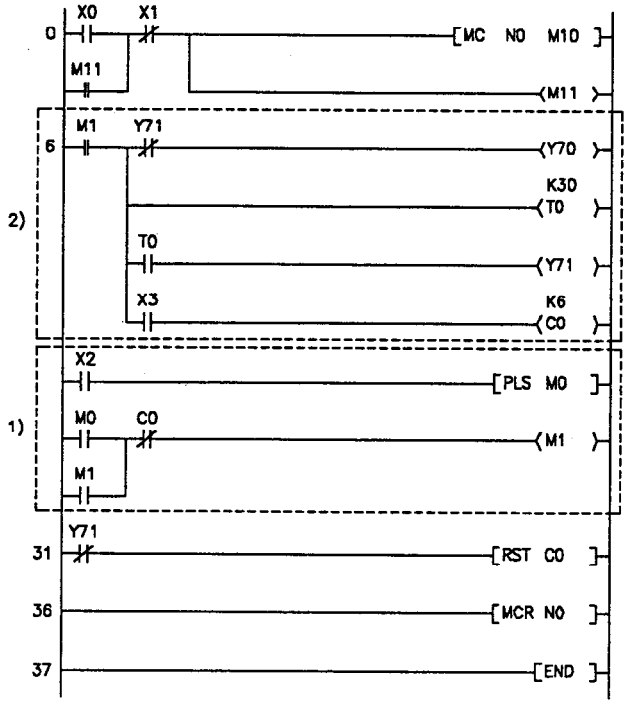
[Alt] [5] [1] [↓] [↓] [Enter] [↓] [↓] [Enter] . . The three lines at 1) are cut.
(Menu)(Edit)(Cut) (Designate cut start) (Designate range) (Cut)



(1)

(1)

[Alt] [5] [3] [↓] [↓] [Enter] ..... Paste after ladder block 2).
(Menu) (Edit) (Paste) (Designate paste destination) (Paste)



[F4]..... Convert
(Convert)

Editing complete

COMMENT

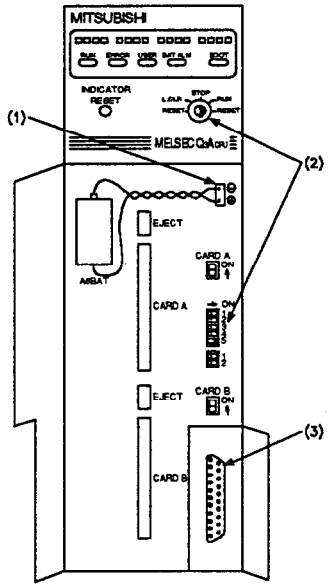
Press [F10] (Undo) key in the ladder write (insert/overwrite) mode during a cut or paste operation to revert to the previous status (to change the cut or paste destination). Only the previous operation can be undone. The undo operation is not possible after conversion. No undo function is available for cut and paste operations on units of ladder blocks in the ladder read mode.

3.5 Operating the CPU for Monitoring and Testing

After setting the switches and internal clock, the program created in Section 3.4.2 is written to the CPU and the CPU is operated to monitor and test the program.

3.5.1 Preparations for operating the CPU

Connect the connectors and set the switches (1) to (3) in the diagram below.



(1) Connect the battery

The battery is unconnected when the CPU is shipped from the factory. Connect the battery, ensuring that the polarity (positive and negative) is correct.

(2) Set the switches

The system setting DIP switches and the RUN/STOP key switch have to be set.

(a) Setting the system setting DIP switches

Set DIP switch #4 to the OFF position.
[DIP switch #4 disables writing to the CPU and control instructions.
It must be set OFF to remove this protection.
All other DIP switches can be in either position.]

(b) Setting the RUN/STOP key switch

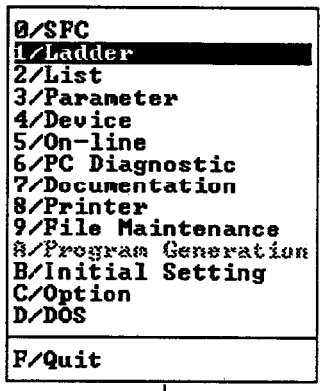
Set to the STOP position.

(3) Connect the RS-422 cable

Connect the cable to the RS-232C/RS-422 converter connected to the IBM PC/AT RS-422 connector.

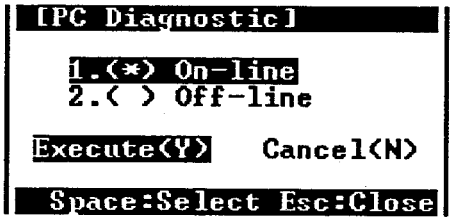
(4) Set the QnACPU internal clock

Set the QnACPU internal clock to the current time using operations from the mode select screen.

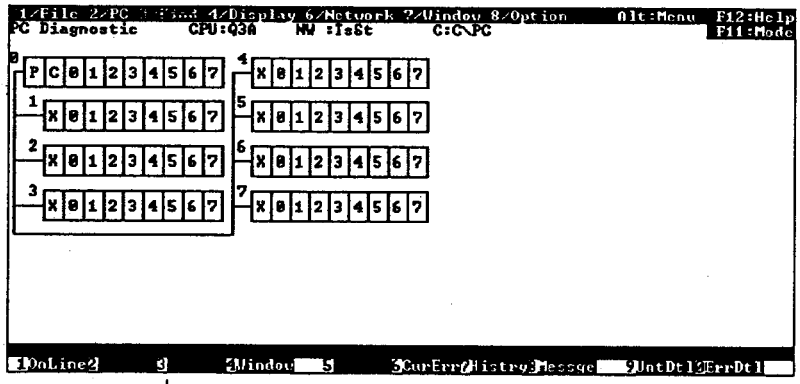


"Mode Select screen"

[6] Select "PC Diagnostic"



[Y] Execute



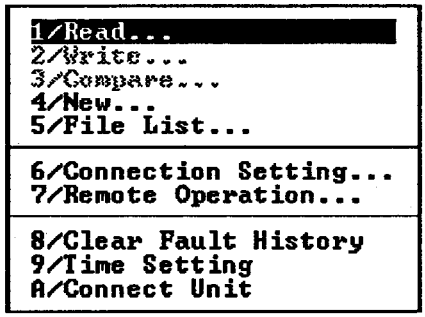
"PC Diagnostic screen"

1)

1)

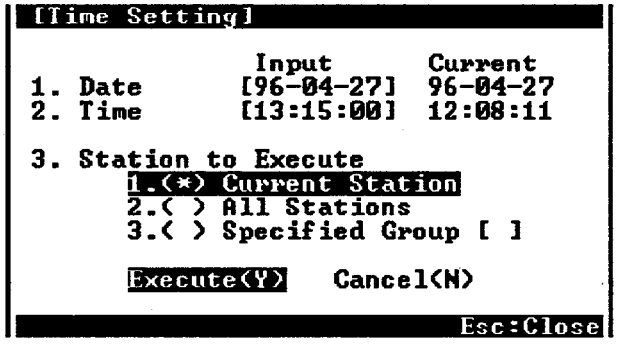
[Alt] Menu

[2] Select "PC"



"PC communications screen"

[9] Select "Time Setting"



"Time Setting screen"

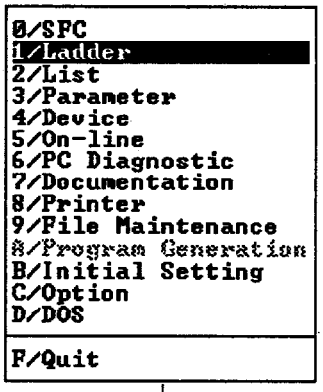
Setting the date
[SP] [9] [6] [-] [0] [4] [-] [2] [7] [Enter]

Setting the time
[SP] [1] [3] [Shift] + [:] [1] [5] [Shift] + [:] [0] [0] [Enter]

Set the time and date for the current designated station (connected PC).
[SP] ([1]) [Y]

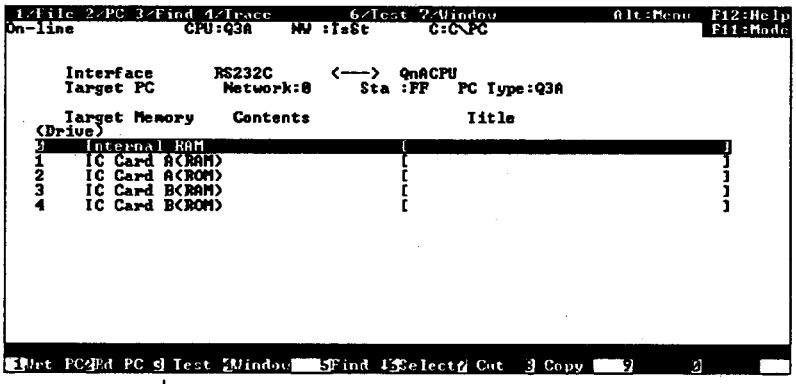
(5) Format QnACPU internal RAM

Format the QnACPU internal RAM using operations from the mode select screen.



"Mode Select screen"

[5] Select "Online"

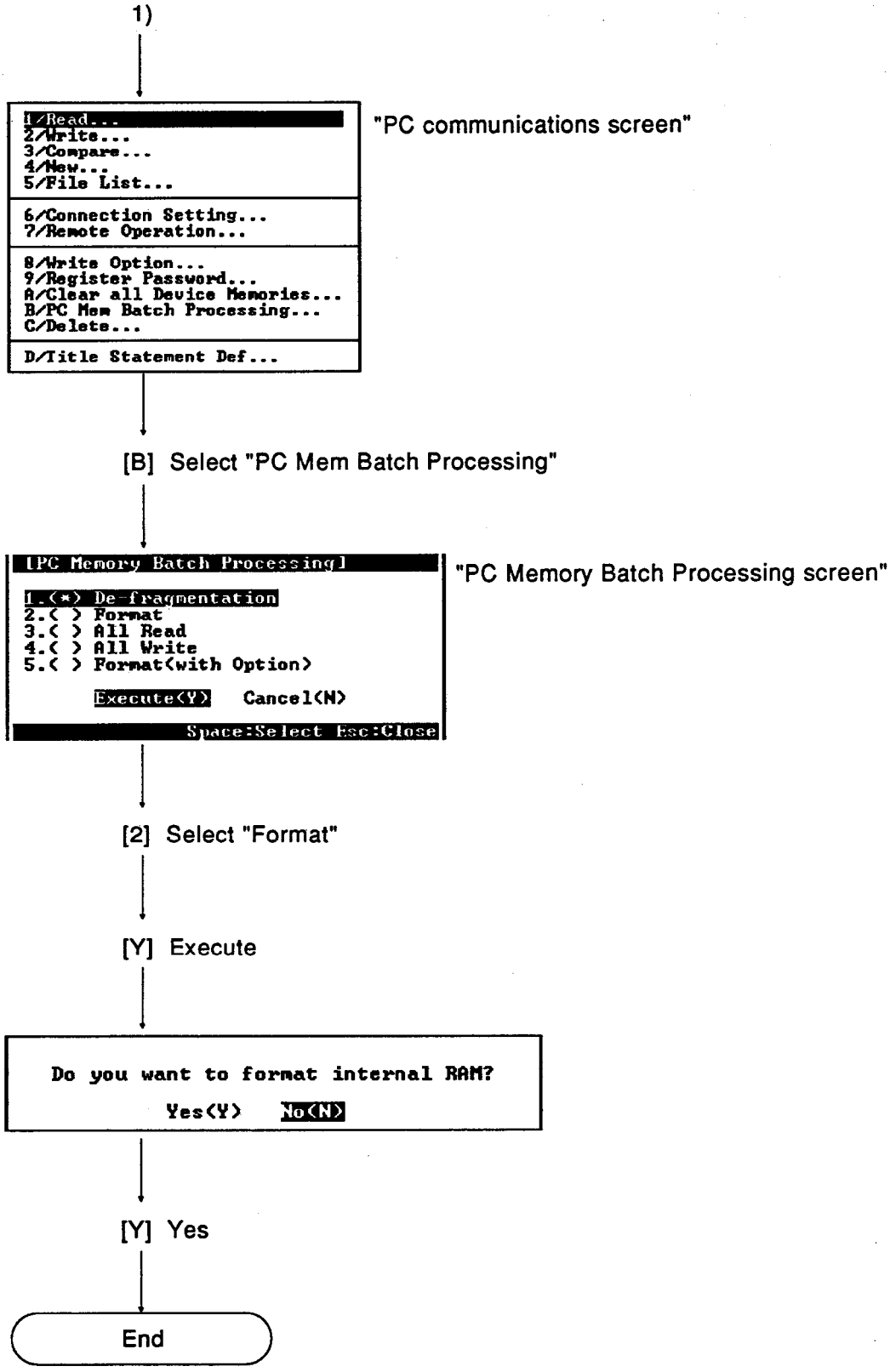


"Online memory select screen"

[Alt] Menu

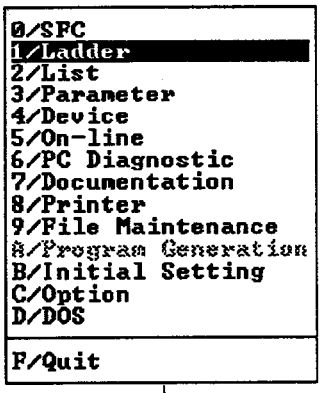
[2] Select "PC"

1)


1)
1/Read...
2/Write...
3/Compare...
4/New...
5/File List...
6/Connection Setting...
7/Remote Operation...
8/Write Option...
9/Register Password...
A/Clear all Device Memories...
B/PC Mem Batch Processing...
C/Delete...
D/Title Statement Def...
"PC communications screen"
[B] Select "PC Mem Batch Processing"
[PC Memory Batch Processing]
1.(*) De-fragmentation
2.( ) Format
3.( ) All Read
4.( ) All Write
5.( ) Format<with Option>
Execute<Y> Cancel<N>
Space:Select Esc:Close
"PC Memory Batch Processing screen"
[2] Select "Format"
[Y] Execute
Do you want to format internal RAM?
Yes<Y> No<N>
[Y] Yes
End

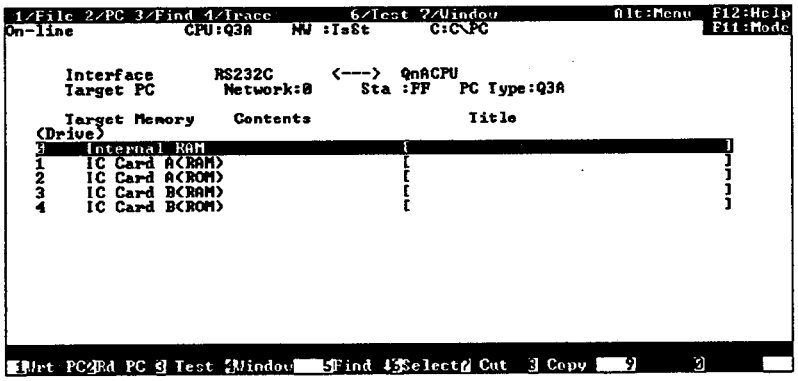
(6) Clear the QnACPU latch devices

Clear the QnACPU latch devices using operations from the mode select screen.



"Mode Select screen"

[5] Select "Online"



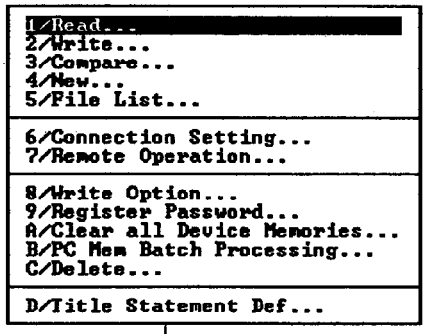
"Online memory select screen"

[Alt] Menu

[2] Select "PC"

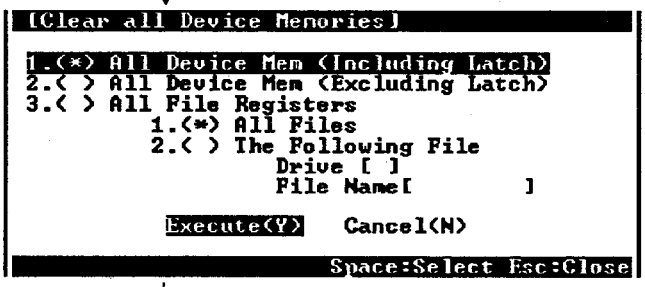
1)

1)



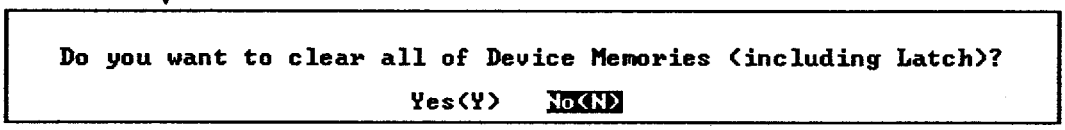
"PC communications screen"

[A] Select "Clear all Device Memories"



"Clear all Device Memories screen"

[Y] Select "All Device Mem (Including Latch)" and execute

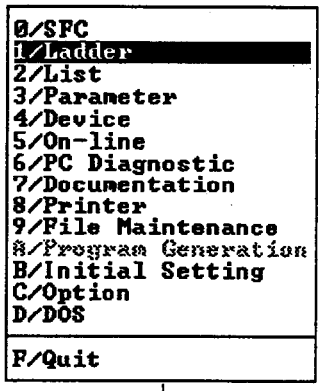


[Y] Clear all

End

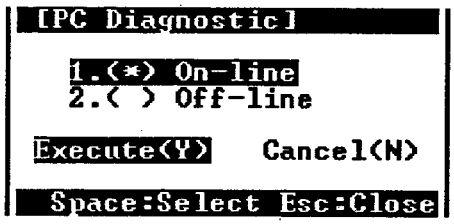
(7) Clear the QnACPU fault history

Clear the fault history data in the QnACPU internal RAM memory using operations from the mode select screen.

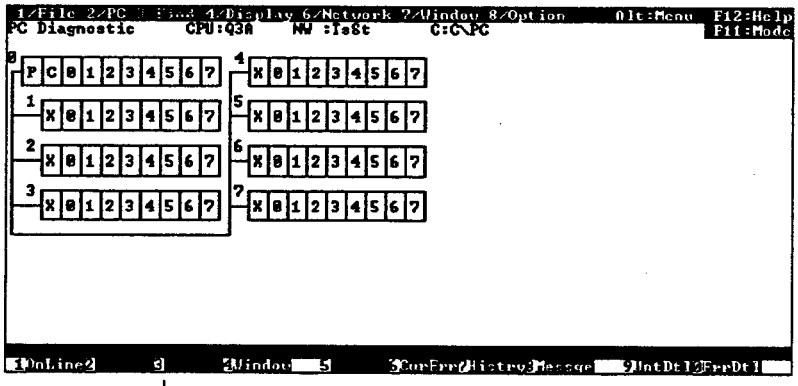


"Mode Select screen"

[6] Select "PC Diagnostic"

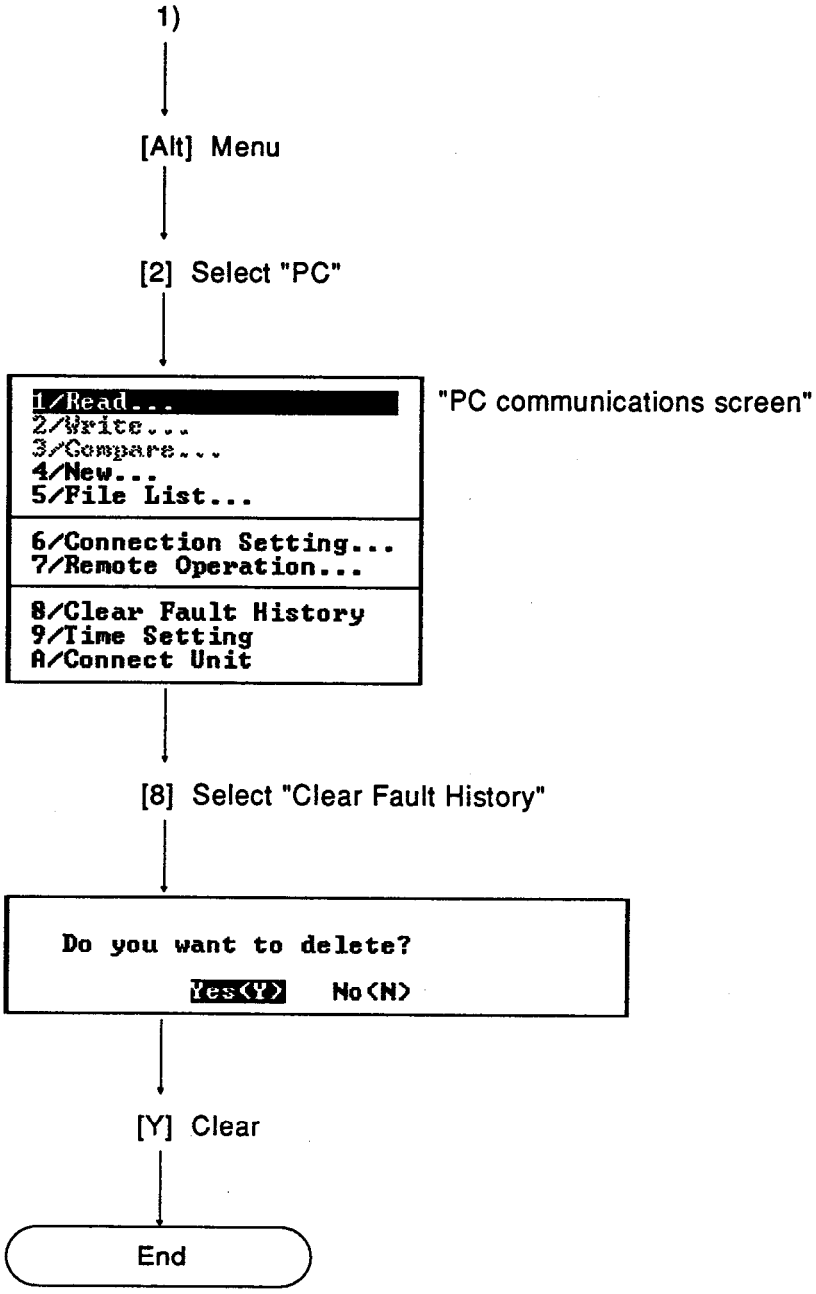


[Y] Execute



"PC Diagnostic screen"

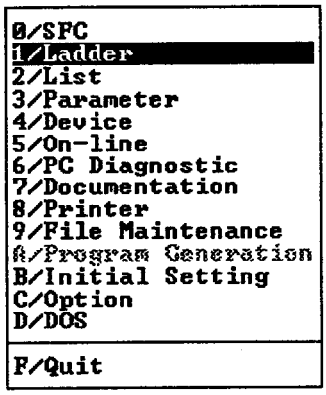
1)


1)
[Alt] Menu
[2] Select "PC"
1/Read...
2/Write...
3/Compare...
4/New...
5/File List...
6/Connection Setting...
7/Remote Operation...
8/Clear Fault History
9/Time Setting
A/Connect Unit
"PC communications screen"
[8] Select "Clear Fault History"
Do you want to delete?
Yes(Y) No(N)
[Y] Clear
End

3.5.2 Writing program to CPU

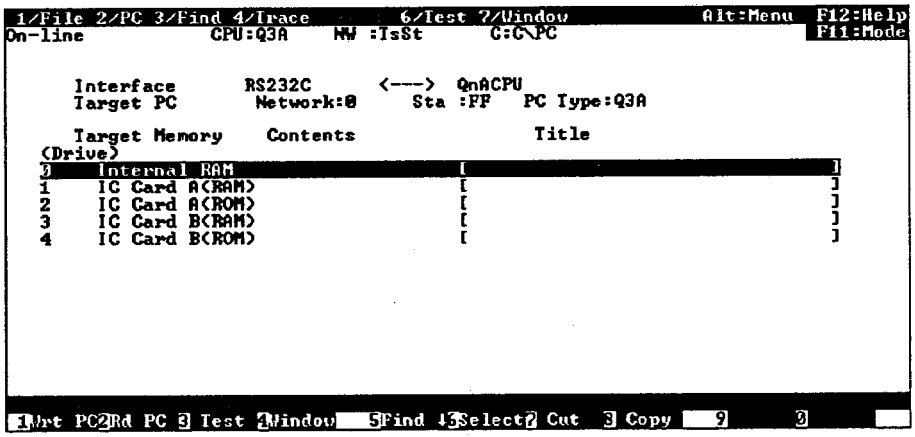
Write the program created in Section 3.4.2 plus the comments created in Section 3.4.3 to the QnACPU using operations from the mode select screen.

Set the QnACPU RUN/STOP key switch to the STOP position.



"Mode Select screen"

[5] Select "On-line"

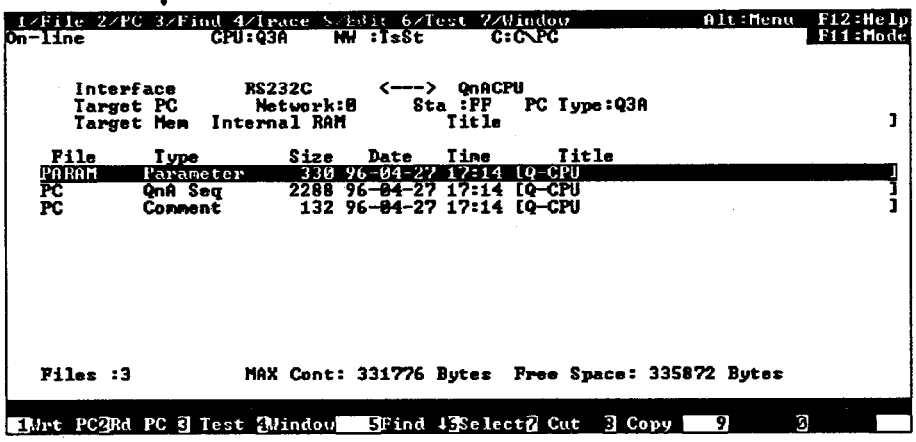


"On-line memory select screen"

[0] Select "Internal RAM"

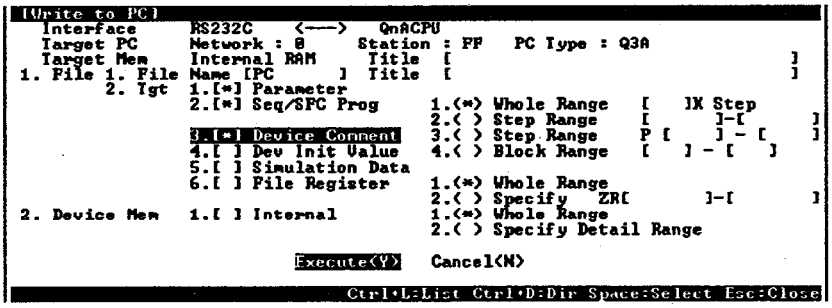
1)

1)



"On-line screen"

[F1] Write PC



"Write to PC screen"
- Write destination is internal RAM.
- The PC selected with "Create..." is automatically displayed as the file name.

- The default values are automatically set for the write data parameters and sequence program (Whole Range).
- Add device comments as write data and write them to the QnACPU.
  [SP] [↓] [SP] [3] [Y]

  A message is displayed in the message field during writing to the PC and another message indicates when the write operation is complete.

Return to the online memory select screen

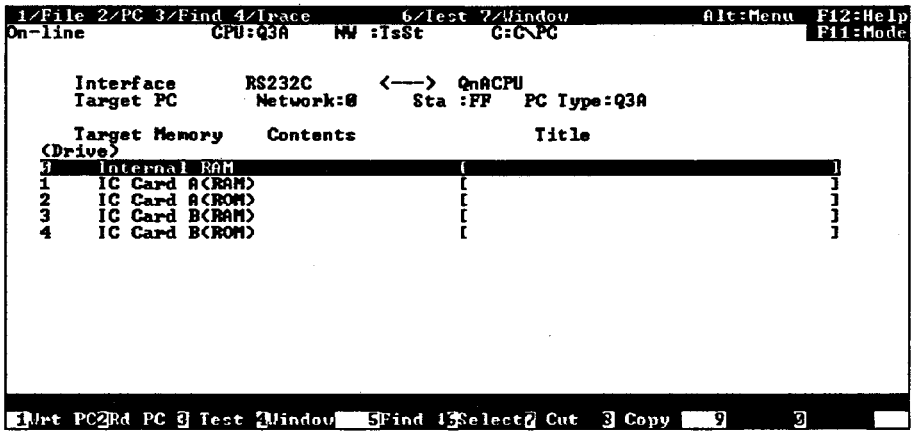
POINT

The QnACPU will not run when the key switch is set to the RUN position immediately after a new program is written to the QnACPU.
The program is first checked and a message is displayed, as follows:
Q2A(-S1) ..... RUN LED flashes
Q3A, Q4A ..... "PRG.CHECK!!" displayed
First set to RESET, then to RUN.

3.5.3 Operating the CPU for monitoring

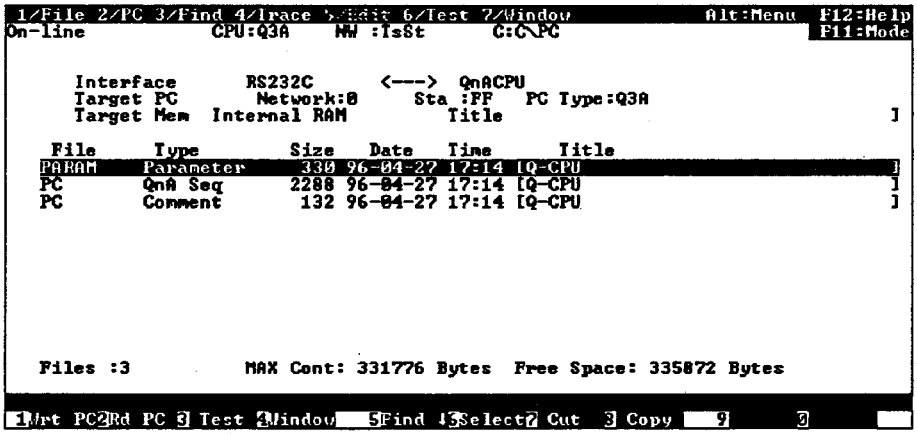
Read the program (written in Section 3.5.2) from the QnACPU before operating and monitoring the QnACPU.

(1) Read the program from the QnACPU using operations from the online memory select screen.



"On-line memory select screen"

[0] Select "Internal RAM"



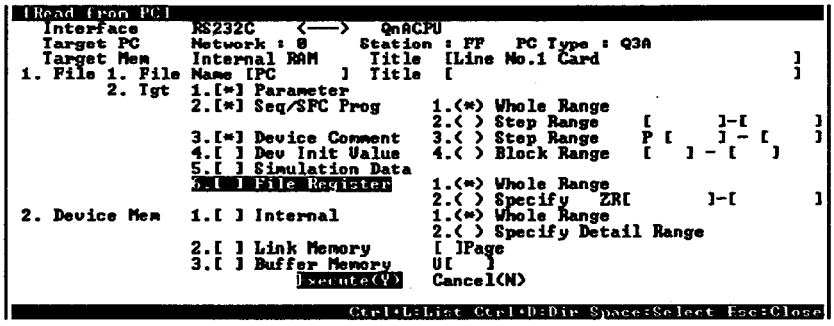
"On-line screen"

Check the file name and data type.

[F2] Read PC

1)

1)



"Read from PC screen"

Setting the file name as "PC".
[SP] [SP] [P] [C]

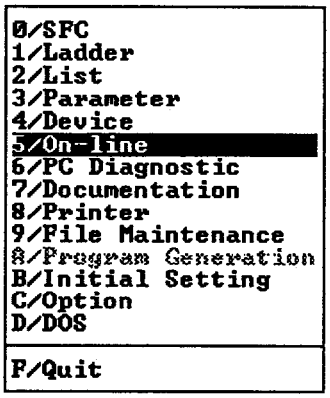
Read program from QnACPU, excluding device initial values and file register data.
[Tab] [Tab] [SP] [4] [5] [Y]

A message is displayed in the message field during reading from the PC and another message indicates when the read operation from QnACPU is complete.

Return to the online memory select screen

(2) Monitor QnACPU using ladder monitor operations from the mode select screen.

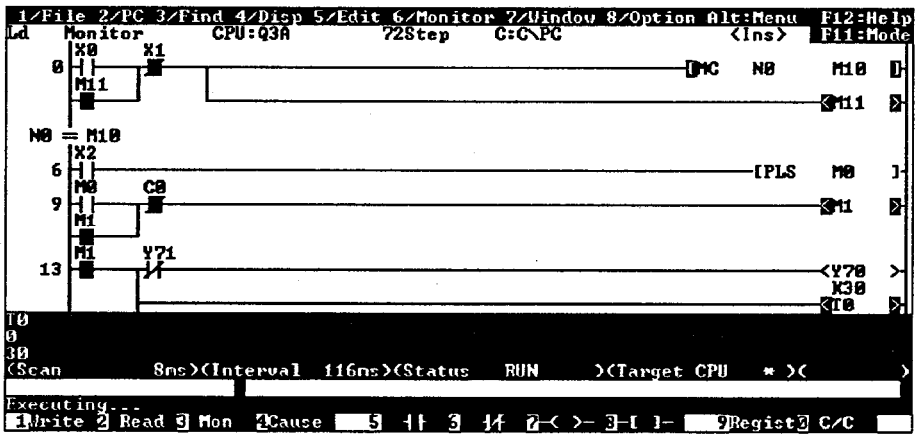
Set the QnACPU RUN/STOP key switch to the RUN position.



"Mode Select screen"

[1] Select "Ladder"

[F3] Monitor



"Ladder Monitor screen"

POINT

Use the registered device monitor to monitor the contents of a word device which is not displayed on the ladder monitor screen.
The example below monitors the counter C0.

Registered device monitor: [F9] [SP] [C] [0] [Enter] [SP] ([1]) [Y]
(Registration) (Counter C0) (16-bit)

Actual System Operation

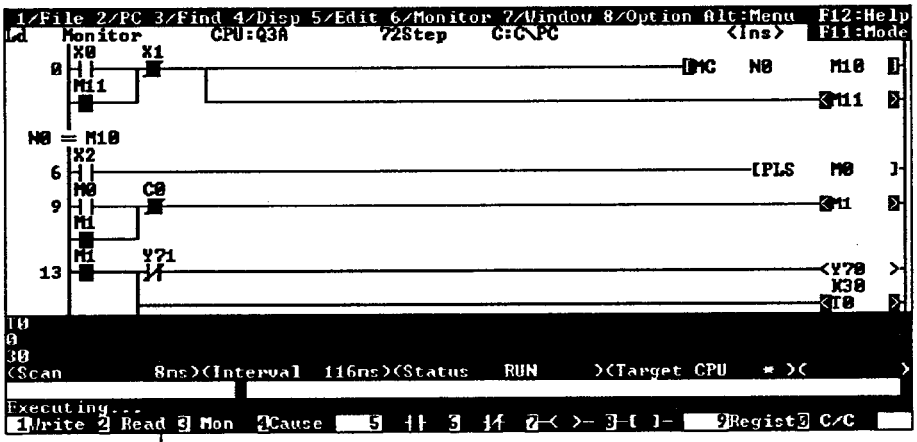
Watch the ladder monitor screen and confirm the following operations.

1) Turn ON X0 to enable the master control program (from MC to MCR) and permit actual system operation. (enable conveyor operation)
2) Y70 (buzzer) lights for 3 seconds after X2 (workpiece sensor) turns ON, then Y71 lights (start conveyor).
3) After X3 is turned from OFF to ON six times (6 products moved to workpiece), Y71 goes out (stop conveyor).
   X2 (no workpiece detected) turns OFF.
4) Repeat steps 2) and 3) (equivalent to cycle operation).
5) Press X1 to disable the master control program (from MC to MCR) and end actual system operation. (stop conveyor operation)
   Operation cannot be restarted after X1 is pressed during operation.
   Repeat the operations from step 1) (enable conveyor operation).

