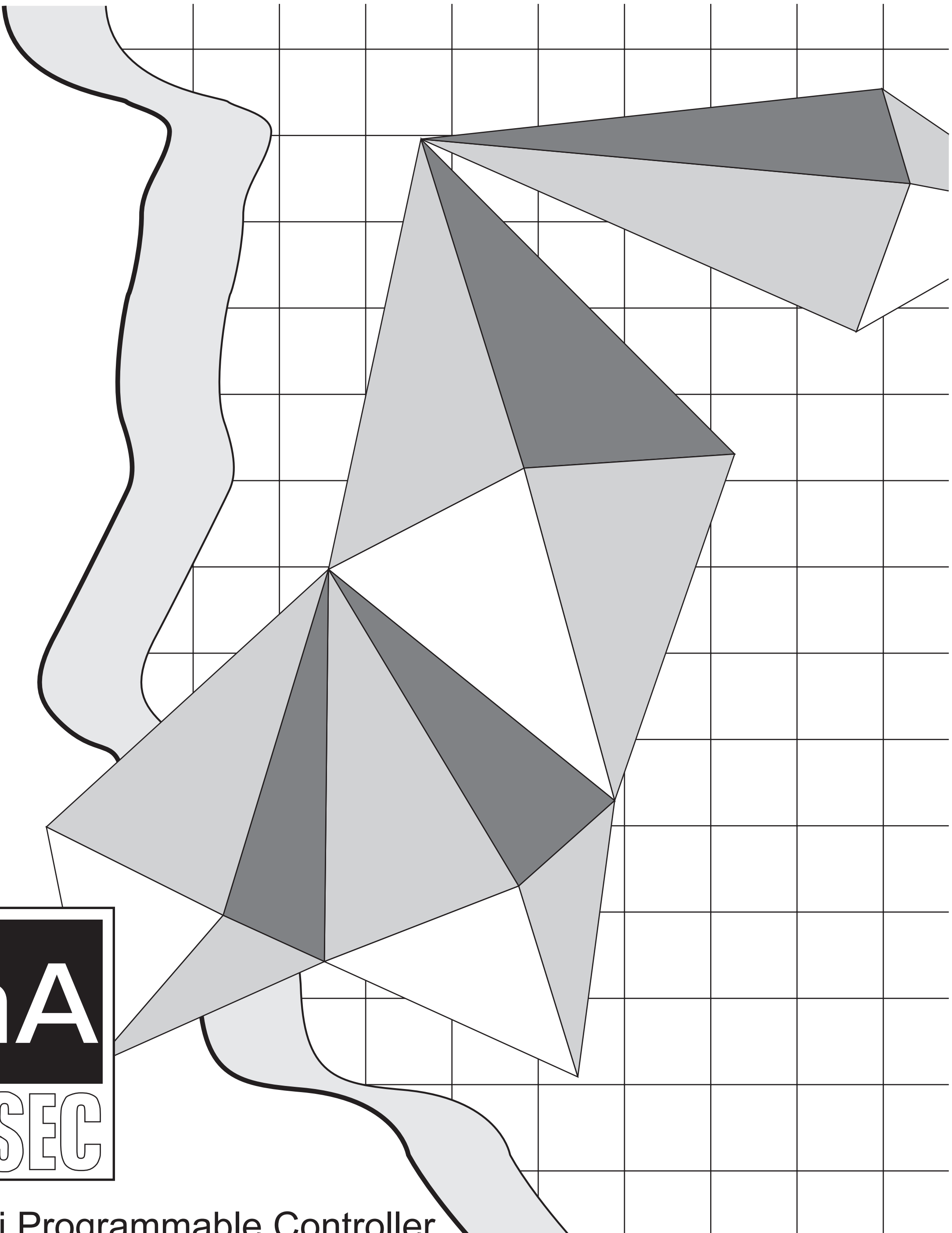
MITSUBISHI

QnA SERIES

Model Q2AS (H) CPU (S1)

User's Manual



QnA
MELSEC

Mitsubishi Programmable Controller

●SAFETY PRECAUTIONS●

(Read these precautions before using this product.)

Before using this product, please read this manual and the relevant manuals carefully and pay full attention to safety to handle the product correctly.

In this manual, the safety precautions are classified into two levels: "⚠ WARNING" and "⚠ CAUTION".

⚠ WARNING	Indicates that incorrect handling may cause hazardous conditions, resulting in death or severe injury.
⚠ CAUTION	Indicates that incorrect handling may cause hazardous conditions, resulting in minor or moderate injury or property damage.

Under some circumstances, failure to observe the precautions given under "⚠ CAUTION" may lead to serious consequences.

Make sure that the end users read this manual and then keep the manual in a safe place for future reference.

[DESIGN PRECAUTIONS]

⚠WARNING

● Create a safety circuit outside the programmable controller to ensure the whole system will operate safely even if an external power failure or a programmable controller failure occurs. Otherwise, incorrect output or malfunction may cause an accident.

(1) When creating an emergency stop circuit, a protection circuit or an interlock circuit for incompatible actions such as forward/reverse rotation or for damage prevention such as the upper/lower limit setting in positioning, create it outside the programmable controller.
Install the emergency stop switch outsid the controlpanel so that workers can operate it easily.

(2) When the programmable controller detects the following error conditions, it stops the operation and turn off all the outputs.
- The overcurrent or overvoltage protector of the power supply module is activated.
- The programmable controller CPU detects an error such as a watchdog timer error by the self-diagnostics function.

In the case of an error undetectable by the programmable controller CPU, such as an I/O control part error, all the outputs may turn on. In order to make all machines operate safely in such a case, set up a fail-safe circuit or a specific mechanism outside the programmable controller. For fail safe circuit example, refer to "OADING AND INSTALLATION" of this manual.

(3) Depending on the failure of the output module's relay or transistor, the output status may remain ON or OFF incorrectly. For output signals that may lead to a serious accident, create an external monitoring circuit.

[DESIGN PRECAUTIONS]

⚠WARNING

- If load current more than the rating or overcurrent due to a short circuit in the load has flowed in the output module for a long time, it may cause a fire and smoke. Provide an external safety device such as a fuse.
- Design a circuit so that the external power will be supplied after power-up of the programmable controller.
  Activating the external power supply prior to the programmable controller may result in an accident due to incorrect output or malfunction.
- For the operation status of each station at a communication error in data link, refer to the respective data link manual.
  Otherwise, incorrect output or malfunction may cause an accident.
- When controlling a running programmable controller (data modification) by connecting a peripheral device to the CPU module or a PC to a special function module, create an interlock circuit on sequence programs so that the whole system functions safely all the time.
  Also, before performing any other controls (e.g. program modification, operating status change (status control)), read the manual carefully and ensure the safety.
  In these controls, especially the one from an external device to a programmable controller in a remote location, some programmable controller side problem may not be resolved immediately due to failure of data communications.
  To prevent this, create an interlock circuit on sequence programs and establish corrective procedures for communication failure between the external device and the programmable controller CPU.
- When setting up the system, do not allow any empty slot on the base unit.
  If any slot is left empty, be sure to use a blank cover (A1SG60) or a dummy module (A1SG62) for it.
  When using the extension base unit, A1S52B(S1), A1S55B(S1) or A1S58B(S1), attach the included dustproof cover to the module in slot 0.
  Otherwise, internal parts of the module may be flied in the short circuit test or when an overcurrent or overvoltage is accidentally applied to the external I/O section.

⚠CAUTION

- Do not install the control lines or communication cables together with the main circuit or power lines, or bring them close to each other.
  Keep a distance of 100mm (3.94inch) or more between them.
  Failure to do so may cause malfunctions due to noise.
- If having read register R outside the allowable range with the MOV instruction, the file register data will be FFFFH. Using this as it is may cause malfunctions.Pay attention not to use any out-of-range file register when designing sequence programs.For instruction details, refer to the programming manual.
- When an output module is used to control the lamp load, heater, solenoid valve, etc., a large current (ten times larger than the normal one) may flow at the time that the output status changes from OFF to ON. Take some preventive measures such as replacing the output module with the one of a suitable current rating.
- Time from when the CPU module is powered on or is reset to when it enters in RUN status depends on the system configuration, parameter settings, and program size.
  Design the program so that the entire system will always operate safely, regardless of the time.

[INSTALLATION PRECAUTIONS]

⚠CAUTION

- Use the programmable controller under the environment specified in the user's manual.
  Otherwise, it may cause electric shocks, fires, malfunctions, product deterioration or damage.

- Install the module after inserting the pegs on the bottom of the module securely into the base unit peg holes.
  Not doing so could cause a malfunction, failure or fall.
  Tightening the screw excessively may damage the screw and/or the module, resulting in a drop of the module, a short circuit or malfunctions.

- Connect the extension cable to the connector of the base unit or module.
  Check for incomplete connection after installing it.
  Poor electrical contact may cause incorrect inputs and/or outputs.

- Insert the memory card and fully press it to the memory card connector.
  Check for incomplete connection after installing it.
  Poor electrical contact may cause malfunctions.

- Be sure to shut off all the phases of the external power supply used by the system before mounting or removing the module.
  Failure to do so may damage the module.

- Do not directly touch the conductive part or electronic components of the module.
  Doing so may cause malfunctions or a failure of the module.

[WIRING PRECAUTIONS]

⚠WARNING

- Be sure to shut off all phases of the external power supply used by the system before wiring.
  Failure to do so may result in an electric shock or damage of the product.
- Before energizing and operating the system after wiring, be sure to attach the terminal cover supplied with the product.
  Failure to do so may cause an electric shock.

⚠CAUTION

- Always ground the FG and LG terminals to the protective ground conductor.
  Failure to do so may cause an electric shock or malfunctions.
- Wire the module correctly after confirming the rated voltage and terminal layout.
  Connecting a power supply of a different voltage rating or incorrect wiring may cause a fire or failure.
- Do not connect multiple power supply modules to one module in parallel.
  The power supply modules may be heated, resulting in a fire or failure.
- Press, crimp or properly solder the connector for external connection with the specified tool.
  Incomplete connection may cause a short circuit, fire or malfunctions.
- Tighten terminal screws within the specified torque range.
  If the screw is too loose, it may cause a short circuit, fire or malfunctions.
  If too tight, it may damage the screw and/or the module, resulting in a short circuit or malfunctions.
- Carefully prevent foreign matter such as dust or wire chips from entering the module.
  Failure to do so may cause a fire, failure or malfunctions.
- Install our programmable controller in a control panel for use.
  Wire the main power supply to the power supply module installed in a control panel through a distribution terminal block.
  Furthermore, the wiring and replacement of a power supply module have to be performed by a maintenance worker who acquainted with shock protection.
  (For the wiring methods, refer to Section 19.7.)

[START AND MAINTENANCE PRECAUTIONS

⚠WARNING

- Do not touch any terminal during power distribution.
  Doing so may cause an electric shock.

- Correctly connect the battery connector.
  Do not charge, disassemble, heat, short-circuit, solder, or throw the battery into the fire.
  Incorrect battery handling may cause personal injuries or a fire due to exothermic heat, burst and/or ignition.

- Be sure to shut off all phases of the external power supply used by the system before cleaning or retightening the terminal screws or module mounting screws.
  Failure to do so may result in an electric shock.
  If they are too loose, it may cause a short circuit or malfunctions.
  Tightening the screw excessively may damage the screw and/or the module, resulting in a drop of the module, a short circuit or malfunctions.

⚠CAUTION

- When performing online operations (especially, program modification, forced output or operating status change) by connecting a peripheral device to the running CPU module, read the manual carefully and ensure the safety.
  Incorrect operation will cause mechanical damage or accidents.
- Do not disassemble or modify each of modules.
  Doing so may cause failure, malfunctions, personal injuries and/or a fire.
- When using a wireless communication device such as a mobile phone, keep a distance of 25cm (9.84inch) or more from the programmable controller in all directions.
  Failure to do so may cause malfunctions.
- Be sure to shut off all the phases of the external power supply used by the system before mounting or removing the module.
  Failure to do so may result in failure or malfunctions of the module.
- Do not drop or apply any impact to the battery.
  Doing so may damage the battery, resulting in electrolyte spillage inside the battery.
  If any impact has been applied, discard the battery and never use it.
- Do not install/remove the terminal block more than 50 times after the first use of the product.
  (IEC 61131-2 compliant)
- Before handling modules, touch a grounded metal object to discharge the static electricity from the human body.
  Failure to do so may cause failure or malfunctions of the module.

[DISPOSAL PRECAUTIONS]

⚠CAUTION

- When disposing of the product, treat it as an industrial waste.
  When disposing of batteries, separate them from other wastes according to the local regulations.
  (For details of the battery directive in EU member states, refer to Appendix 11.)

[TRANSPORTATION PRECAUTIONS]

⚠CAUTION

- When transporting lithium batteries, make sure to treat them based on the transportation regulations.
  (Refer to Appendix 10 for details of the relevant models.)

●CONDITIONS OF USE FOR THE PRODUCT●

(1) Mitsubishi programmable controller ("the PRODUCT") shall be used in conditions;
i) where any problem, fault or failure occurring in the PRODUCT, if any, shall not lead to any major or serious accident; and
ii) where the backup and fail-safe function are systematically or automatically provided outside of the PRODUCT for the case of any problem, fault or failure occurring in the PRODUCT.

(2) The PRODUCT has been designed and manufactured for the purpose of being used in general industries.
MITSUBISHI SHALL HAVE NO RESPONSIBILITY OR LIABILITY (INCLUDING, BUT NOT LIMITED TO ANY AND ALL RESPONSIBILITY OR LIABILITY BASED ON CONTRACT, WARRANTY, TORT, PRODUCT LIABILITY) FOR ANY INJURY OR DEATH TO PERSONS OR LOSS OR DAMAGE TO PROPERTY CAUSED BY the PRODUCT THAT ARE OPERATED OR USED IN APPLICATION NOT INTENDED OR EXCLUDED BY INSTRUCTIONS, PRECAUTIONS, OR WARNING CONTAINED IN MITSUBISHI'S USER, INSTRUCTION AND/OR SAFETY MANUALS, TECHNICAL BULLETINS AND GUIDELINES FOR the PRODUCT.
("Prohibited Application")
Prohibited Applications include, but not limited to, the use of the PRODUCT in;

- Nuclear Power Plants and any other power plants operated by Power companies, and/or any other cases in which the public could be affected if any problem or fault occurs in the PRODUCT.
- Railway companies or Public service purposes, and/or any other cases in which establishment of a special quality assurance system is required by the Purchaser or End User.
- Aircraft or Aerospace, Medical applications, Train equipment, transport equipment such as Elevator and Escalator, Incineration and Fuel devices, Vehicles, Manned transportation, Equipment for Recreation and Amusement, and Safety devices, handling of Nuclear or Hazardous Materials or Chemicals, Mining and Drilling, and/or other applications where there is a significant risk of injury to the public or property.

Notwithstanding the above, restrictions Mitsubishi may in its sole discretion, authorize use of the PRODUCT in one or more of the Prohibited Applications, provided that the usage of the PRODUCT is limited only for the specific applications agreed to by Mitsubishi and provided further that no special quality assurance or fail-safe, redundant or other safety features which exceed the general specifications of the PRODUCTs are required. For details, please contact the Mitsubishi representative in your region.

REVISIONS

*The manual number is noted at the lower left of the back cover.

Print Date	*Manual Number	Revision
Sep.1996	SH (NA)-3599-A	First printing
Feb,1999	SH(NA)-3599-B	Additional model SW□5□-GPPW GPP Function Software Package for Windows Addition The contents of the function version B has been added. Section 2.2, Section 7.2, Sections 8.2.1, 8.2.2, Section 19.8, Appendix 7, 8 Partial correction Safety Precautions, Contents, Section 1.2, Section 3.3.1, Chapter 4, Section 5.3, Section 6.1, Section 8.10.1, Section 15.2, Section 16.1, Section 19.7.1, Section 21.3.1, Appendix 1.6, Appendix 2
Dec., 2002	SH(NA)-3599-C	Equivalent to Japanese version D Partial correction SAFETY PRECAUTIONS, Section 3.1.2, Section 3.3.1, Section 3.3.2, Section 3.3.3, Section 8.2.1, Section 8.5, Section 12.1.5, Section 12.1.6, Chapter 13, Section 14.3, Section 15.1, Section 15.3, Section 16.1, Section 16.2, Section 17.2, Section 17.3, Section 19.1, Section 22.3.2, Appendix 3
Dec., 2003	SH(NA)-3599-D	Additional model A1SY42P Addition Appendix 9, 9.1, 9.2 Partial correction Section 3.3.1, Section 14.3, Section 18.3, Section 19.4.1, Section 19.8, Section 20.1.4, Section 21.3
Oct., 2006	SH(NA)-3599-E	Partial correction SAFETY PRECAUTIONS, Section 1.1, Section 1.2, Section 2.1, 3.1.1, Section 3.1.2, Section 3.3.1, Section 3.3.2, Chapter 4, Section 5.3, Section 7.1, Section 7.2, Section 8.1, Section 8.2.1, Section 8.3, Section 8.4.3, Section 8.6, Section 8.8, Section 8.9, Section 8.10.1, Section 9.3, Section 9.4, Section 10.1, Section 10.5, Section 10.6.3, Section 10.8, Section 12.1, Section 12.1.3, Section 12.1.5, Section 12.4, Chapter 13, Section 14.2, Section 15.1, Section 15.3, Section 16.1.1, Section 16.2, Section 16.3, Section 17.1, Section 17.2, Section 17.3, Section 17.5, Section 18.1, Section 18.2, Section 18.3, Section 19.1, Section 19.4.1, Section 19.4.2, Section 19.5, Section 19.6, Section 19.7, Section 19.7.1, Section 19.7.2, 20, Section 20.2.4, Section 20.2.6, Section 21.2, Section 21.3, Section 21.4, Section 21.5, Section 22.2.5, Section 22.2.6, Section 22.2.8, Section 22.3.3, Section 22.5.2, Appendix 1.1, Appendix 1.6, Appendix 2, Appendix 3, Appendix 4.1, Appendix 4.2, Appendix 5.1, Appendix 5.2, Deletion Section 14.2 Chapter change Section 14.3→Section 14.2

Print Date	*Manual Number	Revision
Apr.,2007	SH(NA)-3599-F	Partial correction Section 3.1.1, 3.3.1, 3.3.2, 17.1, 19.7.1, 19.7.2, 20.1.3, 20.2.7, 22.5.1, 22.5.2, Appendix 2, Appendix 3, Appendix 6.3 Addition Section 20.1.8
Sep.,2008	SH(NA)-3599-G	Partial correction SAFETY PRECAUTIONS, Section 5.2, 5.4, 8.2, 8.2.1, 8.5, 9.2, 9.3, 10.1, 10.3. 10.4, 10.5, 10.6.1, 12.1, 12.1.1, 12.1.7, 12.2, 12.3, 12.4, Chapter 13, Section 15.1, 15.3, 16.1.1, 16.2, 19.1, 19.2, 19.3, 19.4.1, 19.4.2, 19.7.1, 19.8, 20.1, 20.1.2, 20.1.3, 20.2, 20.2.1, 20.2.2, 20.2.3, 20.2.4, 20.2.6, 20.2.7, Chapter 21, Section 21.3.1, 21.3.2, 21.4, 21.5, 22.1, 22.2.2, 22.2.5, 22.3, 22.5.1, Appendix 2 Addition Appendix 11, Appendix 11.1, Appendix 11.2
Aug.,2009	SH(NA)-3599-H	Partial correction Section 15.1, 18.3, 18.6, Chapter 20, Section 21.3.1, 21.3.2, 22.3.3, 22.5.1, Appendix 2, 3
Nov.,2009	SH(NA)-3599-I	Partial correction SAFETY PRECAUTIONS, Section 3.3.1, 21.3.1 Addition CONDITIONS OF USE FOR THE PRODUCT
Mar.,2010	SH(NA)-3599-J	Partial correction SAFETY PRECAUTIONS, Section 3.3.1, 3.3.2, 19.1, 19.7.1, 21.2, 21.3.2, 22.3.7, Appendix 2
Jan., 2011	SH(NA)-3599-K	Partial correction Section 3.2, Section 15.1, Section 19.1, Section 20.1.1, Section 20.1.8 Addition Section 20.1.9

Japanese Manual Version SH-3587-N

Introduction

Thank you for purchasing the Mitsubishi programmable logic controller MELSEC-QnA series.
Before using your new PLC, please read this manual thoroughly to gain an understanding of its functions so that you can use it properly.
Please forward a copy of this manual to the end user.