3.5.4 Modifying program while monitoring CPU operation (online program change (write during RUN))

Use online program change (write during RUN) to modify a program written to the QnACPU.
As an example, the timer T0 set value K30 (3 seconds) is changed to K60 (6 seconds).
Before modifying the program, set the "Write & Conv Setting...".

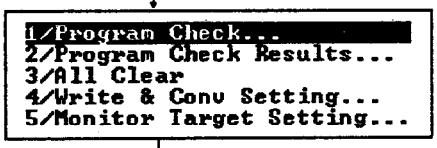
(1) Set the "Write & Conv Setting..." to "Write into PC during RUN state.". Operations from ladder monitor screen (similar to Section 3.5.3 (2)).



"Ladder Monitor screen"

[Alt] Menu

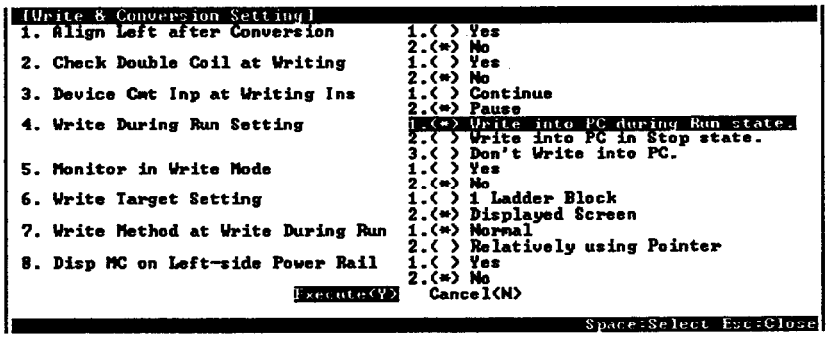
[8] Select "Options"



[4] Select "Writing & Conv Setting...".

1)

1)



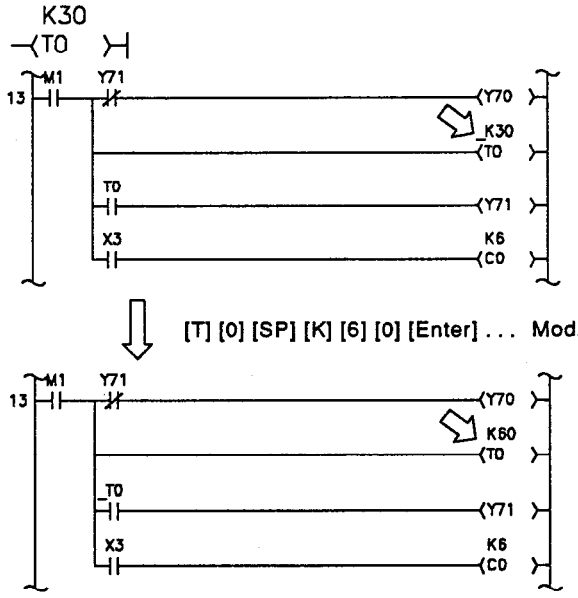
"Write & Conversion Setting screen"

Set "Write into PC during Run state."
[4] [1] [Y].... Revert to ladder monitor screen.

(2) Modify program
Use the following operations to modify the program.

(a) [F1] [Insert]... Set the ladder monitor screen to the write
(Write) (Overwrite) (overwrite) mode.

(b) Press the cursor keys to move the cursor immediately in front of



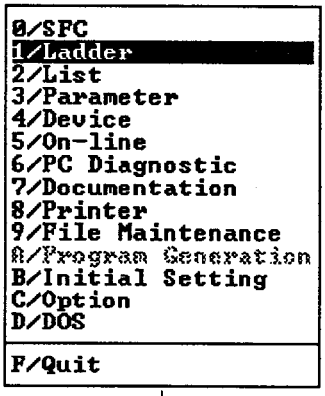
[T] [0] [SP] [K] [6] [0] [Enter] . . . Modify the timer T0 set value.

(c) Press [F4] key to convert the program.
(Convert)
A message is displayed in the message field during conversion. Another message indicates that the operation is complete and the step number was changed "Write into PC during RUN state." ends.

(3) Check actual system operation
Follow step (2) described in Section 3.5.3 (2) "Actual System Operation" to check that the time Y70 lights has changed from 3 seconds to 6 seconds.

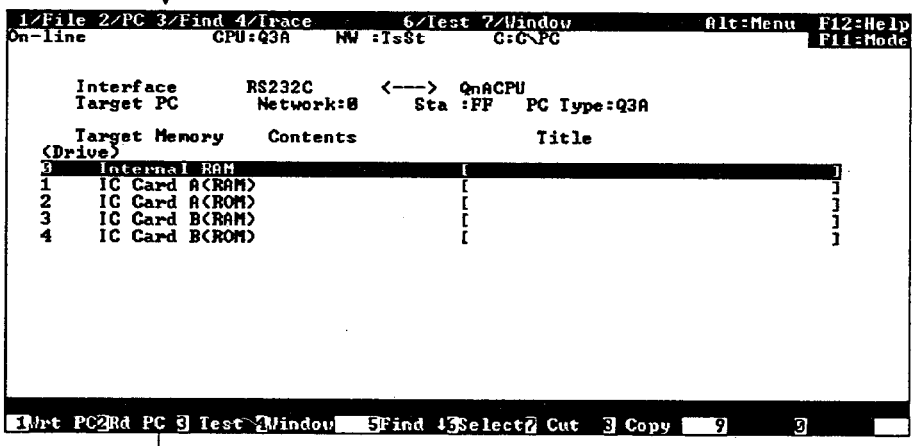
3.5.5 Registering a password

A password can be registered to protect access to the programs in the QnACPU. It can inhibit reading, writing, and display or be used to write protect the programs so that they cannot be accidentally overwritten. Operation from the mode select screen.



"Mode Select screen"

[5] Select "On-line"



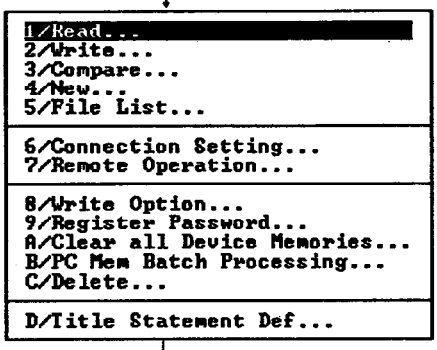
"Online Memory Select screen"

[Alt] Menu

[2] Select "PC"

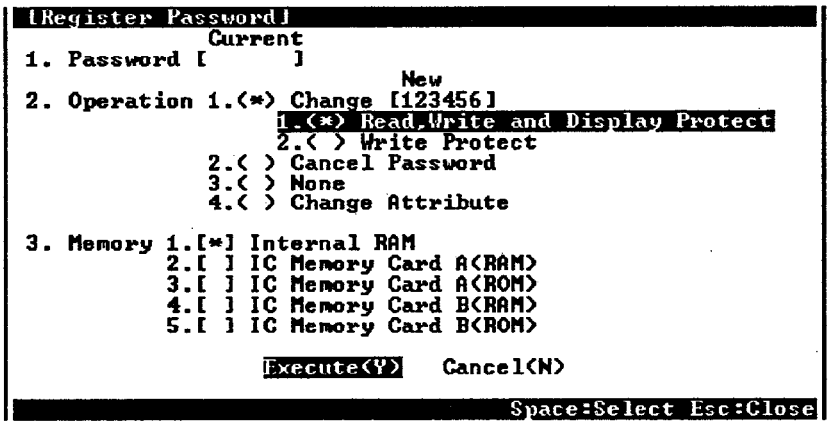
1)

1)



"PC Communications screen"

[9] Select "Register Password"



"Register Password screen"

The password can be designated as up to 6 alphanumeric characters.

Stop the QnACPU

Register the password "123456" in the QnACPU to inhibit reading, writing, and display (default) for the internal RAM memory (default).

[2] [1] [2] [3] [4] [5] [6] [↓] [Y]

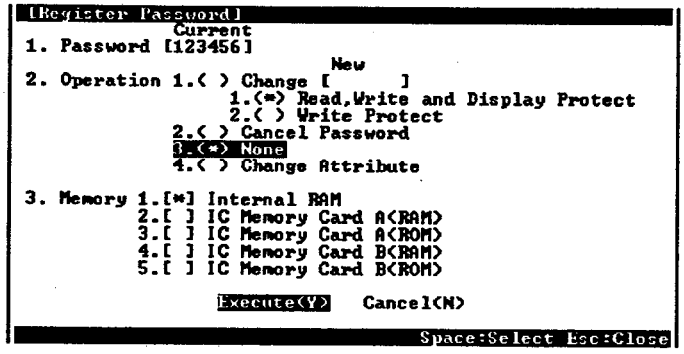
(New) (Password: 123456) (Read, Write and Display Protect) (Execute)

A message is displayed in the message field when the registration operation is complete.

POINTS

(1) Check the password operation after temporarily quitting the IBM PC/AT GPP functions. They password does not become effective until the GPP functions are quit.

(2) Follow the procedures below to initially set the password to access a QnACPU for which an password has been set.
Repeat the operations on the previous page to display the register password screen.



"Register Password screen"

Set the password and select "None"

[1] [1] [2] [3] [4] [5] [6]
(Current) (Registered password: 123456)

[↓] [SP] [↓] [↓] [↓] [↓] [SP] [Y]
(Operation) (Move cursor to "None") (Select) (Execute)

A message is displayed in the message field when the operation is complete.
If the password is incorrect, a message indicates that the file cannot be accessed and file access is not permitted.

(3) If the password is completely forgotten, the PC memory must be formatted, by the method described below.

Set the QnACPU to STOP.

[PC communications screen]
⇨ [B] [2] [Y] Do you want to format internal RAM? [Y]
(Batch operation) (Format) (Execute) (Yes)

A message is displayed in the message field when the formatting of the QnACPU internal RAM is complete (the memory contents have been cleared).

3.5.6 Checking responses to forced faults (PC diagnosis)

The output module is forcibly removed during QnACPU operation to simulate the AY42 (64-point) output module falling out of the PC CPU system in Section 3.1.

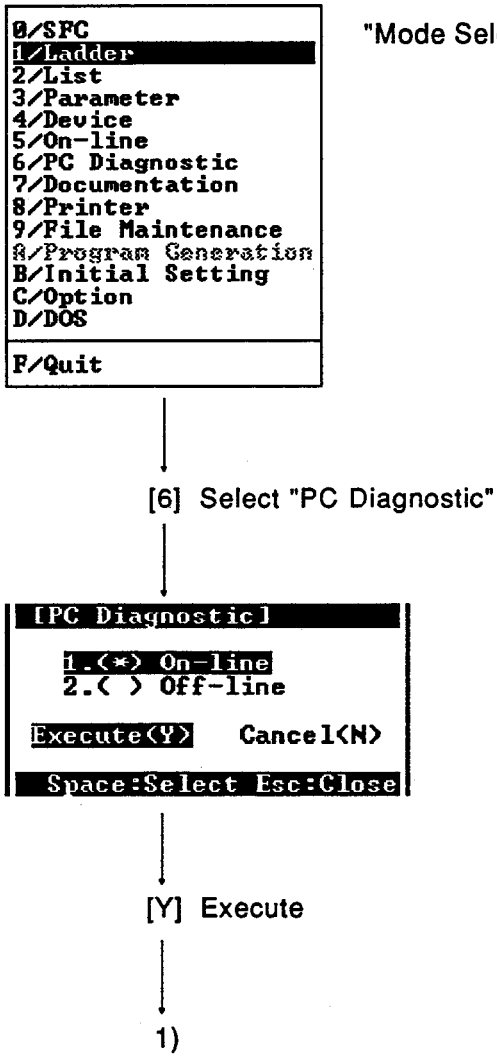
(1) Check QnACPU error indicator flashing and error message display

Check that the error indicator flashes and that the following message appears in the message display window:

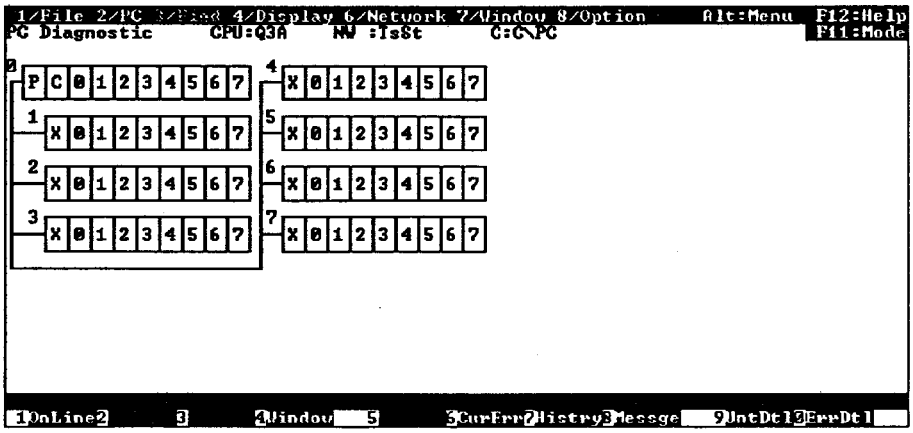
Display message: UNIT VERIFY ERR.

(2) Determine cause using PC diagnosis

Follow the procedure below to determine the cause of the error with PC diagnosis using operations from the mode select screen.



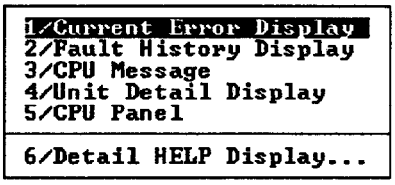
1)



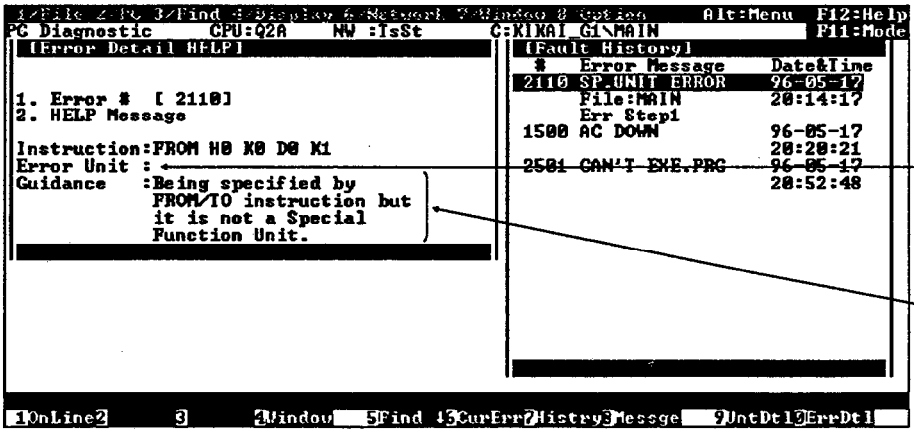
"PC Diagnostic screen"

[Alt] Menu

[4] Select "Dispyay"



[1] Select "Current Error Display"



"Current Error Display screen"

Indicates the module in Slot 1 of the base unit.

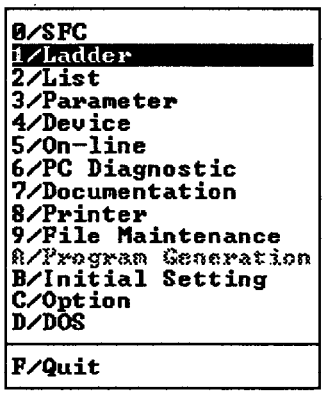
Cause of error

(3) Checking the module causing the I/O module verification error

The PC diagnosis indicated the cause of the error as "Being specified by FROM/TO instruction but it is not a Special Function Unit." (unit verification error). Next, monitor the following special registers to determine the module causing the error.

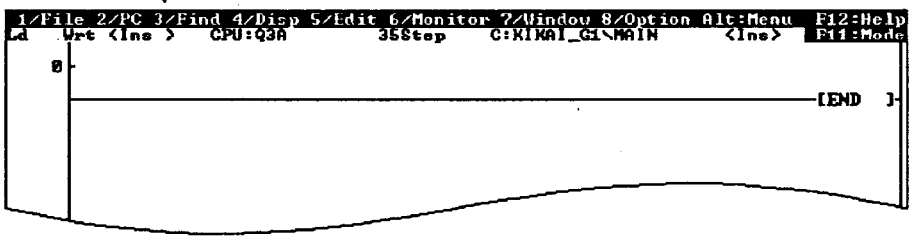
Special Register	Name	Function
SD61	I/O module verification error module number	Stores the I/O signal (latest number) of the module where the I/O module verification error occurred.
SD1400 to SD1431	I/O verification table	Stores the I/O module verification error information in 16-bit units as a bit pattern. 0: no error 1: error

Monitor SD61, then SD1400.
Operate from the mode select screen.



"Mode Select screen"

[1] Select "Ladder"



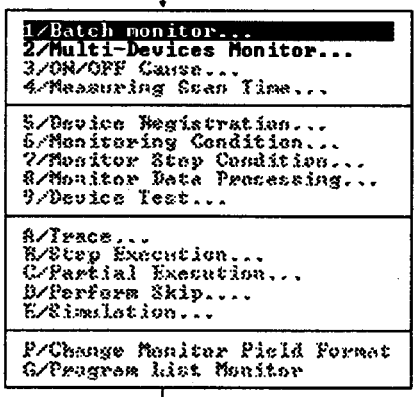
"Ladder Write (insert) screen"

1)

1)

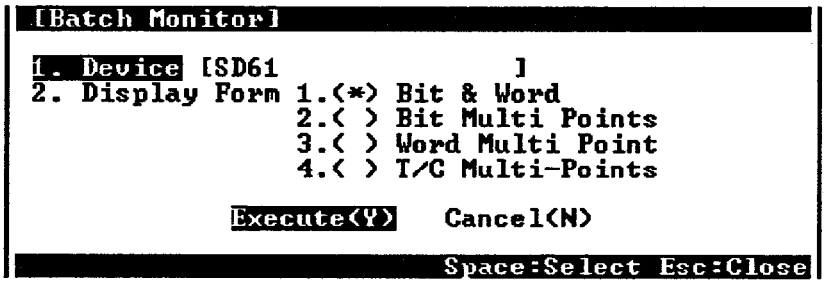
[Alt] Menu

[6] Select "Monitor/Test".



"Monitor/Test Select screen"

[1] Select "Batch monitor"



"Batch Monitor Device Setting screen"

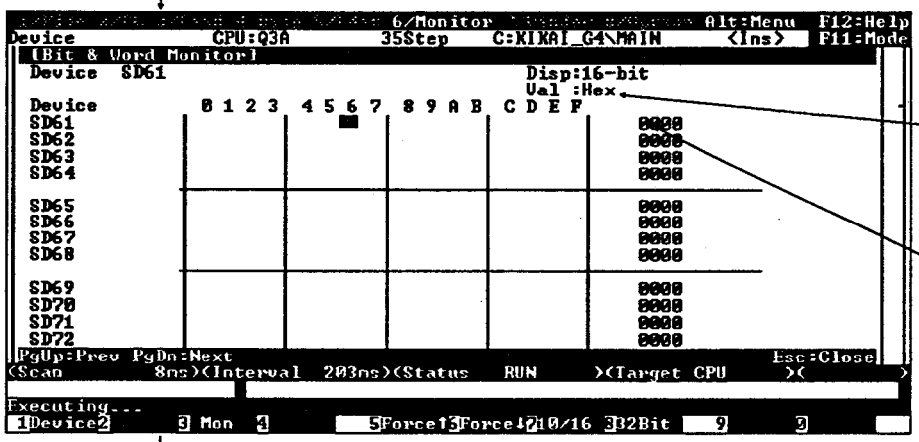
Set special register SD61.
The display format is bit & word monitor (default).
[SP] [S] [D] [6] [1] [Enter]
(Select) (Monitor device SD61)

[Y] Execute

2)

2)

Monitoring SD61

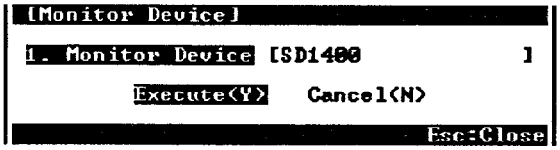


"Bit & Word Device Monitor screen"

Press [F7] key to select the hexadecimal display.

Indicates that the I/O number (latest number) of the module where verification error occurred was X, Y40.
(Corresponds to AY42 mounted in Slot 1.)

[F1] Device



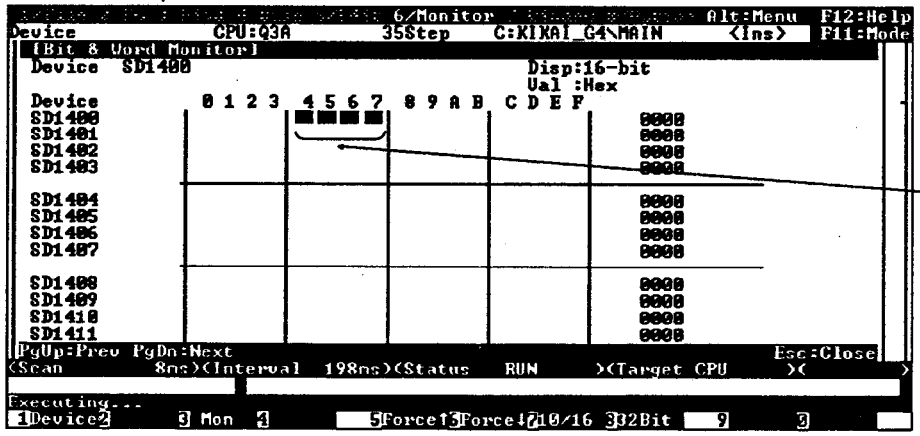
"Monitor Device Setting screen"

Set the next monitor device SD1400.
[BS] [BS] [1] [4] [0] [0] [Enter]

(Previous device No. is cleared.)

[Y] Execute

Monitoring SD1400



"Bit&word Device Monitor screen"

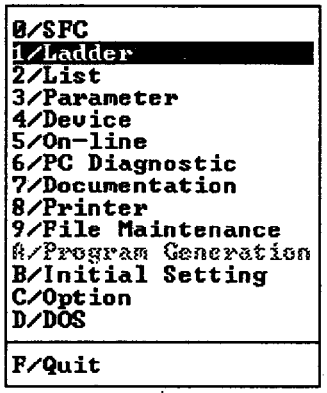
Indicates that the I/O module verification error occurred for XY40 to 4F, XY50 to 5F, XY60 to 6F, and XY70 to 7F.

3.6 Printing the Program

Print out the program created in Section 3.4.2 plus the comments, machine names, and contact/coil destinations created in Section 3.4.3 on a PC-PR201H printer.

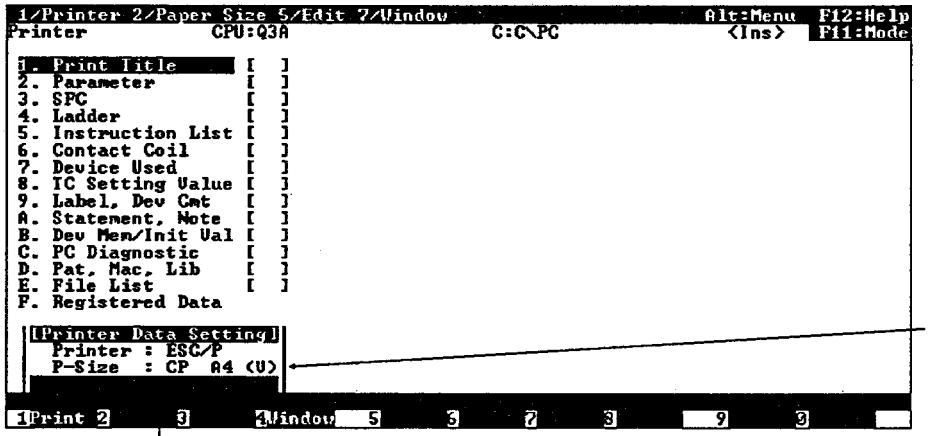
3.6.1 Setting the printer and paper size

Set the type of printer used and the paper size using operations from the mode select screen.



"Mode Select screen"

[8] Select "Printer"



"Printer Mode screen"

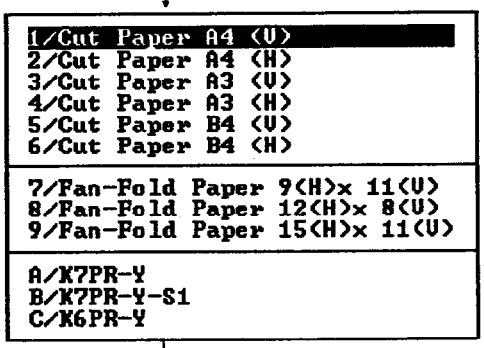
Defalut is cut paper, A4 (V).

[Alt] Menu

[2] Select "Paper size"

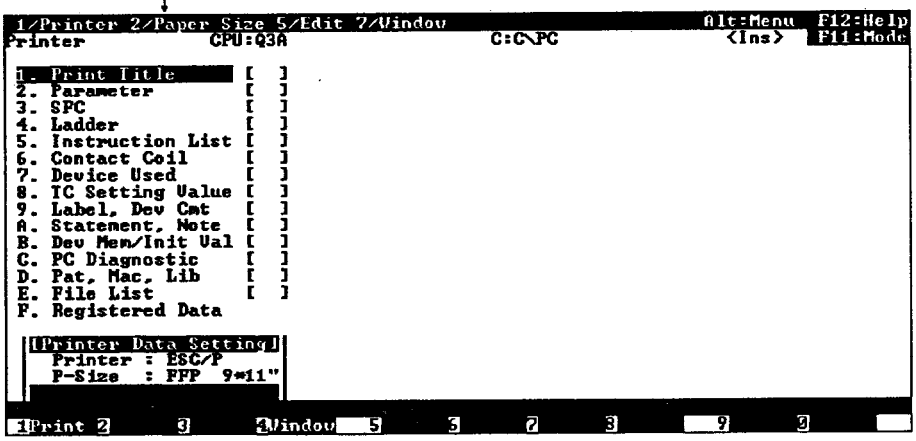
1)

1)



"Paper Size Setting screen"

[7] Select "Fan-Fold Paper, 9 (H) x 11 (V)"



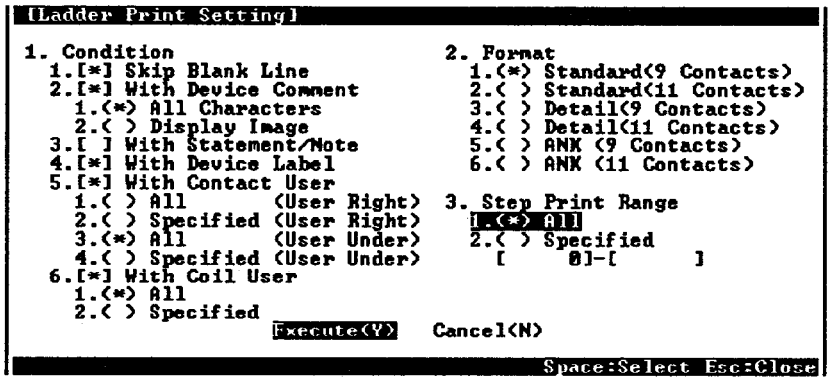
"Printer Mode screen"

3.6.2 Printing out the ladder program

Select "Ladder" as the print item, then set the print format, including printing conditions, printing method, whether headers and footers are added, and the print start position, using operations from the printer mode screen.

"Printer Mode screen"

[↓] ... [↓] [SP] Select "Ladder".
(Press 3 times)



"Ladder Print Setting screen"

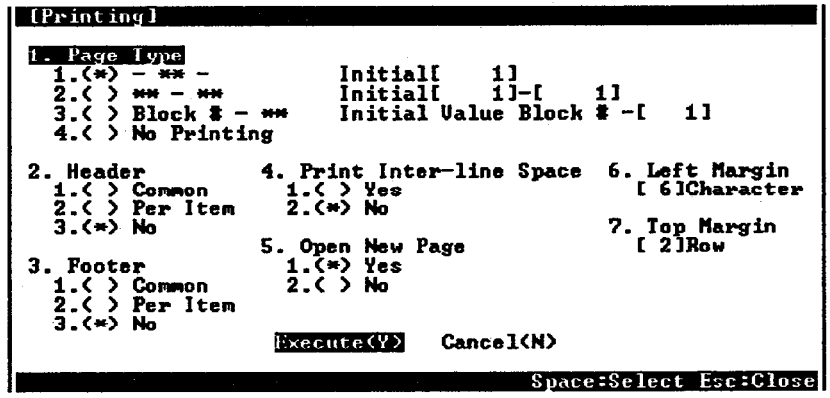
Set the condition, format, and step print range.
Set with [SP] or [↑]/[↓] keys.

> The default settings are: "Skip Blank Line", "No Device Comment", "No Device Label", "No Contact User", "No Coil User" , "Format: Standard (9 Contacts)", and "Step Print Range: A11".

1)

1)

[F1] Print

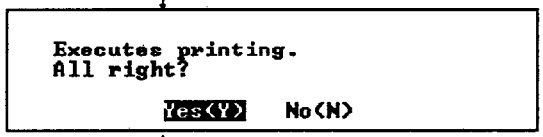


"Printing screen"

Set the print execution conditions. Set the left and top margins only, leaving other items as default values.
Set left margin.
[6] [SP] [Enter]..... 6 characters
Set top margin.
[↓] [SP] [2] [Enter].... 2 rows

The default settings are: "Page Type = -**-", "Header: No", "Footer: No", "Print Inter-line Space: No", "Open New Page: Yes".

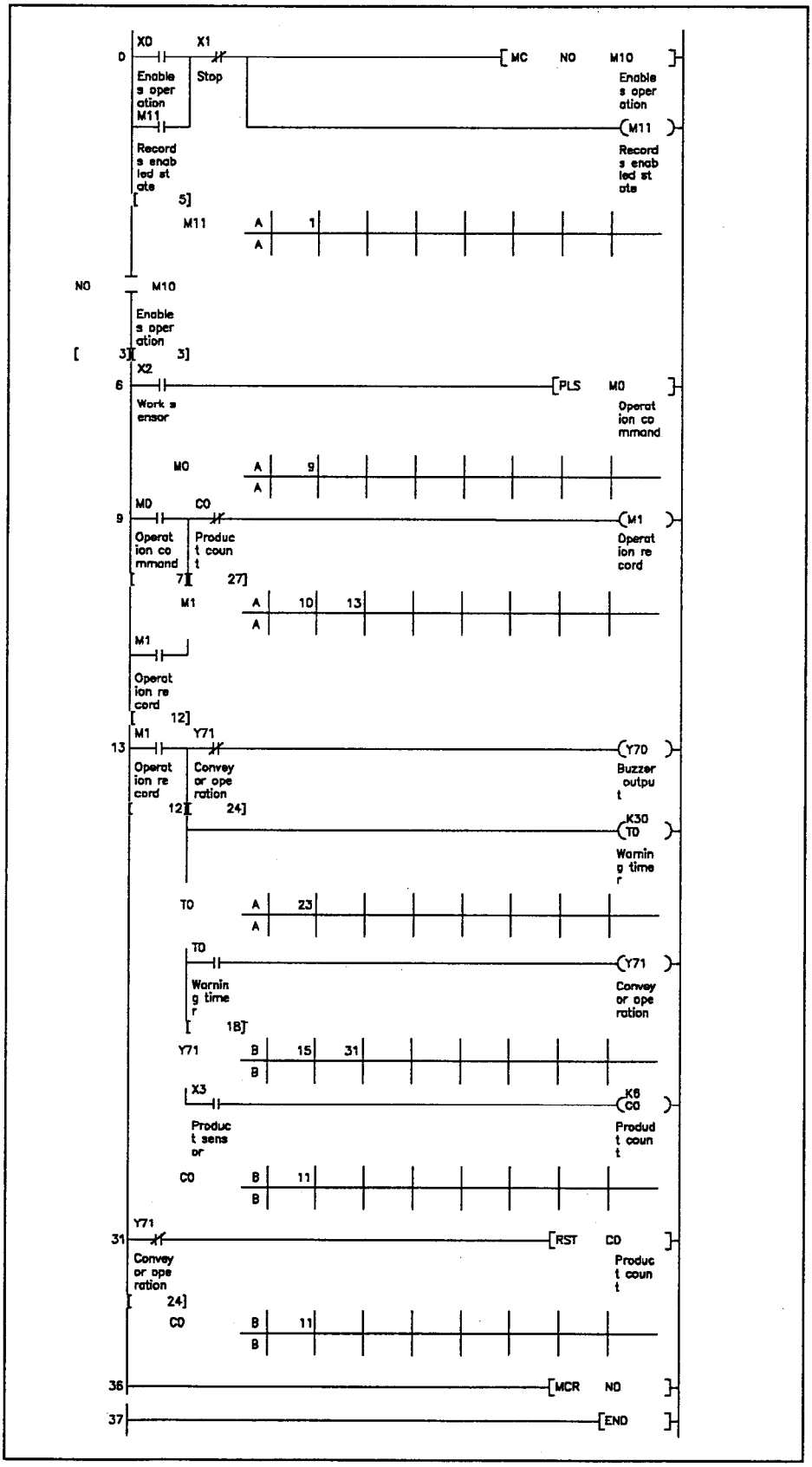
[Y] Print



[Y] Yes . . . Start printing

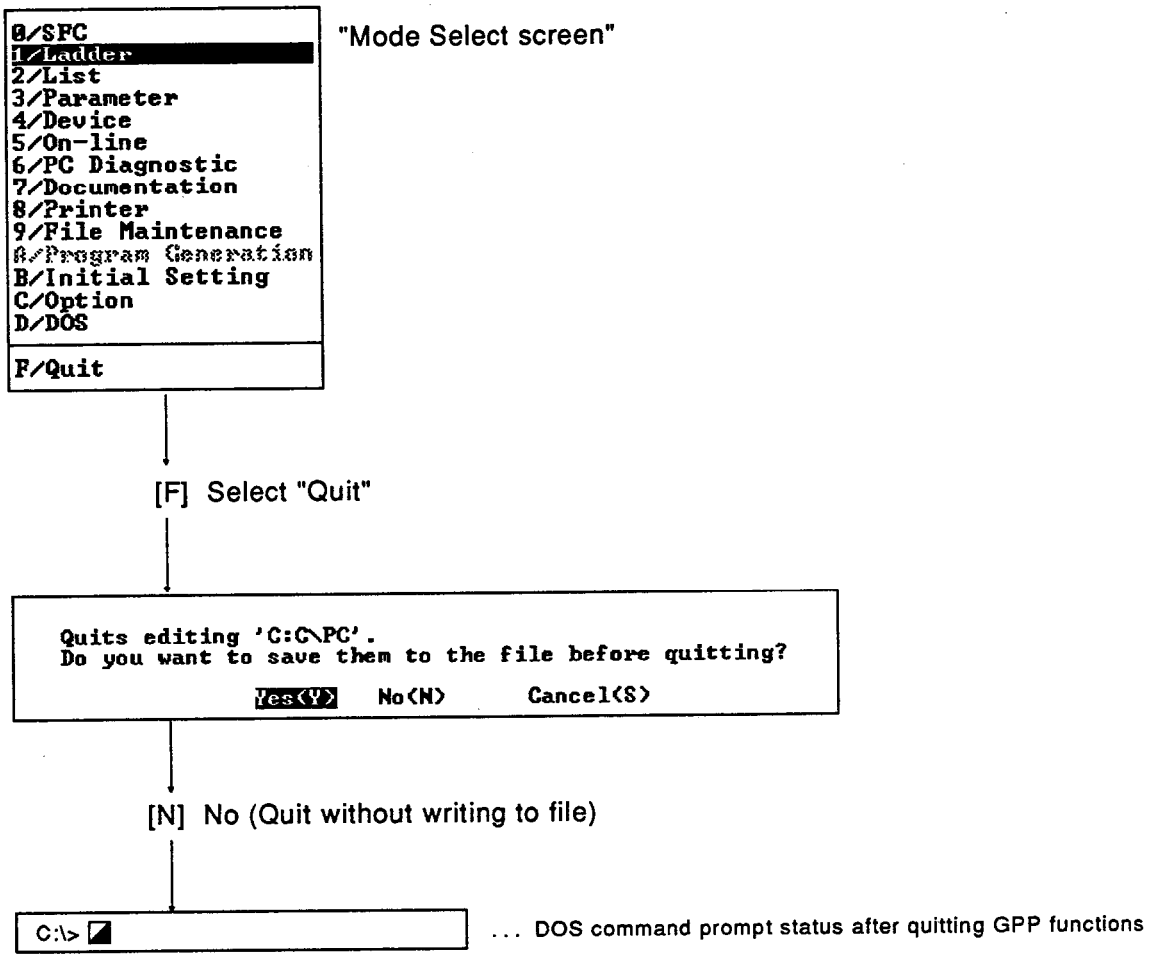
3.6.3 Sample ladder program print-out

An example of a ladder program print-out is shown below.



3.7 Quitting GPP Functions

Quite the GPP functions using operations from the model select screen.



4. MAKING THE MOST OF YOUR QnACPU (1)

This chapter describes methods to make the best use of your QnACPU, which differ from the conventional ACPU operations.

4.1 System Configuration

The system configuration shown below is used for the descriptions in this chapter.

		X00 to X0F	X10 to X1F	Y20 to Y2F	Y30 to Y3F	X/Y40 to X/Y5F	X/Y60 to X/Y7F
A62P	Q3ACPU	AX40	AX40	AY40	AY40	A68AD	A62DA

Memory card: Q1MEM-512SE

4.2 Programming for Individual Control Processes

The QnACPU handles sequence programs as files, so that sequence programs can be divided up into control processes for program design. This section describes this method of programming.

4.2.1 File structure

(1) Program execution can be matched to the type of control.

Programs can be selected from four types: initial execution, scan execution, low-speed execution, and standby.
These programs can be set to run when required, thereby reducing the scan time of the permanently executing scan program.

(a) Initial execution (initial program)

A program executed once only when the QnACPU is set to RUN.
Used for initialization.

(b) Scan execution (scan program)

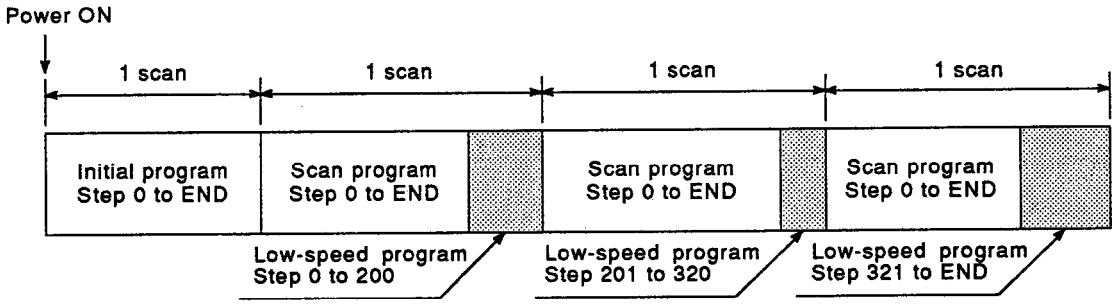
A permanently executing program.
Equivalent to a conventional program running from step 0 to END.
Can include sub-routine programs and interrupt programs.

(c) Low-speed execution (low-speed program)

A program which is not entirely executed each scan but which is executed over several scans, using the constant scan surplus time or a set time.
Used for programs run at low frequency, such as scheduled inspection programs.

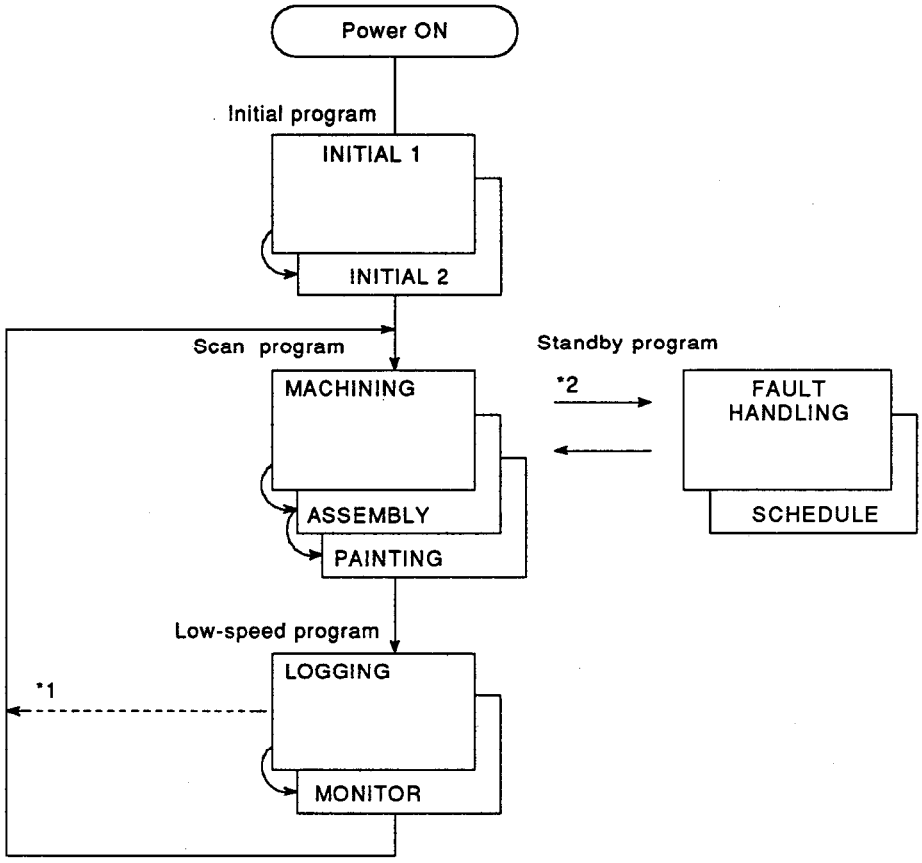
(d) Standby (standby program)

A program which is executed when a certain start condition is met, such as a sub-routine program or an interrupt program.
Used when the sub-routine programs and interrupt programs are held as library data and are handled separately to the main program.



(2) File Management

The sequence program and other data handled by the QnACPU is managed as files. Whereas programs were conventionally handled as main programs and sub-programs, the QnACPU allows structured programming, such that each type of program can be broken down for individual programmers or processes.



*1 If constant-scan or low-speed program execution time is set.
*2 On request

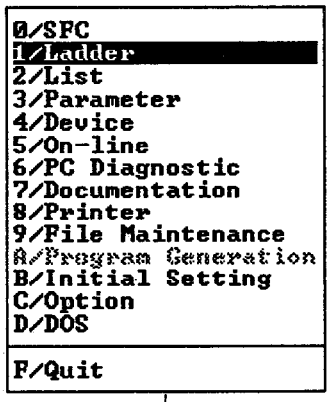
4.2.2 Sample program

The example below executes the files listed in the table.

File Name	Type	Description
INITIAL	Initia	Program to set the initial data.
MACHINING	Scan	Program for machining.
ASSEMBLY	Scan	Program for assembly.
TRANSPORT	Scan	Program for transporting.
INSPECTION	Standby	Program for inspection.
MONITOR	Low-speed	Program for monitoring.

(1) Create each program and write it to the CPU.
Set the RUN/STOP key switch to STOP.
See Chapter 3 for details on programming and writing to the CPU.

(2) Set the parameters to execute multiple files, and write them to the CPU.

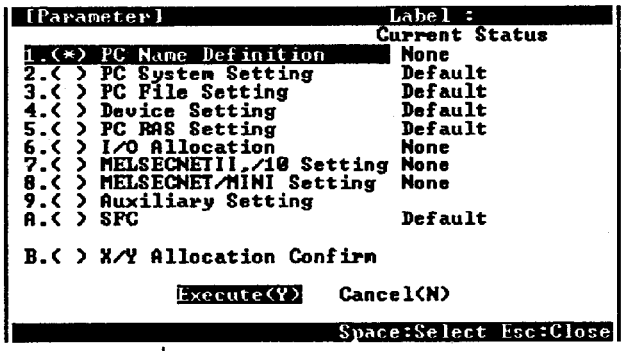


"Mode Select screen"

[3] Select "Parameter"

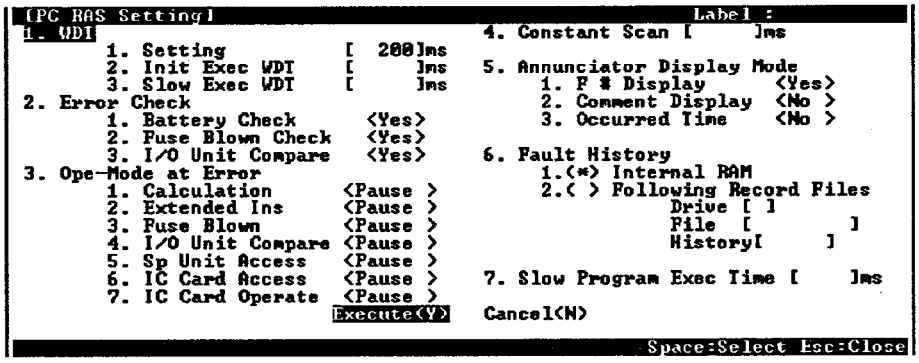
1)

1)



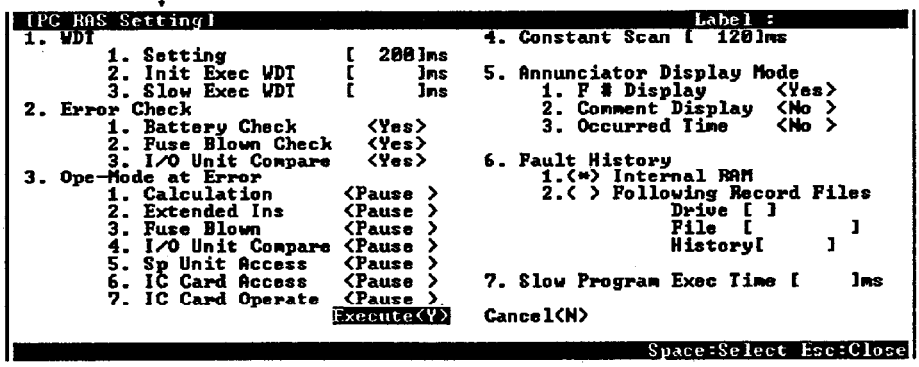
"Parameter screen"

[5] [Y] Select "PC RAS Setting"



"PC RAS Setting screen"

[4] [1] [2] [0] [Enter] Set 120 ms constant scan

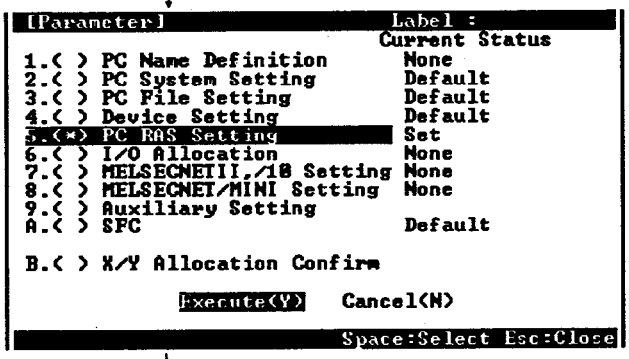


"PC RAS Setting screen"

[Y]

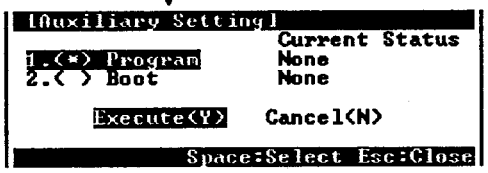
2)

2)



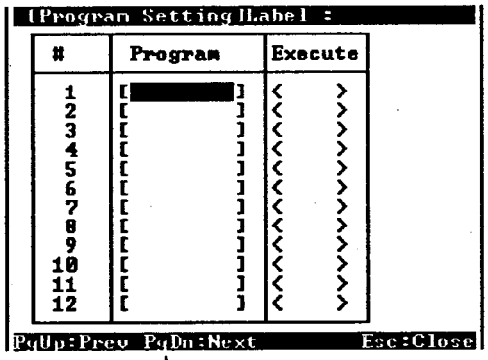
"Parameter screen"

[9] [Y] Select "Auxiliary Setting"



"Auxiliary Setting screen"

[1] [Y]



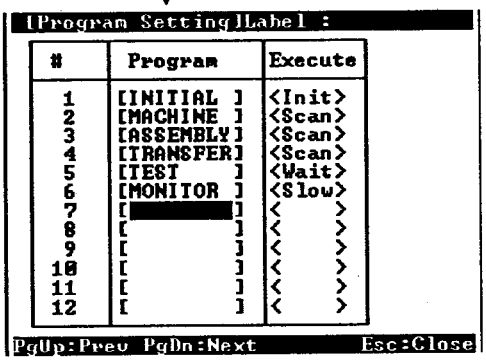
"Program Setting screen"

Sets the programs to execute

[I] [N] [I] [T] [I] [A] [L] [Enter] [SP] [Enter] Set program #1
[M] [A] [C] [H] [I] [N] [E] [Enter] [Enter] Set program #2
[A] [S] [S] [E] [M] [B] [L] [Y] [Enter] [Enter] Set program #3
[T] [R] [A] [N] [S] [F] [E] [R] [Enter] [Enter] Set program #4
[T] [E] [S] [T] [Enter] [SP] [SP] [SP] [Enter] Set program #5
[M] [O] [N] [I] [T] [O] [R] [Enter] [SP] [SP] [Enter] Set program #6

3)

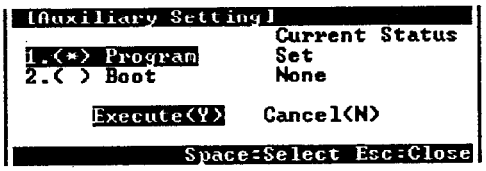
3)



"Program setting screen"

[Esc]

[Y] Register parameters

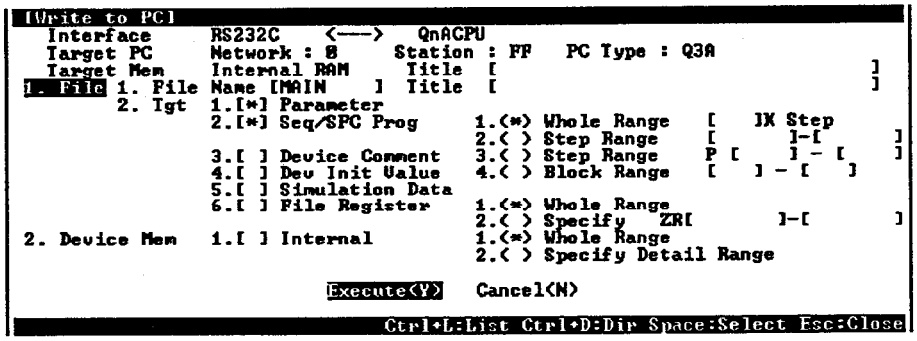


"Auxiliary Setting screen"

[Alt] [2] [2] Select "Write to PC"

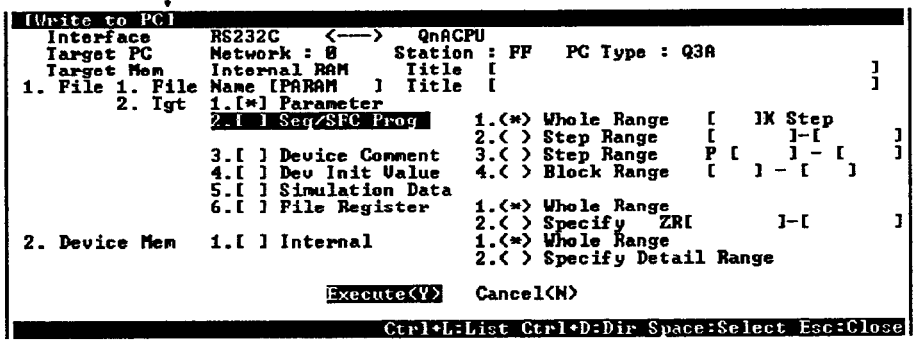
4)

4)



"Write to PC screen"

[SP] [SP] [BS] [BS] [BS] [BS] [P] [A] [R] [A] [M] [↓] [↓] [SP] [↓]



"Write to PC screen"

[Y] Write parameters to CPU

(3) Execute the programs

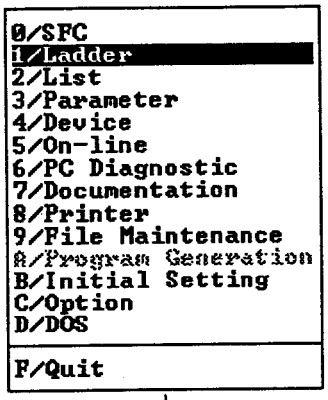
*1

Turn the CPU key switch from "STOP" to "RESET" to "RUN".

*1 If the key switch is turned from STOP to RUN.
The CPU displays a message, as follows, and checks the program.
Q2ACPU, Q2ACPU-S1: RUN LED flashes
Q3ACPU, Q4ACPU : "PRG. CHECK!!" displayed
The CPU runs the programs when the key switch is set from STOP to RESET to RUN.

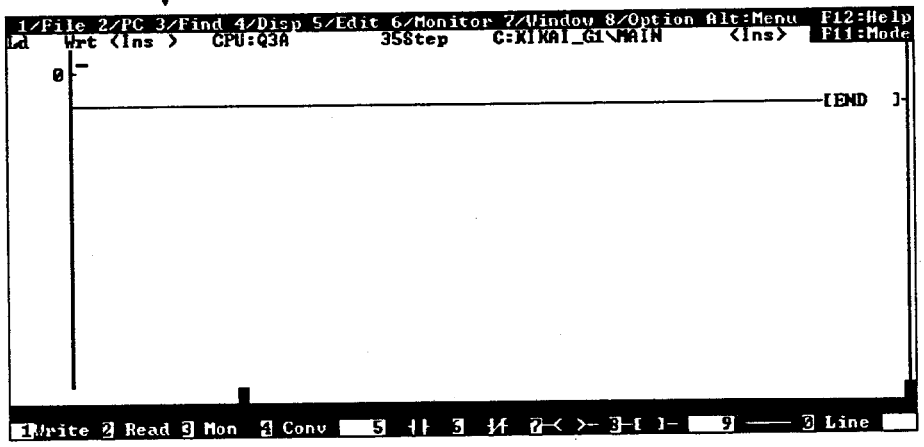
4.2.3 Program list monitor

Displays the processing time for the program executed in Section 4.2.1



"Mode select screen"

[1] Select "Ladder"

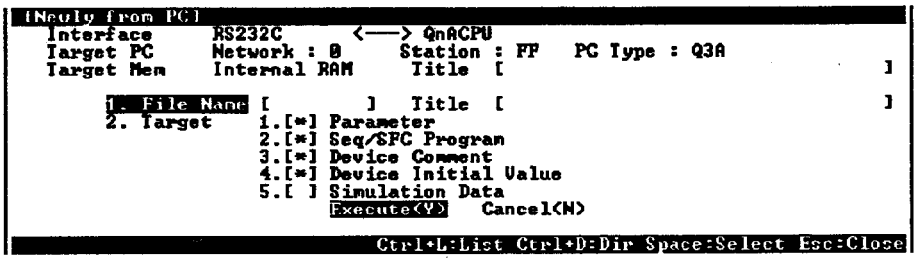


"Ladder Write screen"

[Alt] [2] [4] Select "Newly from PC"
(Match the name of the GPPQ file with the read program name.)

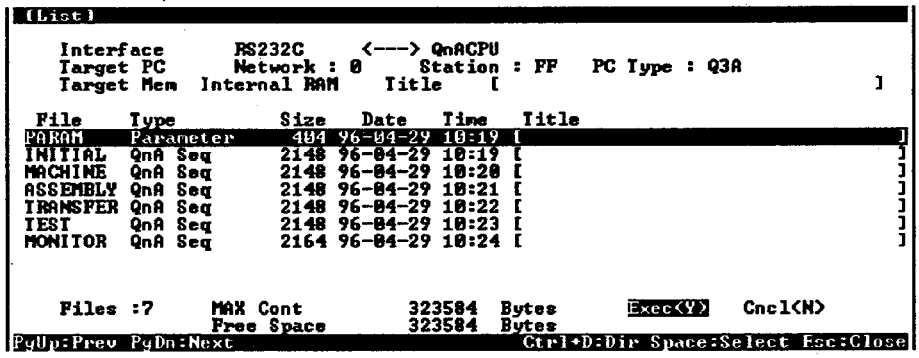
1)

1)



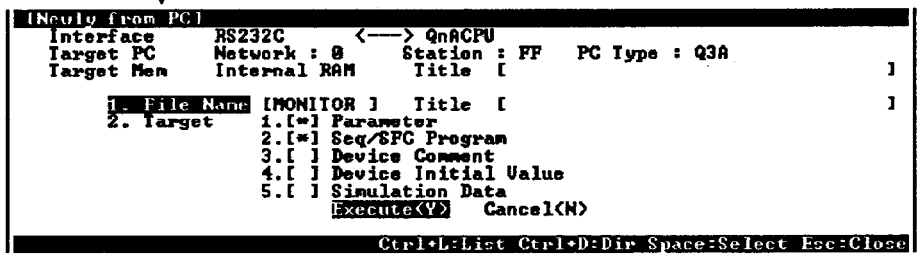
"Newly from PC screen"

[Ctrl] [L] Display file list



"File List screen"

[SP] [↓] [↓] [↓] [↓] [↓] [↓] [SP] [Enter] Select read file

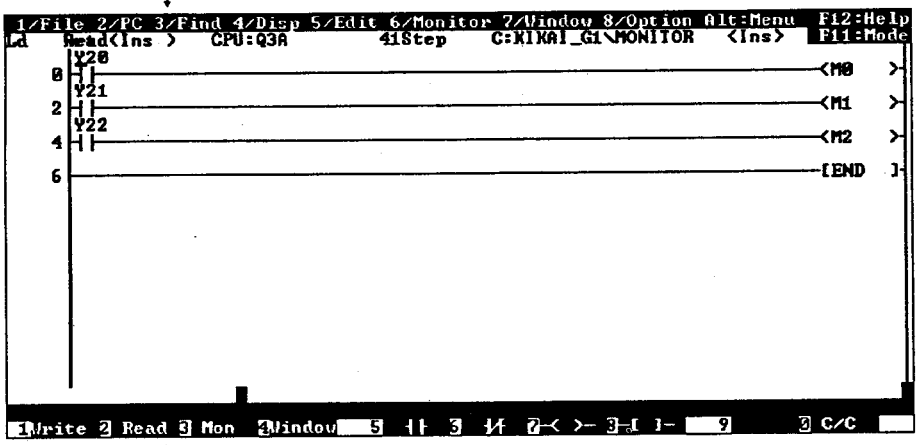


"Newly from PC screen"

[Y] Read parameters and sequence programs from CPU.

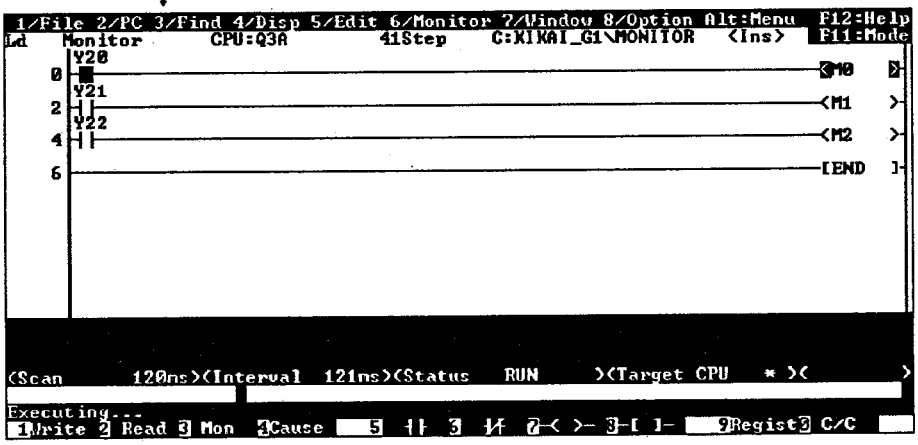
2)

2)



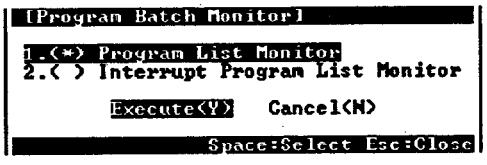
"Ladder Read screen" (example)

[F3] Monitor



"Ladder Monitor screen"

[Alt] [6] [6] Select "Program Batch Monitor"

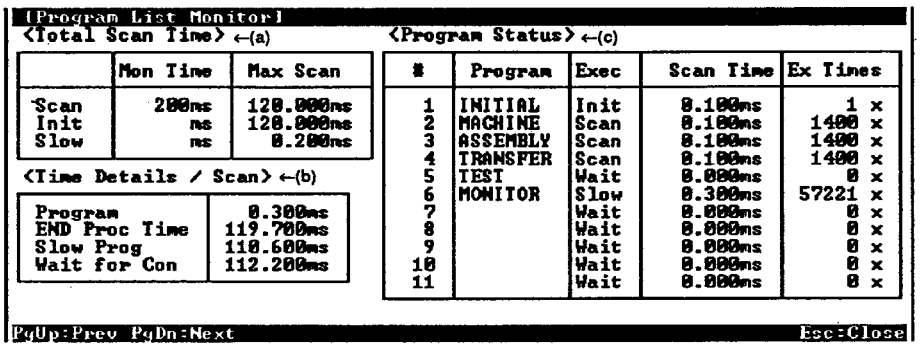


"Program Batch Monitor screen"

[1] [Y] Run program batch monitor

3)

3)



"Program List Monitor screen" (Displayed values for this example only. Actual values may vary.)

Key to the Screen

(a) "Total Scan Time"

Displays the watchdog times set with "5. PC RAS Setting" in the parameter mode and the total scan time for each program type.

1) "Mon time"

Displays the watchdog times for the scan programs, initial programs, and low-speed programs.
A watchdog error occurs if displayed scan time exceeds the CPU watchdog time.

2) "Max Scan"

Displays the total of the times displayed in "Time Details / Scan".

(b) "Time Details / Scan"

Displays the scan time details.

1) "Program"

Displays the total execution time for the scan programs.

2) "END Proc Time"

Displays the END processing time.

3) "Slow Prog"

Displays the total execution times set for low-speed programs, if low-speed program execution times are set.

4) "Wait for Con"

Displays the constant scan wait time if constant scan is set. However, "0.000 ms" is displayed if a low-speed program execution time is also set.

(c) "Program Status"

Displays the execution status of the programs selected with "9. Auxiliary setting" in the parameter mode.

1) "Program"

Displays the program names in the order of the set parameters.

2) "Exec"

Displays the type of program set with the parameters.

3) "Scan Time"

Displays the actual scan time (present value). The scan time is displayed as "0.000 ms" when program operation is stopped (standby status).

4) "Ex Times"

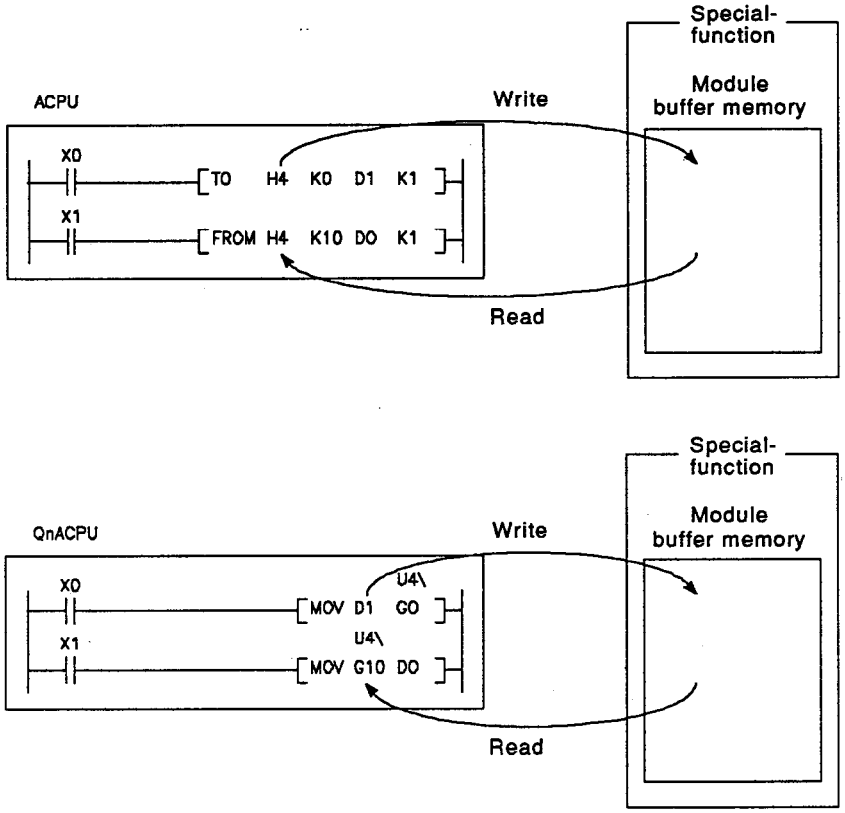
Displays the number of executions as a value from 0 to 65536, starting from zero when measurement was started. The number of executions value is held when program operation is stopped.

4.3 Easy Programming

4.3.1 Direct access of special-function module buffer memory as devices

The ACPU used FROM/TO instructions for buffer memory read and write operations.
The QnACPU simplifies sequence programs by allowing direct access of a special-function module buffer memory as devices.

Example Accessing buffer memory addresses 0 and 10 of the special-function module mounted at X/Y40.

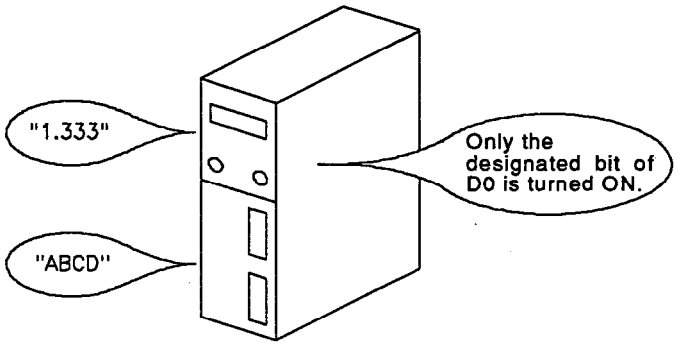


CAUTION

Processing speed with direct access is the same as with the FROM/TO instructions.
Frequently used buffer memory contents should be temporarily moved to a data register to increase processing speed.

4.3.2 Accessing word devices as bit units and using differential contacts

Bit designation can be used for real numbers, character strings, and word devices.

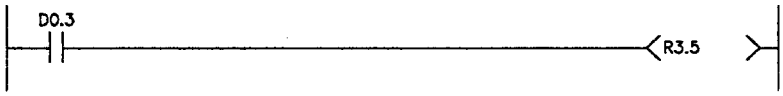


(1) Bit designation

Handles a part of word device data as a bit device.

Example

When D0, bit 3 is ON, R3, bit 5 turns ON.



(2) Differential contact

Uses a leading edge or trailing edge as an input device.

Example

At the leading edge of X0, the 16 bits of data, M0 to M15, are transferred to D0.

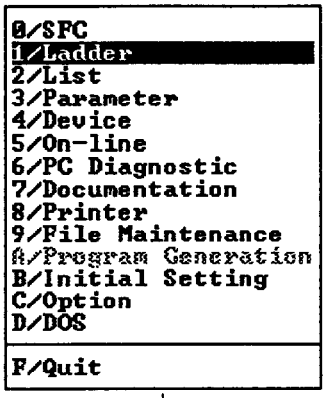
4.4 Convenient Functions When Using QnACPU

4.4.1 Allocation of optimal device points range for the system used

The number of device points was fixed with the conventional ACPU.
However, the QnACPU allows allocation of the optimal device points range to suit the system used.
The settings are shown in the table below.

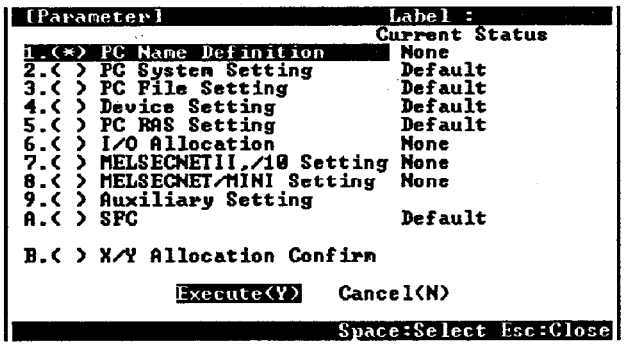
Item	Description	Setting Range	Default Value
Number of devices	Sets the number of internal device points.	32 k points maximum can be set for one device, in a total range of 29 k words (excluding device X, Y, S).	X : 8 k points (fixed) Y : 8 k points (fixed) M : 8 k points L : 8 k points B : 8 k points F : 2 k points SB : 2 k points V : 2 k points S : 8 k points (fixed) T : 2 k points ST : 0 k point C : 1 k point D : 12 k points W : 8 k points SW: 2 k points
Latch range (latch clear key enabled)	Sets the latch range which can be cleared by the latch clear key.	Only one range per device	No setting
Latch range (latch clear key disabled)	Sets the latch range which cannot be cleared by the latch clear key.	Only one range per device	No setting
Local device range	Set the range of local devices enclosed in the a program.	Only one range per device	No setting

(1) Change internal relays, M, to 10 k points, and data registers, D, to 1 k points. (D0 to D500: latch clear key enable; D501 to D1023: latch clear key disable)



"Mode select screen"

[3] Select "Parameter"

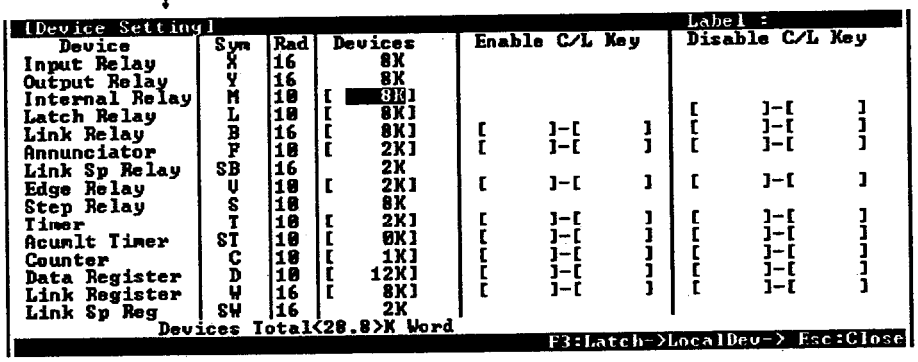


"Parameter screen"

[4] [Y] Select "Device Setting"

1)

1)



"Device Setting screen (default values)"

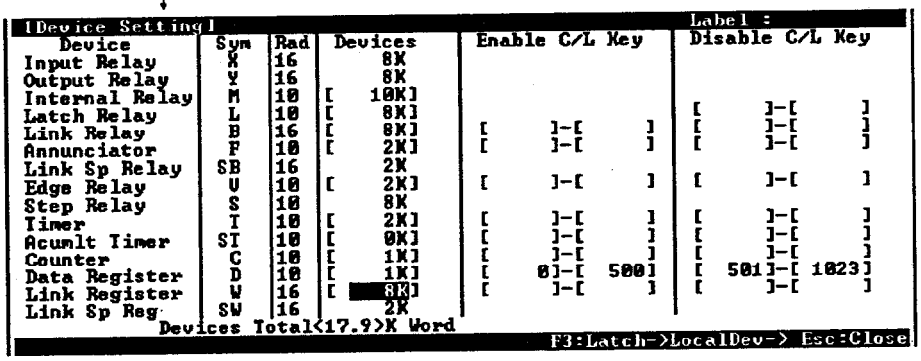
[←] [Del] [1] [0] [Enter] Change internal relays (M) from 8 k (default) to 10 k.

[↓] ... Move cursor to data register item.

[←] [Del] [Del] [1] Change data registers (D) from 12 k (default) to 1 k.