Table of Contents

8 DEBUGGING FUNCTION 8 - 1 to 8 - 64

9 MAINTENANCE FUNCTION 9 - 1 to 9 - 19

10 OTHER FUNCTIONS 10 - 1 to 10 - 24

11 COMMENTS THAT CAN BE STORED IN Q2ASCPU 11 - 1 to 11 - 8

12 OVERVIEW OF PROCESSING PERFORMED BY THE Q2ASCPU12 - 1 to 12 - 28

13 PARAMETER LIST 13 - 1 to 13 - 10

14 SELECTING MEMORY CARD MODELS 14 - 1 to 14 - 4

15 HARDWARE SPECIFICATIONS OF CPU MODULES 15 - 1 to 15 - 6

16 POWER SUPPLY MODULE 16 - 1 to 16 - 6

21 MAINTENANCE AND INSPECTION 21 - 1 to 21 - 16

22 TROUBLESHOOTING 22 - 1 to 22 - 53

APPENDICES App - 1 to App - 165

ABOUT THIS MANUAL

The following are manuals related to this product.
Request for the manuals as needed according to the chart below.

RELATED MANUALS

Manual Name	Manual No. (Type code)
QnACPU GUIDEBOOK For the first-time user of QnACPU, describes steps on creating a program, writing the program in the CPU module, and debugging. Describes usage of QnACPU features. (Sold separately)	IB-66606 (13JF10)
QnACPU PROGRAMMING MANUAL (Fundamentals) Describes programming methods, device names and program types that are necessary in program creation. (Sold separately)	IB-66614 (13JF46)
QnACPU Programming Manual (Common Instructions) Describes how to use sequence instructions, basic instructions and application instructions. (Sold separately)	SH-080810ENG (13JW11)
QnACPU PROGRAMMING MANUAL (Special Functions) Describes dedicated instructions used in special-function modules. (Sold separately)	IB-66616 (13JF48)
QnACPU PROGRAMMING MANUAL (AD57 Instructions) Describes dedicated instructions used to control theA D57(S1)-type CRT controller module. (Sold separately)	IB-66617 (13JF49)
QCPU (Q mode)/QnACPU PROGRAMMLNG MANUAL (PID Control Instructions) Describes dedicated instructions used for PID control in Q2ACPU(S1), Q3ACPU and Q4ACPU. (Sold separately)	SH-080040 (13JF59)
QCPU (Q mode)/QnACPU PROGRAMMING MANUAL (SFC) Describes system components, performance specifications, functions, programming debug going and error codes of MELSAP (Sold separately)	SH-080041 (13JF60)
AnS Module Type I/O User's Manual Describes specification of AnS module as I/O module. (Sold separately)	IB-66541 (13JE81)
Type QnA/Q4AR MELSECNET/10 Network System Reference Manual Descrlbes MELSECNET/10 overview, specifications, part names and settings. (Sold separately)	IB-66620 (13JF77)
Type MELSECNET, MELSECNET/B Data Link System Reference Manual Describes MELSECNET(II) and MELSECNET/B overview, specifications, part names and settings. (Sold separately)	IB-66350 (13JF70)
GX Developer Version 8 Operating Manual Describes the online functions of GX Developer including the programming procedure, printing out procedure, and debugging procedure. (Included with product)	SH-080373 (13JU41)
Type SW2IVD-GPPQ GPP Software package OPERATING MANUAL (Offline) Describes SW2IVD-GPPQ's offline functions such as program creation, printout method and file maintenance. (Included with product)	IB-66774 (13J921)

Manual Name	Manual No. (Type code)
Type SW2IVD-GPPQ GPP Software package OPERATING MANUAL (Online) Describes SW2IVD-GPPQ's online functions such as monitoring and debugging methods. (Included with product)	IB-66775 (13J922)
Type SW2IVD-GPPQ GPP Software package OPERATING MANUAL (SFC) Describes MWLSAP-3 system components, performamce specifications, functions, system start-up procedure, SFC program editing method, monitoring method, printout method and error messages. (Included with product)	IB-66776 (13J923)
Type SW2IVD-GPPQ GPP Software package OPERATING MANUAL (Q6TEL) Describes Q6TEL system configuration, operating methods, etc. (Included with product)	IB-66777 (13J924)

USER PRECAUTONS

PRECAUTIONS WHEN USING THE QNA SERIES

When using a CPU module, format the memory using a peripheral device.
For details of memory format, refer to the following manuals.

- GX Developer Operating Manual
- SW☐IVD-GPPQ Software package Operating Manual (Online)

PRECAUTIONS FOR BATTERY

(1) The operation after removal of a battery
After removing a battery of the CPU module, format the memory using a peripheral device to start next operation. (Refer to Section 21.4)

(2) The operation after excess of a battery life
After removing a battery of the CPU module due to its excess life, format the memory using a peripheral device to start next operation. (Refer to Section 21.5)

1 ABOUT THIS MANUAL

1.1 About this Manual

This manual serves to explain the specifications and functions of the Q2ASCPU, Q2ASCPU-S1, Q2ASHCPU and Q2ASHCPU-S1 (abbreviated as Q2ASCPU hereafter), the specifications of other modules, and the maintenance required for smooth system operation, to users of MELSEC-QnA series programmable controllers.

It is divided into the following three main parts:

(1) Sections 2 and 3	These sections give the general description and system configuration for the Q2ASCPU. Read them to learn the features of Q2ASCPU, and the modules that can be used and points to note when configuring a system.
(2) Sections 4 to 15	These sections give the specifications and functions of Q2ASCPU. They describe each Q2ASCPU function to enable you to use the Q2ASCPU effectively.
(3) Sections 16 to 18	These sections describe the specifications and handling of units/ modules other than the CPU module (power supply module, base units, etc.) Read them to learn how to handle the power supply module, base units, memory cards, etc.
(4) Section 19 to 20	These section describes the loading and installation, EMC・ and low voltage directives.
(5) Section 21 to 22	These sections describe all aspects of maintenance, from installing the Q2ASCPU to daily inspections and troubleshooting. Read them to learn how to install the Q2ASCPU so as to ensure smooth operation, and how to carry out daily inspections and corrective action in the event of trouble.

REMARK

This manual does not cover MELSECNET(II) data link systems, MELSECNET/B data link systems, MELSECNET/10 networks, or the SFC function.
For details on each function, refer to the following manuals.

- MELSECNET(II), MELSECNET/B Data Link
  MELSECNET, MELSECNET/B Data Link System Reference Manual
- MELSECNET/10 Network
  MELSECNET/10 Network System Manual for QnA/Q4AR
- SFC Function
  QCPU (Q Mode)/QnACPU Programming Manual (SFC)

1.2 Abbreviations and Generic Terms Used in this Manual

The following abbreviations and generic terms are used in this manual.

(1)	Q2ASCPU..................................	Abbreviation for Q2ASCPU, Q2ASCPU-S1, Q2ASHCPU, and Q2ASHCPU-S1 type CPU modules.
(2)	Network module.........................	Abbreviation of A1SJ71QLP21, and A1SJ71QBR11 type MELSECNET/10 network modules.
(3)	Ethernet module.........................	Abbreviation of A1SJ71QE71N-B2 and A1SJ71QE71N-B5T type Ethernet interface modules.
(4)	Serial communication module....	Abbreviation of A1SJ71QC24(N), A1SJ71QC24(N)-R2 type serial communication module.
(5)	CC-Link....................................	Abbreviation of Control & Communication Link
(6)	GPP function............................	Abbreviations for the SW□IVD-GPPQ type GPP function software package, GX Developer.
(7)	Personal computer.....................	IBM's PC/AT or completely compatible computers.
(8)	Peripheral device capable .... of GPP functions	Generic term for a peripheral device capable of running the GPP function software, for example an IBM PC/AT.
(9)	Q6PU..	Abbreviation for Q6PU programming unit.
(10)	Peripheral device.......................	Generic term for a device that is connected to a QnACPU and can be used to operate it, for example a personal computer or Q6PU.
(11)	Built-in RAM..............................	A RAM incorporated in the Q2AS CPU that stores sequence programs and other data.
(12)	Memory card.............................	Abbreviation for Q1MEM-□□□ type memory card
(13)	ACPU..	Generic term for a MELSEC-A series programmable controller

2 OVERVIEW

2.1 Features

Q2ASCPU has the following features.

(1) Large memory capacity
  (a) Q2AS(H)CPU-S1 has a program capacity of 60k steps, which means that 60k steps can be used for a single program (Q2AS(H)CPU: 28k steps).
  (b) The device memory capacity is 29k words and the user can change the number of points as required.
  For example, the default number of points for internal relays (M) is 8k points, but this can be expanded up to 32k points.
  (c) One memory card of a maximum of 2M bytes can be installed.
  Memory cards are used to store programs, comments, statements, and file registers.
  (Programs can be stored in the CPU module itself, so a memory card is not essential to run a CPU module.)

(2) High-speed processing
  (a) Higher operation processing speeds have been achieved for basic instructions and application instructions.

	A2USCPU(S1)	Q2ASHCPU(S1)
Basic instructions	0.2μs	→ 0.075μs
Application instructions	1.2μs	→ 0.225μs

  (b) The access time for expansion data memory (file registers: R) has been conformed with the internal devices of the Q2ASCPU (data registers: D, and link registers: W).
  (c) Reading/writing of the buffer memories of special function modules dedicated to QnA (serial communication modules) have been realised processing speed-up by six times compared to AnUCPU.
  (The processing speed of the existing special function modules for ACPU use is about the same as that when using AnUCPU.)
  (d) A high-speed access base unit (A1S38HB/A1S38HBEU) is available to speed up the processing time for accessing special function modules such as network modules and serial communication modules that handle large quantities of data.
  Simply by mounting the special function module on the high-speed access base unit, the access processing speed is increased when the Q2ASCPU accesses the special function module.

(3) Selection of program execution type that is appropriate for the control has been realised.

There are four program execution types to be selected as follows.

(a) Initial execution type

This program type is executed once only when the Q2ASCPU is set to RUN.

(b) Scan execution type

This program type is run continually while the Q2ASCPU is in the RUN status. This is equivalent to a conventional program that runs from step 0 to END instruction.It is possible to create subroutine programs and interrupt programs for this type of program.

(c) Low-speed execution type

This is a program type which is executed only during the surplus constant scan time (process to preset the program execution time for constant scan time) or during the set execution time of the low-speed execution program.

(d) Stand-by type

This type of program consists entirely of a subroutine program or interrupt program.

(4) The SFC language MELSAP3 has been supplied.

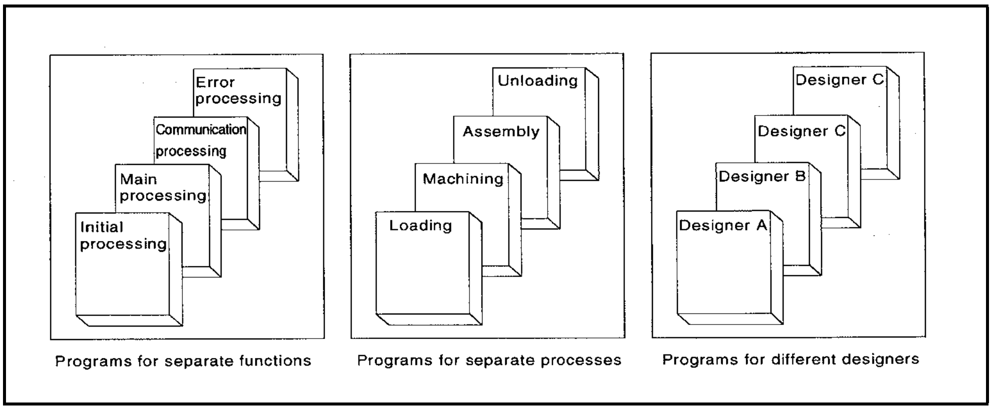
With enhancement of step attributes and SFC control instructions, MELSAP3 makes SFC programming even easier.

(5) A software development environment that improves program productivity has been realized.

(a) In order to enable the design of structured programs, a file format has been adopted for programs.

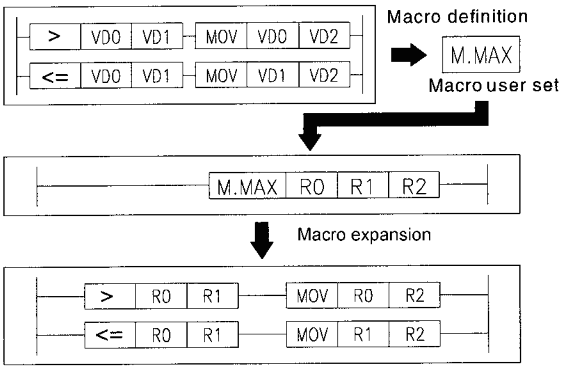
What would conventionally have been a single continuous program can now be handled in a structured way as a number of files.

This allows for design work to be shared by several designers, and allows management of programs in accordance with functions, processes, or designers, etc.



Programs for separate functions Programs for separate processes Programs for different designers

(b) The user can standardize and simplify programs by creating and using macro instructions corresponding to functions.



(c) Devices can be used without restrictions.
  1) Word device bit operations are possible.
  2) Differential contacts can be used.
  3) Buffer memories of special function modules can be accessed directly from a program as devices.
  4) The link data of network modules can be accessed directly from a program as devices.

(d) Ease of operation for GPP function program editing has been improved.
  1) Up to four programs, data, etc., can be edited simultaneously.
     Programs and data can be cut and pasted between edited objects.
  2) Ladder editing is possible while the ladder is displayed with comments.
  3) Familiar operations can be performed with pull-down menus and dialog boxes.

(e) The debugging function at start-up has been perfected.
  1) Ladder modification while performing monitoring is possible.
  2) Coil ON/OFF causes can be searched for.
  3) The timing for monitoring can be set using a step number or device status, allowing debugging to be conducted under the optimum conditions.
  4) Devices for which index qualifications have been set can be monitored.

(f) The GPP function document creation function has been strengthened.
  1) Since comments can now comprise 32 characters, they can be more detailed than before.
  2) Comments can now be set for all devices.
  3) The statements and notes appended to programs can now be managed as an integral part of the program, which makes program modifications and utilization easier.
  4) Printout data can be stored in a file.

(g) A powerful array of support software packages is available for program creation.
   1) Data conversion package
      Comment data, device data, etc., which is created with spreadsheet software and text editors available on the market, can be converted to files for GPP function use.
      Conversely, files created for GPP function use can be converted to data for spreadsheets or text editors.
   2) Macro/library package
      The basic programs for accessing special function modules, and standard programs for error detection, alarm processing, etc., have been brought together as a package of macro and library data.
   3) Ladder sequence linking package
      This package is used to link multiple programs to make a single program.
      This has an automatic allocation function that ensures that devices from each program without duplicating in the created program.
   4) CAD interface program
      This package is used to handle sequence ladders, instruction lists, comment data and SFC diagrams as CAD data and communicate these data to CAD systems.

2.2 Additional Functions of Q2ASCPU

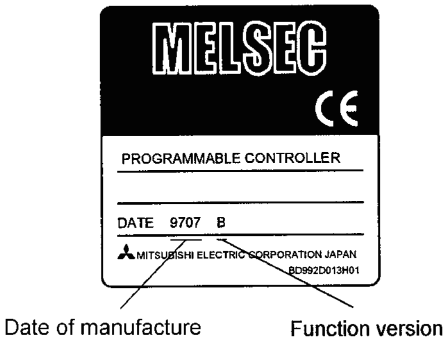
New functions and instructions for special function module are added to the Q2ASCPU.
[Additional functions]

- Variety of local devices.......... Refer to Section 2.2.1 (1)
  - Monitor test of local device.......... Refer to Section 8.2.2
  - Use of local device at the subroutine/interrupt program storage destination.......... Refer to Appendix 7
- Auto refresh setting of CC-Link.......... Refer to Section 2.2.1 (2), Section 7.2.
- MELSECNET/10 relay communication from the Ethernet module (Network relay).......... Refer to Section 2.2.1 (3), Appendix 8.
- Addition of AJ71QC24N-compatible commands .... Section 2.2.1 (4)

[Added instructions for special function module]
The following instructions have been added for function version "B" of the Q2ASCPU:

- A1SJ61QBT11 control instructions 13
- A1SD75 control instructions 19
- A1SJ71ID□-R4 control instructions 12
- A1SJ71QE71 control instructions 10

Additional function/special function module instructions can be used for the Q2ASCPU described function version B in the date column of the rating plate.
Check that function version B is described on the Q2ASCPU rating plate before using the additional function/special function module instructions.
If your Q2ASCPU does not have indication of function version B, skip this item and the description of additional functions.



When using additional function/special function module instructions of the Q2ASCPU, it is necessary to match the GPP function model and the function version/version of the applicable special function module. (Refer to Table 2.1.)

Table 2.1 List of combination between Q2ASCPU and function version/version of special function module

Module/package Name		QnACPU	SW0IVD-GPPQ SW1IVD-GPPQ	SW2IVD-GPPQ	A1SJ71QE71(B2), (B5)	A1SD75P-S3	A1SJ71 ID□-R4	A1SJ61QBT11	A1SJ71QC24(N)(R2)
Condition	Function version	9707B and later	–	ÑüÑü	9707B and later	–	–	9707B and later	–
	Version	–	No restriction	No restriction	–	No restriction	BC and later	–	No restriction
Local device monitor test▪		○	×	○	–	–	–	–	–
Local device switching of subroutine/interrupt program		○	–		–	–	–	–	–
Auto refresh function of CC-Link		○	×	○	–	–	–	○	–
A1SJ61QBT11 control instructions		○	○		–	–	–	○	–
MELSECNET/10 relay communication from Ethernet		○	×	△	○	–	–	–	–
A1SJ71QE71 control instruction		○	○		○	–	–	–	–
A1SD75 control instructions		○	○	○	–	○	–	–	–
ID interface module instructions		○	○		–	–	○	–	–
Compatibility with A1SJ71QC24N commands		○	–		–	–	–	–	○

REMARK

1) Marks ○, -, △ and × in Table 2.1 indicate as follows:
   - ○: Essential for use of function and instruction
   - –: Irrelevant to function and instruction
   - △: Required in the case of access to the QnACPU in other stations from the peripheral device via Ethernet
   - ×: Not available on peripheral devices.
2) GX Developer supports functions of function version B.

2.2.1 Overview of added functions

This section shows an overview of the added functions.

(1) Variety of local device

(a) The device set as the local device at "Device" in Parameter can be monitored and tested with a peripheral device.
This function allows checking and debug of the local device in the program monitored with a peripheral device.

(b) The local device of the file where the subroutine program/interrupt program is stored has made it possible to be used during execution of the subroutine program/interrupt program.
For this function, even if an operation using the local device of the subroutine program is carried out, the original local device cannot be overwritten. In addition, even if an operation using the local device of the interrupt program, the local device which is executed before starting up the interrupt program cannot be overwritten.

(c) The following GPP function software packages are required to perform the monitor test of the local device:

- Personal computer
  GX Developer, SW□IVD-GPPQ type GPP function software package

(2) Auto Refresh Setting of CC-Link

(a) When setting auto refresh of the CC-Link on the peripheral function, cyclic communication with other stations connected to the CC-Link can be automatically performed according to the set auto refresh data.

- Remote I/O station (Communication in ON/OFF data)
- Remote device station (Communication in ON/OFF data and Word data)
- Intelligent device station (Communication in ON/OFF data and Word data)
- Local station/master station (Communication in ON/OFF data and Word data)

The auto refresh setting of the CC-Link allows communication with other stations using the FROM/TO instruction without communicating with the master station of the CC-Link.

(b) Auto refresh is available for up to 8 CC-Link modules for each unit of Q2ASCPU. Communication for 9th CC-Link modules and more can be performed with the CC-Link module using the FROM/TO instruction.

(c) The following GPP function software packages are required to perform the auto refresh setting of the CC-Link:

- Personal computer
  GX Developer, SW□IVD-GPPQ type GPP function software package

It is necessary to upgrade the master station･local station module of CC-Link to function version B or later.

(3) Netwotk relay from Ethernet module

    (a) In the network system with mixture of Ethernet and MELSECNET/10, data can be communicated with the Q2ASCPU of other stations via multiple Ethernets or MELSECNET/10 modules.

    (b) For the network relay from the Ethernet module, the function version of the Ethernet module should be upgraded to "B" or later.

(4) A1SJ71QC24N-compatible commands are possible.

    (a) The following A1SJ71QC24N commands are available:
    - Multiple blocks batch read: Command "0406"
    - Multiple blocks batch write: Command "1406"

    (b) Multiple blocks batch read/batch write is available with A1SJ71QC24N(-R2, R4).
    Multiple blocks batch read/batch write is not available with A1SJ71QC24(-R2, R4).
    For commands of multiple blocks batch read/batch write, refer to the following manual:
    - Corresponding Additional Explanation for A1SJ71QC24N [-R2/R4]

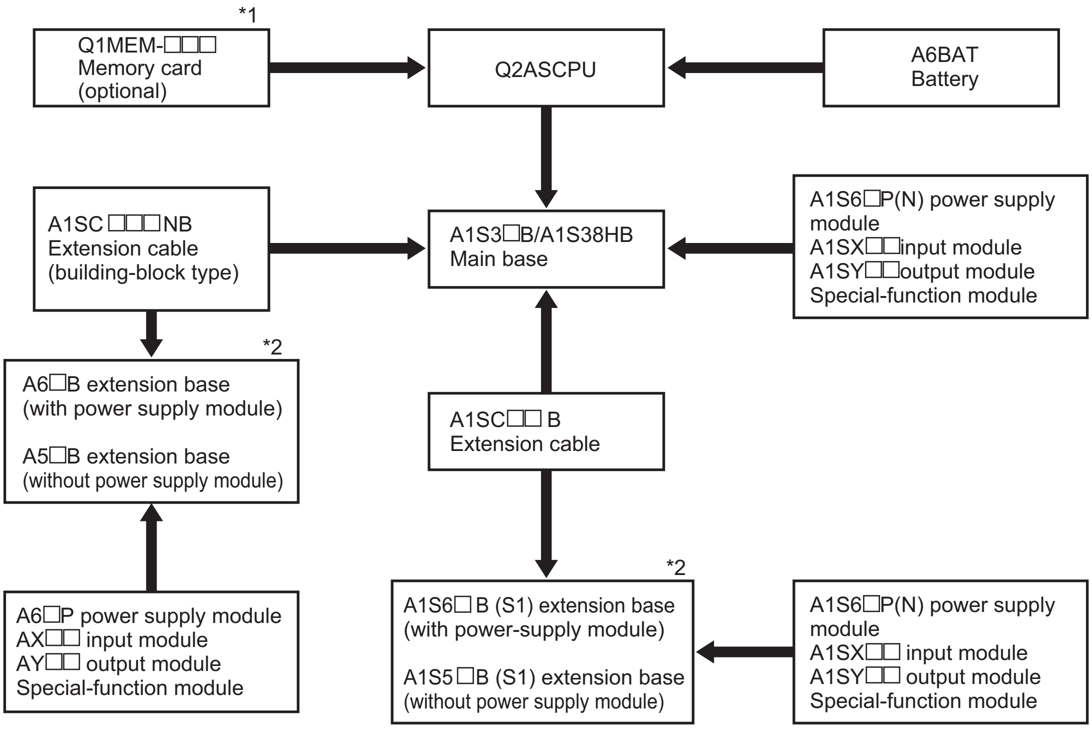
3 SYSTEM CONFIGURATION

This section describes the system configurations that can be used for a system centered on a Q2ASCPU, cautions on configuring the system, and the system equipment.

3.1 System Configuration

The following shows the configuration of equipment and peripheral device when a Q2ASCPU is used in a stand-alone system.

3.1.1 Equipment configuration in a stand-alone system

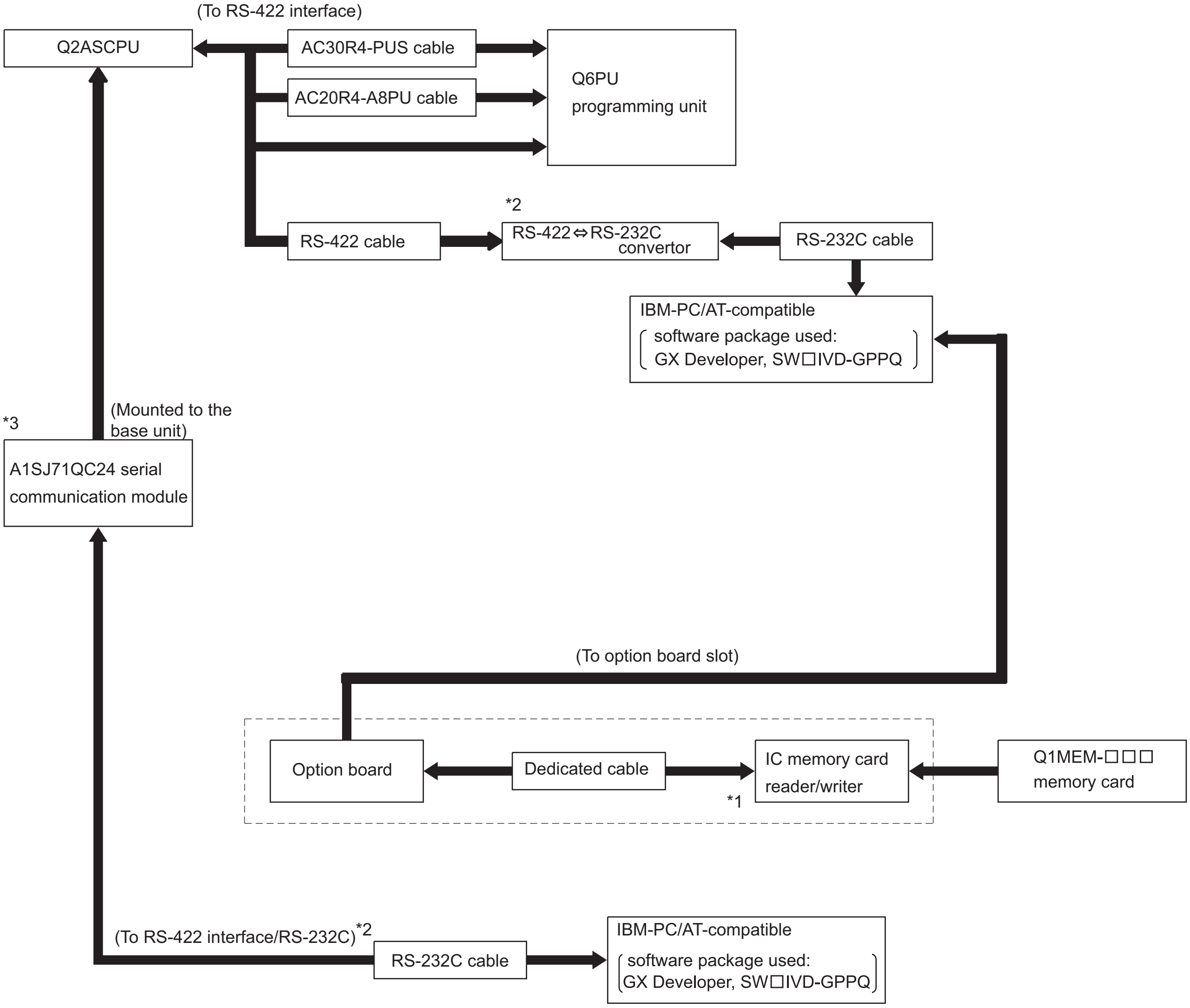


POINT

*1 Up to one memory card can be installed, if required.
SRAM and E^{2}PROM memory cards allow file read/write when mounted on the CPU module.

*2 When using an A1S5□B(S1), A5□B extension base unit, pay particular attention to the power supply capacity of the main base unit.In the case of I/O modules and the special function module with a high internal current consumption, mounting on an A1S6□B(S1), A6□ extension base unit is recommended.
(Refer to Section 16.1 and Section 17.3 for details.)

3.1.2 Configuration of peripheral devices capable of Q2ASCPU



*1 For details on the IC memory card reader/writer setting, refer to Operating Manual for the peripheral device capable of GPP functions.

*2 For connection to RS-422 interface, use the RS-422 ⇔ /RS-232C converter.

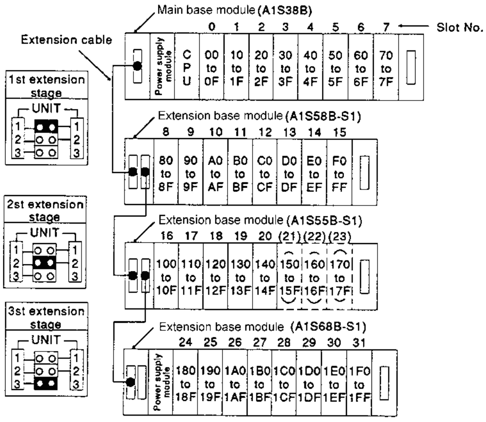
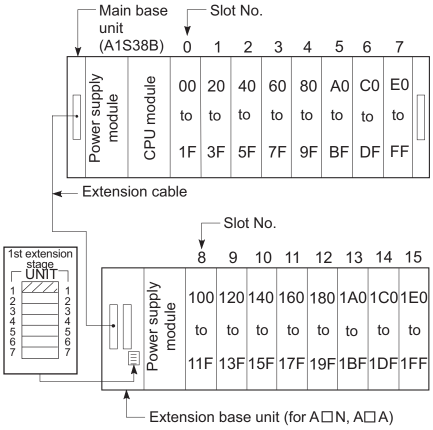
*3 When connecting the serial communication module and the peripheral devices capable of GPP function, see User's Manual of the serial communication modules.

REMARK

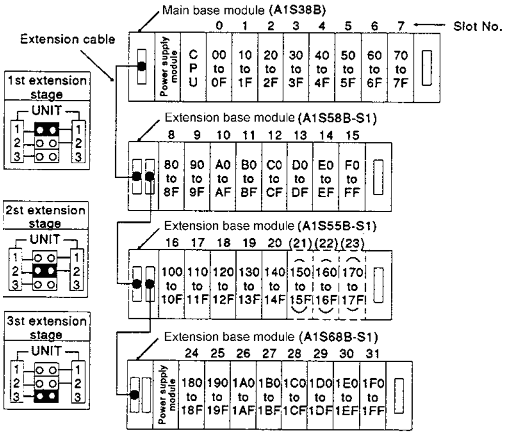
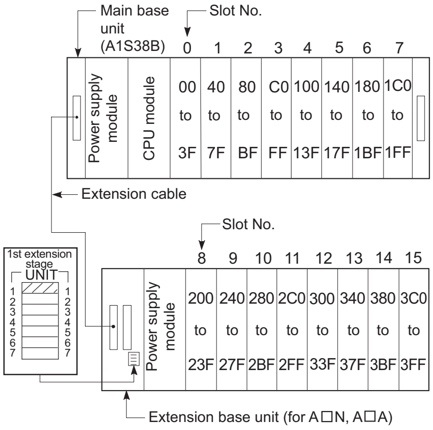
1. For details on the system configuration for each peripheral device, refer to the Operating Manual for each.
2. Q2ASCPU can connect a peripheral device capable of ACPU only when accessing an ACPU in another station via a MELSECNET/10 or MELSECNET data link. (However, Q2ASCPU cannot be accessed.) In this case, set SW1 of system setting switch 2 on the CPU module ON.

3.2 System Configuration Overview

(a) Q2ASCPU, Q2ASHCPU system

System configuration	[When the A1S6☐B(S1), A1S5☐B(S1) extension base is used] The following shows an example that the 16-point module is installed to each slot. 	[When the A6☐B, A5☐B extension base is used] The following shows an example that the 32-point module is installed to each slot. 
Maximum number of extension stages	3rd extension stage	1st extension stage
Maximum number of I/O modules	16 modules	
Maximum number of I/O points	512 points	
Main base unit model name	A1S32B, A1S33B, A1S35B, A1S38B, A1S38HB, A1S38HBEU	
Extension base unit model name	A1S65B(S1), A1S68B(S1), A1S52B(S1), A1S55B(S1), A1S58B(S1)	A62B, A65B, A68B, A52B, A55B, A58B
Extension cable model name	A1SC03B, A1SC07B, A1SC12B, A1SC30B, A1SC01B (right-side installation), A1SC60B	A1SC05NBA1SC07NBA1SC30NBA1SC50NB
Precautions	(1) Only one A6☐B, A5☐B extension base can be used.(The second extension module cannot be used.) (2) When the extension base A1S52B(S1), A1S55B(S1), A58B(S1) or A52B, A55B, A58B are used, the 5VDC power is supplied from the power supply module of the main base unit.Before use, refer to Section 17.3 and examine if it can be used. (3) Limit the length of extension cable to 6m (236inch) or shorter. (4) When using the extension cable, do not install it with the main circuit cables, which has high voltage, large current, or install them close together.	
I/O number assignment (When I/O assignment is not performed)	(1) Assign I/O numbers to the main base unit first, then to the extension base unit. (2) Assign I/O numbers as if both main base unit and extension base unit have 8 slots each.When the A1S32B/A1S33B/A1S35B for 2/3/5 slots are used as the main base unit, add 6/5/3 slots (96 points/80 points/48 points) and assign the extension base unit I/O numbers. (3) 16 points are assigned to an empty slot. (4) When A6☐B or A5☐B is used, be sure to set to a single extension level.If it is set to the number of skipped stages, 16 points/slot are assigned to all of skipped stages× 8 slots, and thus it does not work. (5) Items (2) to (3) can be changed by the I/O assignment. (Refer to Section 5.3)	

(b) Q2ASCPU-S1, Q2ASHCPU-S1 system

System configuration	[When the A1S6□B(S1), A1S5□B(S1) extension base is used] The following shows an example that the 16-point module is installed to each slot.  	[When the A6□B, A5□B extension base is used] The following shows an example that the 64-point module is installed to each slot. 
Maximum number of extension stages	3rd extension stage	1st extension stage
Maximum number of I/O modules	16 modules	
Maximum number of I/O points	1024 points	
Main base unit model name	A1S32B, A1S33B, A1S35B, A1S38B, A1S38HB, A1S38HBEU	
Extension base unit model name	A1S65B(S1), A1S68B(S1), A1S52B(S1), A1S55B(S1), A1S58B(S1)	A62B, A65B, A68B, A52B, A55B, A58B
Extension cable model name	A1SC03B, A1SC07B, A1SC12B, A1SC30B, A1SC01B (right-side installation), A1SC60B	A1SC05NBA1SC07NBA1SC30NBA1SC50NB
Precautions	(1)Only one A6□B, A5□B extension base can be used.(The second extension module cannot be used.) (2) When the extension base A1S52B(S1), A1S55B(S1), A58B(S1) or A52B, A55B, A58B are used, the 5VDC power is supplied from the power supply module of the main base unit.Before use, refer to Section 17.3 and examine if it can be used. (3) Limit the length of extension cable to 6m (236inch) or shorter. (4) When using the extension cable, do not install it with the main circuit cables, which has high voltage, large current, or install them close together.	
I/O number assignment (When I/O assignment is not performed)	(1) Assign I/O numbers to the main base unit first, then to the extension base unit. (2) Assign I/O numbers as if both main base unit and extension base unit have 8 slots each.When the A1S32B/A1S33B/A1S35B for 2/3/5 slots are used as the main base unit, add 6/5/3 slots (96 points/80 points/48 points) and assign the extension base unit I/O numbers. (3) 16 points are assigned to an empty slot. (4)When A6□B or A5□B is used, be sure to set to a single extension level.If it is set to the number of skipped stages, 16 points/slot are assigned to all of skipped stages× 8 slots, and thus it does not work. (5) Items (2) to (3) can be changed by the I/O assignment. (Refer to Section 5.3)	

3.3 System Equipment

3.3.1 System equipment list

The following shows the system equipment (modules and peripheral devices) that can be used in a Q2ASCPU system.

(1) For QnA module

Product Name	Model Name	Description	Number of Occupied Points (points) [I/O Assignment Module Type]	Current consumption		Remark
				5VDC (A)	24VDC (A)	
CPU module	Q2ASCPU	Number of I/O points: 512, built-in RAM: 28k steps	–	0.3	–	Memory card procured separately. Including memory card current consumption.
	Q2ASHCPU	Number of I/O points: 512, built-in RAM: 28k steps		0.7	–	
	Q2ASCPU-S1	Number of I/O points: 1024, built-in RAM: 60k steps		0.3	–	
	Q2ASHCPU-S1	Number of I/O points: 1024, built-in RAM: 60k steps		0.7	–	
Memory card	Q1MEM-64S	SRAM, 64k bytes	–	–	–	
	Q1MEM-128S	SRAM, 128k bytes				
	Q1MEM-256S	SRAM, 256k bytes				
	Q1MEM-512S	SRAM, 512k bytes				
	Q1MEM-1MS	SRAM, 1M bytes				
	Q1MEM-2MS	SRAM, 2M bytes				
	Q1MEM-64SE	SRAM, 32k bytes, E^{2}PROM, 32k bytes				
	Q1MEM-128SE	SRAM, 64k bytes, E^{2}PROM, 64k bytes				
	Q1MEM-256SE	SRAM, 128k bytes, E^{2}PROM, 128k bytes				
	Q1MEM-512SE	SRAM, 256k bytes, E^{2}PROM, 256k bytes				
	Q1MEM-1MSE	SRAM, 512k bytes, E^{2}PROM, 512k bytes				
Power supply module	A1S61PN	5VDC, 5A 100/200VAC input	–	–	–	For power supply slots of main base and extension base
	A1S62PN	5VDC, 3A/24VDC, 0.6A				
	A1S63P	5VDC, 5A 24VDC input				

Product Name	Model Name	Description	Number of Occupied Points (points) [I/O Assignment Module Type]	Current consumption		Remark
				5VDC (A)	24VDC (A)	
Input module	A1SX10	16-point 100 to 120 VAC input module	16 (16 inputs)	0.05	–	
	A1SX10EU	16-point 100 to 120 VAC input module	16 (16 inputs)	0.05	–	
	A1SX20	16-point 200 to 240 VAC input module	16 (16 inputs)	0.05	–	
	A1SX20EU	16-point 200 to 240 VAC input module	16 (16 inputs)	0.05	–	
	A1SX30	16 points 12/24VDC, 12/24VAC input module	16 (16 inputs)	0.05	–	
	A1SX40	16 points 12/24VDC input module	16 (16 inputs)	0.05	–	
	A1SX40-S1	16 points 24VDC input module	16 (16 inputs)	0.05	–	
	A1SX40-S2	16 points 24VDC input module	16 (16 inputs)	0.05	–	
	A1SX41	32 points 12/24VDC input module	32 (32 inputs)	0.08	–	
	A1SX41-S1	32 points 24VDC input module	32 (32 inputs)	0.12	–	
	A1SX41-S2	32 points 24VDC input module	32 (32 inputs)	0.08	–	
	A1SX42	64-input 12/24VDC input module	64 (64 inputs)	0.09	–	
	A1SX42-S1	64 points 24VDC input module	64 (64 inputs)	0.16	–	
	A1SX42-S2	64 points 24VDC input module	64 (64 inputs)	0.09	–	
	A1SX71	32 points 5/12/24VDC input module	32 (32 inputs)	0.075	–	
	A1SX80	16 points 12/24VDC Sink/source input module	16 (16 inputs)	0.05	–	
	A1SX80-S1	16 points 24VDC Sink/source input module	16 (16 inputs)	0.05	–	
	A1SX80-S2	16 points 24VDC Sink/source input module	16 (16 inputs)	0.05	–	
	A1SX81	32 points 12/24VDC Sink/source input module	16 (16 inputs)	0.08	–	
	A1SX81-S2	32 points 24VDC Sink/source input module	32 (32 inputs)	0.08	–	
	A1SX82-S1	64 points 24VDC Sink/source input module	32 (32 inputs)	0.16	–	

Product Name	Model Name	Description	Number of Occupied Points (points) [I/O Assignment Module Type]	Current consumption		Remark
				5VDC (A)	24VDC (A)	
Output module	A1SY10	16-output relay contact output module (2A)	16 (16 outputs)	0.12	0.09	
	A1SY10EU	16-output relay contact output module (2A)	16 (16 outputs)	0.12	0.10	
	A1SY14EU	12-output relay contact output module (2A)	16 (16 outputs)	0.12	0.10	
	A1SY18A	8 points Relay contact output module (2A) for independent contact	16 (16 outputs)	0.24	0.075	
	A1SY18AEU	8 points Relay contact output module (2A) for independent contact	16 (16 outputs)	0.24	0.075	
	A1SY22	16 points triac output module (0.6A)	16 (16 outputs)	0.27	(200VAC) 0.002	
	A1SY28A	8 points triac output module (1A) all points independent	16 (16 outputs)	0.13	–	
	A1SY40	16-output 12/24VDC transistor output module (0.1A) sink type	16 (16 outputs)	0.27	0.008	
	A1SY40P	16-output 12/24VDC transistor output module (0.1A) sink type	16 (16 outputs)	0.08	0.011	
	A1SY41	32-output 12/24VDC transistor output module (0.1A) sink type	32 (32 outputs)	0.50	0.008	
	A1SY41P	32-output 12/24VDC transistor output module (0.1A) sink type	32 (32 outputs)	0.14	0.012	
	A1SY42	64-output 12/24VDC transistor output module (0.1A) sink type	64 (64 outputs)	0.93	0.008	
	A1SY50	16-output 12/24VDC transistor output module (0.5A) sink type	16 (16 outputs)	0.12	0.06	
	A1SY60	16 points 24VDC transistor output module (2A) sink type	16 (16 outputs)	0.12	0.015	
	A1SY60E	16 points 5/12/24VDC transistor output module (2A) source type	16 (16 outputs)	0.20	0.01	
	A1SY68A	8 points 5/12/24/48VDC transistor output module sink/source type all points independent	16 (16 outputs)	0.11	–	
	A1SY71	32 points 5/12VDC transistor output module (0.016A) sink type	32 (32 outputs)	0.40	0.15	
	A1SY80	16-output 12/24VDC transistor output module (0.8A) source type	16 (16 outputs)	0.12	0.02	
	A1SY81	32-output 12/24VDC transistor output module (0.1A) source type	32 (32 outputs)	0.50	0.008	
	A1SY82	64-output 12/24VDC transistor output module (0.1A) source type	64 (64 outputs)	0.93	0.008	