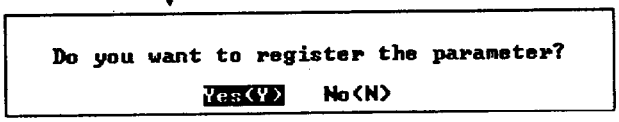
[→] [→] [0] [→] [5] [0] [0] Set latch clear key enable latch range from D0 to D500.

[→] [5] [0] [1] [→] [1] [0] [2] [3] [Enter]
Set latch clear key disable latch range from D501 to D1023.



"Device Setting screen (set values)"

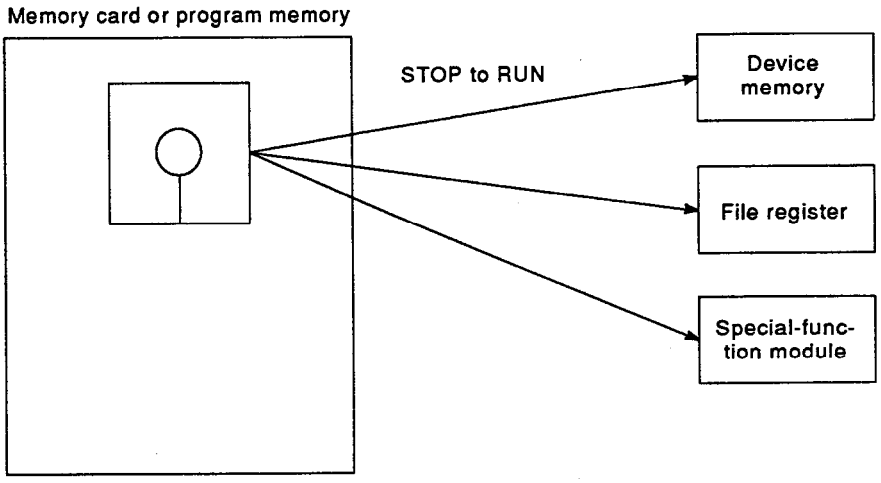
[Esc]



[Y] Register parameter

4.4.2 Setting device values required by the system as initial values

Device initial values preset in the peripheral device can be automatically transferred to the device memory, file registers, and special-function module when the CPU is switched from STOP to RUN. This eliminates the need for a data initial setting program.

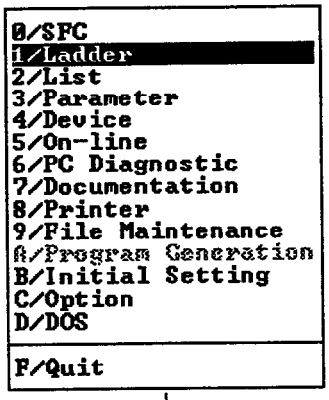


Setting the device initial values

To set the device initial values, the device initial value file must be stored in the CPU program memory or in the memory card.
Initial values can be set for the devices listed in the table below.

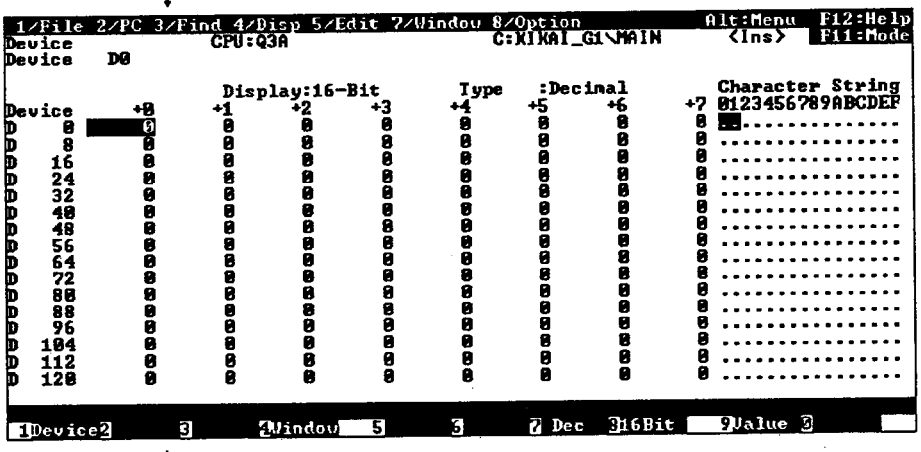
Device Name	Settable/ Not Settable	Device Name	Settable/ Not Settable	Device Name	Settable/ Not Settable	Device Name	Settable/ Not Settable	Device Name	Settable/ Not Settable
X	x	T (contact)	x	FD	x	SZ	x	U□\G	o
Y	x	T (coil)	x	B	x	S	x	J□\X	x
M	x	T (present value)	o	SB	x	TR	x	J□\Y	x
L	x	C (contact)	x	W	o	BL	x	J□\B	x
F	x	C (coil)	x	SW	o	U	x	J□\SB	x
SM	x	C (present value)	o	G	x	J	x	J□\W	o
FX	x	ST (contact)	x	R	o	ZR	o	J□\SW	o
FY	x	ST (coil)	x	P	x			BL□\S	x
V	x	ST (present value)	o	I	x			BL□\TR	x
DX	x	D	o	N	x				
DY	x	SD	o	Z	x				

(1) Set device initial values: D0=0, D1=1, D2=2, D3=3, D4=4, D5=5



"Mode select screen"

[4] Select "Device"

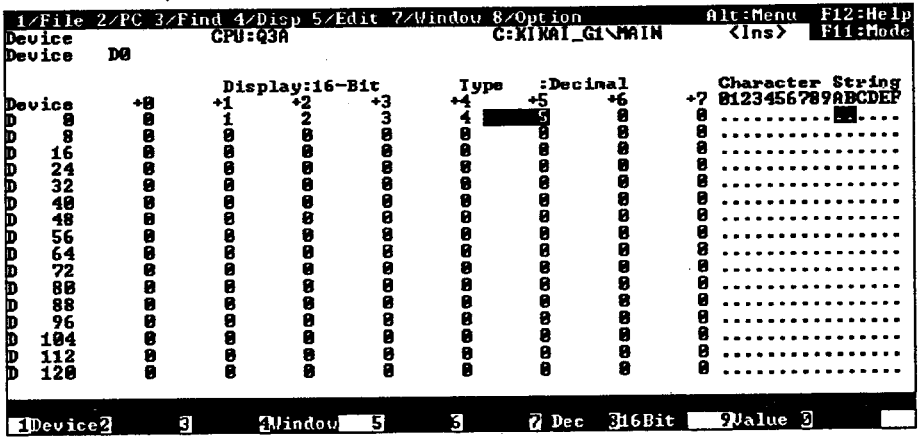


"Device setting screen (before setting)"

[→] [1] [→] [2] [→] [3] [→] [4] [→] [5]
Set D0=0, D1=1, D2=2, D3=3, D4=4, and D5=5.

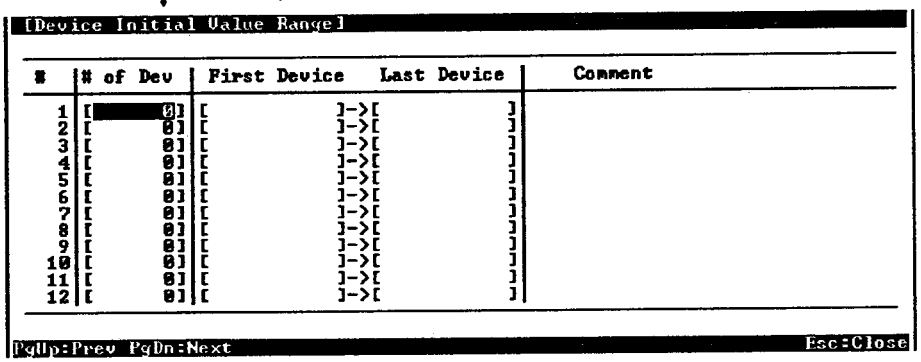
1)

1)



"Device setting screen (after setting)"

[Alt] [5] [4] Select "Device Initial Value Range"

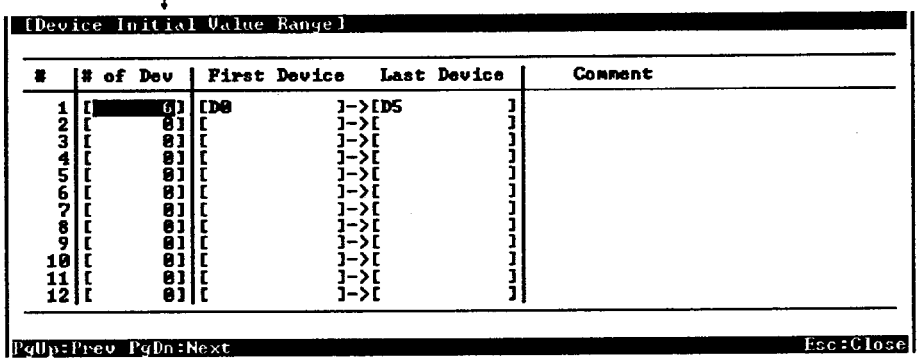


"Device Initial Value Range screen (before setting)"

[6] [→] [D] [0] [Tab] [Enter] Set device initial value range.

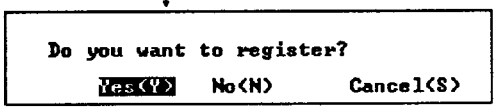
2)

2)



"Device Initial Value Range screen (after setting)"

[Esc]

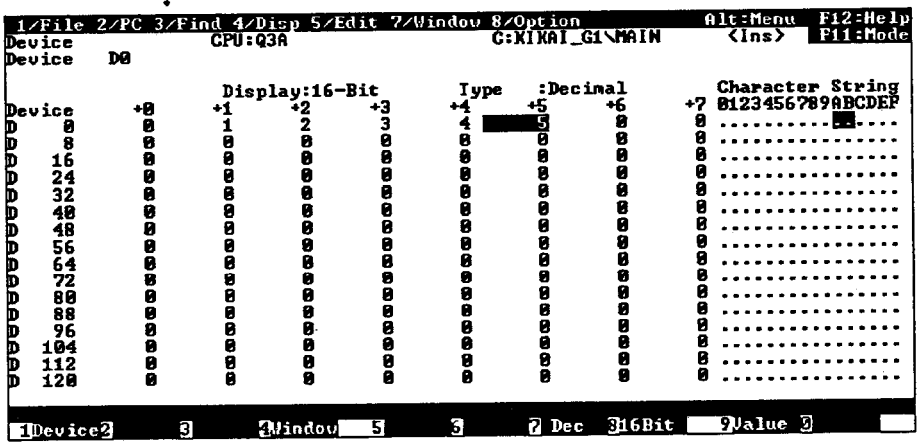


[Y] Register device initial value range settings

3)

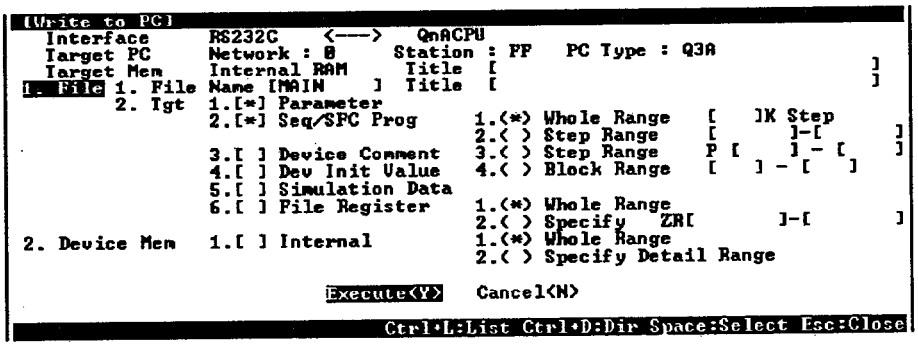
(2) Write device initial values to PC

3)



"Device setting screen"

[Alt] [2] [2] Select "Write to PC"

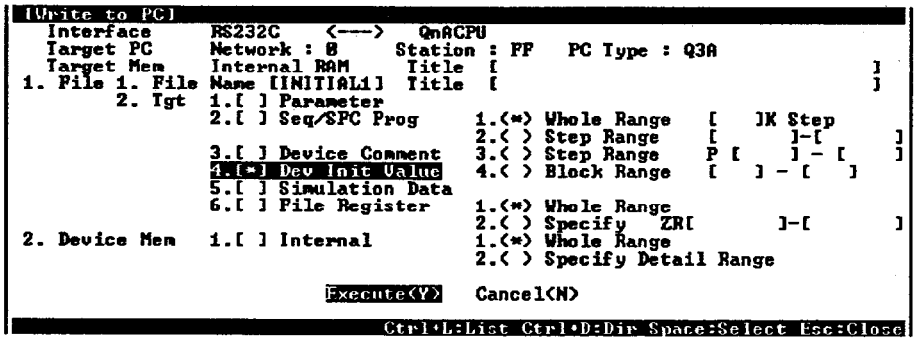


"Write to PC screen"

[SP] [SP] [BS] [BS] [BS] [BS] [I] [N] [I] [T] [I] [A] [L] [1] [Enter] [Enter] [SP] [SP] [↓] [SP] [↓] [↓] [SP]
Set file name to "INITIAL1" and select device initial values

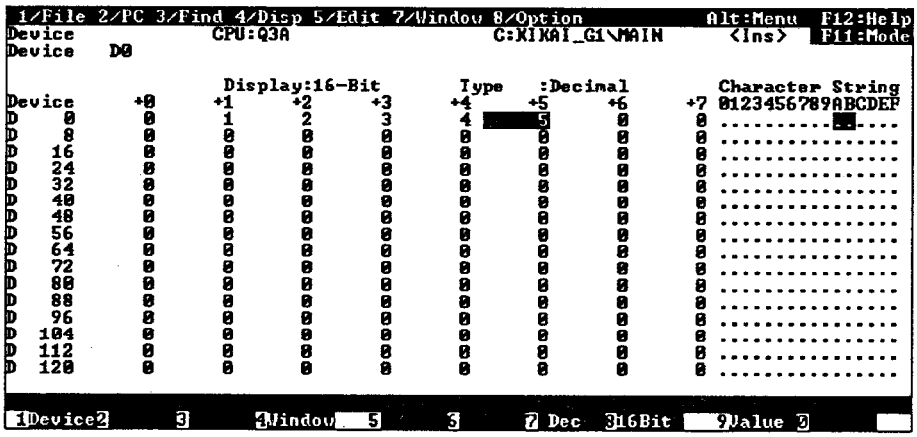
4)

4)



"Write to PC screen"

[Y] Execute "Write to PC"



"Device setting screen"

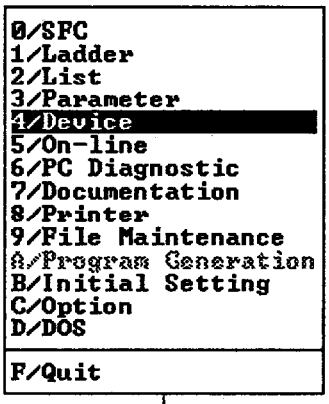
5)

(3) Set parameters

Set with the parameters which device initial value file is effective.

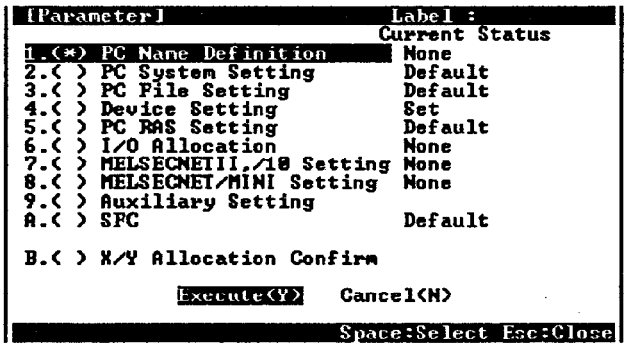
5)

[F11]



"Mode select screen"

[3] Select "Parameter"

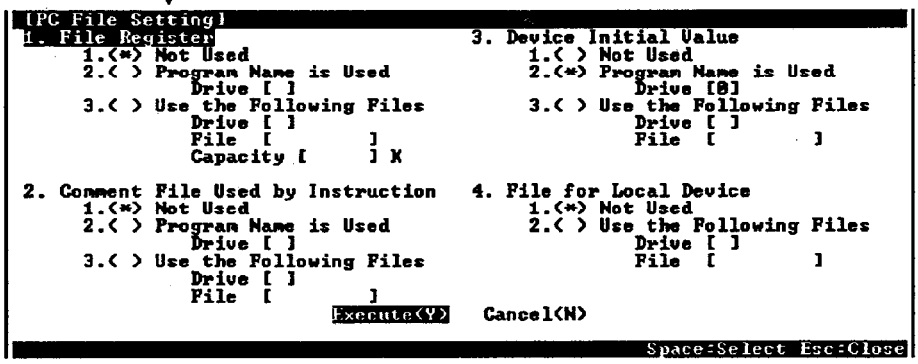


"Parameter screen"

[3] [Y] Select "PC File Setting"

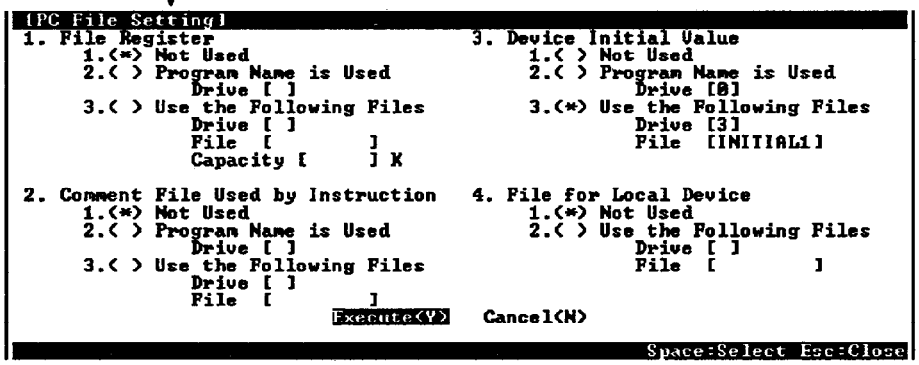
6)

6)



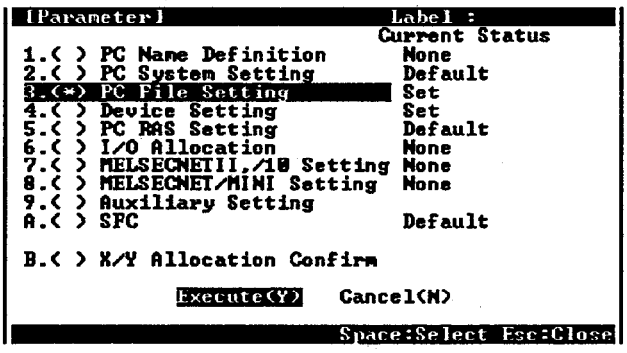
"PC File Setting screen"

[↓] [↓] [SP] [↓] [SP] [3] [↓] [I] [N] [I] [T] [I] [A] [L] [1] [Enter]
Set the memory card B RAM area file name to "INITIAL1"



"PC File Setting screen"

[Y] Set PC file



"Parameter screen"

(4) Execute

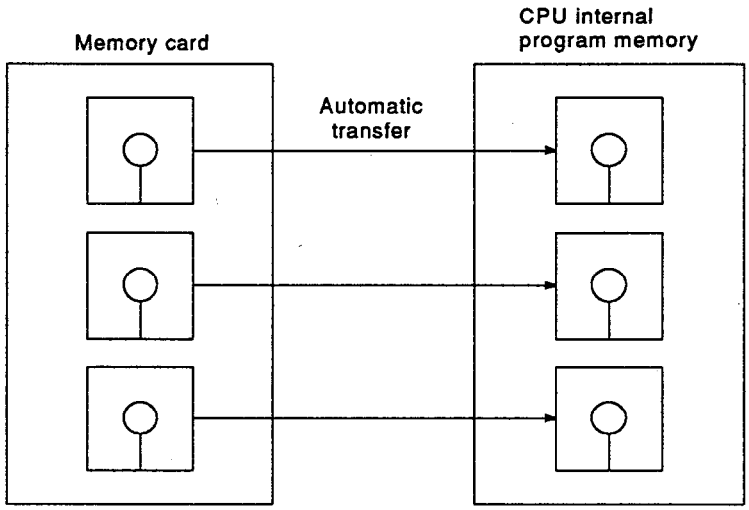
Device initial values written when CPU is switched from STOP to RUN or when the power is turned ON.

(5) Cautions

Device initial values take priority when the settings overlap a latch range.

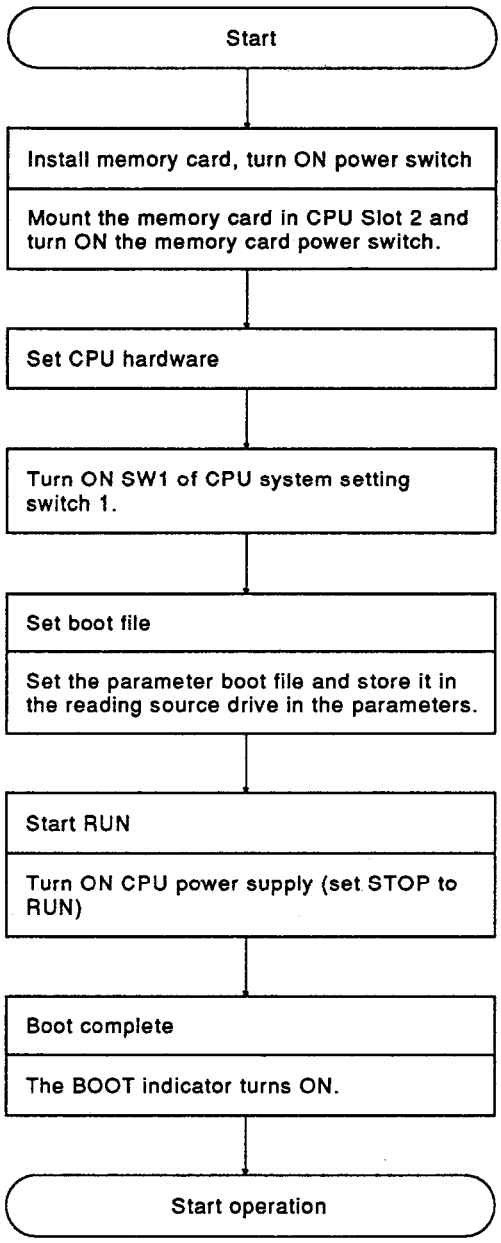
4.4.3 Boot operation with a program transferred from memory card to QnACPU internal memory

A memory card is required for boot operation.
Boot operation is possible by setting the system setting switch and the parameter boot file.
Boot operation is a function for automatic transfer from the memory card to program memory, as defined in the boot file settings, when the power is turned on or the CPU is switched from STOP to RUN.



Procedure to Set Boot Operation

The procedure to set boot operation is shown in the flowchart below.



Boot operation is described using the files in the table below.

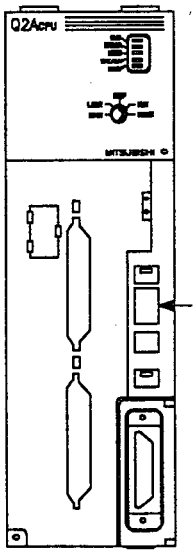
File Name	Type	Description
INITIAL	Device initial value	Device initial values
MACHINING	Sequence	Program for machining
ASSEMBLY	Sequence	Program for assembly
TRANSPORT	Sequence	Program for transporting

(1) Set switches

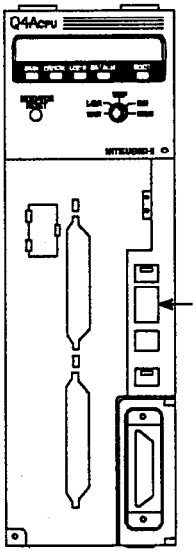
Turn ON SW1 of system setting switch 1.

<table>
<tr><th>Switch</th><th>Setting</th></tr>
<tr><td>NO1</td><td>Turn ON to enable boot operation</td></tr>
<tr><td>NO2
NO3
NO4</td><td>Designate parameter enabled parameter drive
<table>
<tr><th>NO4</th><th>NO3</th><th>NO2</th><th>Storage area</th></tr>
<tr><td>OFF</td><td>OFF</td><td>OFF</td><td>Internal memory (Drive 0:)</td></tr>
<tr><td>OFF</td><td>OFF</td><td>ON</td><td>Slot 1 RAM (Drive 1:)</td></tr>
<tr><td>OFF</td><td>ON</td><td>OFF</td><td>Slot 1 ROM (Drive 2:)</td></tr>
<tr><td>OFF</td><td>ON</td><td>ON</td><td>Slot 2 RAM (Drive 3:)</td></tr>
<tr><td>ON</td><td>OFF</td><td>OFF</td><td>Slot 2 ROM (Drive 4:)</td></tr>
<tr><td>ON</td><td>OFF</td><td>ON</td><td rowspan="3">Do not use</td></tr>
<tr><td>ON</td><td>ON</td><td>OFF</td></tr>
<tr><td>ON</td><td>ON</td><td>ON</td></tr>
</table></td></tr>
</table>

(1) Q2ACPU, Q2ACPU-S1



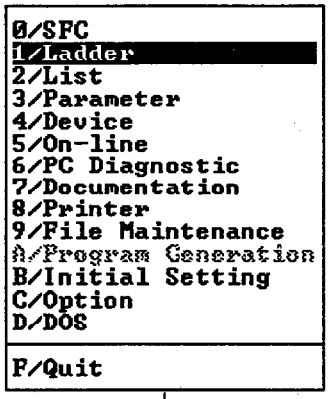
(2) Q3ACPU, Q4ACPU



* View with front cover open

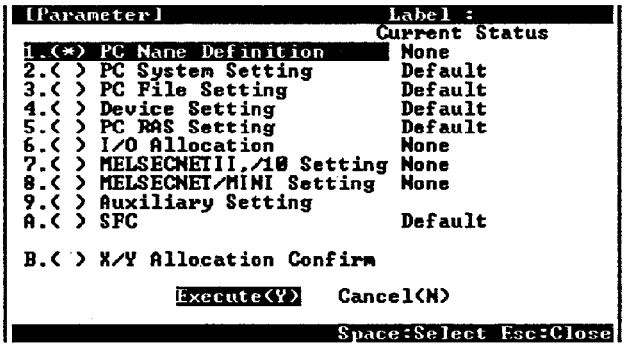
(2) Set boot file

Set the parameter boot file.



"Mode select screen"

[3] Select "Parameter"

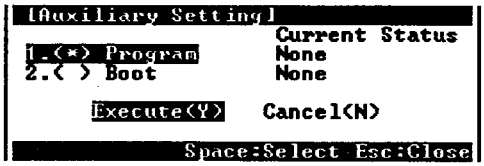


"Parameter screen"

[9] [Y] Select "Auxiliary Setting"

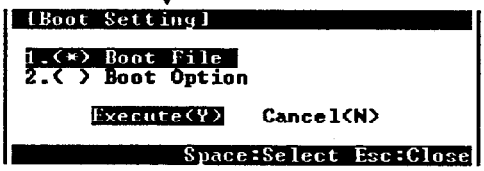
1)

1)



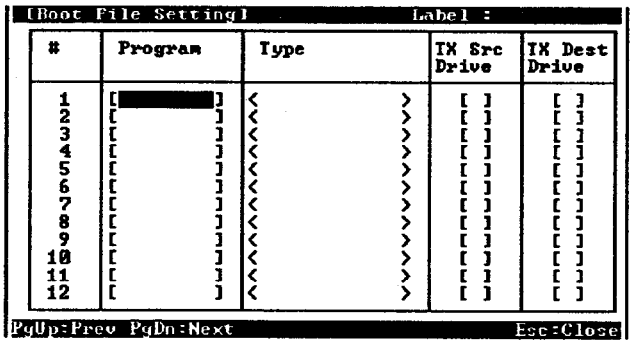
"Auxiliary Setting screen"

[2] [Y] Select "Boot"



"Boot Setting screen"

[1] [Y] Select "Boot File"



"Boot File Setting screen"
(before setting)

[I] [N] [I] [T] [I] [A] [L] [Enter] [SP] [SP] [SP] [SP] [Enter] [3] [Enter] [0] [Enter]
Set program #1

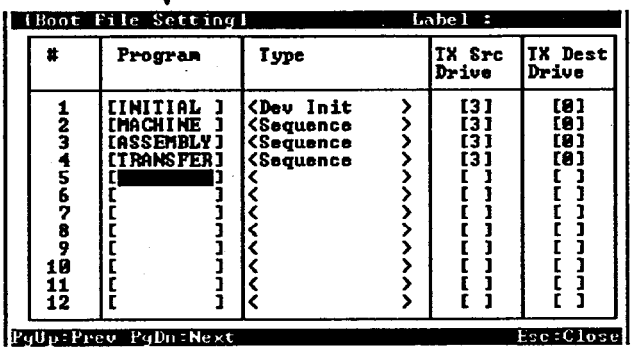
[M] [A] [C] [H] [I] [N] [E] [Enter] [Enter] [3] [Enter] [0] [Enter] Set program #2

[A] [S] [S] [E] [M] [B] [L] [Y] [Enter] [Enter] [3] [Enter] [0] [Enter]
Set program #3

[T] [R] [A] [N] [S] [F] [E] [R] [Enter] [Enter] [3] [Enter] [0] [Enter]
Set program #4

2)

2)



"Boot File Setting screen (after setting)"

[Esc]

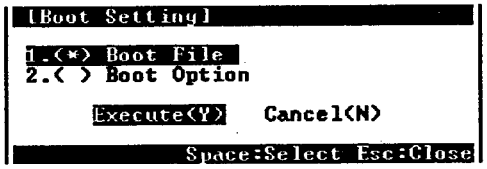
[Y] Register parameter

3)

(3) Set boot options

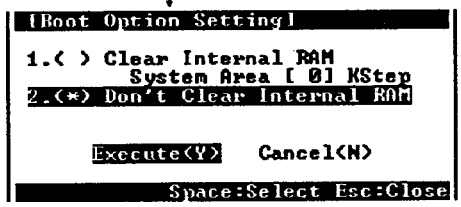
During boot operation the ACPU memory contents are cleared but for the QnACPU, program files are written to a free area of internal memory.
Set the boot options as shown below to clear the internal memory. In this case, the parameters are also cleared, so make sure that the CPU system setting switch 1 does not select the internal memory (drive 0) as the enabled parameter drive.

3)



"Boot Setting screen"

[2] [Y]



"Boot Option Setting screen"

(4) Execution

Boot operation is executed under the following conditions:

- Power turned ON (reset) when the key switch is in the RUN position;
- CPU is switched from STOP to RUN.

The sequence program is executed immediately after boot operation is complete.

(5) Cautions

- During boot operation, a program file with the same name at the transfer destination is overwritten.
- During boot operation, up to 2 or 3 seconds are required to reach RUN status.
- Boot operation is also executed after a power interruption occurs. Therefore, after boot operation has been completed once, turn OFF SW1 of CPU system setting switch 1.
- The transferred file contents may be damaged if the power is turned off or the module is reset during program transfer from the memory card to the internal memory card (boot operation).

4.4.4 Remote operation of the QnACPU from a distant location

Remote operations of the QnACPU are possible, as shown in the table below.

The relationship between the key switch positions and the permitted remote operation is also shown in the table.

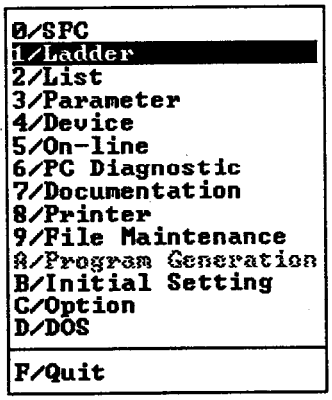
Remote Operation / Key Switch	RUN	STEP-RUN	STOP	PAUSE	RESET	Latch Clear
RUN	RUN	STEP-RUN	STOP	PAUSE	Remote operation NG	Remote operation NG
STOP	STOP	STOP	STOP	STOP	RESET	Latch clear

(1) Remote RUN/STOP

Set the key switch in the RUN position to carry out remote RUN and STOP.
Two methods are available for remote RUN and STOP:

(a) Method using remote RUN contact

Set the remote RUN contact (X) with the parameters. The CPU is in STOP status when the contact is ON or RUN status when the contact is OFF.

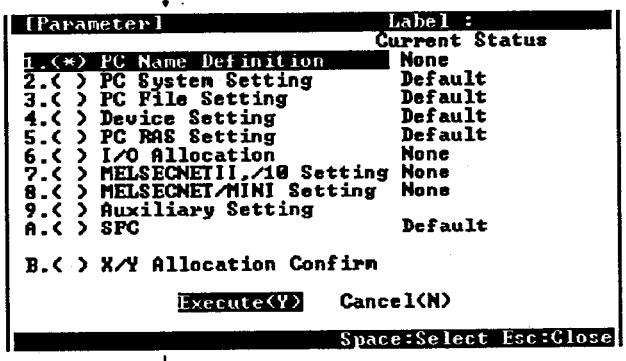


"Mode Select screen"

[3] Select "Parameter"

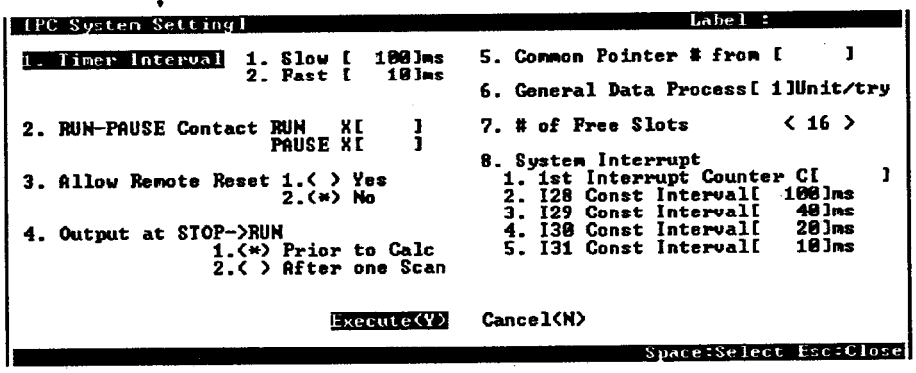
1)

1)



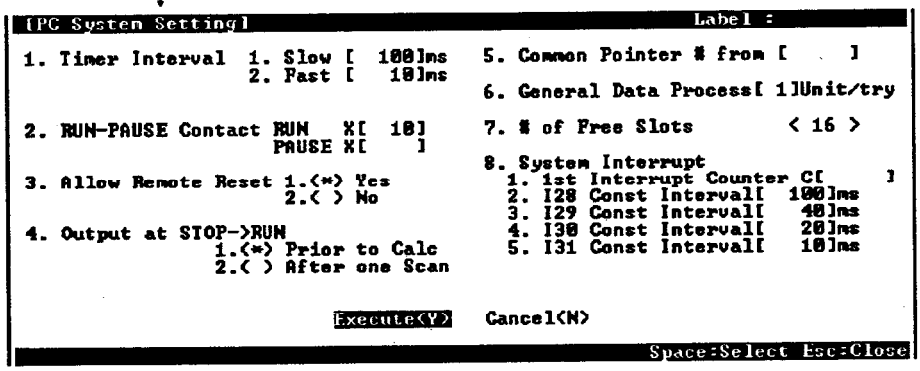
"Parameter screen"

[2] [Y] Select "PC System Setting"



"PC System Setting screen (before setting)"

[2] [1] [0] Set X10 as RUN contact

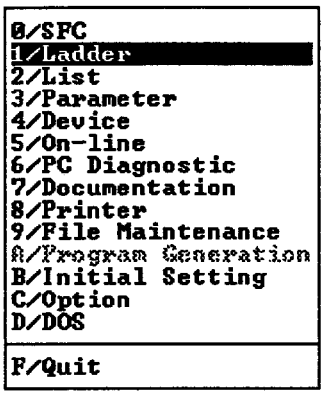


"PC System Setting screen (after setting)"

[Y] Execute PC system setting

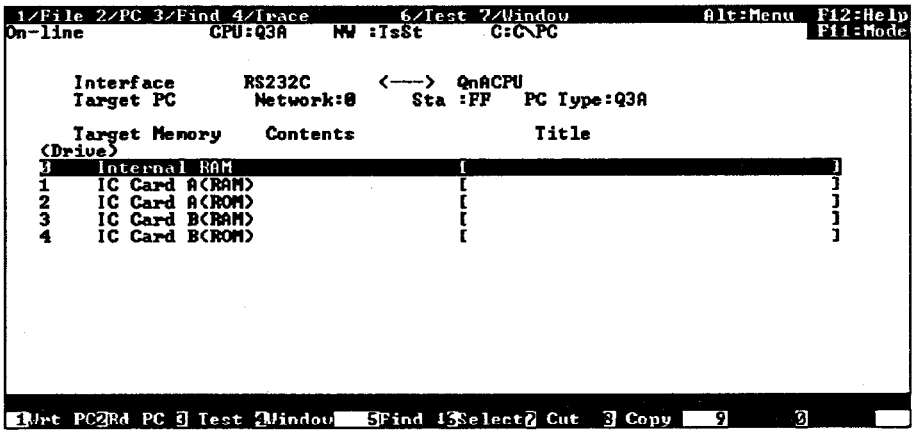
(b) Method using operations from peripheral device or special-function module

Operation is controlled with RUN and STOP instructions from a peripheral device.
The example below shows a remote STOP operation followed by a remote RUN operation.