Product Name	Model Name	Description	Number of Occupied Points (points) [I/O Assignment Module Type]	Current consumption		Remark
				5VDC (A)	24VDC (A)	
I/O module	A1SH42	32 points 12/24VDC input module 32-output 12/24VDC transistor output module (0.1A) sink type	32 (32 outputs)	0.50	0.008	
	A1SH42-S1	32 points 24VDC input module 32-output 12/24VDC transistor output module (0.1A) sink type	32 (32 outputs)	0.50	0.008	
	A1SX48Y18	8 points 24VDC input module 8 points Relay contact output module (2A)	16 (16 outputs)	0.085	0.045	
	A1SX48Y58	8 points 24VDC input module 8-output 12/24VDC transistor output module (0.5A)	16 (16 outputs)	0.06	0.06	
Dynamic input module	A1S42X	16/32/48/64 points 12/24VDC dynamic input module	Specified points [Input Set number of points]	0.08	–	
Dynamic output module	A1S42Y	16/32/48/64 points 12/24VDC dynamic output module	Specified points [Output Set number of points]	0.18	0.055	
Blank cover	A1SG60	Dust-proof cover for unused slot	16 [Empty]	–	–	
Dummy module	A1SG62	16 point, 32 point, 48 point, 64 point selectable module	Specified points [Input Set number of points]	–	–	
Pulse catch module	A1SP60	Pulse input module with short pulse duration (minimum pulse duration:0.5 ms) Input 16 points	16 (16 outputs)	0.055	–	
Analog timer module	A1ST60	Allows for different set timer value (0.1 to 1.0 s, 1 to 10 s, 10 to 60 s, 60 to 600 s) depending on volume Analog timer 8 points	16 (16 outputs)	0.055	–	
Interrupt module	A1SI61	For interrupt program execution Interrupt module (Interrupt input:16 points)	32 [Special 32 points]	0.057	–	
High-speed counter module	A1SD61	32 bit signed binary 50 KPPS, 1 channel	32 [Special 32 points]	0.35	–	
	A1SD62	24 bit signed binary, 2 channels 100 KPPS, Transistor output (sink type)	32 [Special 32 points]	0.1	–	
	A1SD62D	24 bit signed binary, 2 channels 200 KPPS, Transistor output (sink type)	32 [Special 32 points]	0.25	–	
	A1SD62D-S1	24 bit signed binary, 2 channels 200 KPPS, Transistor output (sink type)	32 [Special 32 points]	0.27	–	
	A1SD62E	24 bit signed binary, 2 channels 100 KPPS, Transistor output (source type)	32 [Special 32 points]	0.1	–	
A/D converter module	A1S64AD	4 to 20mA/0 to 10V 4 analog channels	32 [Special 32 points]	0.4	–	
	A1S68AD	4 to 20mA/0 to 10V 8 analog channels	32 [Special 32 points]	0.4	–	

Product Name	Model Name	Description	Number of Occupied Points (points) [I/O Assignment Module Type]	Current consumption		Remark
				5VDC (A)	24VDC (A)	
D/A converter module	A1S62DA	4 to 20mA/0 to 10V Analog output, 2 channels	32 [Special 32 points]	0.8	–	
	A1S68DAV	-10 to 10V input Analog output, 8 channels	32 [Special 32 points]	0.65	–	
	A1S68DAI	4 to 20mA input Analog output, 8 channels	32 [Special 32 points]	0.85	–	
Analog I/O module	A1S63ADA	Analog input, Two channels Controllable simplified loop, Analog output. 1 channel	32 [Special 32 points]	0.8	–	
	A1S66ADA	Analog input, Four channels Controllable simplified loop, Analog output, 2 channels	64 (64 outputs)	0.21	0.16	
Temperature digital converter module	A1S62RD3	For connecting Pt 100 (3-wire type) Temperature input 2 channels	32 [Special 32 points]	0.49	–	
	A1S62RD4	For connecting Pt 100 (4-wire type) Temperature input 2 channels	32 [Special 32 points]	0.39	–	
	A1S68TD	Temperature input 8 channels	32 [Special 32 points]	0.32	–	
Temperature control module	A1S62TCTT-S2	Transistor output, temperature input 2 channels/module PID control : ON/OFF pulse	32 [Special 32 points]	0.19	–	
	A1S62TCTTBW-S2	Transistor output, temperature input 2 channels/module PID control : ON/OFF pulse, heater wire breakage detection function	32 [Special 32 points]	0.28	–	
	A1S62TCRT-S2	Transistor output, platinum RTD(Resistance Temperature Detector) input 2 channels/module PID control : ON/OFF pulse	32 [Special 32 points]	0.19	–	
	A1S62TCRTBW-S2	Transistor output, platinum RTD input 2 channels/module PID control : ON/OFF pulse, heater wire breakage detection function	32 [Special 32 points]	0.28	–	
	A1S64TCTT-S1	Transistor output, temperature input 4 channels/module PID control : ON/OFF pulse or 2- position control	32 [Special 32 points]	0.33	–	
	A1S64TCTTBW-S1	Transistor output, temperature input 4 channels/module PID control : ON/OFF pulse or 2- position control Heater wire breakage detection function	32 [Special 32 points]	0.42	–	
	A1S64TCRT-S1	Transistor output, temperature input 4 channels/module PID control : ON/OFF pulse or 2- position control	32 [Special 32 points]	0.33	–	
	A1S64TCRTBW-S1	Transistor output, temperature input 4 channels/module PID control : ON/OFF pulse or 2- position control Heater wire breakage detection function	32 [Special 32 points]	0.42	–	

Product Name	Model Name	Description	Number of Occupied Points (points) [I/O Assignment Module Type]	Current consumption		Remark
				5VDC (A)	24VDC (A)	
Temperature control module	A1S64TCTRT	Transistor output, thermocouple input, or platinum RTD input [For standard contro] 4 channels/module PID control: ON/OFF pulse or 2 positioning control [For heating-cooling control] 2 channels/module PID control: ON/OFF pulse	32 [Special 32 points]	0.33 (0.19)*	–	*:When the temperature conversion function of unused channels are not used in the heating-cooling control
	A1S64TCTRT BW	Transistor output, thermocouple input, or platinum RTD input [For standard control] 4channels/module PID control: ON/OFF pulse or 2 positioning control [For heating-cooling control] 2channels/module PID control: ON/OFF pulse, wire breakage detection function	32 [Special 32 points]	0.39 (0.25)*	–	
Computer link module	A1SJ71QC24	Computer link function 300 to 19,200bps RS-232C 1 channel, RS-422/485 1 channel	32 [Special 32 points]	0.24	–	• Dedicated to QnACPU
	A1SJ71QC24-R2	Computer link & printer function RS-232C 2 channel 300 to 19,200bps	32 [Special 32 points]	0.155	–	
	A1SJ71QC24 N	Computer link function 300 to 115,200bps RS-232C 1 channel, RS-422/485 1 channel	32 [Special 32 points]	0.35	–	
	A1SJ71QC24 N1			0.38		
	A1SJ71QC24 N-R2	Computer link & printer function RS-232C 2 channel 300 to 115,200bps	32 [Special 32 points]	0.3	–	
	A1SJ71QC24 N1-R2			0.3		
	A1SJ71UC24-R2[*2]	Computer link function RC-232C 1 channel	32 [Special 32 points]	0.1	–	
	A1SJ71UC24-PRF[*2]	Computer link & printer function RS-232C 1 channel	32 [Special 32 points]	0.1	–	
	A1SJ71UC24-R4[*2]	Computer link function, multidrop link function RS-422 / RS-485 1 channel	32 [Special 32 points]	0.1	–	
Ethernet interface module	A1SJ71E71N3-T[*2]	10 BASE-T	32 [Special 32 points]	0.69	–	
	A1SJ71E71N-B2[*2]	10 BASE2	32 [Special 32 points]	0.66	–	
	A1SJ71E71N-B5[*2]	10BASE5	32 [Special 32 points]	0.57	–	

Product Name	Model Name	Description	Number of Occupied Points (points) [I/O Assignment Module Type]	Current consumption		Remark
				5VDC (A)	24VDC (A)	
Ethernet interface module	A1SJ71QE71N-B2	10 BASE2	32 [Special 32 points]	0.53	–	• Dedicated to QnACPU • Maximum 4 modules can be used for one CPU module. (Refer to Section 3.3.2)
	A1SJ71QE71N-B5	10BASE5	32 [Special 32 points]	0.40	–	
	A1SJ71QE71N3-T	10 BASE-T	32 [Special 32 points]	0.53	–	
Intelligent communication module	A1SD51S[*2]	Interpreter BASIC, Compiler BASIC. RS-232C 2 channel. RS-422/485 1 channel	32 [Special 32 points]	0.4	–	
Positioning module	A1SD70	1-axis positioning control, speed control and speed-positioning control, analog voltage output for speed-positioning control (0 to ± 10V)	48 (First half empty 16 points Second half special32 points)	0.3	–	
	A1SD75P1-S3	For positioning control, Pulse chain output, One axis	32 [Special 32 points]	0.7	–	
	A1SD75P2-S3	Used for positioning control Pulse output, 2 axes (independent, dual-axis simultaneous, linear interpolation, circular interpolation)	32 [Special 32 points]	0.7	–	
	A1SD75P3-S3	Used for positioning control Pulse output, 3 axes (independent, triple-axis simultaneous, dual-axis linear interpolation, dual-axis circular interpolation)	32 [Special 32 points]	0.7[*]	–	* When Differential driver is connected: 0.78
	A1SD75M1	Used for positioning control, Digital output. For MR-H-B/MR-J-B/MR-J2-B.	32 [Special 32 points]	0.7	–	
	A1SD75M2	Used for positioning control, Digital output. For MR-H-B/MR-J-B/MR-J2-B. (independent, dual-axis simultaneous, linear interpolation, circular interpolation)	32 [Special 32 points]	0.7	–	
	A1SD75M3	Used for positioning control, Digital output. For MR-H-B/MR-J-B/MR-J2-B. (independent, triple-axis simultaneous, dual-axis linear interpolation, dual-axis circular interpolation)	32 [Special 32 points]	0.7	–	
ID interface module	A1SD35ID1	ID interface module Connectable reader/writer unit: one	32 [Special 32 points]	0.25	0.17	
	A1SD35ID2	ID interface module Connectable reader/writer units: two	32 [Special 32 points]	0.25	0.33	
MELSECNET (II) data link module	A1SJ71AP21[*2]	For MELSECNET(II) data link system master station and local station	32 [Special 32 points]	0.33	–	Maximum 2 modules can be used for one CPU module. (Refer to Section 3.3.2)
	A1SJ71AP21-S3[*2]	For MELSECNET(II) data link system master station and local station (for G1-type optical fiber cable)	32 [Special 32 points]	0.33	–	
	A1SJ71AR21[*2]	For MELSECNET(II) data link system master station and local station (for coaxial cable)	32 [Special 32 points]	0.8	–	

Product Name	Model Name	Description	Number of Occupied Points (points) [I/O Assignment Module Type]	Current consumption		Remark
				5VDC (A)	24VDC (A)	
MELSECNET/B data link module	A1SJ71AT21B[*2]	For MELSECNET/B data link system master station and local station	32 [Special 32 points]	0.66	–	
	A1SJ72T25B[*1]	For MELSECNET/B data link system remote I/O station	–	0.3	–	
B/NET interface module	A1SJ71B62-S3	Master module for B/NET	32 [Special 32 points]	0.08	–	
MELSECNET/10 data link module	A1SJ71QLP21	For control station , master station , and local station of the MELSECNET/10 data link module system (For SI type optical fiber cable, double loop)	32 [Special 32 points]	0.40	–	Maximum 4 modules can be used for one CPU module. (Refer to Section 3.3.2)
	A1SJ71QLP21GE	For control station , master station, and local station of the MELSECNET/10 data link module system (For GI type optical fiber cable, double loop)	32 [Special 32 points]	0.47	–	
	A1SJ71QLP21S	For control station , master station , and local station of the MELSECNET/10 data link module system With external power supply function (For SI type optical fiber cable, double loop)	48 (First half empty 16 points Second half special32 points)	0.40	0.17	
	A1SJ71QBR11	For control station , master station, and local station of the MELSECNET/10 data link module system (For the single bus coaxial cable)	32 [Special 32 points]	0.80	–	
	A1SJ71QLR21	For control station , master station, and local station of the MELSECNET/10 data link module system (For the coaxial cable dual loop)	32 [Special 32 points]	1.14	–	
CC-Link system master module	A1SJ61QBT11	For the master and local stations of the CC-Link data link system (For the twisted pair shield cable only)	32 [Special 32 points]	0.40	–	
MELSECNET/MI NI-S3 master module	A1SJ71PT32-S3[*1]	For MELSECNET/MINI-S3 master station, 64 stations maximum, Controls remote I/O with a total of 512 I/O points, and remote terminal	I/O dedicated mode 32 (special 32 points)	0.35	–	
			Expanded mode 48 (special 48 points)			
MELSEC - I/O LINK master module	A1SJ51T64	For MELSECNET - I/O master station, 16 stations maximum, Controls I/O LINK with a total of 128 I/O points, and remote I/O module If only a few remote I/O units are used, perform I/O assignment with a peripheral device to decrease the number of occupied I/O points to 16, 32, or 48.	64 [Output 64 points]	0.115	0.09	
S-LINK interface module	A1SJ71SL92N	Master module for S-LINK I/O total 128 points	32 [Special 32 points]	0.20	–	
AS-I interface module	A1SJ71AS92	Master module for AS-I, total I/O: 496 points	32 [Special 32 points]	0.15	–	
Position detection module	A1S62LS	Absolute detection system	32 [Special 32 points]	0.55	–	
PLC easier monitoring module	A1SS91	PC easier monitoring module	32 [Output 32 points]	0.08	–	

Product Name	Model Name	Description	Number of Occupied Points (points) [I/O Assignment Module Type]		Current consumption		Remark
					5VDC (A)	24VDC (A)	
Memory card interface module	A1SD59J-S2	Memory card interface module	32	[Special 32 points]	0.05	–	Power consumption assumes connection of A1SD59J-MIF.
Simulation module	A6SIM-X64Y64	I/O simulation module for connection to the main base,Allows desk debugging without connecting I/O module to the base module. Use an expansion cable of the AnS series ←→ between the main base of the AnS series and the A6SIM-X64Y64.	64 64	(64 inputs) (64 outputs)	TYP. 0.3 (When all points "ON")	–	
PROFIBUS Interface Module	A1SJ71PB92D	PROFIBUS-DPmaster module	32	[Special 32 points]	0.56	–	
	A1SJ71PB96F	PROFIBUS-FMSinterface module	32	[Special 32 points]	0.56	–	
DeviceNet Interface Module	A1SJ71DN91	Device Net master module	32	[Special 32 points]	0.24	–	
MODBUS Interface Module	A1SJ71UC24-R2-S2	RS-232Ctype MODBUS interface module	32	[Special 32 points]	0.1	–	
	A1SJ71UC24-R4-S2	RS-422/485type MODBUS interface module	32	[Special 32 points]	0.1	–	

*1 Discontinued model

*2 This module can access devices within the device range of the AnACPU (cannot access file register). (Refer to Section 3.3.2.)

Product Name	Model Name	Description	Number of Occupied Points (points) [I/O Assignment Module Type]	Current consumption		Remark
				5VDC (A)	24VDC (A)	
Graphic operation terminal	A985GOT	Large-size graphic operation terminal 256 colors, TFT color, 800×600 dots, high intensity	32 [Special 32 points]*	0.22 *	–	*When bus connected
	A975GOT	Large-size graphic operation terminal 256 colors, TFT color, 640×480 dots, high intensity				
	A970GOT	Large-size graphic operation terminal 16 colors, TFT color, 640×480 dots, high intensity/ 16 colors, TFT color, 640×480 dots, wide viewing angle/ 8 colors, STN color, 640×480 dots/ 2 colors, STN monochrome, 640×480 dots				
	A960GOT	Large-size graphic operation terminal 2 colors, EL, 640×400 dots				
	A956GOT	Medium-size graphic operation terminal 8 colors, STN color, 320×240 dots/ STN monochrome, 320×240 dots/ 256 colors, STN color, 320×240 dots				
	A956WGOT	Medium-size graphic operation terminal 256 colors, TFT color, 480×234 dots				
	A953GOT	Medium-size graphic operation terminal 8 colors, STN color, 320×240 dots/ STN monochrome, 320×240 dots/ 256 colors, STN color, 320×240 dots	–	–	–	For RS-232C connected only
	A951GOT	Medium-size graphic operation terminal 8 colors, STN color, 320×240 dots/ STN monochrome, 320×240 dots/ 256 colors, STN color, 320×240 dots	32 [Special 32 points]*	0.22 *	–	*When bus connected
	A950GOT	Medium-size graphic operation terminal 8 colors, STN color, 320×240 dots/ STN monochrome, 320×240 dots/ 256 colors, STN color, 320×240 dots	–	–	–	For RS-422 connected only
	GT1565-VTBA	Large-size graphic operation terminal 8.4" 256/65536 colors, TFT color, 640×480 dots (When installing a multi color display board, 65536 colors can be displayed.)	32 [Special 32 points]*	0.12	–	*When bus connected
	GT1575-VTBA	Large-size graphic operation terminal 10.4" 256/65536 colors, TFT color, 640×480 dots (When installing a multi color display board, 65536 colors can be displayed.)				
Main Base Unit	A1S32B	2 I/O modules can be installed.	–	–	–	Extension connector attached to one on each side
	A1S33B	3 I/O modules can be installed.				
	A1S35B	5 I/O modules can be installed.				
	A1S38B	8 I/O modules can be installed.				

Product Name	Model Name	Description	Number of Occupied Points (points) [I/O Assignment Module Type]	Current consumption		Remark
				5VDC (A)	24VDC (A)	
Extension Base Unit	A1S52B	2 I/O modules can be installed.	–	–	–	Does not take power supply module. (Power supplied from the main base module).
	A1S52B-S1					
	A1S55B	5 I/O modules can be installed.				
	A1S55B-S1					
	A1S58B	8 I/O modules can be installed.	–	–	–	
	A1S58B-S1					
	A1S65B	5 I/O modules can be installed.	–	–	–	The power supply module is required.
	A1S65B-S1					
	A1S68B	8 I/O modules can be installed.				
	A1S68B-S1					
Extension Cables	A1SC01B	Flat cable, 55 mm (2.17 inch) long	–	–	–	For extension to right side
	A1SC03B	330mm (118.11in) long	–	–	–	Extension base module connecting cable
	A1SCO7B	Extension base module connecting cable				
	A1SC12B	1200 mm (47.24 inch) long				
	A1SC30B	1200 mm (47.24 inch) long				
	A1SC60B	6000 mm (236.22 inch) long				
	A1SC05NB	450 mm (17.72 inch) long	–	–	–	Cable for A□N and A□A extension bases
	A1SCO7NB	Extension base module connecting cable				
	A1SC30NB	1200 mm (47.24 inch) long				
	A1SC50NB	5000 mm (197.1 inch) long				

Product Name		Model Name	Description	Applicable Model
Memory cassette	E^2PROM	A2SNMCA-30KE	With 30k-step E^2PROM (direct connection)	Direct writing to and reading from a peripheral device is feasible.
Battery		A6BAT	IC-RAM memory backup	Installed in the Q2ASCPU, Q2ASHCPU, Q2ASCPU-S1, Q2ASHCPU-S1 main module
Connector/ terminal block conversion module		A6TBXY36	For sink type input module and sink type output module (standard type)	A1SX41(S1/S2), A1SX42(S1/S2), A1SY41, A1SY41P, A1SY42, A1SY82, A1SH42(S1)
		A6TBXY54	For sink type input module and sink type output module (2 wire type)	
		A6TBX70	For sink type input module (3 wire type)	A1SX41(S1/S2), A1SX42(S1/S2), A1SH42(S1)
		A6TBX36-E	For source-type input module (standard type)	A1SX71, A1SX82-S1, A1SX81(S2)
		A6TBY36-E	For source-type output module (standard type)	A1SY81, A1SY82
		A6TBX54-E	For source type input module (2 wire type)	A1SX71, A1SX82-S1, A1SX81(S2)
		A6TBY54-E	For source type input module (2 wire type)	A1SY81, A1SY82
		A6TBX70-E	For source type input module (3 wire type)	A1SX71, A1SX82-S1, A1SX81(S2)
Cable for connector/ terminal block conversion module		AC05TB	0.5 m (1.64 ft.) long, for sink module	A6TBXY36 A6TBXY54 A6TBX70
		AC10TB	1 m (3.28 ft.) long, for sink module	
		AC20TB	2 m (6.56 ft.) long, for sink module	
		AC30TB	3 m (9.84 ft.) long, for sink module	
		AC50TB	5 m (16.4 ft.) long, for sink module	
		AC80TB	5 m (16.4 ft.) long, for sink module	
		AC100TB	10 m (32.8 ft.) long, for sink module	
		AC05TB-E	0.5 m (1.64 ft.) long, for source module	A6TBX36-E A6TBY36-E A6TBX54-E A6TBY54-E A6TBX70-E
		AC10TB-E	1 m (3.28 ft.) long, for source module	
		AC20TB-E	2 m (6.56 ft.) long, for source module	
		AC30TB-E	3 m (9.84 ft.) long, for source module	
		AC50TB-E	5 m (16.4 ft.) long, for source module	
Relay terminal module		A6TE2-16SRN	For sink-type output module	A1SY41, A1SY41P, A1SY42, A1SH42(S1)
Cable for relay terminal module connection		AC06TE	0.6 m (1.97 ft.) long	A6TE2-16SRN
		AC10TE	0.6 m (1.97 ft.) long	
		AC30TE	3 m (9.84 ft.) long	
		AC50TE	5 m (16.4 ft.) long	
		AC100TE	10 m (32.8 ft.) long	
Terminal block cover for A1S I/O module and special module		A1STEC-S	Slim type terminal block cover for A1S I/O module and special module	All terminal block connector type modules

Product Name	Model Name	Description	Applicable Model
IDC terminal block adapter	A1S-TA32	IDC terminal block adapter for 32 points 0.5mm^2 (AWG20)	A1SX41(S1/S2), A1SX71, A1SY41, A1SY41P, A1SY71
	A1S-TA32-3	IDC terminal block adapter for 32 points 0.3mm^2 (AWG22)	
	A1S-TA32-7	IDC terminal block adapter for 32 points 0.75mm^2 (AWG18)	
Terminal block adapter	A1S-TB32	For 32 points, conversion into Europe type terminal block	A1SX41(S1/S2), A1SX71, A1SY41, A1SY41P, A1SY71
40-pin connector	A6C0N1	Soldering-type, straight out	Sink type (40p FCN)
	A6C0N2	Solderless-type, straight out	
	A6C0N3	Press-fit type, flat cable	
	A6CON4	Soldering-type, straight/diagonal out	
3-pin D-sub connector	A6C0N1E	Soldering-type, straight out	Source type (37p D-sub)
	A6C0N2E	Solderless-type, straight out	
	A6C0N3E	Press-fit type, flat cable	

REMARK

Toa Electric Industrial CO., LTD. provides I/O cables with connectors, which can connect to 40-pin connector (A1SX41,A1SX42,A1SY41,A1SY41P,A1SY42, etc.) or 37-pin D-sub connector (A1SX81,A1SY81) of I/O modules.

Contact:
TOA ELECTRIC INDUSTRIAL CO., LTD.

(2) Peripheral device

Product Name	Model Name	Remark
Programming unit	Q6PU	Connected to the CPU module by an RS-422 cable (AC30R4-PUS, AC20R4-A8PU); for program writing and reading. (5VDC 0.4A)
RS-422 cable	AC30R4-PUS	Cable for connection between CPU module and Q6PU 3 m (9.84 ft.) long
	AC20R4-A8PU	Cable for connection between CPU module and Q6PU 2m (78.74 in) long

3.3.2 Precautions when configuring the system

The following shows the hardware and software packages which can be used for Q2ASCPU.

(1) Hardware

(a) The number of modules that can be mounted is restricted depending on the module type.

Applicable Module	For Q2ASCPU only	For AnSCPU	Remark
I/O module	–	No limit	–
Special function module	No limit	No limit	–
Intelligent special function module	No limit	Total 6 modules	Including GOT-A900 Series (Only when the bus connection is used.), and GOT1000 Series (Only when the bus connection is used.)
Interrupt module	–	Only 1 module	–
Link module · Ethernet module	Ethernet module for network, total of 4 units	Total 2 for data link use	Total 4 for network, Ethernet and data link use

REMARK

The modules described above are categorized as follows.

1) I/O module:	Standard input modules and output modules
2) Special function module:	Special function modules that perform processing in accordance with FROM/TO instructions from the Q2ASCPU (for example: A1S64AD, A1S62DA, etc.)
3) Intelligent special function module:	Special function modules that can process not only by executing FROM/TO instruction of Q2ASCPU but also by accessing Q2ASCPU from special function module (Example: A1SJ71UC24-R2, A1SJ71QC24N, etc.)
4) Interrupt module:	Modules that issue interrupts to the Q2ASCPU (A1SI61)
5) Link module:	Special function modules for MELSECNET II, /B data links and MELSECNET/10 networks. (Example: A1SJ71AP21, A1SJ71QLP21, etc.)
6) Ethernet module:	Dedicated Ethernet interface modules for Q2ASCPU (A1SJ71QE71N-B2, A1SJ71QE71N-B5)

(b) The following shows special function modules that cannot be used with Q2ASCPU:

- AJ71C23 (Host controller high-speed link module)
- AD57-S2 (A6MD controller module)
- AJ71C24 (Computer link module): Manufactured through February 1987.
  (Products manufactured in March 1987 or later, and products marked "H" (corresponding to A3H) can be used.)
- AD51 (Intelligent communication module)
  : Manufactured thorugh March 1987.
  (Products manufactured in April 1987 or later, and products marked "H" (corresponding to A3H) can be used.)
- A7GT-BUS (Bus connection interface module for A77GOT and A870GOT): Manufactured through January 1996.
  (Products manufactured in February 1996 or later, and products marked "C" (corresponding to A3H) can be used.)
- AJ71LP21(G), AJ71BR11, AJ71LR21, A1SJ71LP21, A1SJ71BR11, A1SJ71LR21 (MELSECNET/10 network modules)

(c) When using a special function module with Q2ASCPU, the device range to be used is depending on models of special function modules.

Device	Access range			
	Device range equivalent to the A3HCPU^{*1}	Device range equivalent to the AnACPU^{*1}		
	AD51(S3), AJ71C24-S3, AJ71P41	A1SD51S, A1SJ71UC24-R2/PRF/R4, A1SJ71AP21(S3)^{*2}, A1SJ71AR21^{*2}, A1SJ71AT21B^{*2}, AD51H(S3), AD51FD-S3, AJ71C23-S3, AJ71C24-S6/S8, AJ71UC24, AJ71AP21(S3)^{*2}, AJ71AR21^{*2}, AJ71AT21B^{*2}, AJ71ME81	A1SJ71E71N(3)-T, A1SJ71E71N3-B5T/B2/B5, AJ71E71(S3), AJ71E71N(3)-T, AJ71E71N-B5T/B2/B5	
			Q2AS(H)	Q2AS(H)-S1
I/O device (X/Y)	X/Y0 to X/Y7FF	X/Y0 to X/Y7FF	X/Y0 to X/Y1FF	X/Y0 to X/Y3FF
Internal relays (M, L, S) ^{*3}	M0 to M2047	M0 to M8191	M0 to M8191	
Link relay (B)	B0 to B3FF	B0 to BFFF	B0 to BFFF	
Timer (T)	T0 to T255	T0 to T2047	T0 to T2047	
Counter (C)	C0 to C255	C0 to C1023	C0 to C1023	
Data register (D)	D0 to D1023	D0 to D6143	D0 to D6143	
Link register (W)	W0 to W3FF	W0 to WFFF	W0 to WFFF	
Annunciator (F)	F0 to F255	F0 to F2047	F0 to F2047	

*1 Reading/Writing of file registers, programs, etc. are not possible.

*2 Only I/O devices (X/Y), link relay (B), and link register (W) are available.

*3 Even when L or S is specified, the device becomes M.
(Example: Even when L10 is specified, the device becomes M10.)

(d) When a QnACPU is mounted on a main base unit for A1S38HB/A1S38HBEU high-speed access, the Q2ASCPU can access special function modules, intelligent special function modules and link modules to write/read at greater speeds.
QnACPU cannot input/output to the I/O module at greater spped.

(e) The following shows how to connect graphic operation terminal units to a Q2ASCPU.

Model	Connection Method	Accessible Device Range
GOT1000 series	Direct connection to CPU Computer link connection CC-Link connection MELSECNET/10 connection Bus connection	Access is available for all device ranges of Q2ASCPU. (Refer to the GT Works2/GT Designer2 Reference Manual for details.)
GOT-A900 series	Direct connection to CPU Computer link connection CC-Link connection MELSECNET(II), /B, /10 connection Bus connection	Access is available for all device ranges of Q2ASCPU. (Refer to the GT Works2/GT Designer2 Reference Manual for details.)

(f) The accessible range for an A1SJ71UC24 computer link module comprises the CPU to which the A1SJ71UC24 is mounted (the host station) and the other stations in the network to which the host station is connected.
It is not possible to access other stations in other networks by using the MELSECNET/10 network system routing function.

The access range for an A1SJ71QC24N serial communication module is the host station, other stations in the network connected to the host station, and other stations in other networks accessed through up to 7 relay stations by using the routing function.

(g) When accessing from intelligent communication module A1SD51S to other station Q2ASCPU/QnACPU on the network, only Format 1 control table can be used.Format 2 control table cannot be used.
Access to other network is not allowed on Format 1 control table.

(h) In Q2ASCPU, I/O assignment set with the parameter cannot be valid for MELSECNET (II) and MELSECNET/B.When setting the I/O assignment for a remote I/O station, build the remote I/O network with MELSECNET/10.

(2) Software package
The following shows the system start-up software packages to create programs for Q2ASCPU.

Peripheral Device Capable of GPP Functions	Software Package for System Start-up
Personal computer	GX Developer, SW□IVD-GPPQ

Set to the following PC CPU type using peripheral device.

PC CPU model	PC CPU model
Q2AS(H)CPU	Q2A
Q2AS(H)CPU-S1	Q2AS1

Apart from the above, the following software packages can be used.

• CAD interface package	SW□IVD-CADQ
• Data conversion package	SW□IVD-CNVQ
• Macro/library package	SW□IVD-MSDQ SW□IVD-MSPQ
• Ladder sequence linking package	SW□IVD-LNKQ

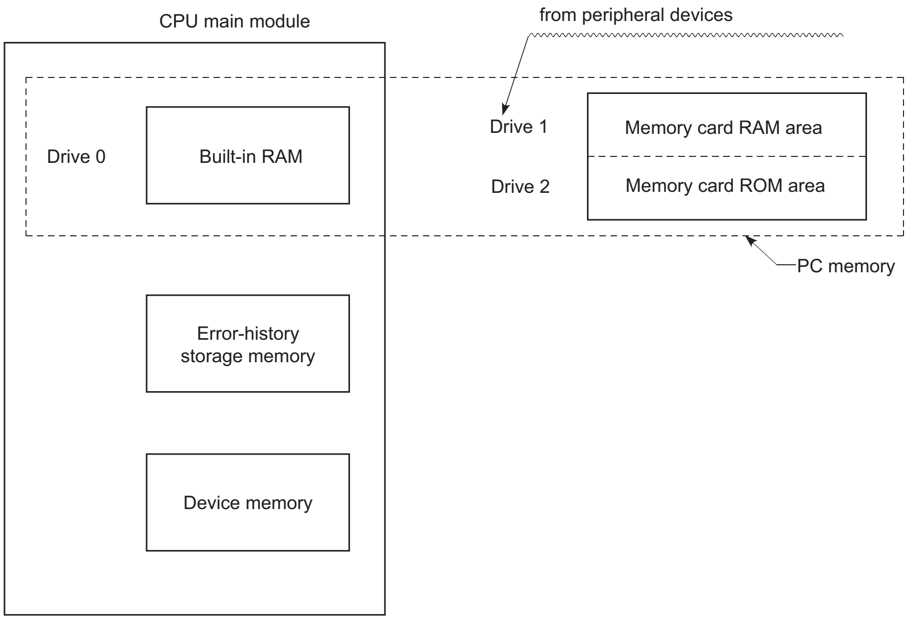
REMARK

The following shows the peripheral devices and software packages that cannot be used with Q2ASCPU:

- A□PUProgramming unit)
- A6WU (ROM writer unit)
- A6DU-B (Data access unit)
- A6TEL (Modem interface unit)
- A6GPP (Intelligent GPP)
- A6HGP (Hand-held graphic programmer)
- A6PHP (Plasma hand-held graphic programmer)
- System start-up software package for ACPU
  SW□□□-GPPA, SW□□□-SAP2
- Utility software package for ACPU
  SW□□□-GPPATEL, SW□□□-CADIF, SW□□□-DRWA,
  SW□□□-FUNP, SW□□□-TSAP2

3.3.3 Q2ASCPU memory block diagram

The following block diagram shows the Q2ASCPU memory configuration.



- Built-in RAM : Memory that stores parameters, sequence programs, etc.
- Error history storage memory : Memory that stores error history data
- Device memory : Memory that stores device data
- Memory card (RAM, ROM area) : Memory that stores the files, comments, etc., for parameters, sequence programs, sampling traces, etc.
- PLC memory : Indicates all the memories of drives 0 through 2.

For file types stored in each memory, refer to "FireTypes & Storage Destinations of Files Managed by QnACPU" in the QnACPU Programming Manual (Fundamentals).

4 PERFORMANCE SPECIFICATIONS

This section shows the performance specifications of the Q2ASCPU.

Item		Model Name				Remark
		Q2ASCPU	Q2ASCPU-S1	Q2ASHCPU	Q2ASHCPU-S1	
Control method		Sequence program control method				
I/O control mode		Refresh mode				Direct input/output is allowed by specifying direct input/output (DX□, DY□).
Programming language		Language dedicated to sequence control				
		Relay symbol language, logic symbolic language, MELSAP-3 (SFC)				
Processing speed (Sequence instruction)	LD	0.2μs/step		0.075μs/step		
	MOV	0.6μs/step		0.225μs/step		
Constant Scan (Function that makes the scan time constant)		5ms to 2000ms (Possible to set in 5ms units)				Possible to set in the parameters
Memory capacity		Capacity of the installed memory card (Max. 2036k bytes)				
Program capacity	Number of steps	Maximum 28k step	Maximum 60k step	Maximum 28k step	Maximum 60k step	
	Number of files	28	60	28	60	
Number of I/O device points		8192 points X/Y to 1FFF				The number of points usable in the program
Number of I/O points		512 points X/Y0 to 1FF	1024 points X/Y0 to 3FF	512 points X/Y0 to 7FF	1024 points X/Y0 to FFF	The number of accessible points to actual I/O module

Item		Model Name				Remark
		Q2ASCPU	Q2ASCPU-S1	Q2ASHCPU	Q2ASHCPU-S1	
Device points	Internal relay [M]	Default 8192 points M0 to 8191				Possible to set the number of points to be used by the parameter
	Latch relay [L]	Default 8192 points L0 to 8191				
	Link relay [B]	Default 8192 points B0 to 1FFF				
	Timer [T]	Default: 2048 points (T0 to T2047) (Low-speed timers and high-speed timers sharing) Set low-speed timers/high-speed timers switching with instructions. Set low-speed/high-speed measurement units by parameter. (Low-speed timers: 10ms to 1000ms, 10ms units, Default: 100ms) (High-speed timers: 1ms to 100 ms, 1ms units, default: 10 ms)				
	Retentive timer [ST]	Default 0 points ST0 to 2047 (Low-speed timers and high-speed timers sharing) Set Low-speed timers/high-speed timers switching with instructions. Set low-speed/high-speed measurement units by parameter. (Low-speed timers: 10ms to 1000ms, 10ms units, Default: 100ms) (High-speed timers: 1ms to 100 ms, 1ms units, default: 10 ms)				
	Counter [C]	• Normal counter Default 1024 points C0 to 1023 • Interrupt counters Max. 48 points (Default: 0 point, Can be set by the parameter.)				
	Data register [D]	Default 12288 points D0 to 12287				
	Link register [W]	Default 8192 points W0 to 1FFF				
	Annunciator [F]	Default 2048 points F0 to 2047				
	Edge relay [V]	Default 2048 points V0 to 2047				
	File register [R]	32768 points R0 to 32767 Up to 1042432 points can be used by block switching.				Possible to set the number of points to be used by the parameter
		1042432 points ZR0 to 1042431 Block switching is not necessary.				
	Special link relay [SB]	Default 2048 points SB0 to 7FF				The number of device points is fixed.
	Special link register [SW]	Default 2048 points SW0 to 7FF				
	Step relay [S]	8192 points S0 to 8191				
	Index register [Z]	16 points (Z0 to 15)				
	Pointer [P]	4096 points P0 to 4095 Possible to set Ranges for pointers in files and common pointers by the parameter.				
	Interrupt pointer [I]	48 points I0 to 47 The fixed-cycle interval for system interrupt pointers I28 to I31 is set by the parameter. (5ms to 1000ms, in 5ms units)				
	Special relay [SM]	2048 points SM0 to 2047				
	Special register [SD]	2048 points SD0 to 2047)				
	Function input [FX]	16 points (FX0 to F)				
	Function output [FY]	16 points (FY0 to F)				
	Function register [FD]	5 points (FD0 to 4)				
Link direct device		Devices that access link devices directly. Dedicated to MELSECNET/10Designation format: J□ □\□ □				

Item	Model Name				Remark
	Q2ASCPU	Q2ASCPU-S1	Q2ASHCPU	Q2ASHCPU-S1	
Special function module direct device	Devices that directly access the buffer memories of special function modules. Designation format: U□ □\G□ □				
Latch (power failure compensation) range	L0 to L8191 (Default) (Latch ranges can be set for B, F, V, T, ST, C, D, W devices.)				Possible to set in the parameters
Remote RUN/PAUSE contact	Possible to setup one contact poin for each of RUN/PAUSE from X0 to X1FFF.				
Clock Function	Year, month, day, hour, minute, second, day of the week (automatic detection of the leap year) Accuracy -1.7 to +4.9s (TYP. +1.7s)/d at 0℃ Accuracy -1.0 to +5.2s (TYP. +2.2s)/d at 25℃ Accuracy -7.3 to +2.5s (TYP. -1.9s)/d at 55℃				
Allowable momentary power failure period	Depends on the power supply modules				See Section Section 16.1.
5VDC internal current consumption*	0.3A	0.3A	0.7A	0.7A	
Weight	0.5kg	0.5kg	0.5kg	0.5kg	
External dimensions	130 × 54.5 × 110 (5.12 × 2.15 × 4.33)				

REMARK

* Indicates current consumption of the Q2ASCPU with function version "B" (9707B).The following shows the current consumption values of theQ2ASCPU without the function version:

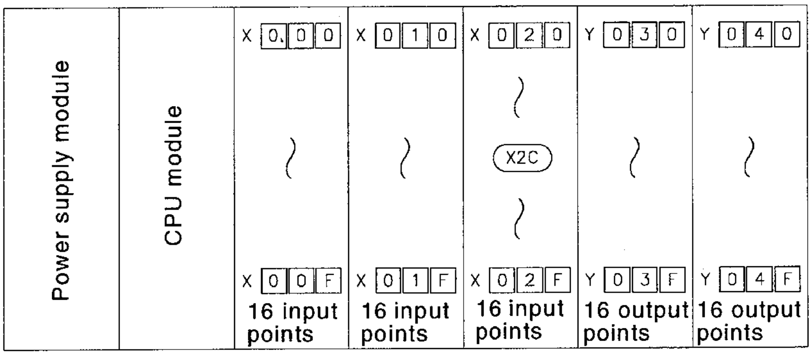
- Q2ASCPU, Q2ASCPU-S1 : 0.3A
- Q2ASHCPU, Q2ASHCPU-S1 : 0.7A

5 I/O NUMBER ASSIGNMENT

This section explains the method for I/O number assignment using the Q2ASCPU to enable data communications with a I/O modules and a special function module.