"Mode Select screen"

[5] Select "On-line"

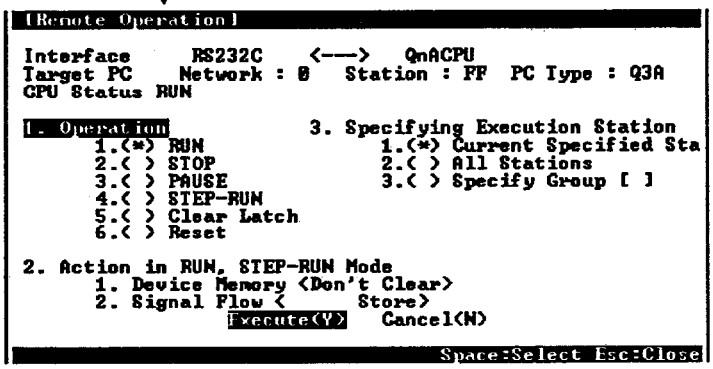


"On-line screen"

[Alt] [2] [7] Select "Remote Operation"

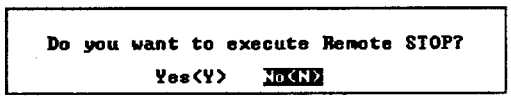
1)

1)

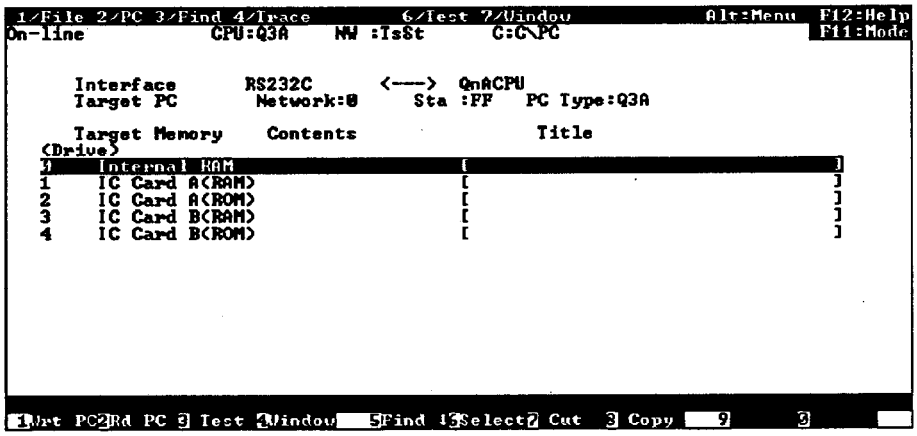


"Remote Operation screen"

[1] [2] [Y] Select "STOP"



[Y] Execute remote STOP

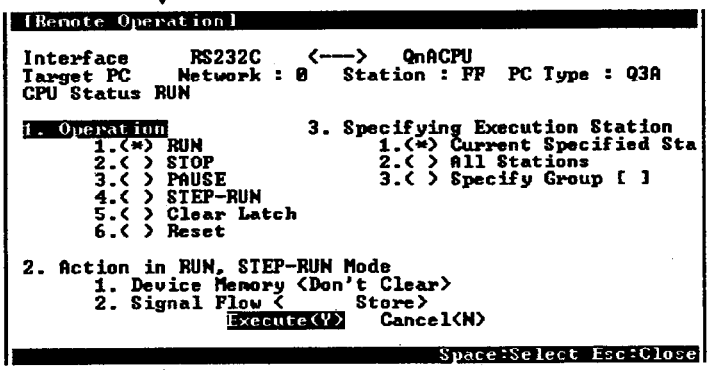


"On-line screen"

[Alt] [2] [7] Select "Remote Operation"

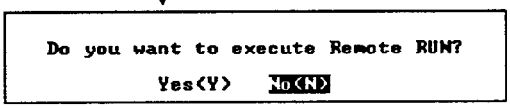
2)

2)



"Remote Operation screen"

[1] [1] [Y] Select "RUN"



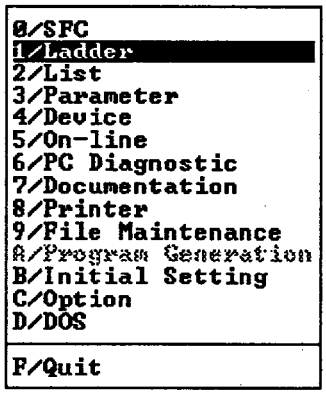
[Y] Execute remote RUN

(2) Remote PAUSE

When the remote PAUSE is executed, the CPU stops and the I/O status is held.
Set the key switch in the RUN position to carry out remote PAUSE.
Two methods are available for remote PAUSE.

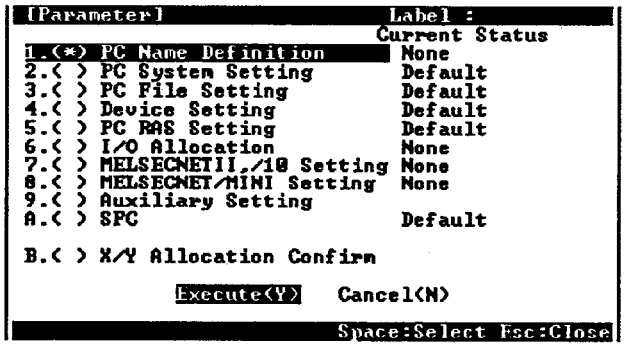
(a) Method using remote PAUSE contact

To use the remote PAUSE contact, turn ON the remote PAUSE enable coil (SM206) with the sequence program.
Set the remote PAUSE contact (X) with the parameters. The CPU is in PAUSE status when the contact is ON or RUN status when the contact is OFF.



"Mode Select screen"

[3] Select "Parameter"

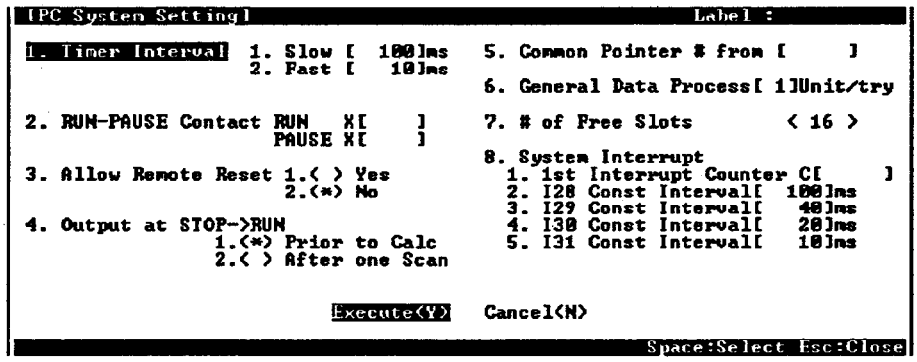


"Parameter screen"

[2] [Y] Select "PC System Setting"

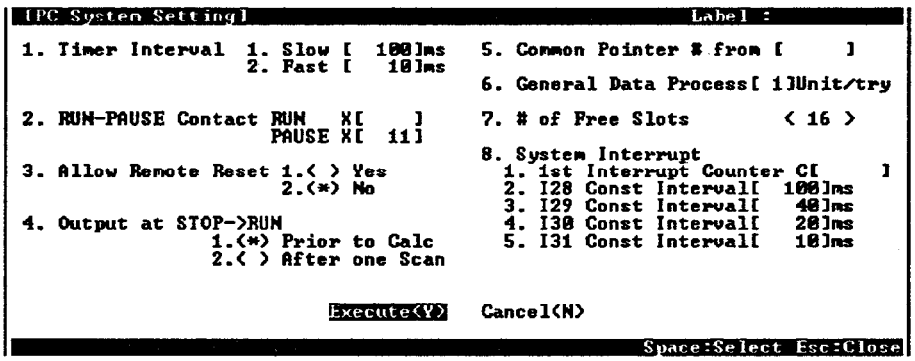
1)

1)



"PC System Setting screen (before setting)"

[2] [↓] [1] [1] Set X11 as the PAUSE contact

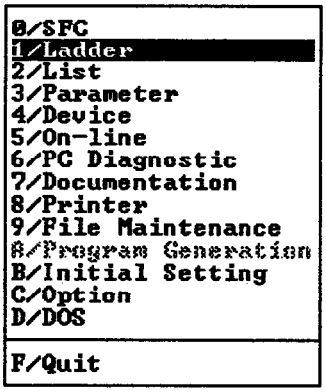


"PC System Setting screen (after setting)"

[Y] Set PAUSE contact

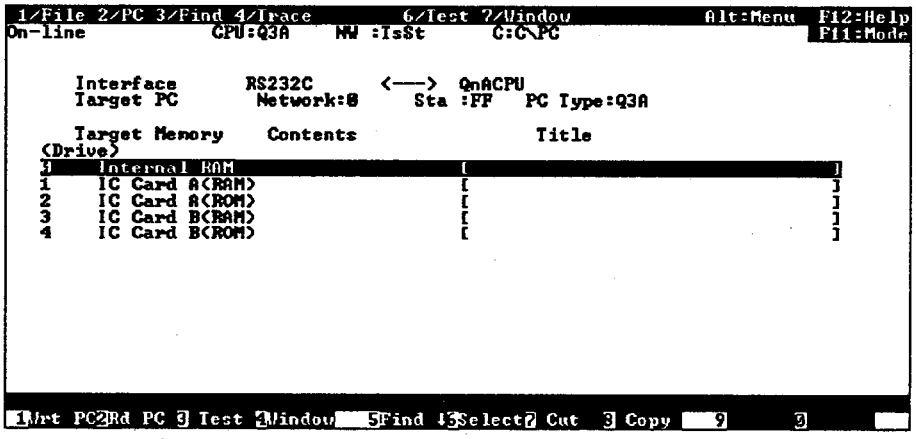
(b) Method using operations from peripheral device

Operation is controlled with the remote PAUSE instruction from a peripheral device.



"Mode Select screen"

[5] Select "On-line"

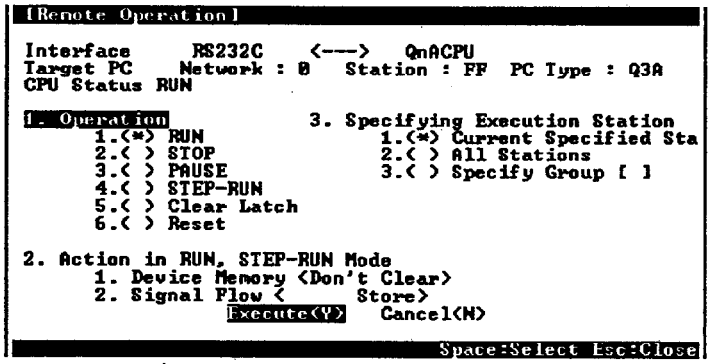


"On-line screen"

[Alt] [2] [7] Select "Remote Operation"

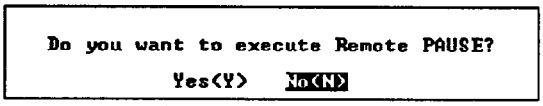
1)

1)



"Remote Operation screen"

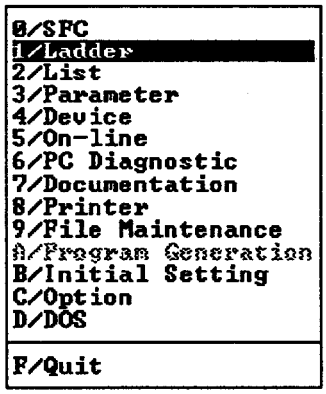
[SP] [3] [Y] Select "PAUSE"



[Y] Execute remote PAUSE

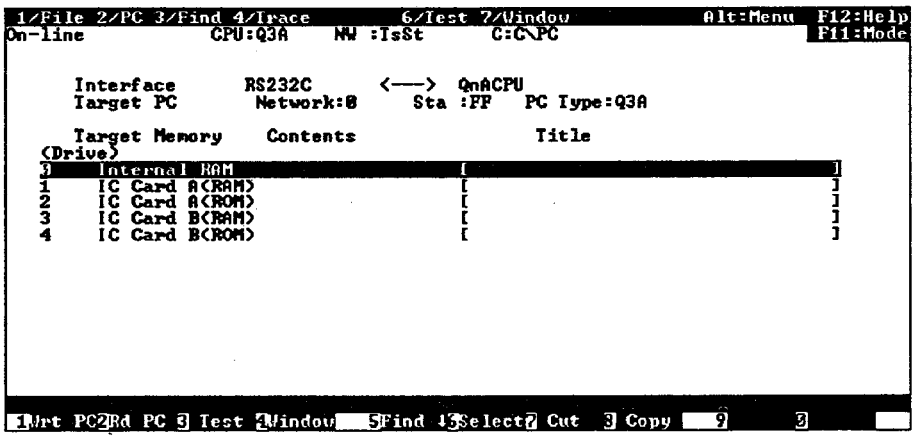
(3) Remote STEP-RUN

Operation is controlled with the remote STEP-RUN instruction from a peripheral device.



"Mode Select screen"

[5] Select "On-line"

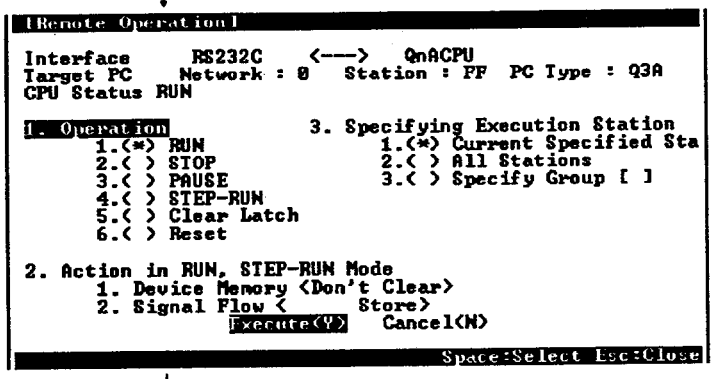


"On-line screen"

[Alt] [2] [7] Select "Remote Operation"

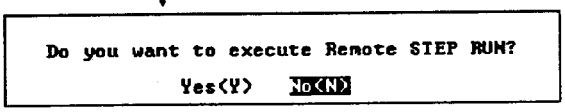
1)

1)



"Remote Operation screen"

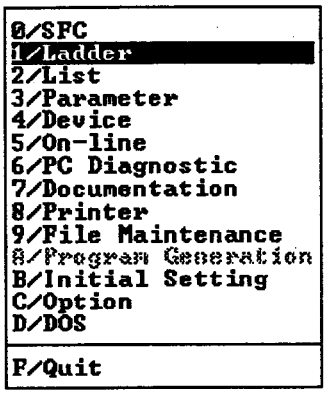
[SP] [4] [Y] Select "STEP-RUN"



[Y] Execute remote STEP-RUN

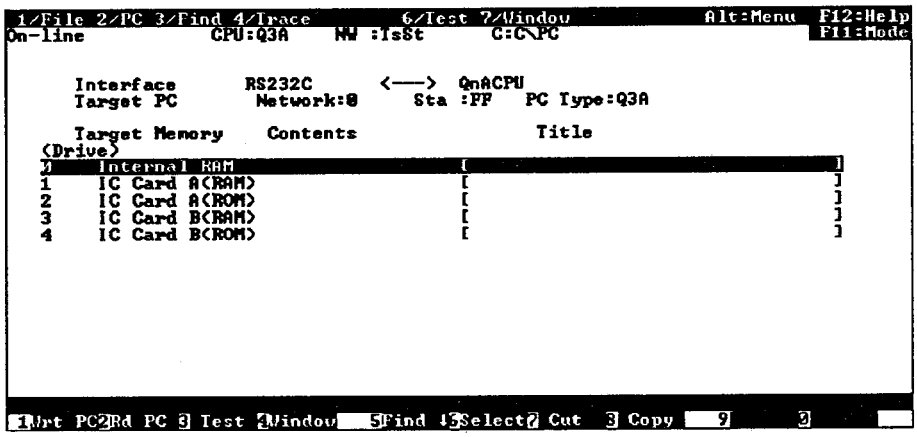
(4) Remote latch clear

Set the key switch in the STOP position to carry out remote latch clear. The remote latch clear operation is carried out from the peripheral device.



"Mode Select screen"

[5] Select "On-line"

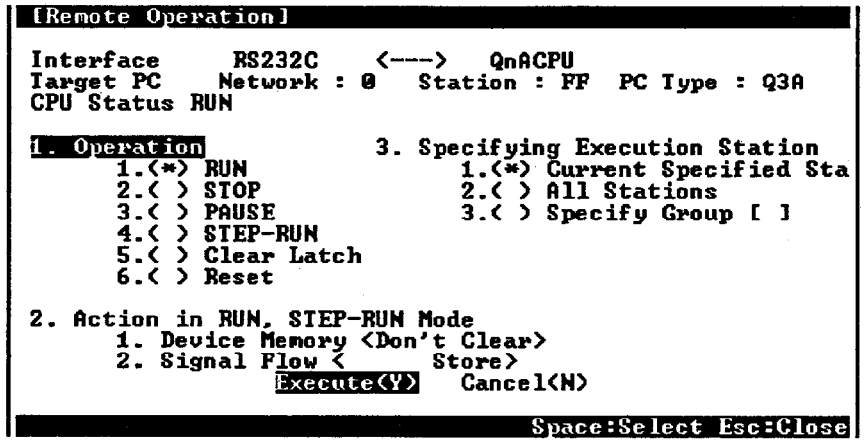


"On-line screen"

[Alt] [2] [7] Select "Remote Operation"

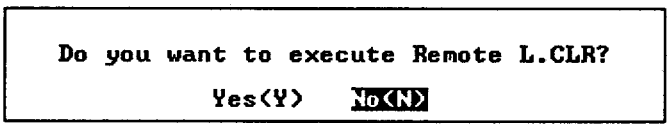
1)

1)



"Remote Operation screen"

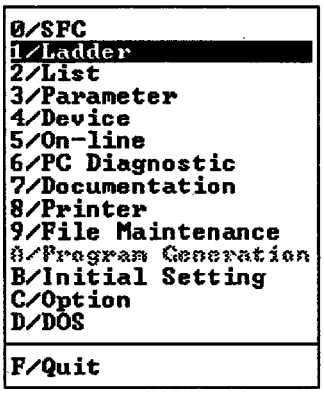
[SP] [5] [Y] Select "Clear Latch"



[Y] Execute Remote L.CLR

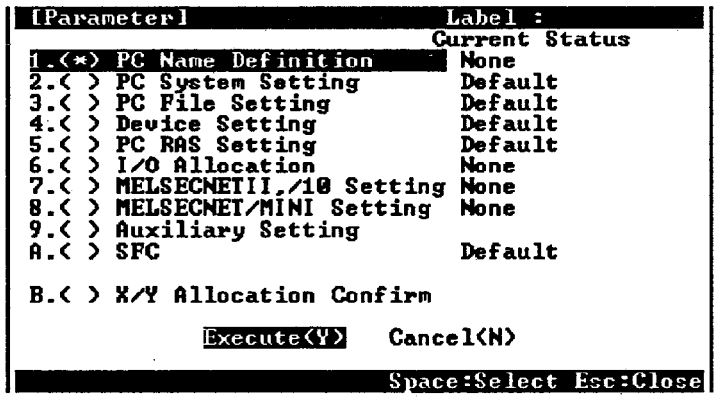
(5) Remote RESET

Set the key switch in the STOP position to carry out remote RESET. Enable remote RESET with the parameters and carry out the remote RESET operation from the peripheral device.



"Mode Select screen"

[3] Select "Parameter"

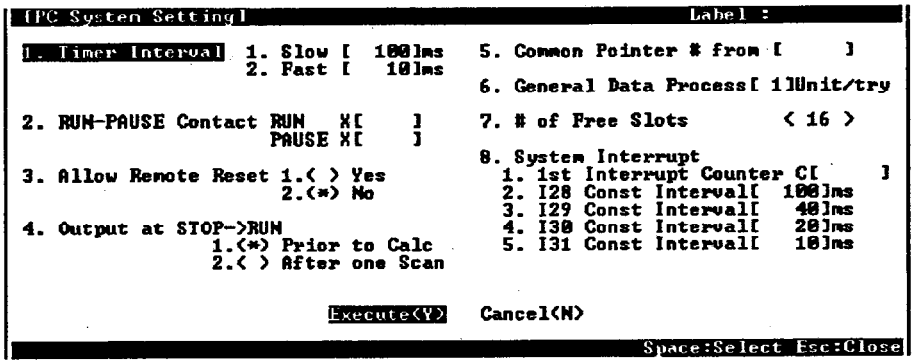


"Parameter screen"

[2] [Y] Select "PC System Setting"

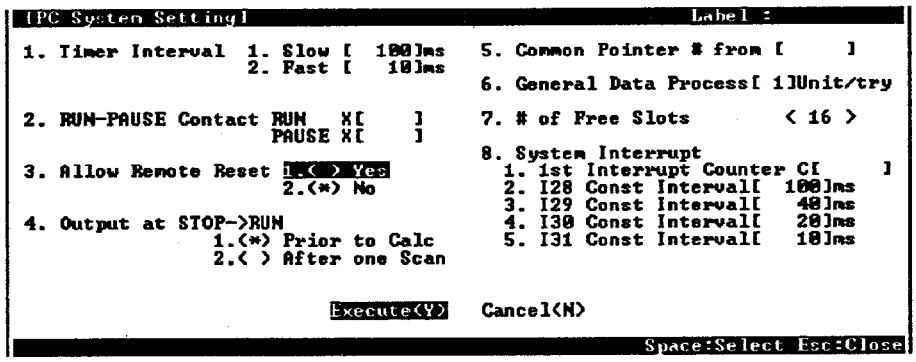
1)

1)



"PC System Setting screen (before setting)"

[3] [1] Select "Allow Remote Reset Yes"

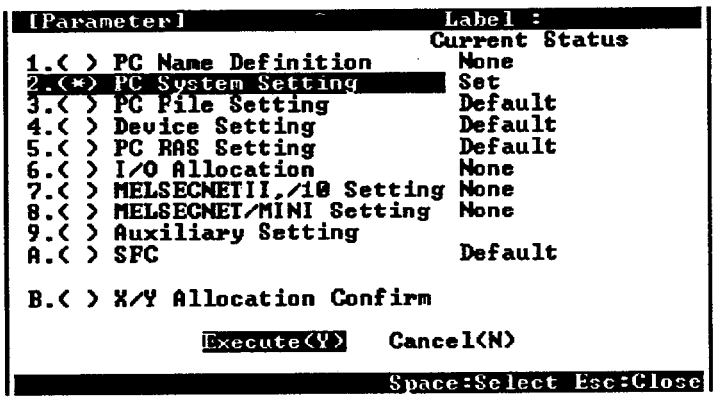


"PC System Setting screen (after setting)"

[Y] Set "Allow Remote Reset Yes"

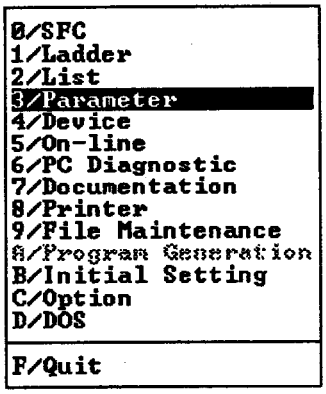
2)

2)



"Parameter screen"

[F11] Select "Mode Select screen"

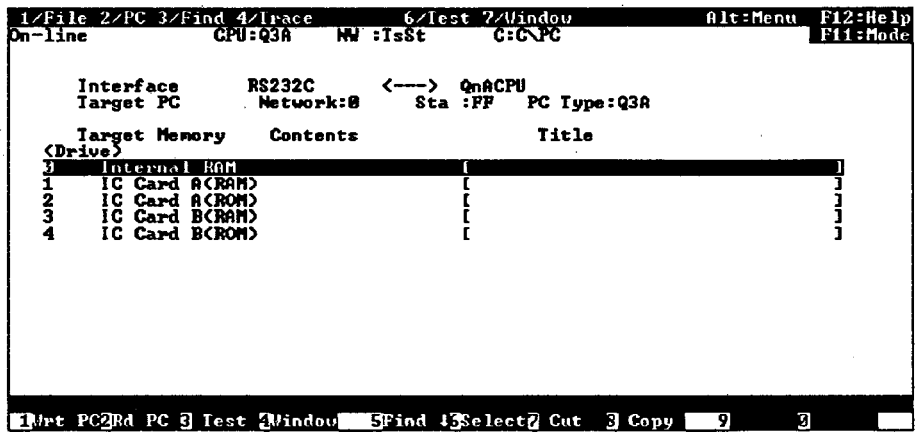


"Mode Select screen"

[5] Select "On-line"

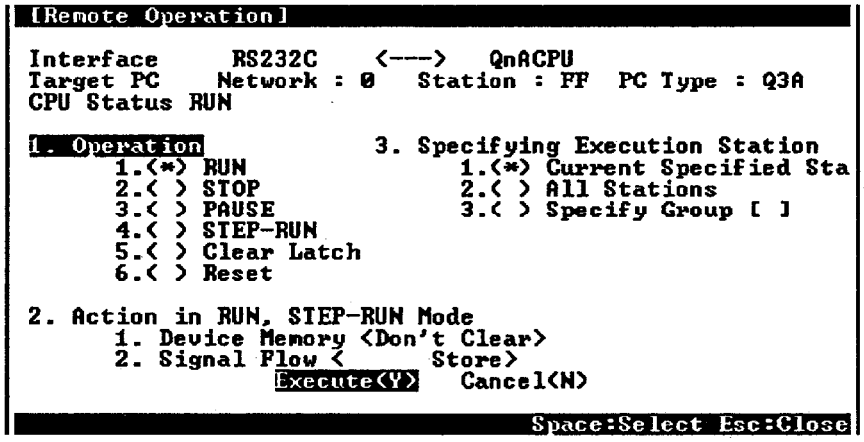
3)

3)

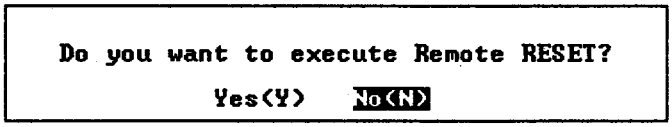


"On-line screen"

[Alt] [2] [7] Select "Remote Operation"



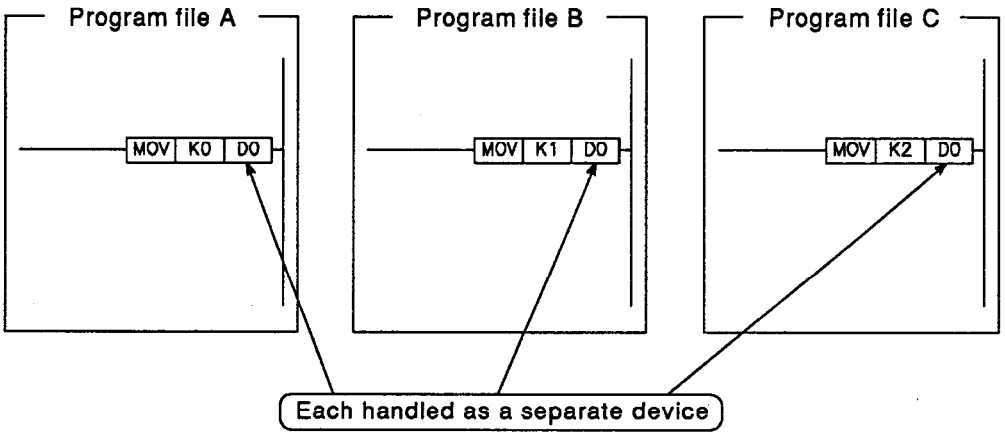
[SP] [6] Select "Reset"



[Y] Execute Remote RESET

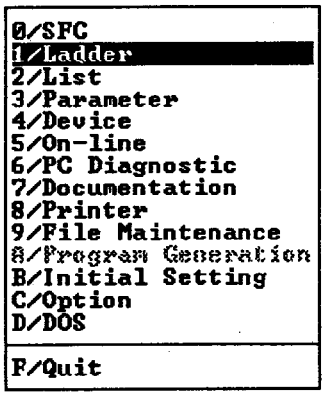
4.4.5 What are local devices?

The QnACPU is able to handle each device in each program file as a separate device. These devices are called "local devices."



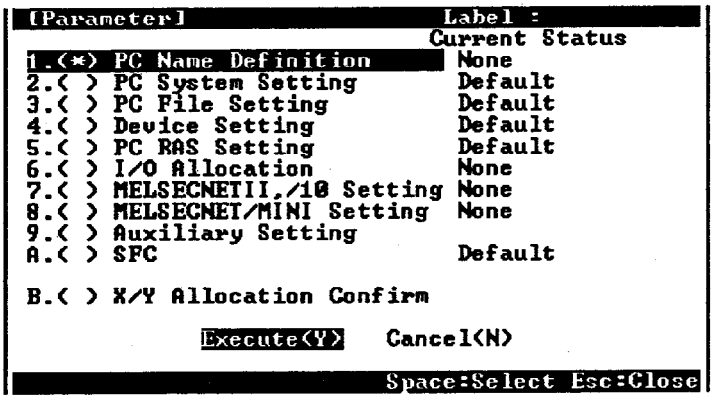
- A memory card is required to use local devices.
- Parameter settings are required to use local devices.
- The following devices can be used as local devices.
  - M : internal relay
  - V : edge relay
  - T : normal timer
  - ST : retentive timer
  - C : counter
  - D : data register

(1) Set 1 k point (M0 to M1023) of the internal relays as local devices



"Mode Select screen"

[3] Select "Parameter"

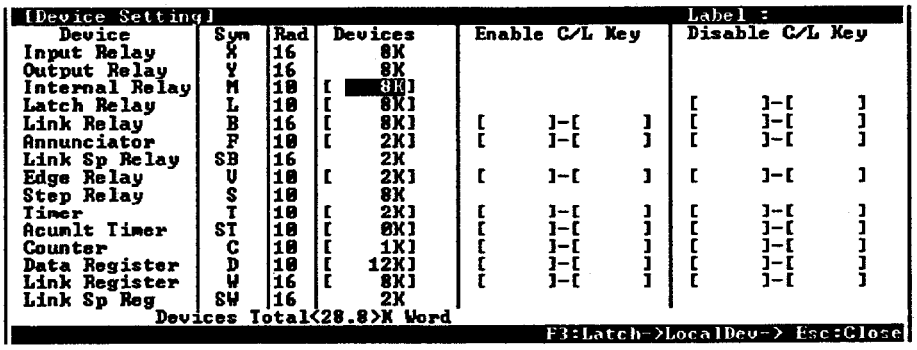


"Parameter screen"

[4] [Y] Select "Device Setting"

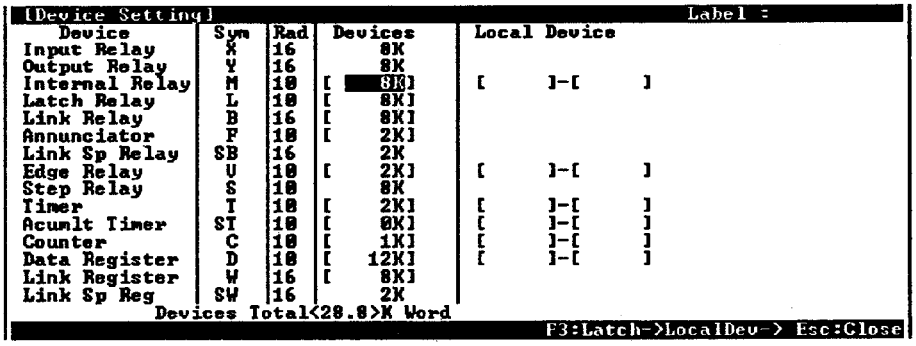
1)

1)



"Device Setting screen (before setting)"

[F3] Change from latch to local devices



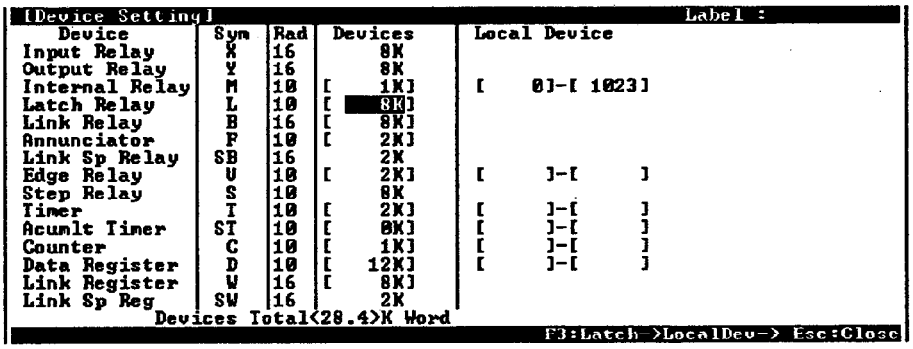
"Device Setting screen (before setting)"

[←] [Del] [Del] [1] [Tab] [0] [Tab] [1] [0] [2] [3] [Enter]

Set M0 to M1023 as local devices.

2)

2)



"Device Setting screen (after setting)"

[Esc]

[Y] Register the parameter

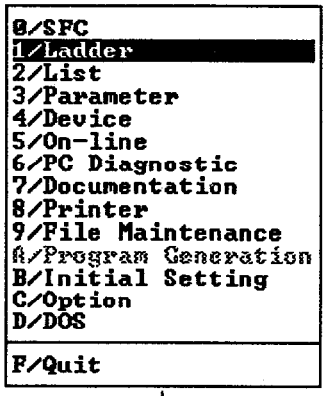
4.5 Easy Program Debugging

This section describes the easy debugging functions.

4.5.1 Simultaneous monitoring by multiple operators

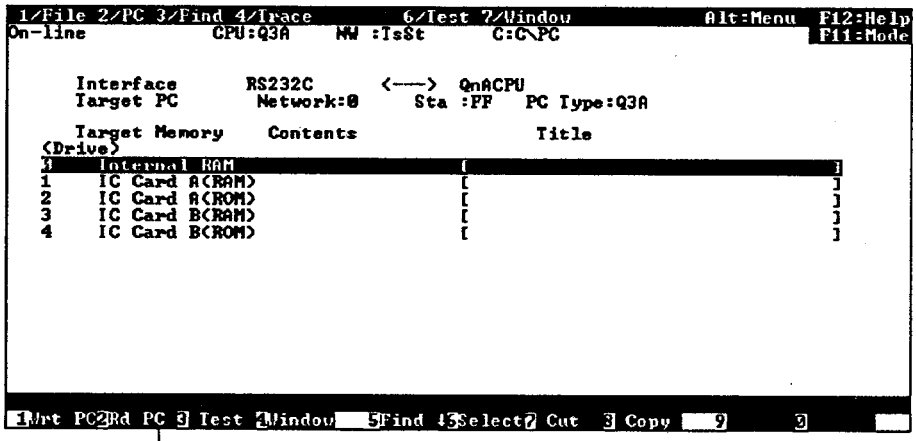
Monitoring is possible by multiple operators. Setting the other station monitor file in the system settings allows high-speed monitoring. (Setting of the host station monitor file is not required).
Each other station monitor file occupies 1 k step.