5.1 I/O Numbers

The I/O number is used in the sequence program to input data from a input module and to output data to an output module.
The I/O number is expressed as three-digit hexadecimal numbers.
The I/O numbers when all the I/O modules are occupied in 16 points are indicated below.



Concept of I/O numbers

REMARK

When programming with a peripheral device for GPP function, I/O numbers can be input in 2 digits.

I/O numbers		Input with a peripheral device
X010	→	X10
Y020	→	Y20

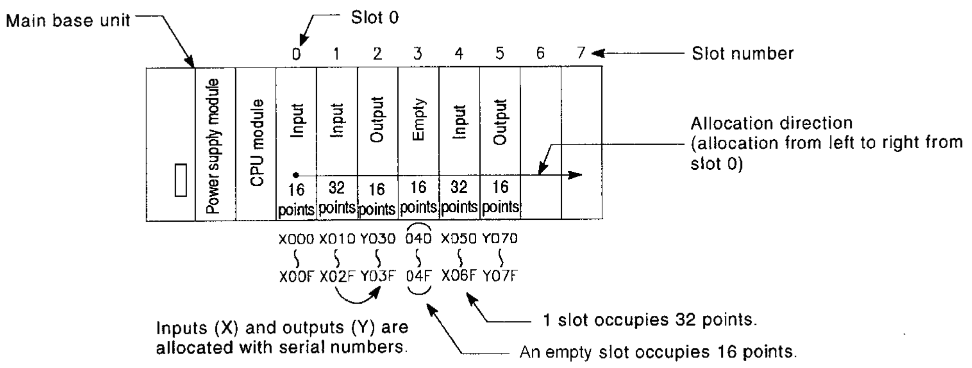
5.2 I/O Number Assignment Concept

When the programmable controller power is ON or the CPU module is reset, the I/O assignment described below is performed.
In the sequence program, designate the I/O numbers assigned in accordance with the following.

(1) I/O numbers are sequentially assigned from left to right, taking slot 0 (The slot to the right of the CPU module) of the main base unit to be "0".

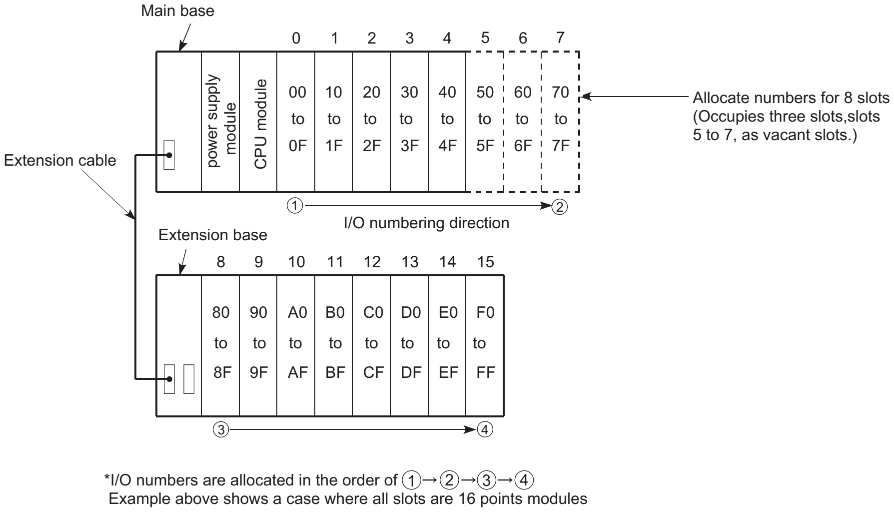
(2) The I/O modules and special function modules mounted to the main base unit occupy the I/O numbers corresponding to the number of I/O points for each module.

(3) 16 points are assigned to the empty slots where no I/O module or special function module is mounted.



(4) If an extension base unit is connected, its assignment starts from the number immediately after the number assigned to a main base unit.

(5) I/O numbers are assigned assuming that every base unit has 8 slots.
If a 5-slot type base unit is used, an I/O number obtained by adding points equivalent to 3 slots (48 points) to the final I/O number of the 5-slot base unit is assigned to the next extension base unit.



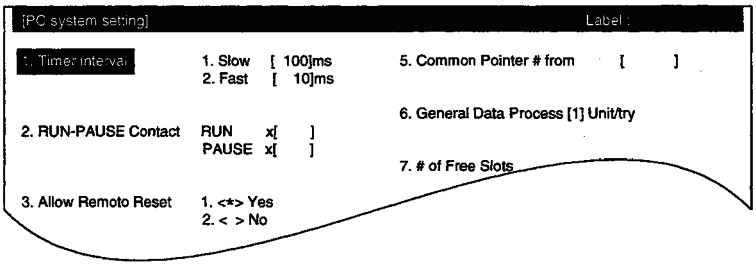
*I/O numbers are allocated in the order of ①→②→③→④
Example above shows a case where all slots are 16 points modules

5.3 I/O Assignment with GPP Function

When using the Q2ASCPU, I/O modules and a special function module can be controlled even if I/O assignment with GPP function is not performed.
I/O assignment with GPP function are valid in the following cases.

(1) The purpose of I/O assignment with GPP function
   (a) When using a base unit for 5 slots, set 0 point for 3 slots for efficient use of number of I/O points.
   (b) Reserve the points when changing a module to other than a 16-point module for future system extension.
   (c) The I/O assignment prevents the I/O numbers from changing if an I/O module or special function module that occupies other than 16 points has to be removed due to failure.
   (d) The I/O assignment reduces the I/O number modification in a program since it enables to match with the I/O numbers of the designed program and to change the I/O numbers assigned to each module on the base unit per slot.

(2) The concept of I/O assignment with GPP function
   The following two methods are available for I/O assignment with GPP function.
      1) Set the number of points for the empty slots on a main base unit and extension base unit.(Points occupied by empty slot)
      2) Set the I/O assignment per slot of main base unit or extenstion base unit to each module type.(I/O assignment)
   Parameter settings are used for both of these methods.If both 1) and 2) are set, the setting of 2) takes priority.
   (a) Setting points occupied by empty slot
      Set the number of points for all slots that are empty on the base unit.
      In the systems in which this setting is not made in the parameters, 16 points are set for empty slots.
      Make this setting in "8. Number of empty slots" on the "PLC system" screen in the parameter mode.



[PC system setting] Label :

1. Timer interval 1. Slow [100]ms
2. Fast [10]ms

2. RUN-PAUSE Contact RUN x[ ]
PAUSE x[ ]

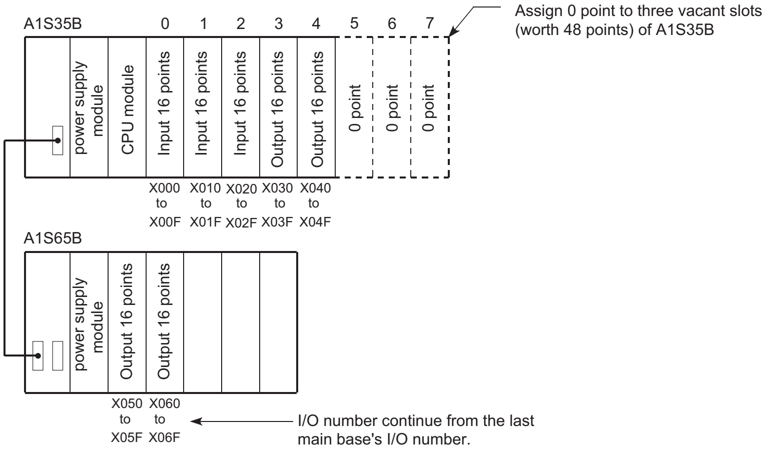
3. Allow Remoto Reset 1. <*> Yes
2. < > No

5. Common Pointer # from [ ]

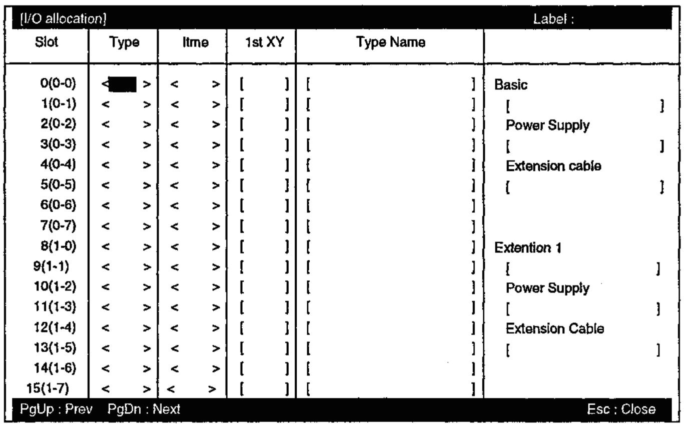
6. General Data Process [1] Unit/try

7. # of Free Slots

The setting is made in units of 16 point within the range of 0 to 64.The default is 16 points.
Example: When the points occupied by empty slot is set to 0 points



(b) I/O assignment settings
Set the I/O assignment per slot of main base unit or extenstion base unit to each module type.
Make this setting in the "I/O Assign" screen in the parameter mode.



```
[I/O allocation]                                            Label :
 Slot      Type     Itme    1st XY      Type Name
 0(0-0)   <    >   <    >  [    ]  [               ]  Basic
 1(0-1)   <    >   <    >  [    ]  [               ]   [              ]
 2(0-2)   <    >   <    >  [    ]  [               ]   Power Supply
 3(0-3)   <    >   <    >  [    ]  [               ]   [              ]
 4(0-4)   <    >   <    >  [    ]  [               ]   Extension cable
 5(0-5)   <    >   <    >  [    ]  [               ]   [              ]
 6(0-6)   <    >   <    >  [    ]  [               ]
 7(0-7)   <    >   <    >  [    ]  [               ]
 8(1-0)   <    >   <    >  [    ]  [               ]  Extention 1
 9(1-1)   <    >   <    >  [    ]  [               ]   [              ]
10(1-2)   <    >   <    >  [    ]  [               ]   Power Supply
11(1-3)   <    >   <    >  [    ]  [               ]   [              ]
12(1-4)   <    >   <    >  [    ]  [               ]   Extension Cable
13(1-5)   <    >   <    >  [    ]  [               ]   [              ]
14(1-6)   <    >   <    >  [    ]  [               ]
15(1-7)   <    >   <    >  [    ]  [               ]
PgUp : Prev   PgDn : Next                                   Esc : Close
```

The setting details are as follows:

Item		Setting	Setting range	Default value
Slot setting		Set data for each slot. (Not necessary to set all data).	Empty/input/output/special	No setting
	Classification	Set the module type.		
	Number of points	Set the number of points for the module.	0 to 64 points (in 16 point units)	
	Start XY	Set the start number of XY devices of the module.	0 to 1FFF (in 16 point units)	
	Model Name	Set the model name of the module.	Up to 16 characters	
Base specification		Set data for each base unit.	Up to 16 characters	No setting
	POWER SUPPLY MODULE	Set the model name of the power supply module.		
	Extension cable	Set the model name of the extension cable.	Up to 16 characters	

The items without settings are handled as follows:

- Type and Points : In accordance with the loaded module.
- Start XY : The number following the total points obtained by adding the number of points of the modules already set.
  If there is any duplication, an error (SP.UNIT LAY ERROR) is detected.

POINT

The power supply module names set in the base specification is only used for the current capacity check in the PLC diagnostics mode and not used for a CPU module. Therefore, even if they are not set, any problem does not occur.

The CPU module performs the following processing when I/O assignment is set.

1) Any of the following assignment can be performed per slot of each base unit.

Assigned number of points			
Empty slot	Input module	Output module	Special function module
0	-	-	-
16	16	16	16
32	32	32	32
48	48	48	48
64	64	64	64

2) The slots for which I/O assignment has been performed with GPP function, the I/O assignment setting takes priority regardless of the loaded module.
   - If a number of points fewer than the that of the loaded I/O module is set, the actual number of points of the loaded I/O module is reduced.
     For example, if the loaded module is a 32-point input module but I/O assignment is set for a 16-point input module using GPP function, the latter 16 points for the input module cannot be used.
   - If a number of points is greater than the that of the loaded I/O module is set, the number of points in excess of the actual number of points is occupied with dummy points.
   - If the slot where an I/O module is loaded is set as a empty slot, the I/O module will be unusable.

3) The slots for which I/Oassignment is not performed using GPP function are assigned with the number of points of the loaded module.

4) The slots for which I/O assignment is not performed using GPP function are assigned I/O numbers that are consecutive to those of modules for which I/O assignment has been performed.

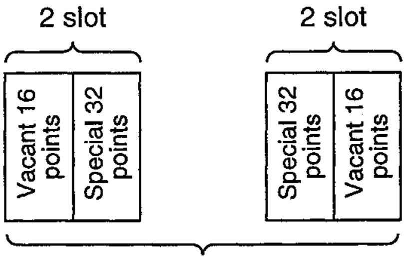
(3) Precautions

(a) If there is a disparity between the I/O assignment made in the parameter settings and the actually loaded I/O modules, the input and output is not normally performed.

Loaded module	I/O assignment	Result
Input	Output	No input
Output	Input	No output
Input/Output	Special	CPU module error
Special	Input/Output	CPU module error

(b) The I/O assignment of a slot to which a special function module is loaded has to be the same setting with the module.Not doing so may cause an error.

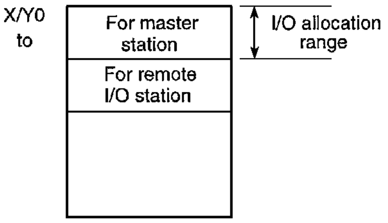
1)	A11VC......................................	Special: 16 points
2)	AI61	Special: 32 points
3)	AG62...	Input: Set number of points
4)	Modules that occupy 2 slots......	Set "Empty, 16 points" and "Special, 32 points".



Refer to users' manual for special module being used.

(c) When operating MELSECNET data link, perform I/O assignment as follows.

1) As for a master station, I/O assignment has to be performed for the master station and all remote I/O stations.
   I/O assignment of MELSECNET (II)/B to the remote I/O station is invalid.



2) As for a local station, perform I/O assignment only for the local station.
3) Assign the I/O for the I/O hybrid module (e.g. A42XY) as an output module.

(d) When the MELSECNET/10 network is established, assign the I/O only for the host station (master station).
Since the I/O assignment of MELSECNET/10 to the remote I/O station is irrelevant, the I/O assignment is not allowed.
For I/O assignment of MELSECNET/10 to the remote I/O station, use the I/O assignment settings in the "Network param".

REMARK

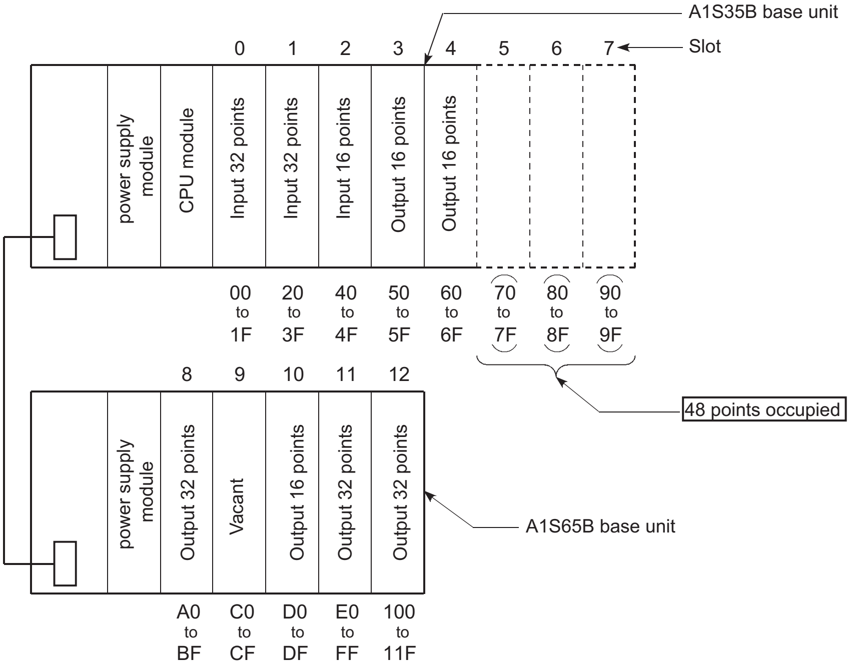
As for the remote I/O station of MELSECNET (II)/B, I/O assignment settings in the "Network param" is irrelevant, therefore, the I/O assignment is not allowed.

5.4 Example of I/O Number Assignment

The following shows the example of I/O number assignment when I/O assignment is performed using GPP function.

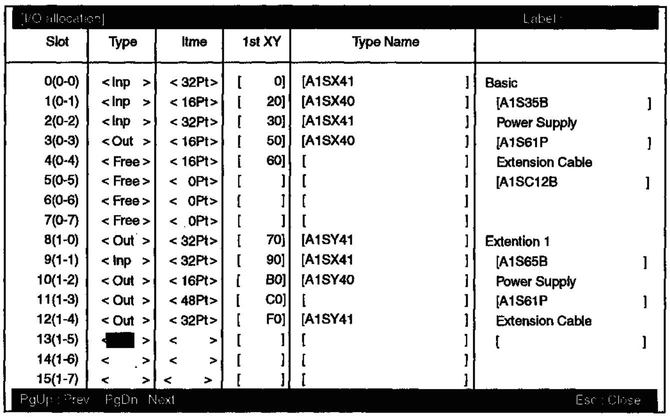
(1) When changing the assignment for an empty slot from 16 points to 0 or 32 points
When the A1S35B is used, there are three empty slots. When setting the assignment for these to 0 points in order to increase the number of I/O points that can be used by the CPU module
When reserving 32 I/O points for a current empty slot to which a 32-point input module is loaded later, in order to prevent the I/O number assignment change
To achieve these operations, perform I/O assignment as follows.

(a) Loading status and I/O numbers



(b) I/O numbers when I/O assignment is performed using GPP function

1) I/O assignment example



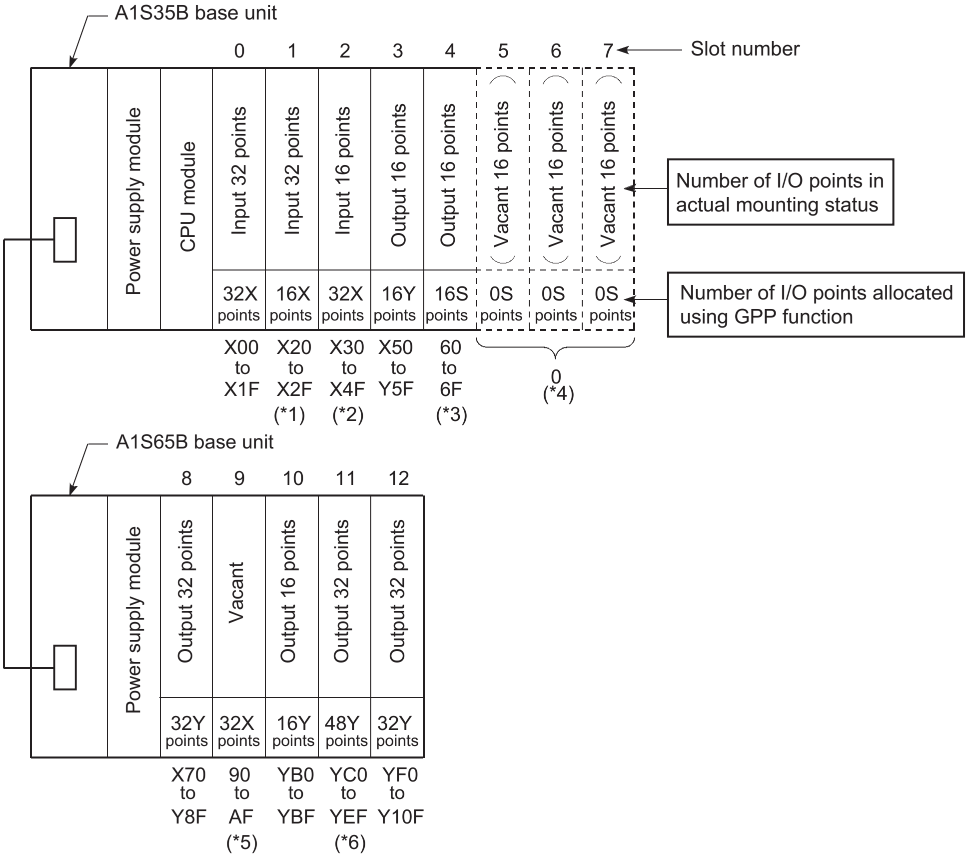
[I/O allocation] Label :

Slot	Type	Itme	1st XY	Type Name	
0(0-0)	<Inp >	< 32Pt>	[0]	[A1SX41]	Basic
1(0-1)	<Inp >	< 16Pt>	[20]	[A1SX40]	[A1S35B]
2(0-2)	<Inp >	< 32Pt>	[30]	[A1SX41]	Power Supply
3(0-3)	<Out >	< 16Pt>	[50]	[A1SX40]	[A1S61P]
4(0-4)	< Free >	< 16Pt>	[60]	[ ]	Extension Cable
5(0-5)	< Free >	< 0Pt>	[ ]	[ ]	[A1SC12B]
6(0-6)	< Free >	< 0Pt>	[ ]	[ ]	
7(0-7)	< Free >	< 0Pt>	[ ]	[ ]	
8(1-0)	< Out >	< 32Pt>	[70]	[A1SY41]	Extention 1
9(1-1)	< Inp >	< 32Pt>	[90]	[A1SX41]	[A1S65B]
10(1-2)	< Out >	< 16Pt>	[B0]	[A1SY40]	Power Supply
11(1-3)	< Out >	< 48Pt>	[C0]	[ ]	[A1S61P]
12(1-4)	< Out >	< 32Pt>	[F0]	[A1SY41]	Extension Cable
13(1-5)	<■ >	< >	[ ]	[ ]	[ ]
14(1-6)	< >	< >	[ ]	[ ]	
15(1-7)	< >	< >	[ ]	[ ]	

PgUp : Prev PgDn Next Esc : Close

The example of I/O assignment with GPP function

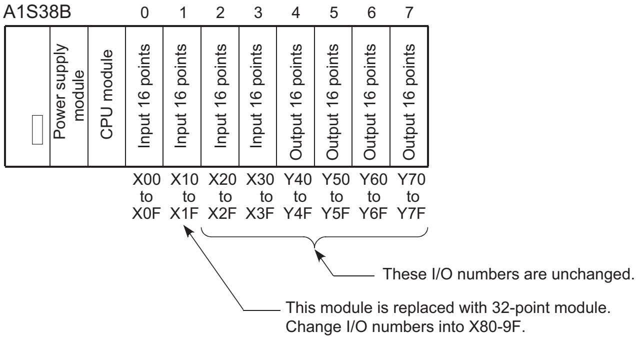
2) I/O numbers after performing I/O assignment using GPP functon



*1 Since 16 points is set, the latter 16 points of inputs cannot be used.
*2 Since 32 points is set, the points from 40 to 4F is occupied with dummy points.
*3 Since "Empty (S), 16 points" is set, the points cannot be used for outputs.
*4 Since "Empty (S), 0 points" is set, the number of I/O points for the three slots are not lost.
*5 Since "input (S), 32 points" is set, there are 32 input points.
*6 Since 48 points are set, E0 to EF is occupied with dummy points.

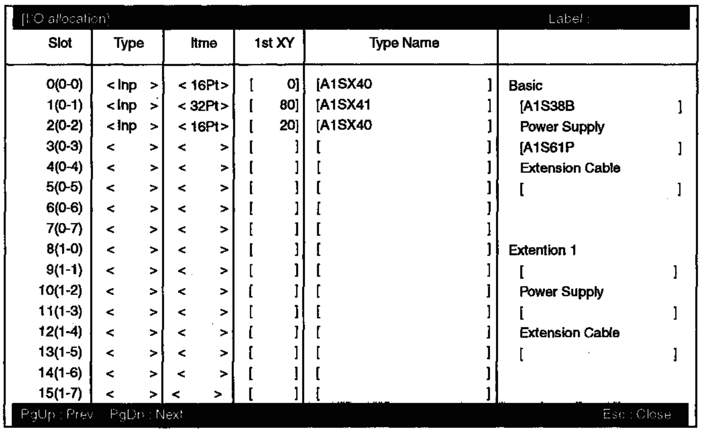
(2) Replacing a 16-point input module with a 32-point input module
When replacing the 16-point input module with a 32-point input module without changing the all I/O number assignment in a system to which a 16-point input module is designedTo achieve this operation, perform I/O assignment as follows.

(a) Loading status and I/O numbers before the replacement



(b) I/O numbers when I/O assignment is performed using GPP function

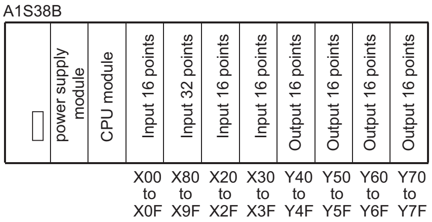
1) I/O assignment example



Slot	Type	Itme	1st XY	Type Name	
0(0-0)	<Inp >	< 16Pt>	[0]	[A1SX40]	Basic
1(0-1)	<Inp >	< 32Pt>	[80]	[A1SX41]	[A1S38B]
2(0-2)	<Inp >	< 16Pt>	[20]	[A1SX40]	Power Supply
3(0-3)	< >	< >	[ ]	[ ]	[A1S61P]
4(0-4)	< >	< >	[ ]	[ ]	Extension Cable
5(0-5)	< >	< >	[ ]	[ ]	[ ]
6(0-6)	< >	< >	[ ]	[ ]	
7(0-7)	< >	< >	[ ]	[ ]	
8(1-0)	< >	< >	[ ]	[ ]	Extention 1
9(1-1)	< >	< >	[ ]	[ ]	[ ]
10(1-2)	< >	< >	[ ]	[ ]	Power Supply
11(1-3)	< >	< >	[ ]	[ ]	[ ]
12(1-4)	< >	< >	[ ]	[ ]	Extension Cable
13(1-5)	< >	< >	[ ]	[ ]	[ ]
14(1-6)	< >	< >	[ ]	[ ]	
15(1-7)	< >	< >	[ ]	[ ]	

The example of I/O assignment with GPP function

2) I/O numbers after performing I/O assignment using GPP function and replacing the module



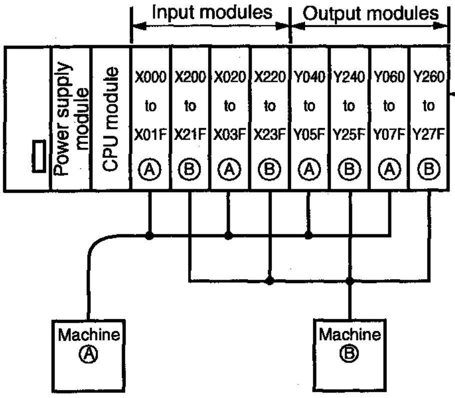
POINT

When the I/O number set for "Start XY" in the "I/OAssign" is changed, also set the "Start XY" for the next module to avoid changing the I/O numbers of the module for which the change was made and the subsequent modules.
In the example above, since "20" is set for the "Start XY" for the second slot, consecutive I/O numbers starting from X30 are set for slot 3 and later.

(3) When combining an input module and output module having non-consecutive I/O numbers on a base unit

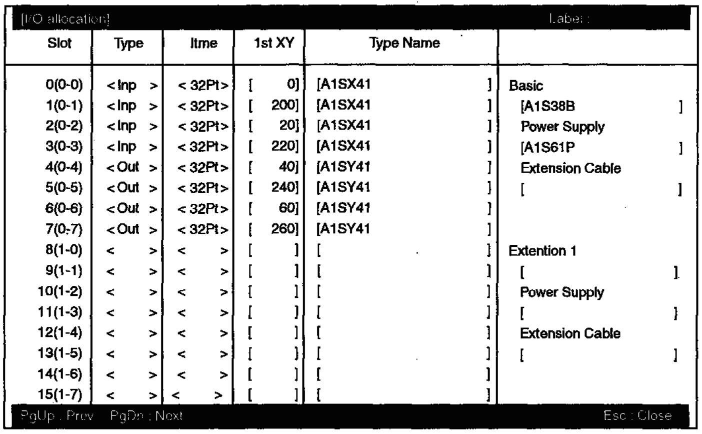
When controlling the machine Ⓐ (I/O numbers X0 to X3F, Y40 to Y7F) and machineⒷ (I/O numbers X200 to X23F and Y240 to X27F) with a single programmable controller, it is desired to combine input modules and output modules on the base unit.To achieve this operation, perform I/O assignment as follows.

(a) Loading status and I/O numbers to be set



Mount input modules of machines Ⓐand Ⓑalternately to mount them together.
This also applies to output modules.

(b) The example of I/O assignment with GPP function



[I/O allocation] Label :

Slot	Type	Itme	1st XY	Type Name	
0(0-0)	<Inp >	< 32Pt>	[0]	[A1SX41]	Basic
1(0-1)	<Inp >	< 32Pt>	[200]	[A1SX41]	[A1S38B]
2(0-2)	<Inp >	< 32Pt>	[20]	[A1SX41]	Power Supply
3(0-3)	<Inp >	< 32Pt>	[220]	[A1SX41]	[A1S61P]
4(0-4)	<Out >	< 32Pt>	[40]	[A1SY41]	Extension Cable
5(0-5)	<Out >	< 32Pt>	[240]	[A1SY41]	[ ]
6(0-6)	<Out >	< 32Pt>	[60]	[A1SY41]	
7(0-7)	<Out >	< 32Pt>	[260]	[A1SY41]	
8(1-0)	< >	< >	[ ]	[ ]	Extention 1
9(1-1)	< >	< >	[ ]	[ ]	[ ]
10(1-2)	< >	< >	[ ]	[ ]	Power Supply
11(1-3)	< >	< >	[ ]	[ ]	[ ]
12(1-4)	< >	< >	[ ]	[ ]	Extension Cable
13(1-5)	< >	< >	[ ]	[ ]	[ ]
14(1-6)	< >	< >	[ ]	[ ]	
15(1-7)	< >	< >	[ ]	[ ]	

PgUp : Prev PgDn : Next Esc : Close

6 DATA COMMUNICATIONS WITH SPECIAL FUNCTION MODULES

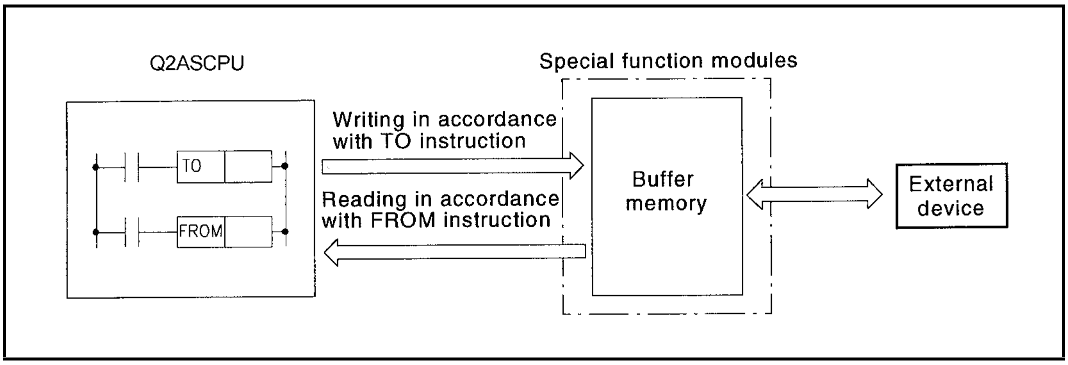
This chapter explains the methods for reading data from a special function module, and writing data to a special function module with the Q2ASCPU.
The special function module is a module that allows analog quantity, high-speed pulse, etc., which cannot be processed with I/O module alone, to be handled by the Q2ASCPU.
For example, analog quantity is converted to a digital value by an analog/digital converter module (which is a special function module) so that they can be used by the Q2ASCPU.
The special function module has buffer memory in which data input from external sources and data to be output to external destinations are stored.
The folloiwng two methods are available for reading data from a special function module, and writing data to a special function module with the Q2ASCPU.

1) Using the FROM/TO instruction
2) Using special direct devices

These methods are explained in the following sections.

6.1 Reading/Writing Data from/to the Q2ASCPU Using the FROM/TO Instruction

When the FROM/TO instruction is performed, data stored in the buffer memory of a special function module is read, or data is written to the buffer memory of a special function module.



Data communications with a special function module

When the FROM instruction is performed, the data read from the buffer memory is stored in the specified device. When the TO instruction is performed, the data in the specified device is written to the buffer memory.

REMARK

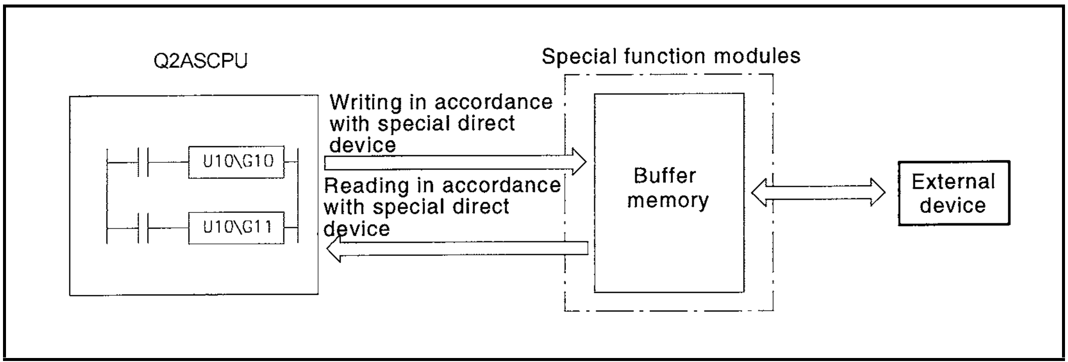
1) For details on the FROM/TO instructions, refer to the QCPU (Q mode)/ QnACPU Programming Manual (Common Instructions).
2) For details on the buffer memory of a special function module, refer to the manual of the special function module in use.

POINT

When executing the FROM/TO instruction for the special function module frequently in short scan time, it may cause the target special function module operation error.
When executing the FROM/TO instruction, match the processing time and conversion time of the special function module using timer or constant scanning.

6.2 Reading/Writing Data from/to the Q2ASCPU Using Special Direct Devices

As the FROM/TO instruction, the special direct device reads data stored in the buffer memory of a special function module or writes data to the buffer memory of a special function module.



The special direct device represents the buffer memory in a special function module as the Q2ASCPU device.

Example: U10\G10: U10→ Indicates the head I/O No.100 of the special function module. (Hexadecimal)
G10→ Indicates the buffer memory address 10. (Decimal)

REMARK

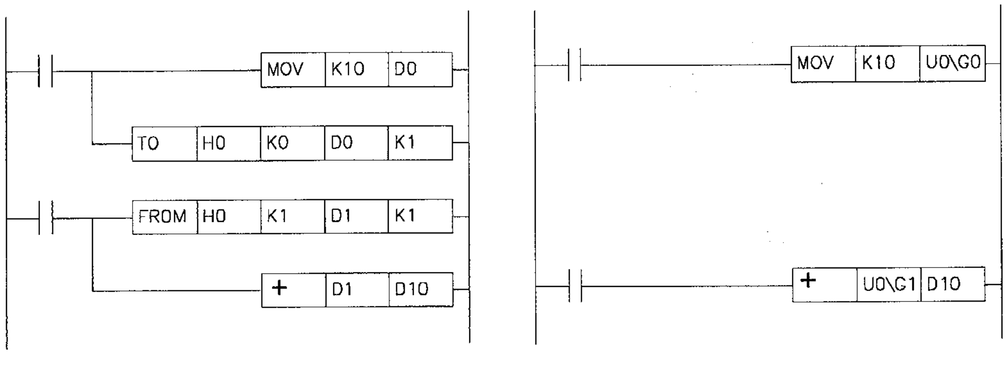
For details on a special direct device, refer to the QnACPU Programming Manual (Fundamentals).

The special direct device differs from the FROM/TO instruction in that the CPU module can handle the buffer memory of a special function module as a direct device.
This can reduce the total number of steps in the program. However, the instruction processing speed is the same with the FROM/TO instruction.

Example: Writing data to address 0 of the buffer memory in the special function module loaded at X/Y0, and reading the data of address 1.

(a) Using the FROM/TO instruction

(b) Using special direct device

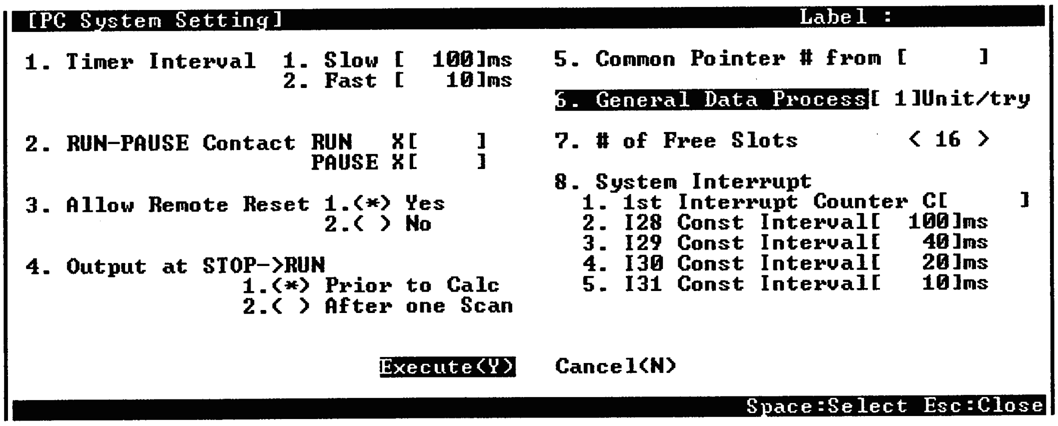


POINT

1. When reading data from the special function module frequently during the programming, store the special direct device to a data register after reading in an area of the program by using the FROM instruction rather than by using them at each instruction.
   This is because programming scan interval is added due to an access processing to the special function module for each instruction.
2. When executing the instruction using a special direct device for the special function module frequently in short scan time, it may cause the target special function module operation error.
   When performing the instruction using a special direct device, match the processing time and conversion time of the special function module using timer or constant scanning.

6.3 Processing for Data Communication Requests from a Special Function Module

When a data communication request is received from a special function module such as a serial communication module, the Q2ASCPU performs the processing for the data communication request at the END processing.
The Q2ASCPU can process all the data communication requests received in one scan with one END processing, according to the parameter settings. In this case, the data lag to each module is eliminated, but the END processing is extended by the data communications request processing.
Data communications request batch processing is set in the "6. General Data Processing"on the "PC system" screen in the GPP function parameter mode.
The setting range is 1 to 6 modules, and the processsing can be set per module.



```
[PC System Setting]                                         Label :
 1. Timer Interval  1. Slow [  100]ms    5. Common Pointer # from [      ]
                    2. Fast [   10]ms
                                         6. General Data Process[ 1]Unit/try

 2. RUN-PAUSE Contact RUN   X[     ]     7. # of Free Slots         < 16 >
                      PAUSE X[     ]
                                         8. System Interrupt
 3. Allow Remote Reset 1.(*) Yes           1. 1st Interrupt Counter C[      ]
                       2.( ) No            2. I28 Const Interval[  100]ms
                                           3. I29 Const Interval[   40]ms
 4. Output at STOP->RUN                    4. I30 Const Interval[   20]ms
                    1.(*) Prior to Calc    5. I31 Const Interval[   10]ms
                    2.( ) After one Scan

                              Execute(Y)  Cancel(N)
                                                 Space:Select Esc:Close
```

7 AUTO REFRESH FUNCTION

7.1 For MELSECNET/MINI-S3

By setting link information, I/O storage device, etc. of the MELSECNET/MINI-S3 to the parameters, the module automatically communicates with the buffer memory area for the batch refresh send/received data of the type A1SJ71PT32 MELSECNET/MINI-S3 master module (abbreviated as the MINI master module hereafter).
The settings are made on the MELSECNET/MINI setting in the parameter mode of GPP function.

Sequence programs can be created using the I/O devices allocated to send/received by the MELSECNET/MINI-S3 setting. (The FROM/TO instructions are not required.)

POINT

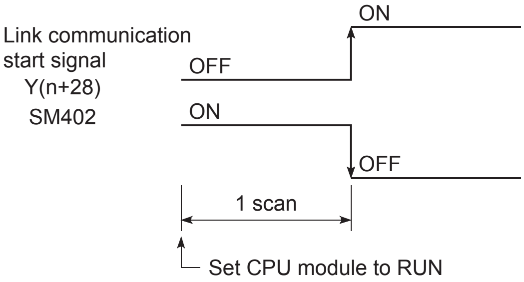
(1) Since up to 8 master modules can be set for auto refresh by the parameter, auto refresh is possible for up to 8 modules.
When 9 or more modules are desired, use the FROM/TO instruction in the sequence program from the 9th module.

(2) Since auto refresh is not possible with send/received data for the separate refresh I/O modules and for the remote terminal units No.1 to No.14, use them by the FROM/TO instructions.
However, the remote terminal units shown below are subject of auto refresh in the limited area:
- AJ35PTF-R2 RS-232C interface module
- AJ35PT-OPB-M1-S3 mount-type tool box
- AJ35PT-OPB-P1-S3 portable type tool box

(3) For the master modules set up for auto refresh, since the Q2ASCPU automatically turns ON the link communication start signal Y(n+18) or Y(n+28), it is not necessary to turn it on from the sequence program.