(1) Setting the other station monitor file



"Mode Select screen"

[5] Select "On-line"

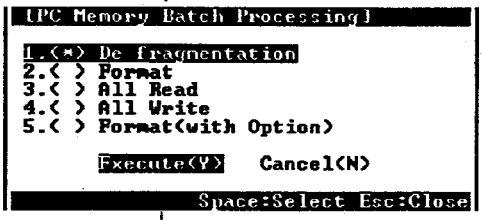


"On-line screen"

[Alt] [2] [B] Select "PC Memory Batch Processing"

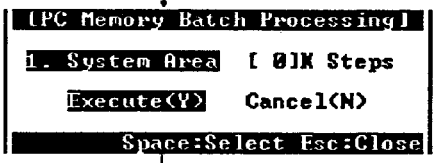
1)

1)



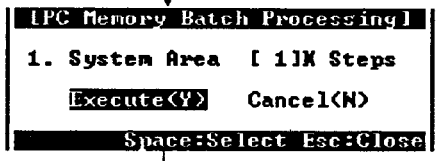
"PC Memory Batch Processing screen"

[5] [Y] Select "Format (with Option)"



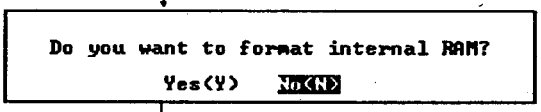
"System Area Setting screen (before setting)"

[SP] [1] [Enter] Set system area to 1 k step



"System Area Setting screen"

[Y]



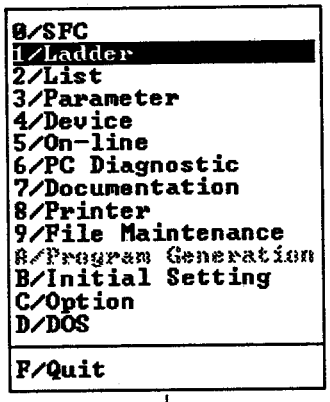
[Y]

(2) Monitor
See Chapter 3 for information about monitoring.

4.5.2 Monitoring devices at any step

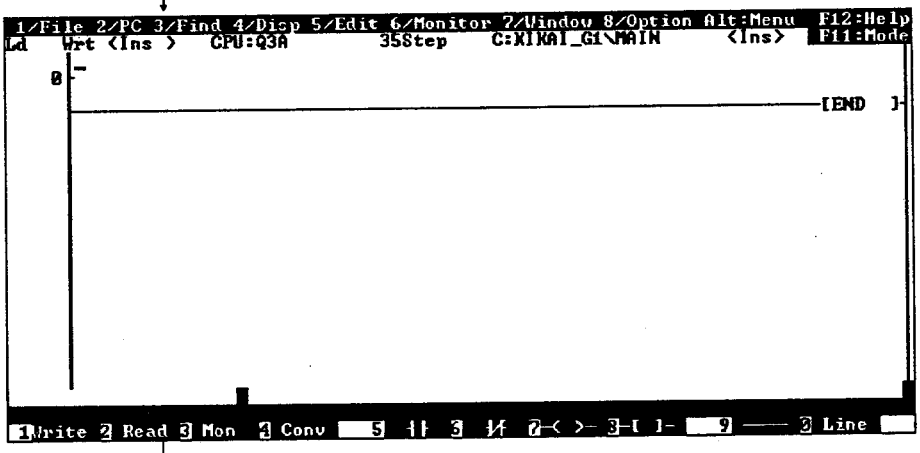
Monitor conditions can be set for precise monitoring of the PC operating status or monitoring of local devices.
In addition to monitoring during the END processing, the monitor conditions allow monitoring to be set at a step number, a step continuity state, or a device state.

(1) Read the monitored file from the PC CPU



"Mode Select screen"

[1] Select "Ladder"



"Ladder Write screen"

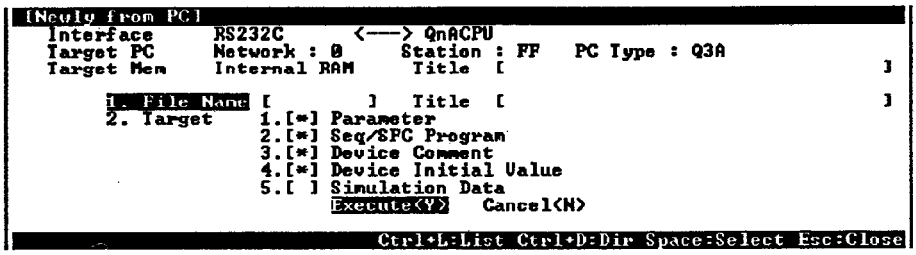
[Alt] [2] [4] Select "Newly from PC"

1)

POINT

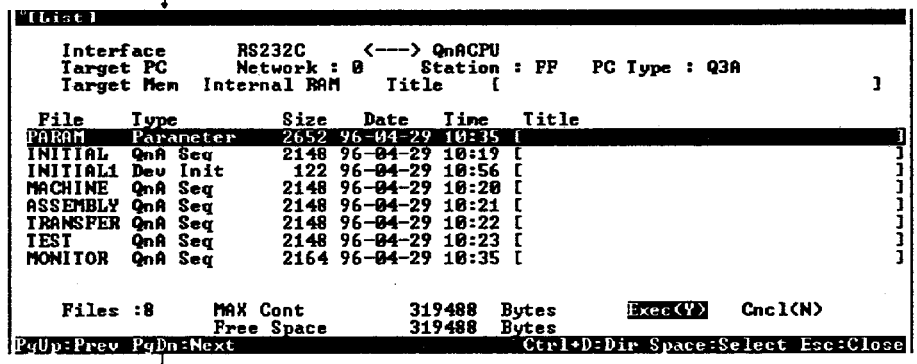
Select "Newly from PC" to match the GPP file name with the CPU file name.

1)



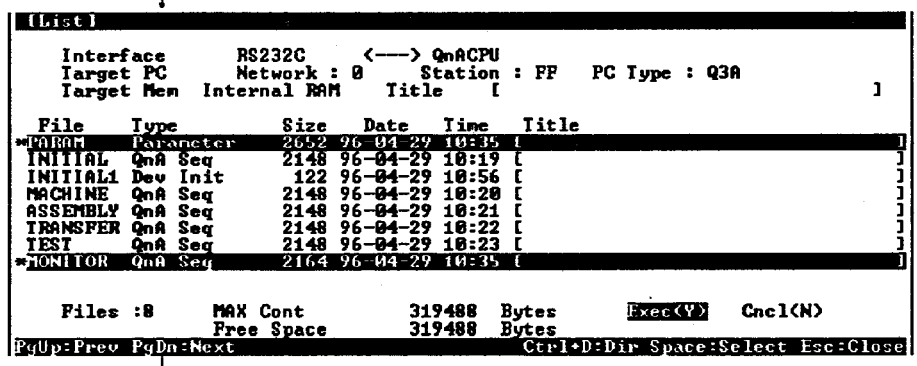
"Newly from PC screen"

[Ctrl] [L] Display file list



"File name display screen (before selecting)"

[SP] [↓] [↓] [↓] [↓] [↓] [↓] [SP]
Select parameter and the file name "MONITOR"

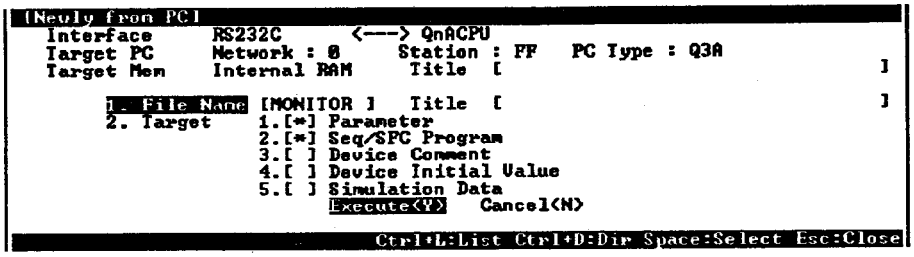


"File name display screen (after selecting)"

[Enter]

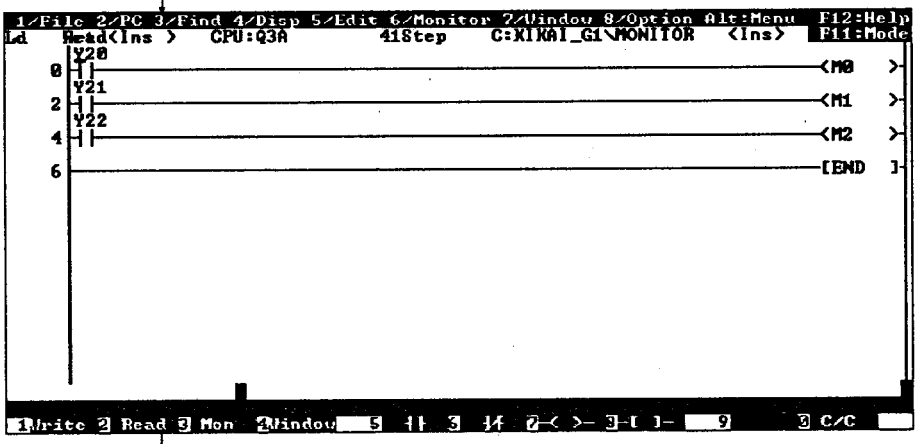
2)

2)



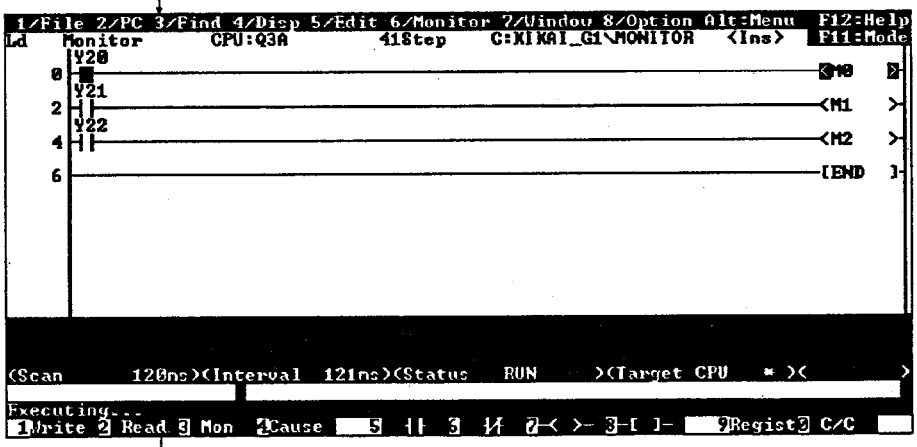
"Newly from PC screen"

[Y]



"Ladder Read screen"

[F3] Start monitoring

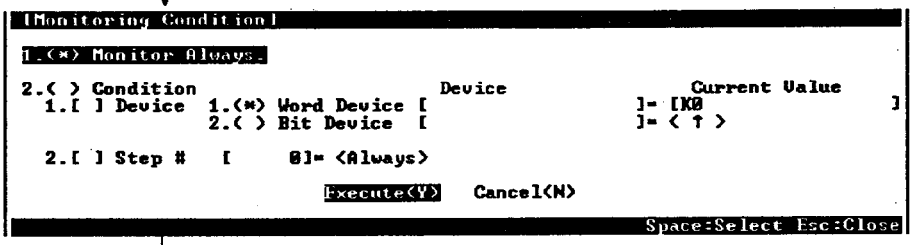


"Ladder Monitor screen"

[Alt] [6] [6] Select "Monitoring Condition"

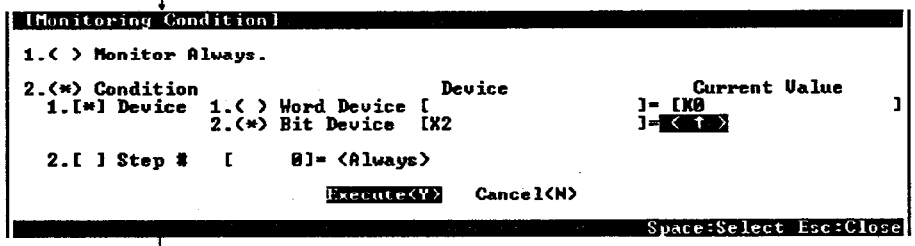
3)

3)



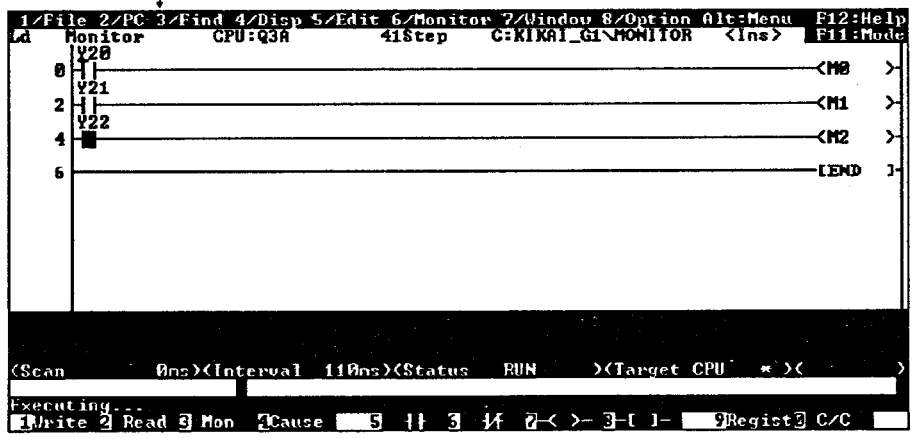
"Monitoring Condition screen (before setting)"

[2] [1] [Tab] [2] [X] [2] [Tab]
Set leading edge of bit device X2 as the monitor condition



"Monitoring Condition screen (after setting)"

[Y] Monitor with detailed conditions



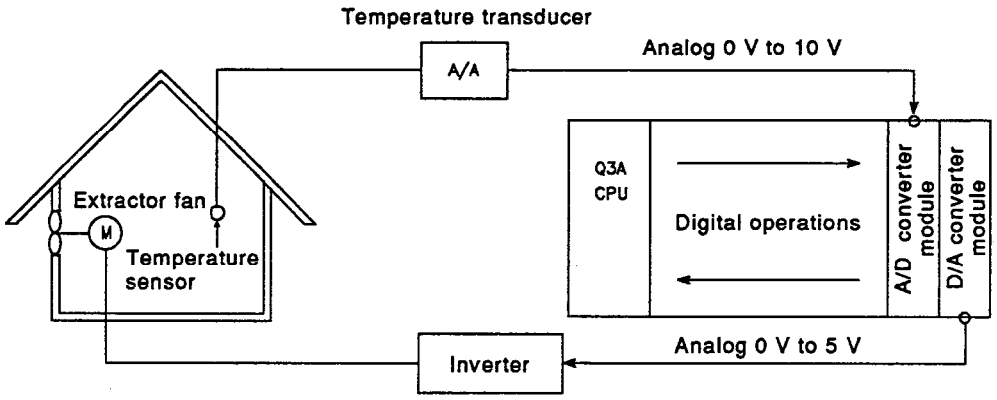
"Ladder Monitor screen"

4.6 Programming

4.6.1 Creating one program

This section shows how to create a program for the imaginary system below.

		X00 to X0F	X10 to X1F	Y20 to Y2F	Y30 to Y3F	X/Y40 to X/Y5F	X/Y60 to X/Y7F
A62P	Q3ACPU	AX40	AX40	AY40	AY40	A68AD	A62DA

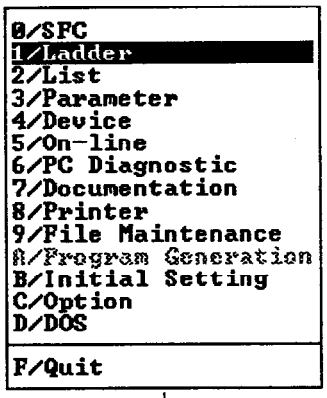


This system maintains a constant temperature in the cabin.

Cabin	• The temperature sensor measures the cabin temperature in the range from 0 °C to 50 °C.
Tempera-ture transducer	• Converts the measured temperature to a voltage between 0 V and 10 V.
A/D converter module	• Converts the 0 V to 10 V voltage to a digital value from 0 to 2000.
Q3ACPU	• Digital values are written to the D/A converter module to stop the extractor fan when the temperature is less than 20 °C, run it at high setting when the temperature exceeds 30 °C, and at low setting when the temperature is between 20 °C and 30 °C.
D/A converter module	• The digital value is converted to a voltage between 0 V and 5 V and sent to the inverter.
Inverter	• This voltage drives the extractor fan.

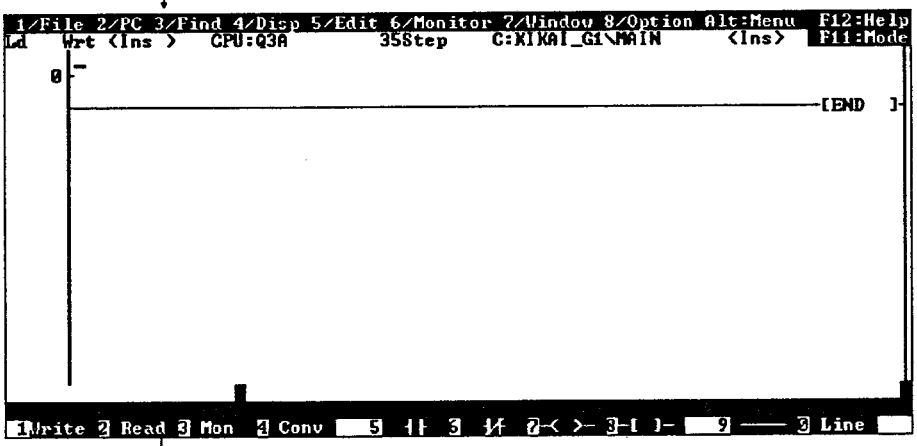
Allocation of devices used
X0 ..... Stop switch
X41 ..... A/D converter READY signal
D0 ..... Stores cabin temperature
X61 ..... D/A converter READY signal
Y20 to Y2F... Cabin temperature display

(1) Create temperature regulation program



"Mode Select screen"

[1] Select "Ladder"



"Ladder write screen (before programming)"

[Ins] Change Write (insert) to Write (overwrite)

[F6] [X] [0] [Enter] [F5] [X] [4] [1] [Enter] " X0 X14 " Write

( ) ( )

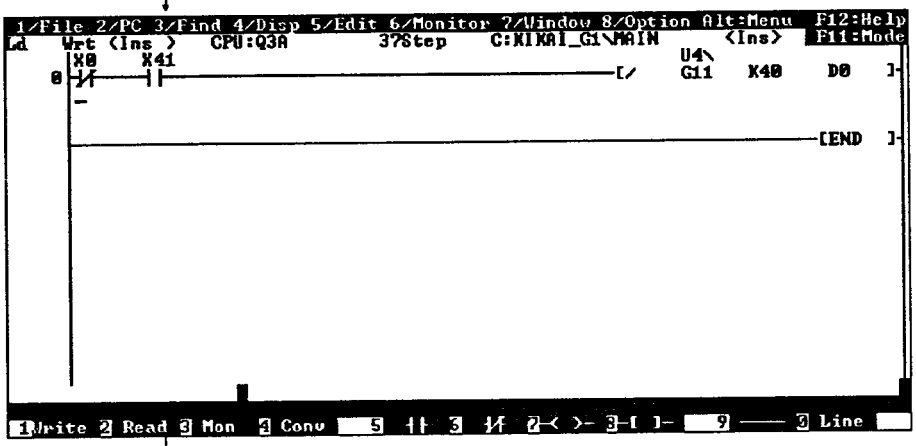
[F8] [/] [SP] [U] [4] [\] [G] [1] [1] [SP]

( )

[K] [4] [0] [SP] [D] [0] [Enter] " U4\G1 K40 D0 " Write

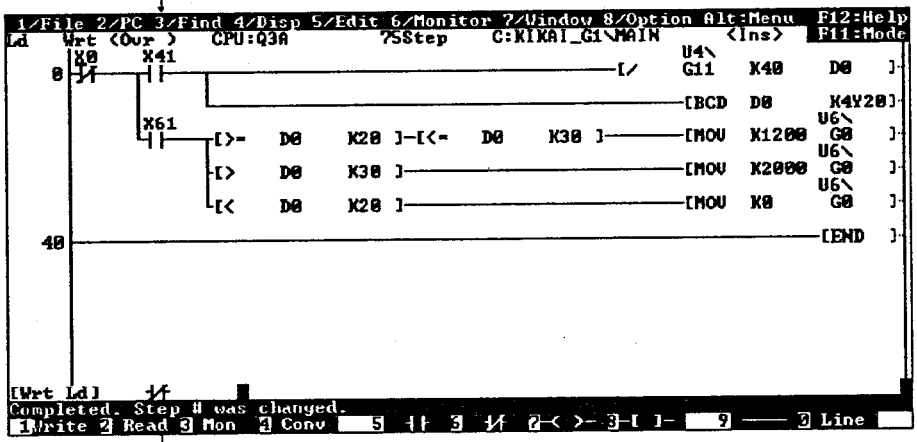
1)

1)



"Ladder Write screen (during programming)"

Continue this process to create the following ladder program.
[4] Convert



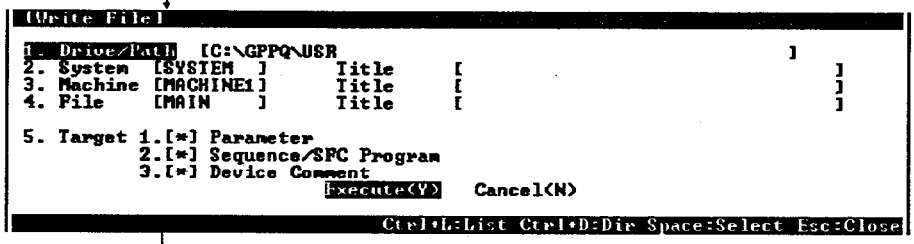
"Ladder Write screen (after programming)"

2)

(2) Write created program to file

2)

[Alt] [1] [2] Select "Write File"

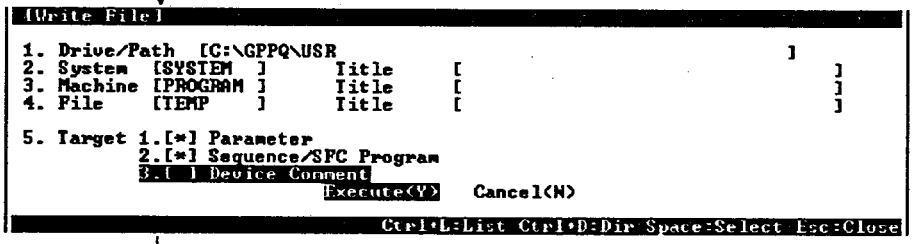


"Write File screen (before setting)"

[3] [BS] [BS] [BS] [BS] [BS] [BS] [BS] [BS]
[P] [R] [O] [G] [R] [A] [M] [Enter] Change machine name to "PROGRAM"

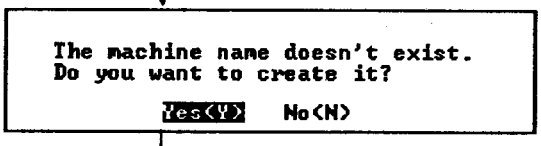
[4] [BS] [BS] [BS] [BS] [T] [E] [M] [P] [Enter] Change file name to "TEMP"

[5] [1] [3] Change write object to sequence program



"File Write screen (after setting)"

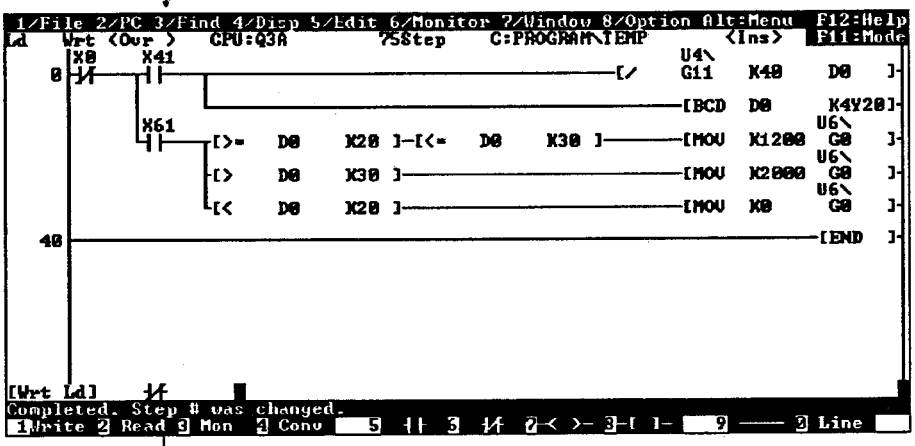
[Y] Write file



[Y] Create new machine name

3)

3)



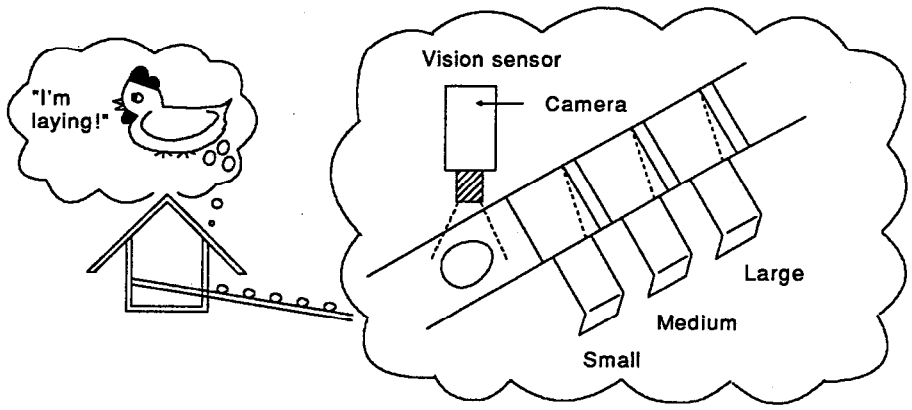
"Ladder Write screen"

[F11] Display mode select screen

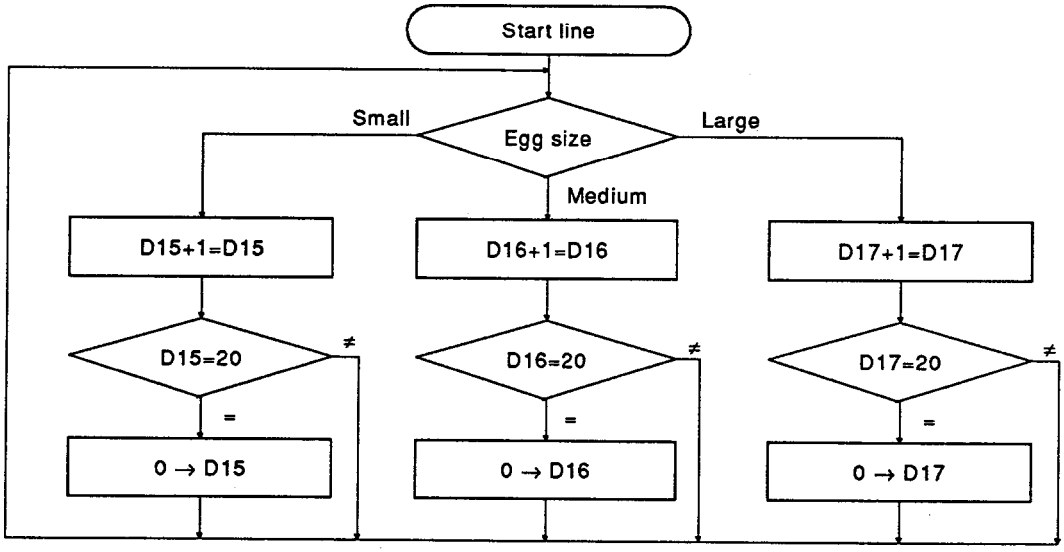
4)

4.6.2 Creating multiple programs

This section shows how to create a program to add extra functions to the system described in section 4.6.1.



Classifies the eggs laid by the chickens into small, medium, and large sizes, and packs them as tens into boxes.

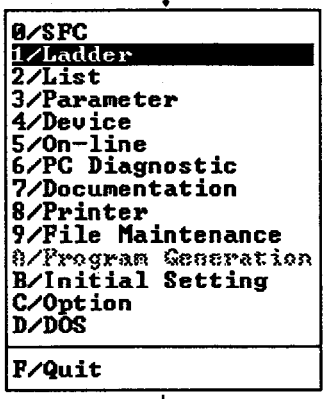


Allocation of devices used

- X10... Line start switch
- X11... Line stop switch
- Y30... Line start output
- M15 .. Small egg counter sensor
- M16 .. Medium egg counter sensor
- M17 .. Large egg counter sensor
- Y32... Extract small eggs
- Y33... Extract medium eggs
- Y34... Extract large eggs
- M10... Line start internal relay
- M11... Line stop internal relay
- D15... Number of small eggs storage
- D16... Number of medium eggs storage
- D17... Number of large eggs storage
- Y35... Change to small egg pack
- Y36... Change to medium egg pack
- Y37... Change to large egg pack

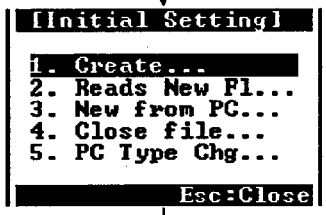
(1) Create size-selection program

4)



"Mode Select screen"

[B] Select "Initial Setting"

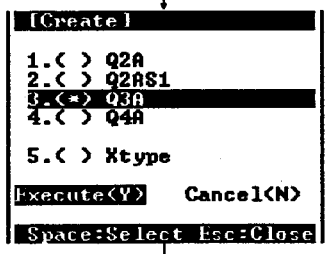


"Initial Setting screen"

POINT

The program created in Section 4.6.1 remains because up to four programs can be created simultaneously with the GPP.

[1] Select "Create..."

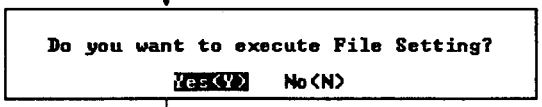


"Create screen"

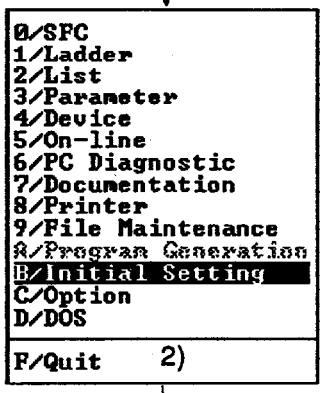
[3] [Y] Select "Q3A"

5)

5)



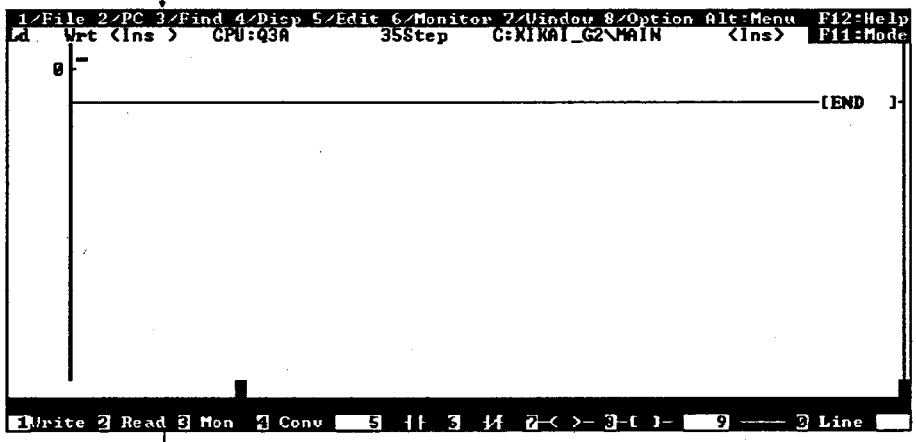
[N]



"Mode Select screen"

2)

[1] Select "Ladder"



"Ladder Write screen (before programming)"

[Ins] Change Write (insert) to Write (overwrite)

[F5] [Shift] + [F7] [X] [1] [0] [Enter] " X10 " Write

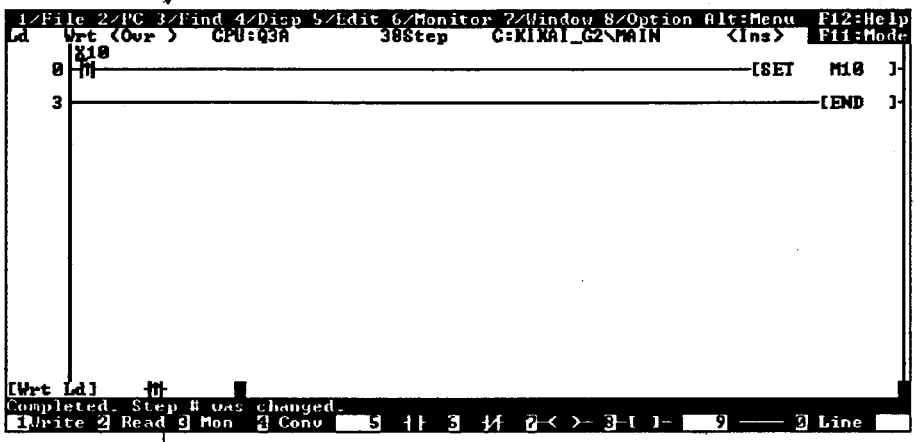
(⊣⊢) (↑)

[F8] [S] [E] [T] [SP] [M] [1] [0] [Enter]

(⊣⊢) [SET M10] " Write

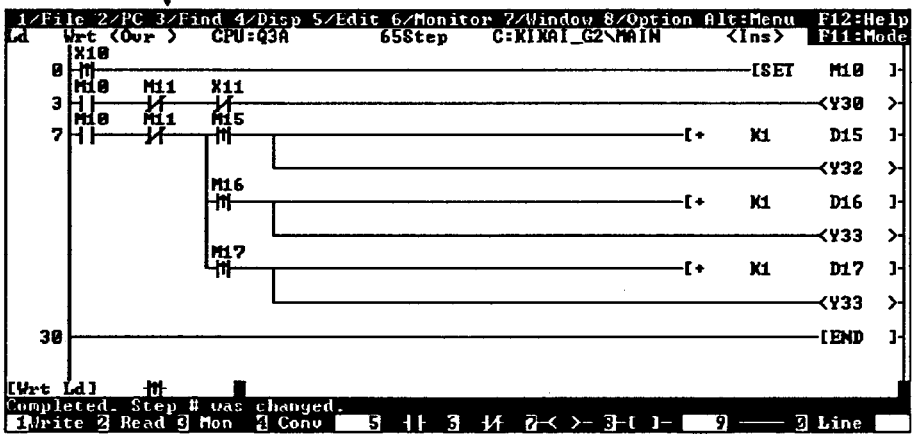
6)

6)



"Ladder Write screen (during programming)"

Continue this process to create the following ladder program.
[F4] Convert

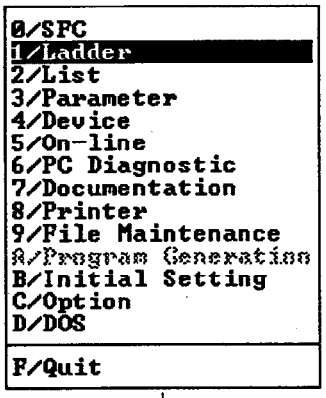


"Ladder Write screen (after programming)"

(2) Set the file name to "SIZE" and write the file.
The operation is the same as described in Section 4.6.1(2).

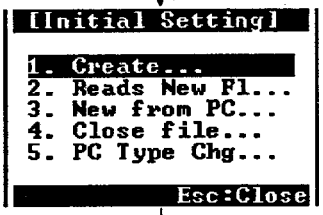
(3) Write program to the CPU.
See Section 3.5.2 for details about the write operation.

(4) Create the packing program



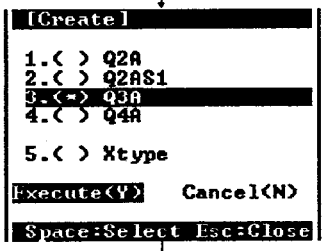
"Mode Select screen"

[B] Select "Initial Setting"



"Initial Setting screen"

[1] Select "Create..."

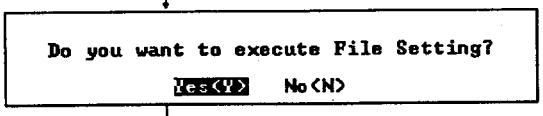


"Create screen"

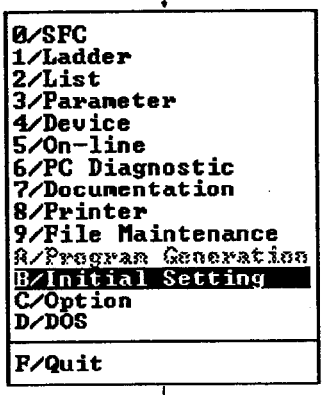
[3] [Y] Select "Q3A"

1)

1)

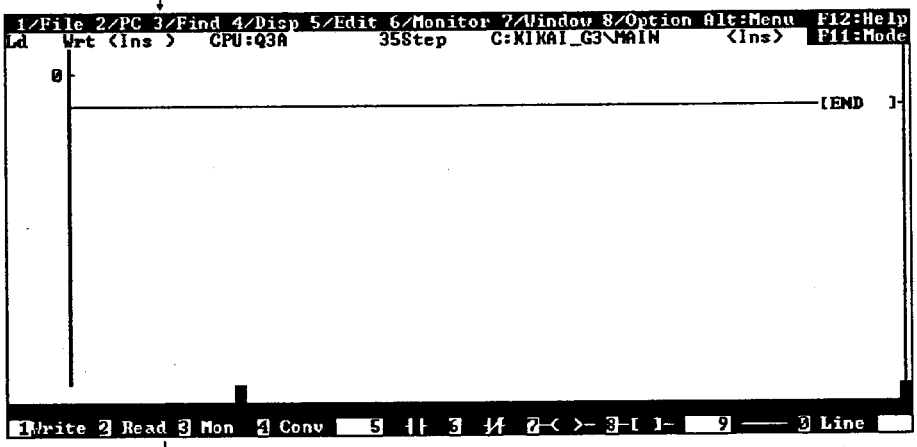


[N]



"Mode Select screen"

[1] Select "Ladder"



"Ladder Write screen (before programming)"

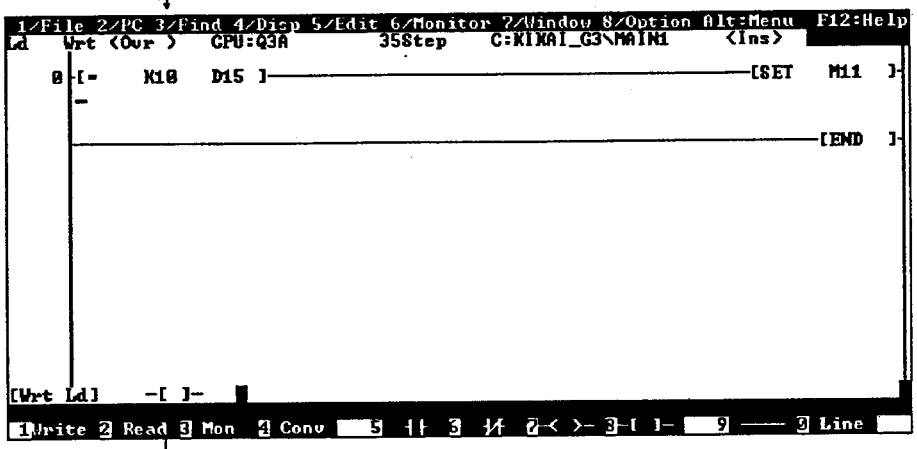
[Ins] Change Write (insert) to Write (overwrite)

[F8] [Shift] [=] [SP] [K] [1] [0] [SP] [D] [1] [5] [Enter]
(↵)
"-[= K10 D15]-" Write

[F8] [S] [E] [T] [SP] [M] [1] [1] [Enter]
(↵)
"-[SET M11]-" Write

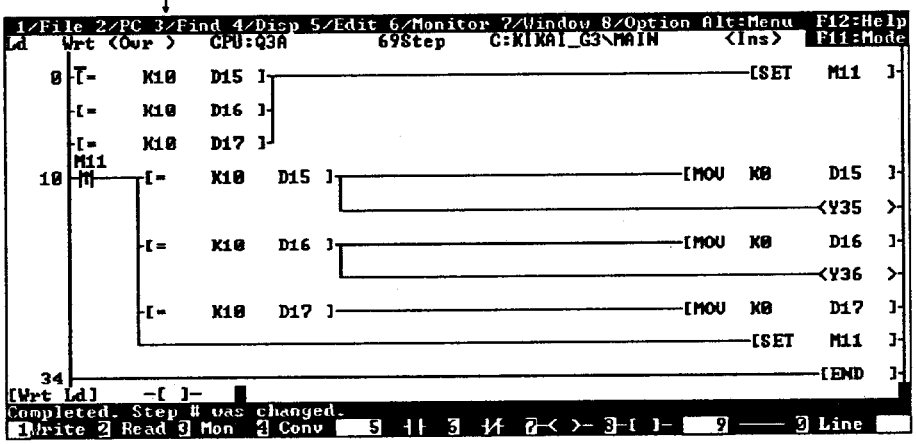
2)

2)



"Ladder Write screen (during programming)"

Continue this process to create the following ladder program
[F4] Convert

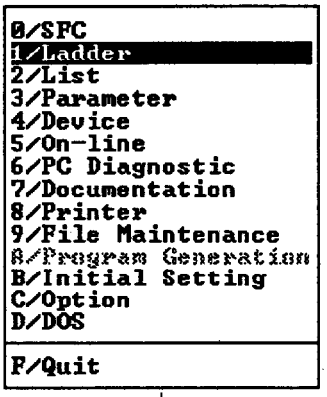


"Ladder Write screen (after programming)"

(5) Set the file name to "PACKING" and write the file.
The operation is the same as described in Section 4.6.1 (2).

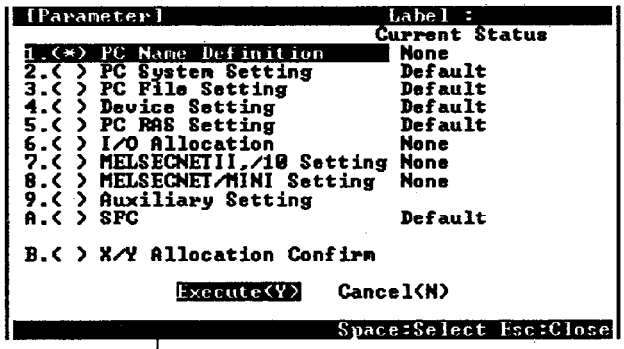
(6) Write program to the CPU.
See Section 3.5.2 for details about the write operation.

(7) Set parameters



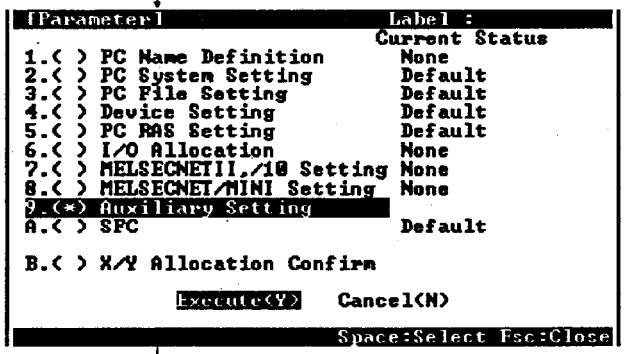
"Mode Select screen"

[3] Select "Parameter"



"Parameter screen (before setting)"

[9] Select "Auxiliary Setting"

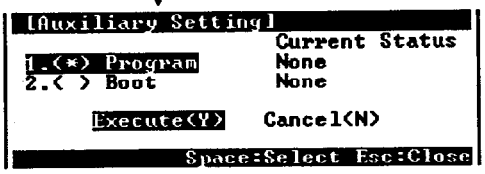


"Parameter screen (after setting)"

[Y]

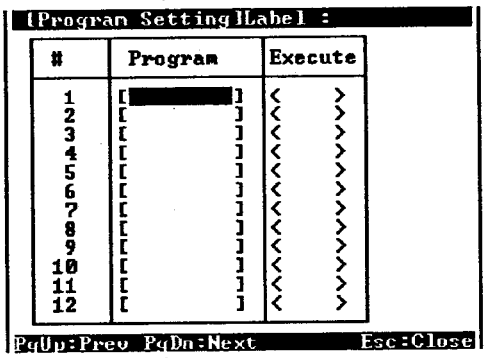
1)

1)



"Auxiliary Setting screen"

[1] [Y] Select "Program"

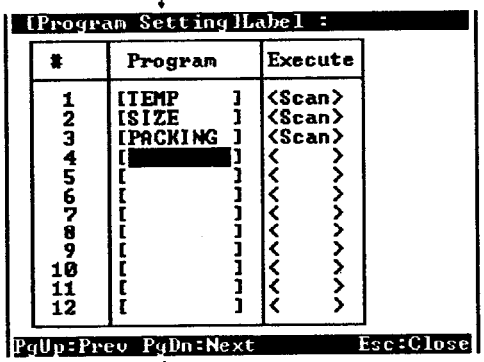


"Program Setting screen (before setting)"

[T] [E] [M] [P] [Tab] [Enter] Set program name "TEMP" as a scan program

[S] [I] [Z] [E] [Tab] [Enter] Set program name "SIZE" as a scan program

[P] [A] [C] [K] [I] [N] [G] [Tab] [Enter] Set program name "PACKING" as a scan program

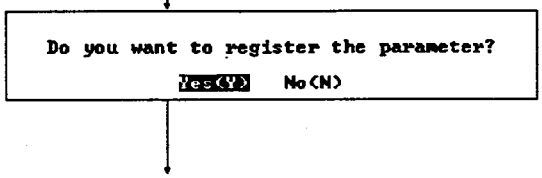


"Program Setting screen (after setting)"

[Esc]

2)

2)



[Y] Register parameters

[Esc] Close parameter screen

(8) Write to CPU.
See Section 4.2.2.

5. MAKING THE MOST OF YOUR QnACPU (2)

5.1 Programming Before I/O Module Configuration is Determined

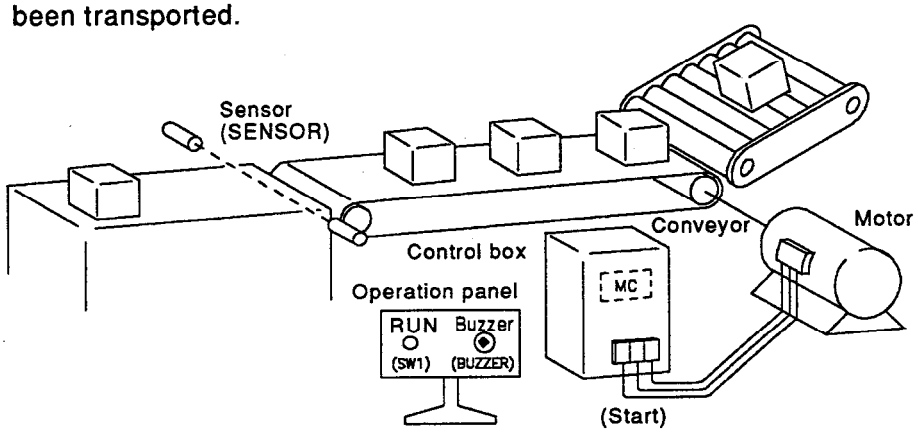
By adding labels to the sequence program, a QnACPU program can be created before the I/O module configuration is determined.

5.1.1 Programming using labels

The example shown below is used to explain programming with labels.

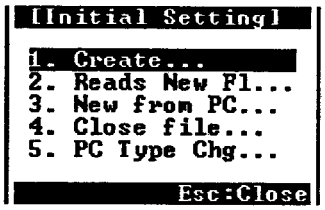
General Description of the System

When the conveyor RUN switch is pressed, the warning buzzer sounds for 3 seconds and the conveyor automatically starts. The conveyor stops when the system detects that 6 products have been transported.



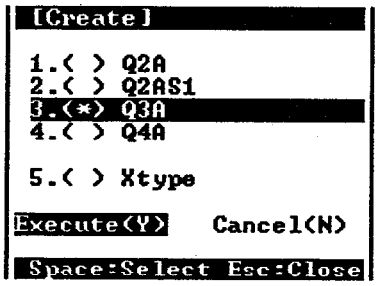
(1) Create the sequence program using the labels below.

Label	Device	Comment
SW1	X0	Operation command for conveyor
COUNT	C0	6-product product count
OPERATION	M0	Operation in progress
START	Y71	Operation start command
BUZZER	Y70	Operation start buzzer
TIMER	T0	3 second timer for buzzer
SENSOR	X1	Product detection



"Initial Setting screen"

[1] Select "Create..."

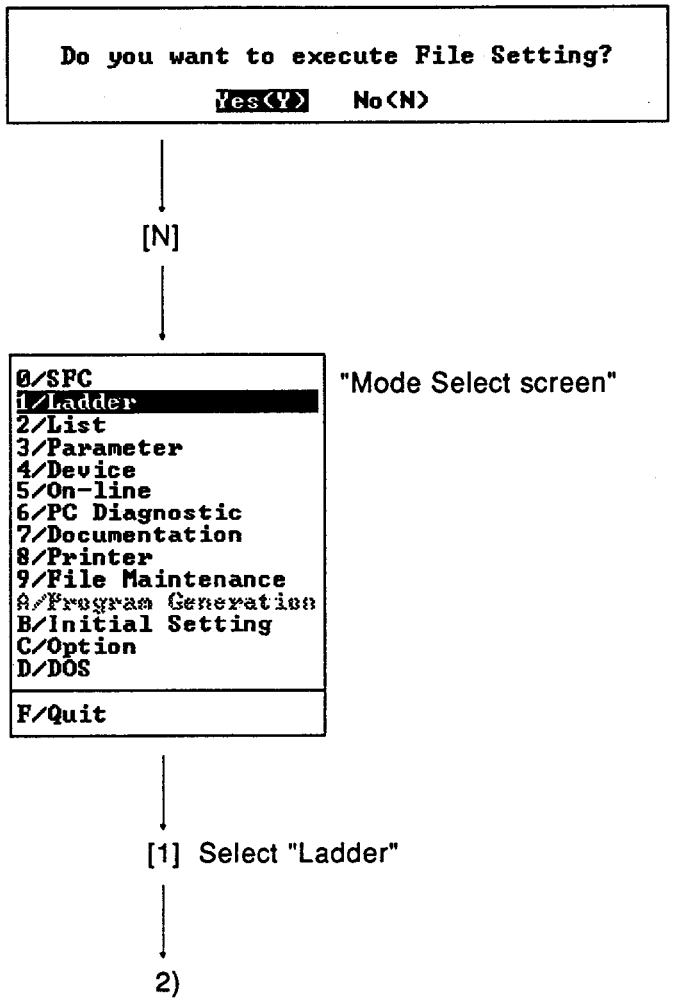


"Create screen"

[5] [Y] Select PC type "Xtype"

1)

1)


Do you want to execute File Setting?
Yes(Y) No(N)
[N]
0/SFC
1/Ladder
2/List
3/Parameter
4/Device
5/On-line
6/PC Diagnostic
7/Documentation
8/Printer
9/File Maintenance
A/Program Generation
B/Initial Setting
C/Option
D/DOS
F/Quit
"Mode Select screen"
[1] Select "Ladder"

2)

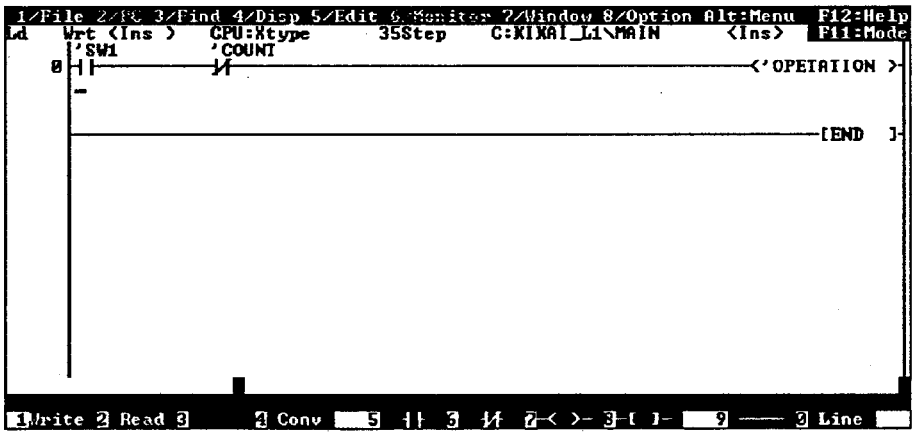
2)

"Ladder Write screen (before programming)"

POINT

To define a label, insert an apostrophe (') at the beginning.

[F5]+[Shift] ['] [S] [W] [1] [Enter]
(⊣⊢)
[F6]+[Shift] ['] [C] [O] [U] [N] [T] [Enter]
(⊣/⊢)
[F7]+[Shift] ['] [O] [P] [E] [R] [A] [T] [I] [O] [N] [Enter]
(⊣↑⊢)



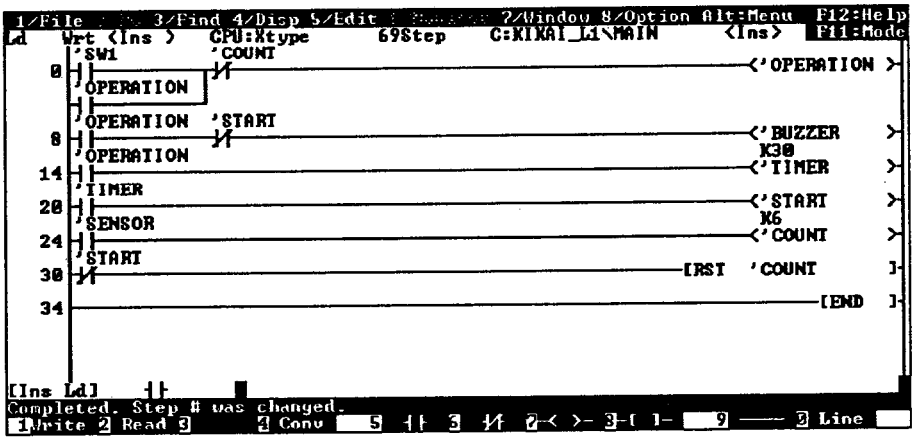
"Ladder Write screen (during programming)"

3)

3)

Continue this process to create the following ladder program.

[F4] Convert



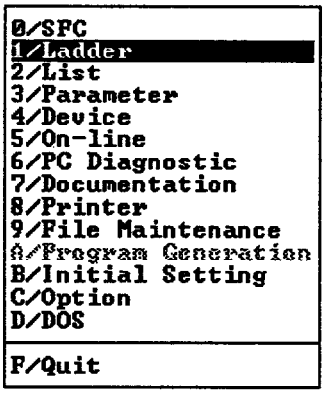
"Ladder Write screen (after programming)"

[F11] Select "Mode"

4)

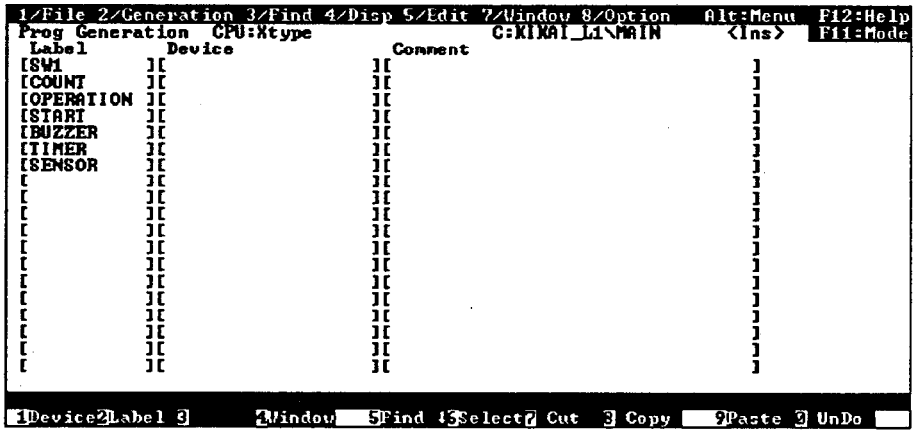
(2) Assign devices and comments to the labels

4)



"Mode Select screen"

[A] Select "Program Generation"

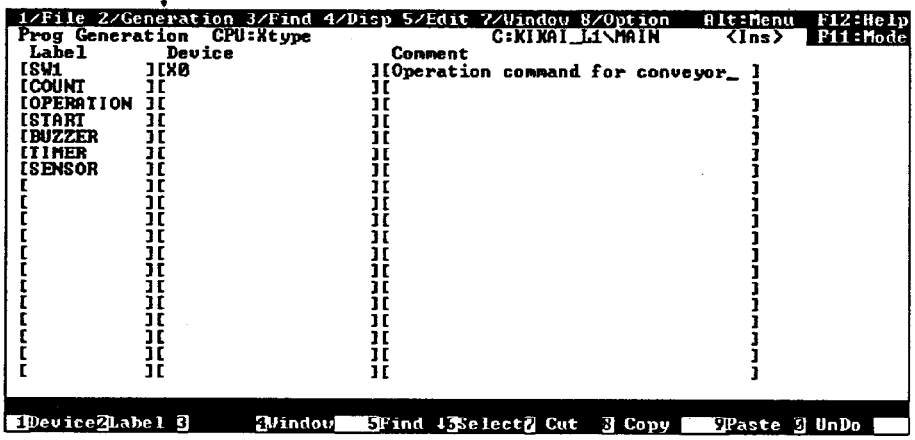


"Program Generation screen (before setting)"

5)

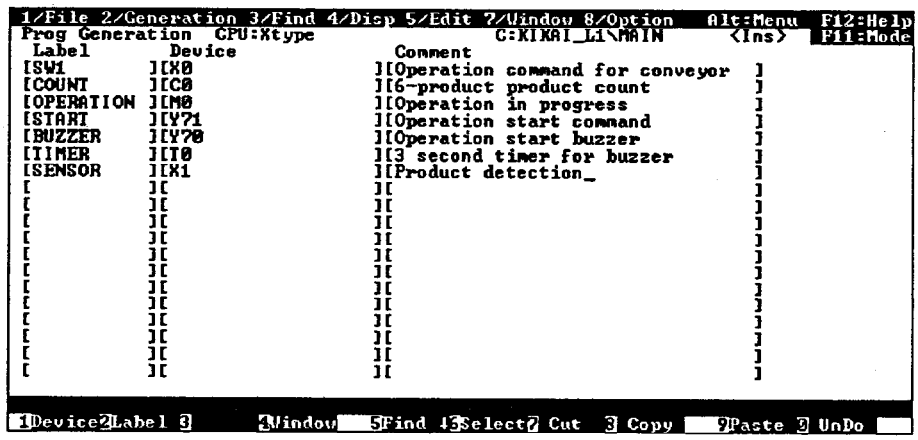
5)

[Tab] [X] [0] [Tab] [O] [P] [E] [R] [A] [T] [I] [O] [N] [SP]
[C] [O] [M] [M] [A] [N] [D] [SP] [F] [O] [R] [SP]
[C] [O] [N] [V] [E] [Y] [O] [R]



"Program Generation screen (during setting)"

Continue this process to assign devices and comments to the labels.



"Program Generation screen (after setting)"

6)

POINT

The comments set in the Program Generation screen become the comments in the program after program expansion.

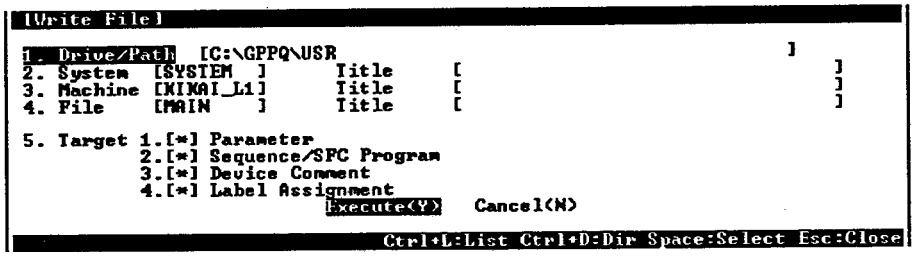
(3) Write the program to file

6)

↓

[Alt] [1] [2] Select "Write File"

↓



"Write File screen (before setting)"

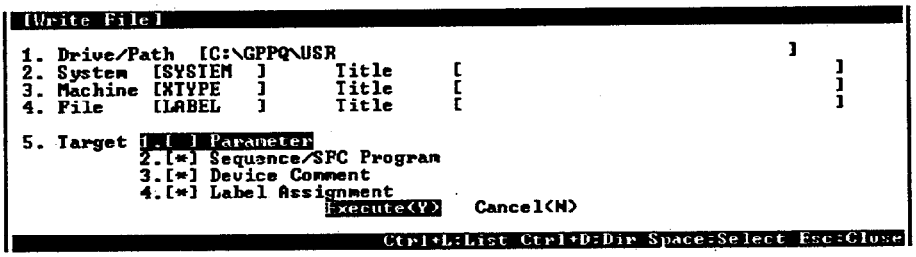
↓

[3] [BS] [BS] [BS] [BS] [BS] [BS] [BS] [BS] [X] [T] [Y] [P] [E] [Enter]
Change machine name to "XTYPE"

[4] [BS] [BS] [BS] [BS] [L] [A] [B] [E] [L] [Enter]
Change file name to "LABEL"

[5] [1] Change write object to sequence program and device comment.

↓



"Write File screen (after setting)"

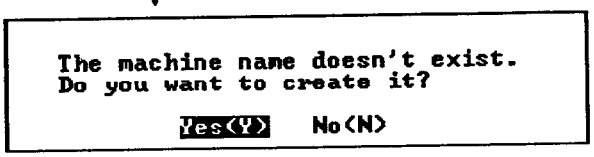
↓

[Y] Write file Execte

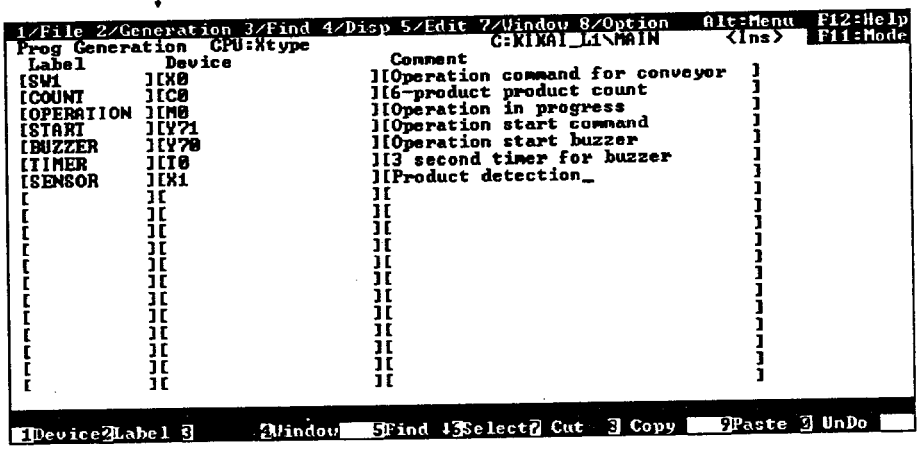
↓

7)

7)



[Y] Create new machine name



"Program Generation screen"

8)

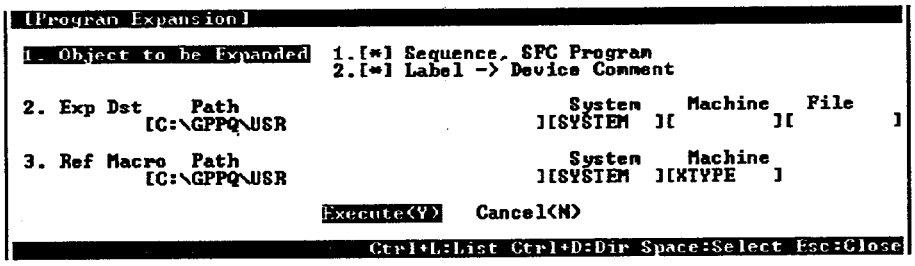
(4) Expand general-purpose program to an actual program

8)

[Alt] [2] Select "Program Generation"

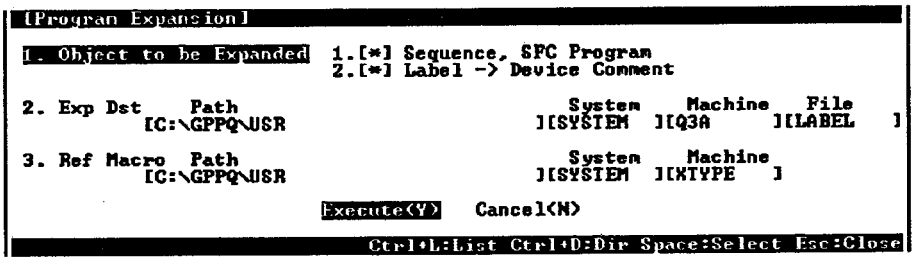
"Program Generation screen"

[1] Select "Program Expansion"



"Program Expansion screen (before setting)"

[2] [Tab] [Tab]
[Q] [3] [A] [Tab]
[L] [A] [B] [E] [L] [Enter] Change machine name and file name

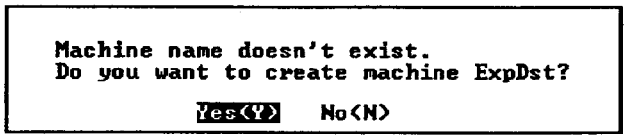


"Program Expansion screen (after setting)"

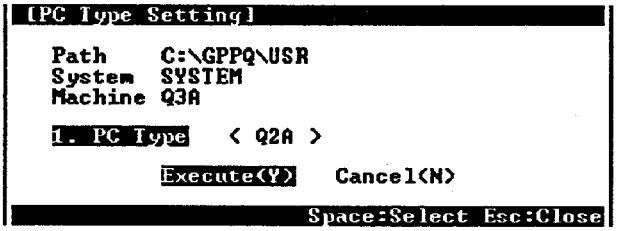
[Y]

9)

9)

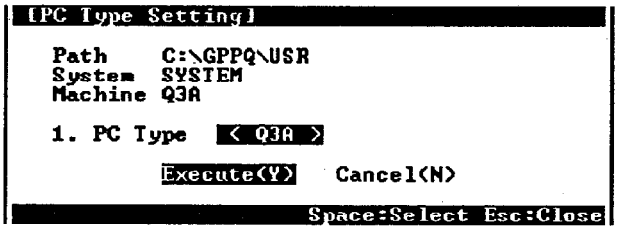


[Y]



"PC Type Setting screen (before setting)"

[SP] [SP] [SP] Select PC type "Q3A"

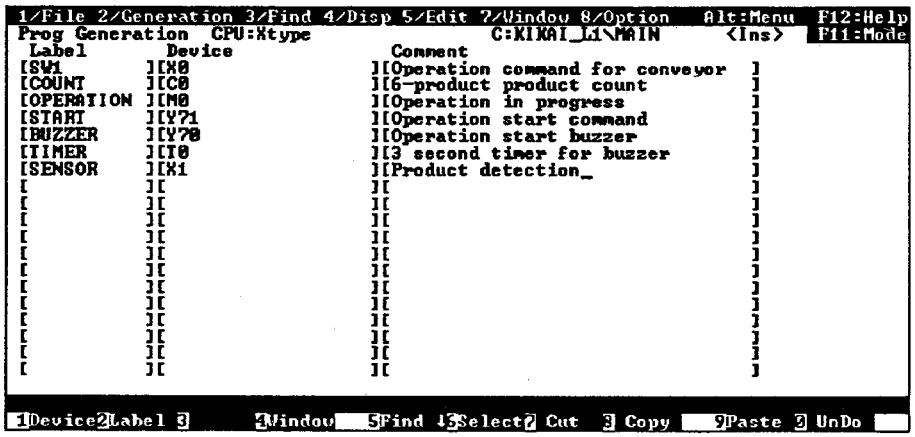


"PC Type Setting screen (after setting)"

[Y]

10)

10)



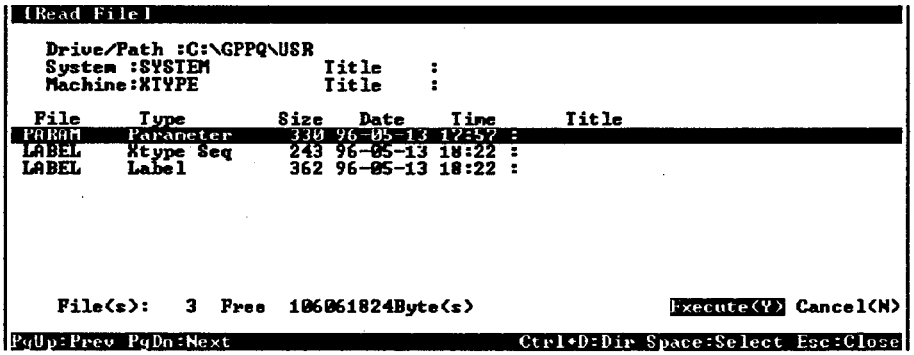
"Program Generation screen"

11)

(5) Read the expanded program

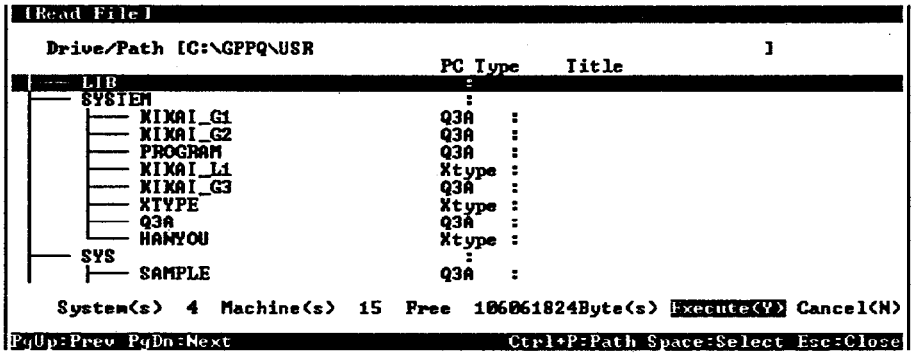
11)

[Alt] [1] [1] Select "Read File"



"Read File screen"

[Ctrl]+[D] Directory display

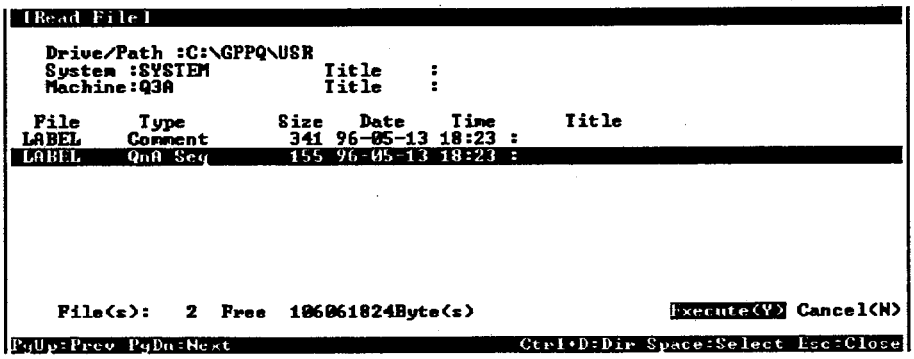


"Read File screen (directory display)"

[↓] [↓] [↓] [↓] [↓] [↓] [↓] [↓] [Y] Select machine name "Q3A"

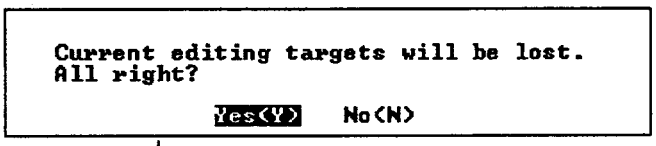
12)

12)

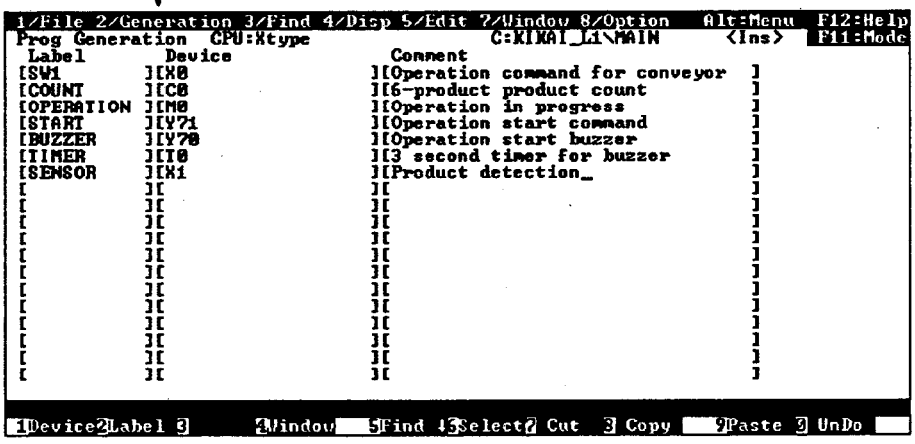


"Read File screen"

[SP] [Y] Select file name "LABEL"



[Y]

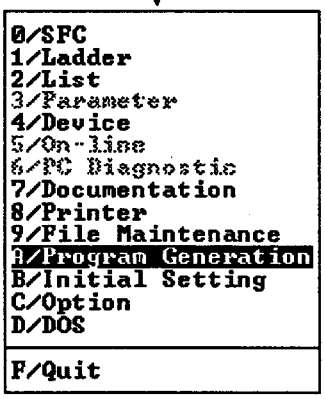


"Program Generation screen"

[F11] Display "Mode Select screen"

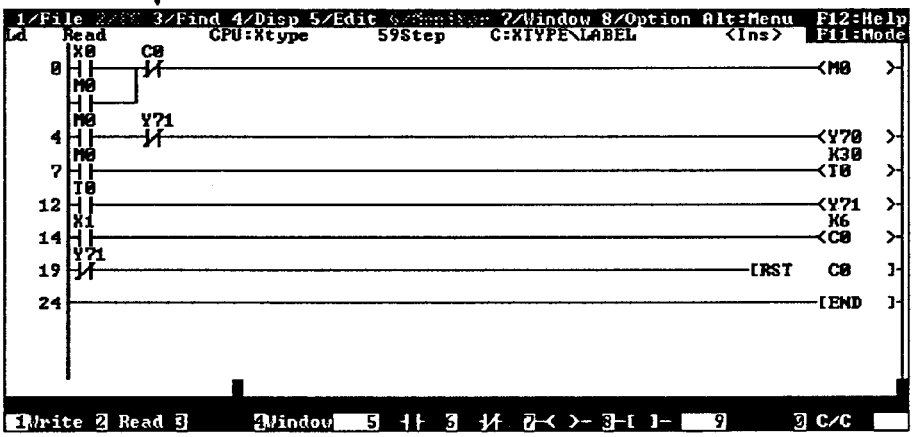
13)

13)



"Mode Select screen"

[1] Select "Ladder"



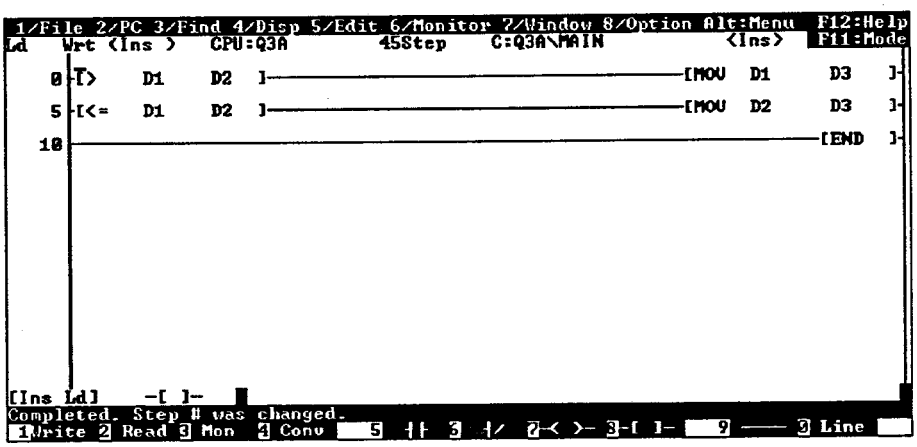
"Ladder Read screen"

5.2 Utilizing Program Resources

5.2.1 Program standardization with user-created macros

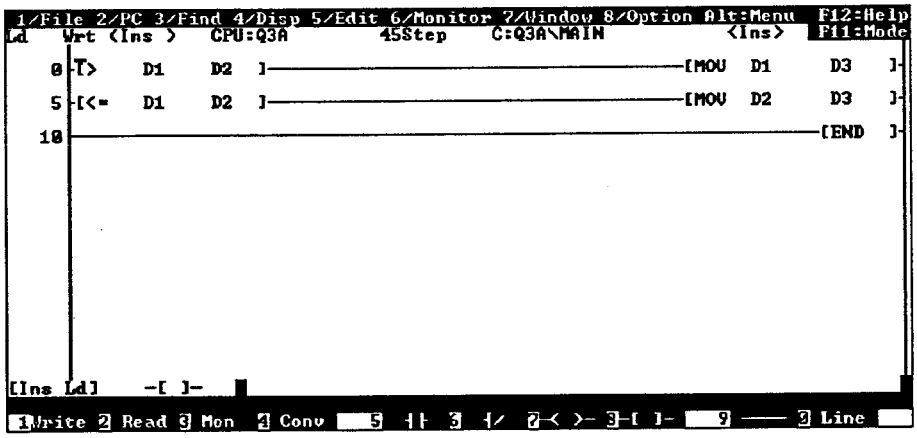
Macro instructions are instructions which read and utilize ladders pre-saved in a peripheral device.
Using macro instructions improves program productivity.
This section describes how to use the macro instructions.

(1) Create the following sequence program and write it to file



"Ladder Write screen (before programming)"

(2) Read the file and register the macro



"Ladder Write screen"

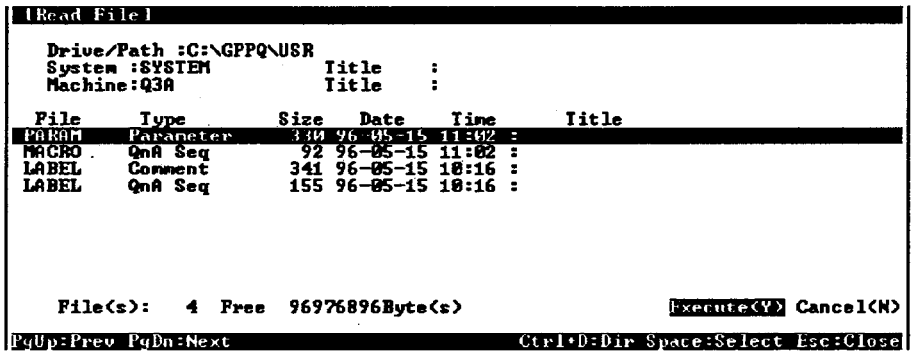
↓

[Alt] [1] [1] Select "Read File"

↓

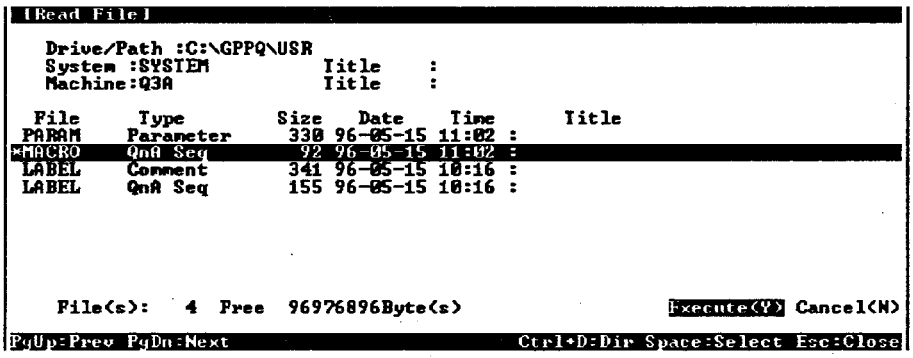
1)

1)



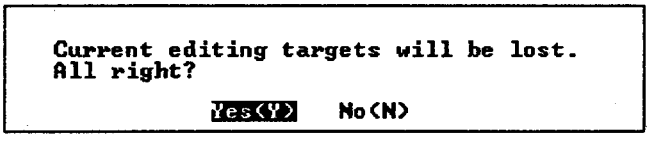
"Read File screen (before setting)"

[↓] [SP] Select file name "MACRO"



"Read File screen (after setting)"

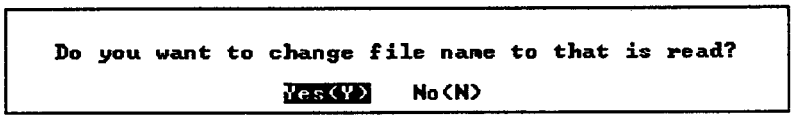
[Y]



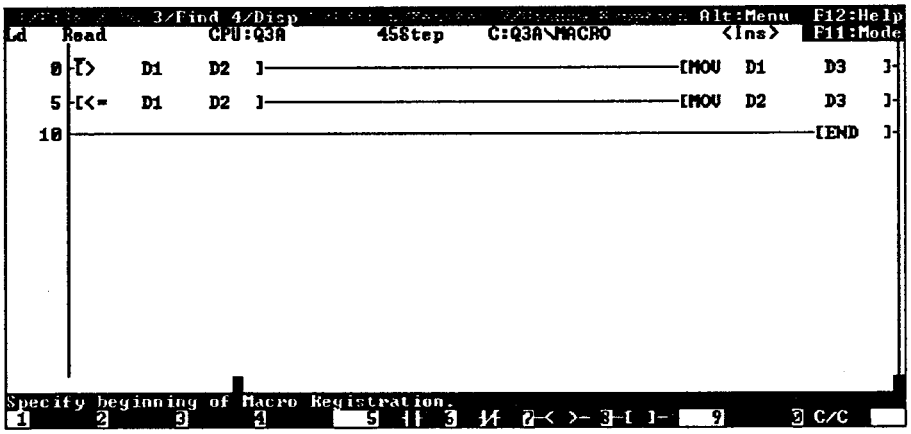
[Y]

2)

2)

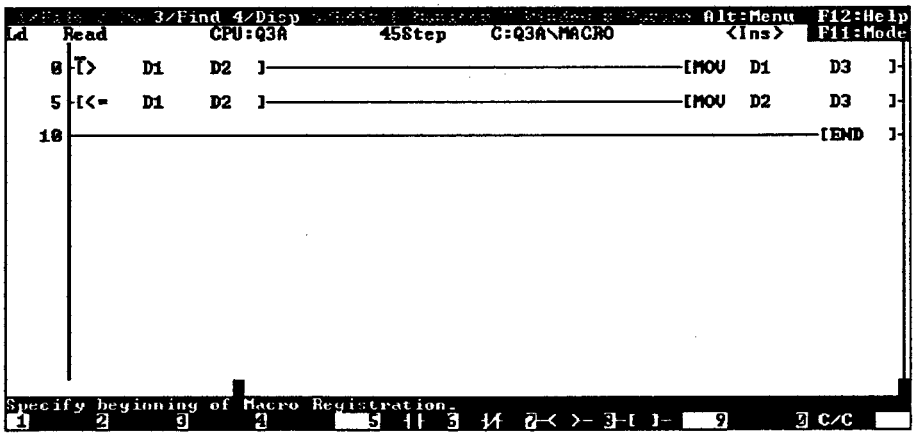


[Y]



"Ladder Read screen"

[Alt] [1] [8] Select "Macro Registration"

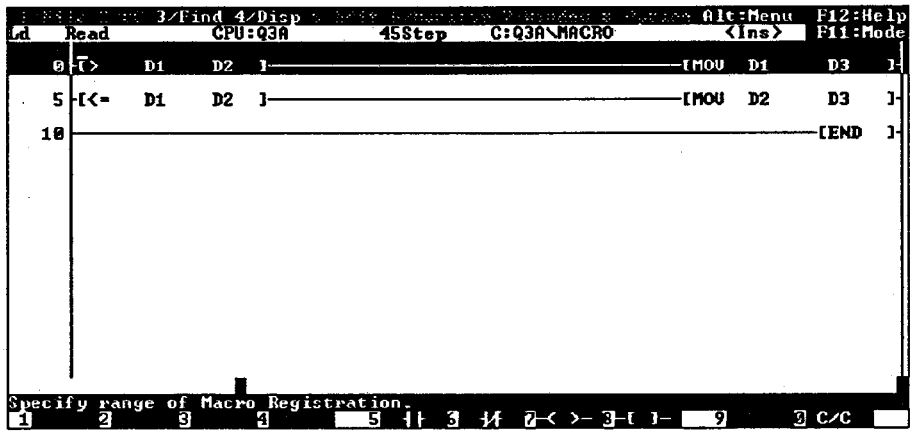


"Ladder Read screen (designate macro registration destination)"

[Enter] Designate step 0 as beginning of Macro Registration destination

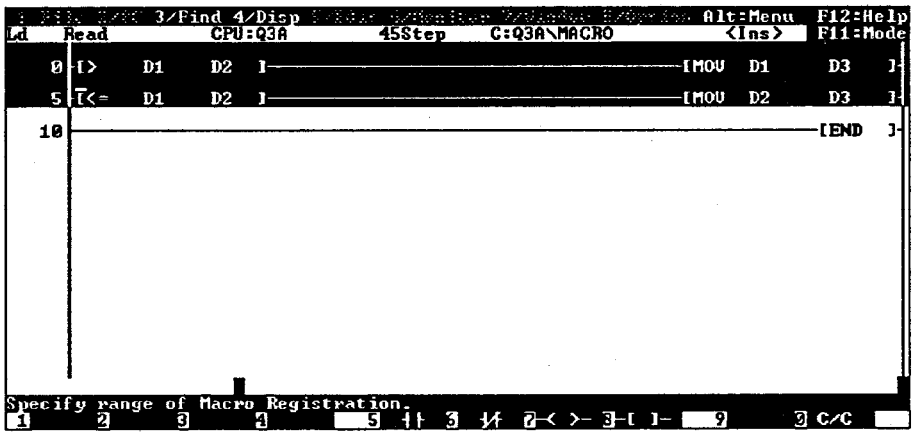
3)

3)



"Ladder Read screen (designate macro registation range)"

[↓] Designate the macro registration range up to step 9

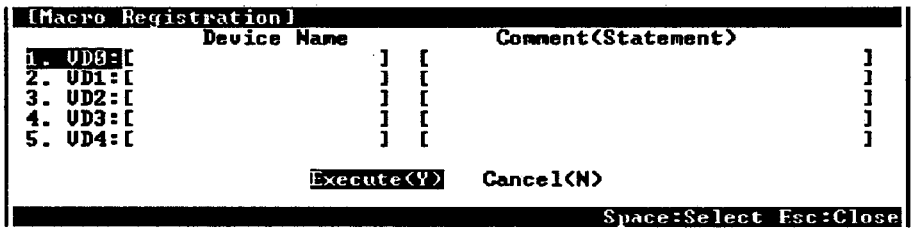


"Ladder Read screen (designate macro registration range)"

[Enter] Enter macro registration range

4)

4)



"Macro Registration screen (before setting)"

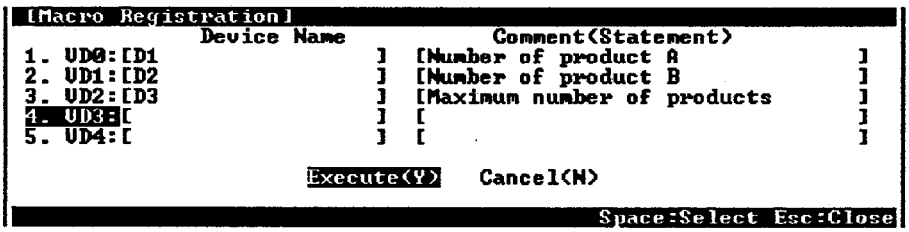
[1] [D] [1] [Enter] [Enter] Set D1 as VD0

[2] [D] [2] [Enter] [Enter] Set D2 as VD1

[3] [D] [3] [Enter] [Enter] Set D3 as VD2

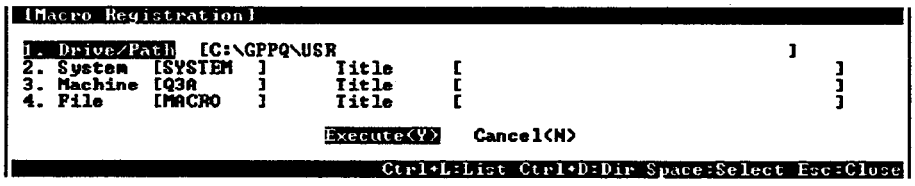
POINT

A maximum of five variable devices (VD) can be set.



"Macro Registration screen (after setting)"

[Y]

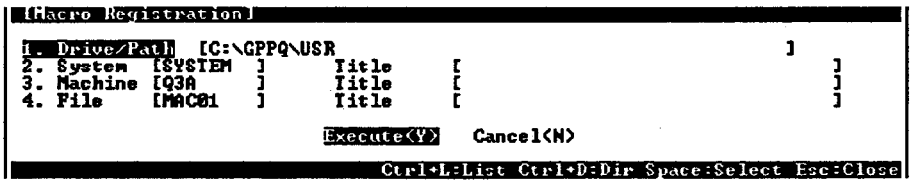


"Write File screen (before setting)"

[4] [BS] [BS] [0] [1] [Enter]
Select file name "MAC01"

5)

5)



" Write File screen (after setting)"

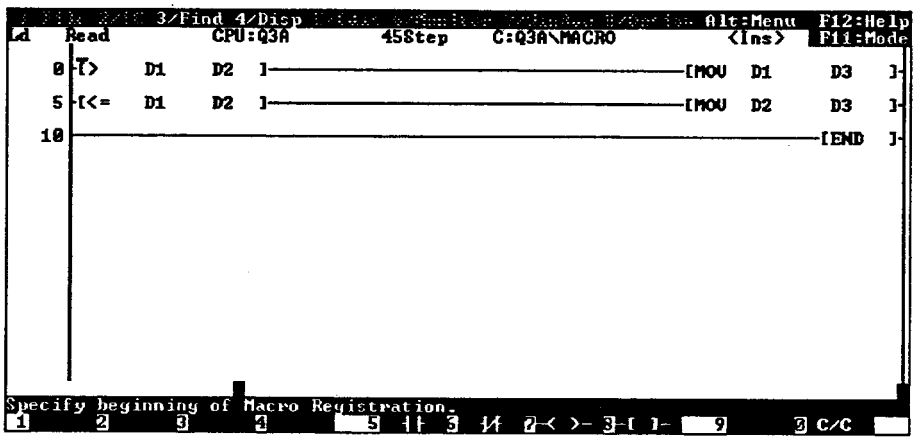
[Y]

[F1] Select "Ladder Write"

6)

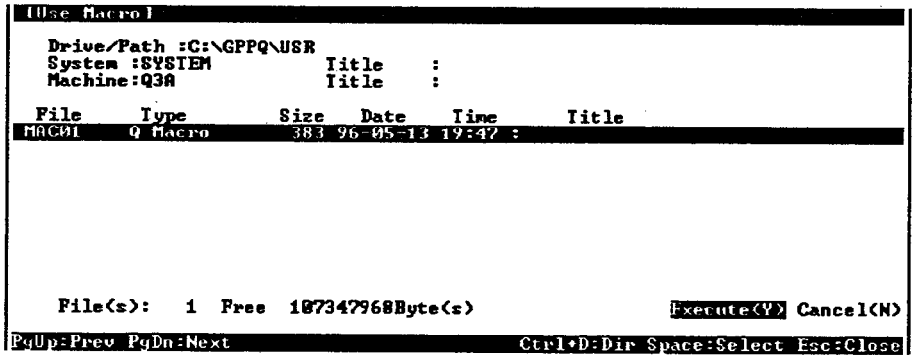
(3) Read a registered macro file and utilize the macro.

6)



"Ladder Write screen"

[Alt] [1] [9] Select "Use Macro"

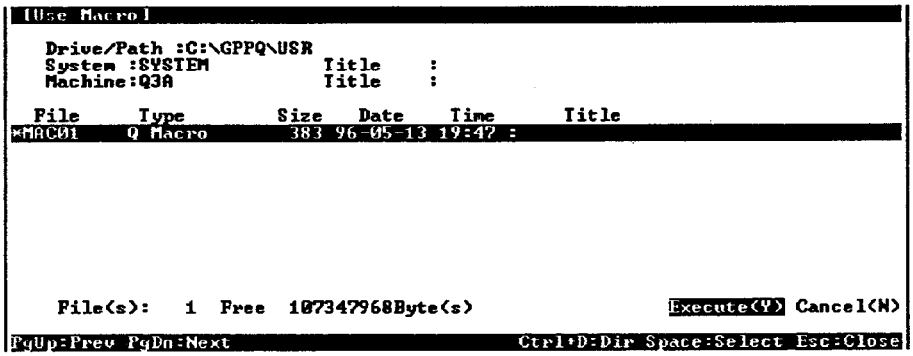


"Read File screen (before setting)"

[SP] Select file "MAC01"

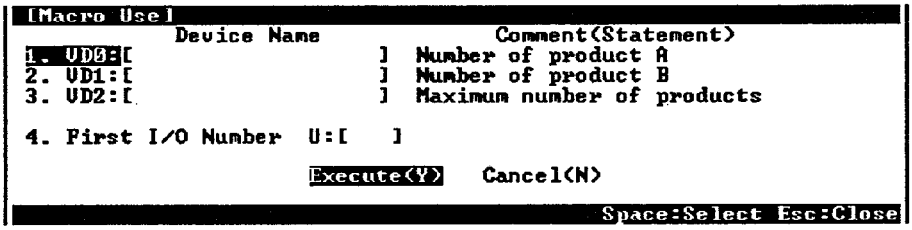
7)

7)



"Read File screen (after setting)"

[Y]



"Macro Use screen (before setting)"

[1] [D] [1] [0] [Enter] Set D10 as VD0

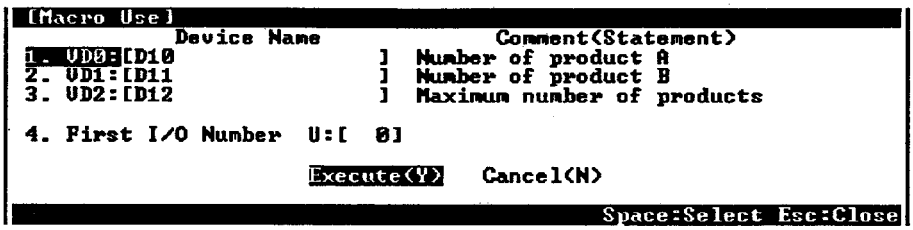
[2] [D] [1] [1] [Enter] Set D11 as VD1

[3] [D] [1] [2] [Enter] Set D12 as VD2

[4] [0] [Enter] Set 0 as the first I/O number

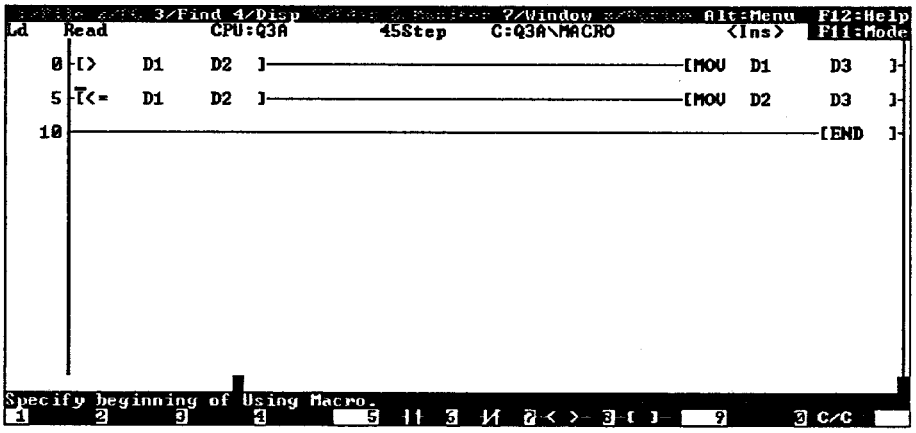
8)

8)



"Macro Use screen (after setting)"

[Y]

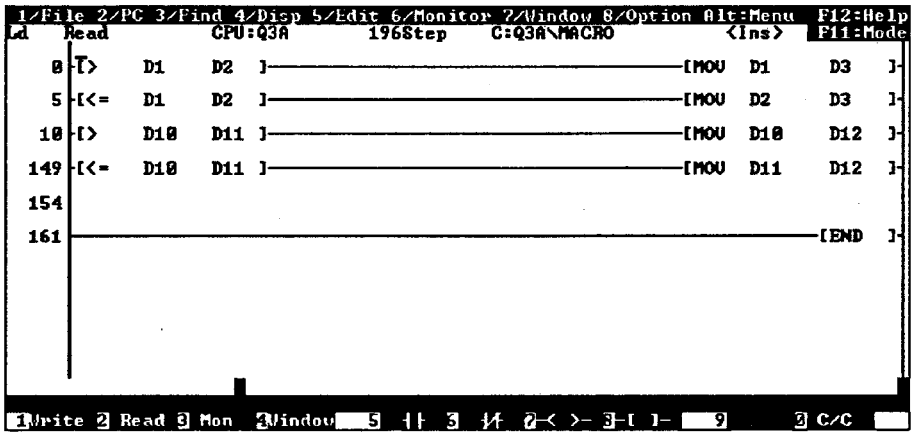


"Ladder Write screen (before macro use)"

[↓] [Enter] Set step 10 as the first step to macro use

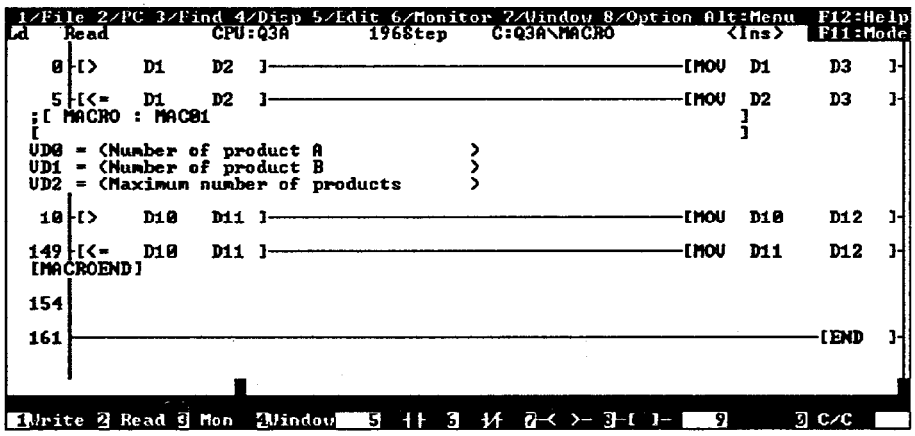
9)

9)



"Ladder Write screen (after macro use)"

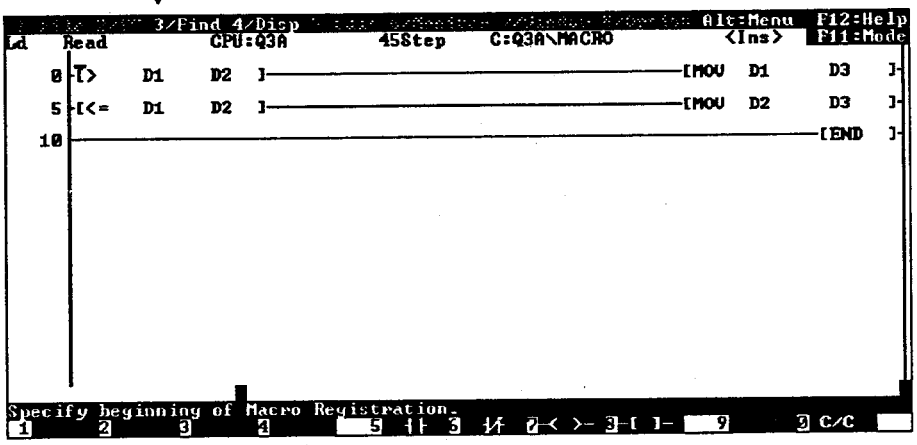
[Alt] [4] [3] Turn on statement display (OFF→ON)



"Ladder Write screen (with statement display ON)"

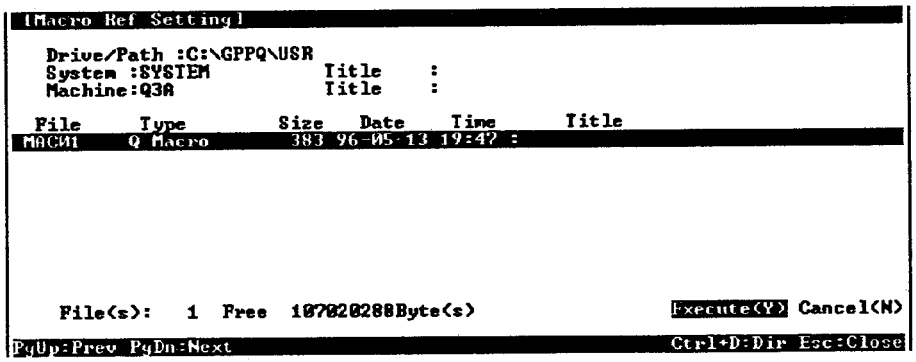
(4) Macro use with macro instruction "M. file name ␣ VD1 ␣ VD2 ␣ D3".

Read the file with file name "MACRO" from machine name "Q3A".



"Ladder Read screen"

[F1] Change Read to Write
[Alt] [1] [A] Select "Macro Ref Setting"

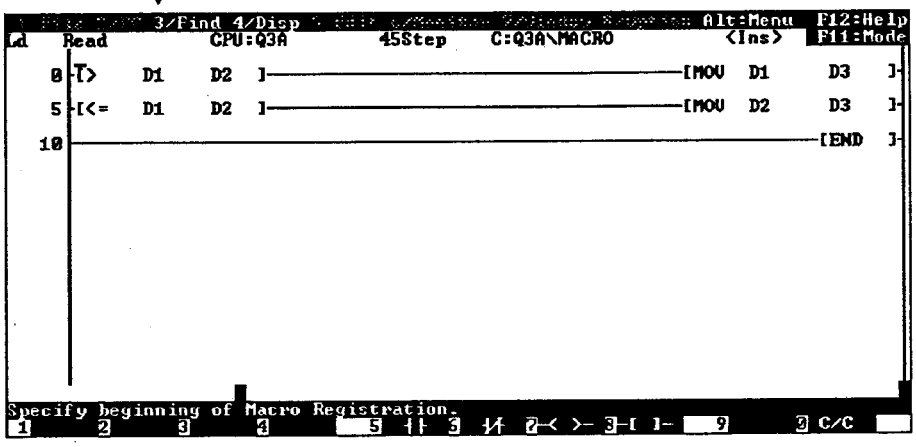


"Macro Ref Setting screen"

[Y]

10)

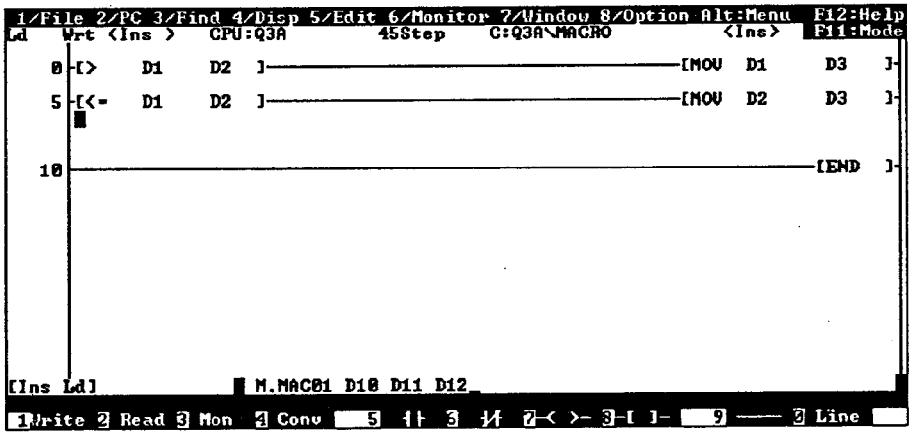
10)



"Ladder Write screen (before macro use)"

[↓] [↓] [Shift]+[Ins] [M] [.] [M] [A] [C] [0] [1] [SP] [D] [1] [0] [SP] [D] [1] [1] [SP] [D] [1] [2] [Enter] Set step 10 as the first step to macro use

"-[M.MACO1 D10 D11 D12]- " Write

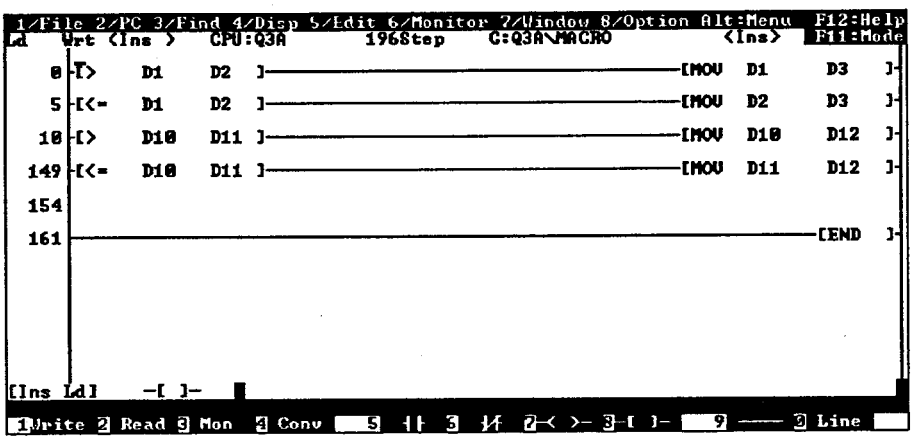


"Ladder Write screen (after macro use)"

[F4] Convert

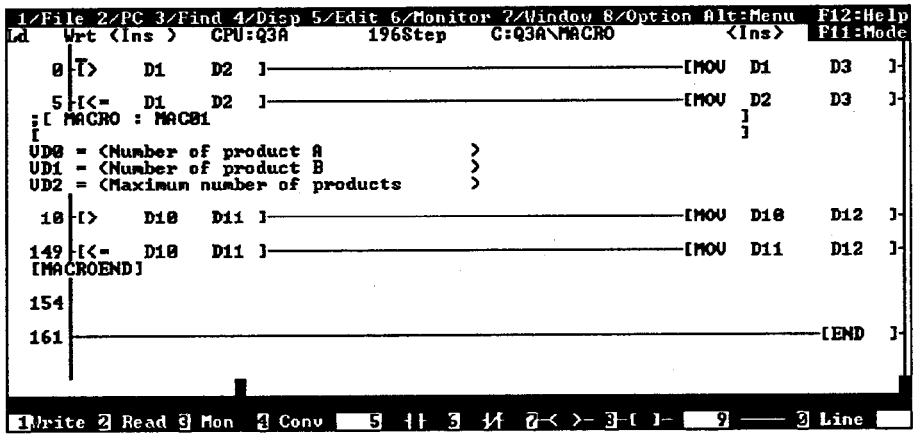
11)

11)



"Ladder Write screen (after conversion)"

[Alt] [4] [3] Turn on statement display (OFF→ON)



"Ladder Write screen (with statement display ON)"

QnACPU GUIDEBOOK

MODEL	QNACPU-G-E
MODEL CODE	13JF10
IB(NA)66606-A(9607)MEE	

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE : MITSUBISHI DENKI BLDG MARUNOUCHI TOKYO 100-0005 TELEX : J24532 CABLE MELCO TOKYO
NAGOYA WORKS : 1-14 , YADA-MINAMI 5 , HIGASHI-KU, NAGOYA , JAPAN