(4) Auto refresh of I/O data is performed by the batch after the Q2ASCPU performs the END instruction.
(Auto refresh processing is performed when the CPU module is in the RUN/ PAUSE/STEP-RUN status.)

(5) The master module may perform the processing while the link communication start signal Y(n+18) or Y(n+28) is OFF depending on the remote terminal units connected.
For instance, if the AJ35PTF-R2 RS-232C interface unit is used without protocol, it is necessary to write parameters to the parameter area (buffer memory address 860 to 929) while the link communication start signal is OFF.
Since the link communication start signal becomes ON after the CPU module enters the RUN status and one scan is performed, write the parameters during the first 1 scan.



(6) If the hardware error signal X(n+0)or X(n+20) or ROM error signal X(n+8)or X(n+28) of a master module for which auto refresh has been set comes ON, the Q2ASCPU does not perform auto refresh processing.

(7) When making the settings, ensure that there is no duplication between receive data refresh devices and send data refresh devices.

(1) Parameter setting items, setting ranges and contents of auto refresh, as well as the buffer memory address of the master module which is used for exchanging data with the Q2ASCPU are shown below.
Set the parameters for the number of the master modules used.

I/O signal from a master module	Buffer memory address of a master module	Item	Setting range	Description	Default value
–	–	Number of master modules	0, 1 to 8 module(s)	• Sets the total number of the master modules used. Set "0" if auto refresh is not to be used.	Follow the settings made in the "I/O Assign" in the parameter mode.*3
–	–	Start I/O No.	Number of I/O points of CPU module	• Sets the head I/O number where the master module is installed.	
–	–	Model classification of MINI/MINI-S3	• MINI or MINI-S3	• MINI In I/O mode (occupies 32 points) • MINI-S3 In expansion mode (occupies 48 points)	
–	0	Total number of remote I/O stations	0 to 64 stations	• Set only when MINI is set. • In MINI-S3, since the number of master module's initial ROMs becomes valid, the setting is not necessary .(When the setting is executed, ignore it).	
–	110 to 141	Storage device for received data *4	• X • M, L, B, T, ST, C, D, W, R, ZR, none (Bit device: multiples of 16)	• Sets the devices to store the received/send data for batch refresh. • Specify the head number of the device. • The total number of remote I/O stations, set starting from the first device number, is occupied as a auto refresh area. (8 points/station × 64 stations = 512 points⋯: Bit device)*2 • Use of X/Y remote I/O range is recommended for devices.	X1000 to X11FF
–	10 to 41	Send data storage device	• Y • M, L, B, T, ST, C, D, W, R, ZR, none (Bit device: multiples of 16)		Y1000 to Y11FF
–	1	Number of retries	0 to 32 times	• Sets the number of retries upon the communication errors occurrence. • Error is not output when the communication is restored within the number of the retries set.	5 times
Y(n+1A)*1	–	FROM/TO response specification	Link priority; CPU priority (Priority selection of access to the master module buffer memory)	1) Link priority • Link access by MINI-S3 has the priority. During the link access, FROM/TO is caused to wait. • Possible to read out the received data refreshed at the same timing. • The maximum wait time (0.3ms + 0.2ms × number of separate refresh stations) for the FROM/TO instruction may be generated. 2) CPU priority • The FROM/TO instructions from a CPU module are given access priority. Even during the link access, it interrupts and accesses. • Depending on the timing, received data in the midst of I/O refresh may be read. • No wait time for the FROM/TO instruction.	CPU priority
Y(n+1B)*1	–	Data clear specification for communication faulty station	Retention, clear (received data)	• Retention Retains the received data for batch and separate refresh. • Clear Sets all points to OFF	Clear

I/O signal from a master module	Buffer memory address of a master module	Item	Setting range	Description	Default value
–	100 to 103 195	Faulty station detection	M, L, B, T, ST, C, D, W, R, ZR, none (Bit device: multiples of 16)	• Sets the head device to store the faulty stations detected data. • MINI occupies 4 words; MINI-S3 occupies 5 words.	No setting
–	107 196 to 203	Error No.	T, ST, C, D, W, R, ZR	• Sets the head device to store the error code at the error occurrence. • MINI occupies 1 word; MINI-S3 occupies (1 + Number of remote terminal modules) words.	No setting
–	4	Line error check setting (Line error)	• Test message sending (Test) • OFF data sending (OFF) • Immediate data transmission before line errors (Retention)	• Sets data sending method for verification of faulty area when the line errors occur.	Retention
–	–	Operation at CPU STOP	Stop/Continue	• Sets the operating status when the CPU module is in the STOP state.	Stop

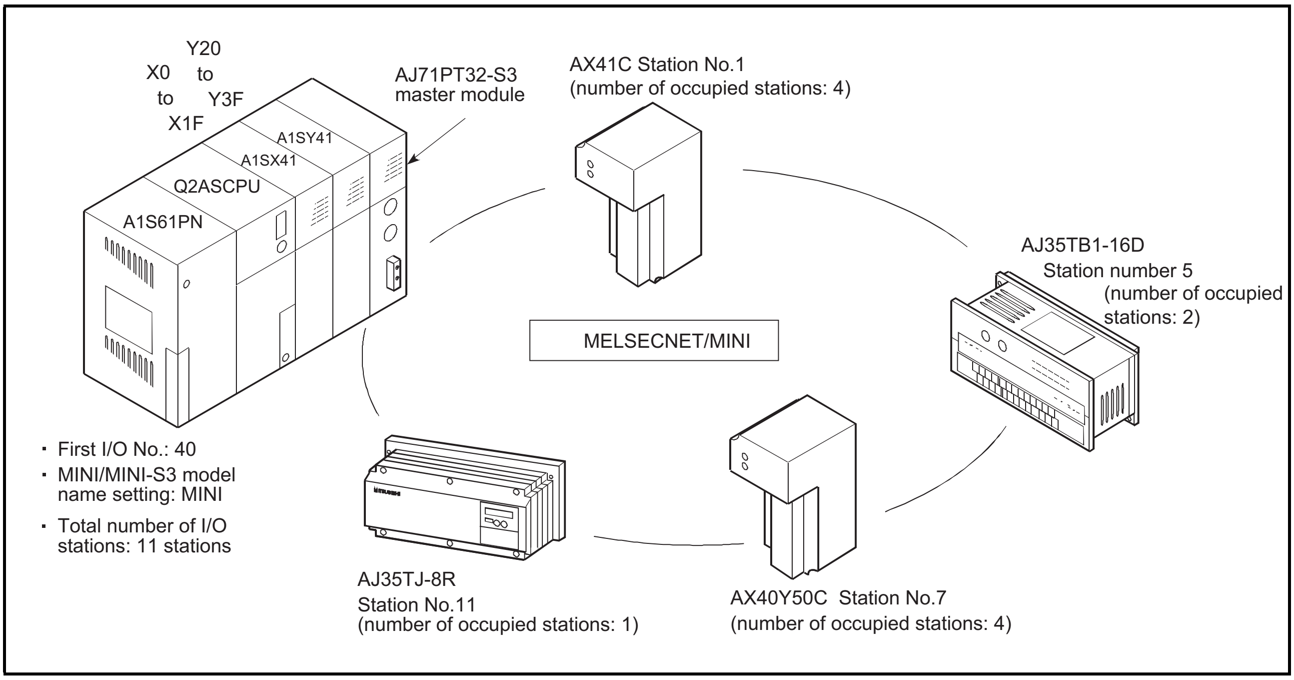
*1 "n" is determined by the installation location of the master modules.

*2 When the total number of remote I/O station is odd, add 1 to the station number to obtain the occupied storage devices.

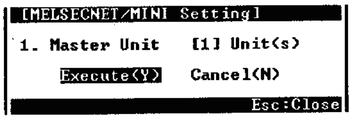
*3 When the master module number setting column is made blank in parameter setting, auto refresh can be used without this setting.
However, model name registration is required in the "I/O Assign". (MINI mode: AJ71PT32, MINI-S3 mode: AJ71PT32-S3)

*4 When the input (X) is specified in the received data storage device, use the I/O number later than the number used for the module loaded on the main base unit and the extension base unit. When the I/O number usage range for the module loaded on the main base and the extension base is used for input/output of the received data storage device, the CPU module imports both the input ON/OFF data from the input module and the ON/OFF data from auto refresh of MELSECNET/MINI-S3. Therefore, input (X) of the CPU module is not operated as desired.

(2) Setting of the send/received data storage devices is explained using the system example shown below.
(Example) When the device X/Y400 and later are used as the remote I/O stations:



Sample parameter setting of the GPP function for the above system configuration is shown below:

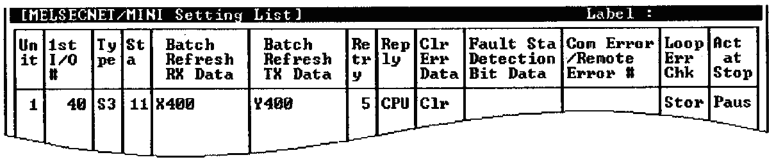


[MELSECNET/MINI Setting]

1. Master Unit [1] Unit(s)

Execute(Y) Cancel(N)

Esc:Close

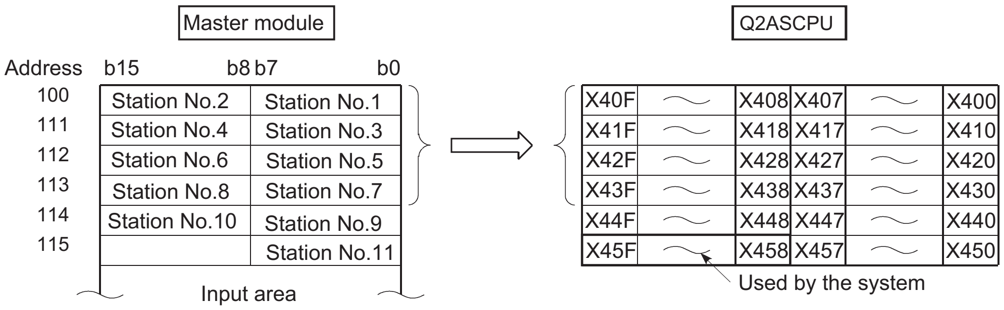


[MELSECNET/MINI Setting List] Label :

Un it	1st I/O #	Ty pe	St a	Batch Refresh RX Data	Batch Refresh TX Data	Re tr y	Rep ly	Clr Err Data	Fault Sta Detection Bit Data	Com Error /Remote Error #	Loop Err Chk	Act at Stop
1	40	S3	11	X400	Y400	5	CPU	Clr			Stor	Paus

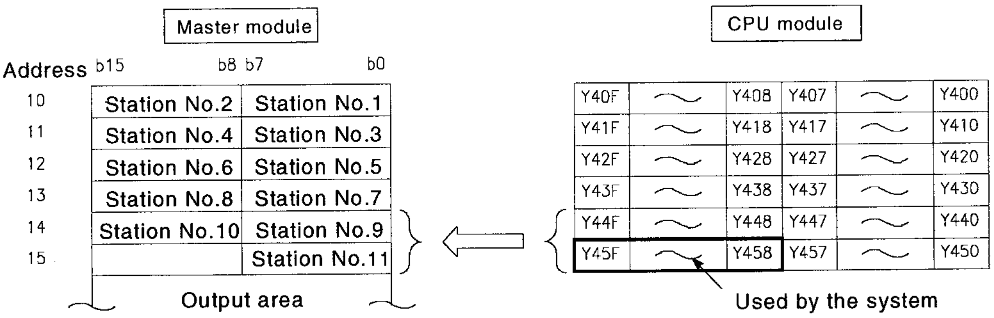
The storage devices for the send/received data for the present system example are as follows:

(a) Storage device for received data



1) Set the device number (X400) for b0 of the station 1 as a received data storage device.
2) The received data storage device occupies from X400 to X45F.
   For the present system example, since the total number of stations is odd, it is occupied for one extra station.
3) The device numbers of input modules connected are as follows:
   Stations 1 to 4 AX41C ⟶ X400 to X41F
   Stations 5 and 6 AJ35TB-16D ⟶ X420 to X42F
   Stations 7 and 8 AX40Y50C ⟶ X430 to X43F
   With respect to X440 to X45F, they are simultaneously refreshed, and set to OFF at any time.
   Do not use X440 to X45F in the sequence program.

(b) Send data storage device



1) Set the device number (Y400) for b0 of the station 1 as a send data storage device.
2) The send data storage device occupies from Y400 to Y45F.
   For the present system example, since the total number of stations is odd, it is occupied for one extra station.
3) The device numbers of output modules connected are as follows:
   Station 9 to 10 AX40Y50C ⟶ Y440 to Y44F
   Station 11 AJ35TJ-8R ⟶ Y450 to Y457
   With respect to Y400 to Y43F and Y458 to Y45F, they are simultaneously refreshed, but are not output.

POINT

(1) If the same device type is used for the send data storage devices and received data storage devices, make sure that there is no duplication of device numbers.
When the received data storage device is set to B0 in the system configuration example, it occupies B0 to B5F as the device range.
Set the send data storage device to B60 or later.
When the send data storage device is set to B60, the device range will be B60 to BBF.

(2) If a bit device is specified as the send/received data storage device, the device number set must be a multiple of 16.
Example:
X0, X10, ······ X100, ······
M0, M16, ······ M256, ······
B0, B10, ······ B100, ······

(3) Device range used is (8 points) × (Number of stations).
When the number of stations is an odd number, extra 8 points are necessary.

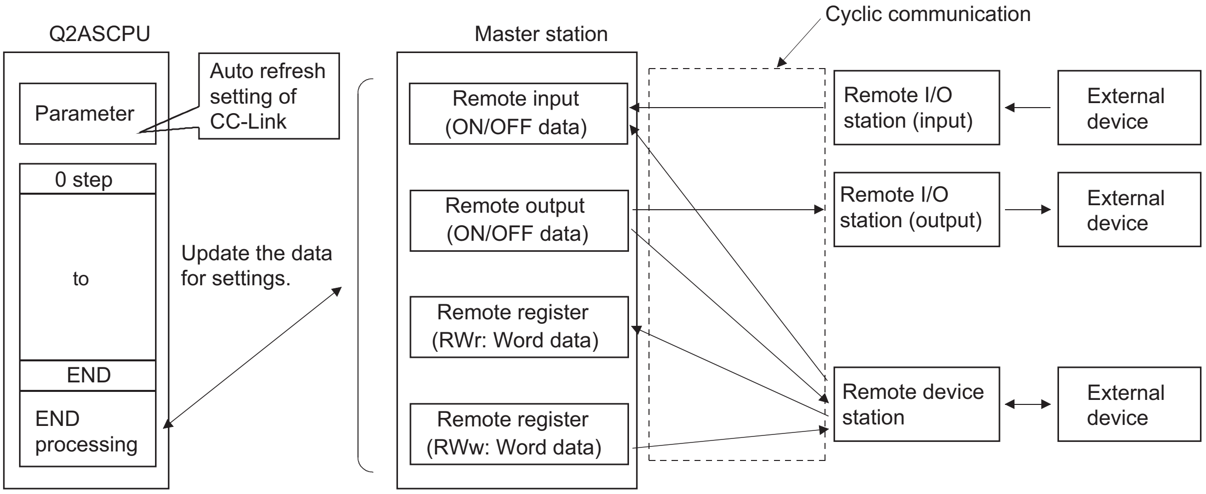
(4) When specifying input (X) for the received data storage device, specify the device number out of the actual input (X) range.

7.2 Auto Refresh Setting of CC-Link

Auto refresh of the CC-Link designates automatic communications between the Q2ASCPU and the buffer memory for cyclic communication of CC-Link master stations/ local stations.
Data for communication varies depending on the remote station connected.

- Remote I/O station (Communication in ON/OFF data)
- Remote device station (Communication in ON/OFF data and Word data)
- Intelligent device station (Communication in ON/OFF data and Word data)
- Master station/local station (Communication in ON/OFF data and Word data)

The auto refresh setting of the CC-Link allows communication with other stations of CC-Link using the FROM/TO instruction without communicating with the master station of the CC-Link.



(1) Settings for auto refresh
The Table 7.1 shows the setting items for auto refresh parameters of the Q2ASCPU.

Table 7.1 List of auto refresh settings

Item	Description	Setting range	Setting station		
			M	L	T
Number of modules	The number of CC-Link modules is set.	1 to 8	○	○	○
Module head I/O number	The head I/O number of a CC-Link module is set.	0000H to 0FE0H	○	○	○
Module type	The loaded CC-Link module type (Master station, local station, stand-by station) is set.	• M: Master station • L: Local station • T: Stand-by station	○	○	○
Receiving data batch refresh bit device (Input data)	• The device that stores the batch refresh received data from the remote station is set. • When the head device number is set, the points corresponding to the specified number of stations (Total number of stations) are obtained to refresh all areas. The output module area is also refreshed. • The settings are made in units of 16 points.	X, M, L, B, T, ST, C, D, W, R, ZR*	○	○	○
Transmission data batch refresh bit device (Output data)	• The device that stores the batch refresh send data to the remote station is set. • When the head device number is set, the points corresponding to the specified number of stations (Total number of stations) are obtained to refresh all areas. The input module area is also refreshed. • The settings are made in units of 16 points.	X, M, L, B, T, ST, C, D, W, R, ZR*	○	○	○
Receiving data batch refresh word device (Remote device: RWr)	• The device that stores the batch refresh received data from the remote station is set. • When the head device number is set, the points corresponding to the specified number of stations (Total number of stations) are obtained to refresh all areas. The I/O module area is also refreshed. • The settings are made per point.	M, L, B, T ,ST, C, D, W, R, ZR*	○	○	○
Transmission data batch refresh device (Remote device: RWw)	• The device that stores the batch refresh send data to the remote station is set. • When the head device number is set, the points corresponding to the specified number of stations (Total number of stations) are obtained to refresh all areas. The I/O module area is also refreshed. • The settings are made per point.	M, L, B, T ,ST, C, D, W, R, ZR*	○	○	○
Receiving buffer specification for transient station	• The receive buffer capacity for transient station is set.	80 to 4096	○	×	×
Transmission buffer specification for transient station	• The send buffer capacity for transient station is set.	80 to 4096	○	×	×
Batch refresh device for special relay	• The destination device for special relay is set.	M, L, B, T ,ST, C, D, W, R, ZR*	○	○	○

* Only when the file register is set to "Use the designated file" with the "Parameter", R and ZR can be used as the auto refresh devices.
When "Use same file name as program" is set, R and ZR cannot be used.

REMARK

1) In "Setting station" in the table above, M refers to the master station, L to the local station, and T to the stand-by station.
2) In the table above, ○ means that the setting can be made and × means that the setting is not required.

Table 7.1 List of auto refresh settings (Continued)

Item	Description	Setting range	Setting station		
			M	L	T
Batch refresh device for special register	• The destination device for special register is set.	T, ST, C, D, W, R, ZR*	○	○	○
Auto update buffer specification	• The buffer capacity for automatic update is set.	128 to 4096	○	×	×
Total number of slave stations	• The last station number of the remote station connected to the master station is set.	1 to 64	○	×	×
Delay timer	• The delay time of link scan is set.	1 to 100 (0 is invalid.)	○	×	×
Standby station specification	• The use status of the stand-by master function is set.	• Not used • Used	○	×	×
Number of retries	• The number of retries at the occurrence of a transient transmission error is set.	1 to 7	○	×	×
Number of automatic return stations	• The number of automatic return stations is set to one link scan.	1 to 10	○	×	×
Operation specification for CPU stop	• When the CPU module has stopped, continuation/stop of the data link is set.	• Stop • Continue	○	×	×
Scan mode setting	• Synchronization/Non-synchronization is set to the CPU module scan.	• Non-synchronization • Synchronization	○	×	×
Station type	• The model for each remote station is set.	• Remote I/O station • Remote device station • Intelligent device station	○	×	×
Number of occupied stations	• The number of occupied stations for each remote station is set.	• 1 station • 2 stations • 3 stations • 4 stations	○	×	×
Specification of reserved station	• Reservation for remote station is set.	• Not reserved • Reserved	○	×	×
Specification of invalid station	• Validity/Invalidity for error detection of the remote station is set.	• Invalid • Valid	○	×	×

* Only when the file register is set to "Use the designated file" with the "Parameter", R and ZR can be used as the auto refresh devices. When "Use same file name as program" is set, R and ZR cannot be used.

REMARK

1) In "Setting station" in the table above, M refers to the master station, L to the local station, and T to the stand-by station.
2) In the table above, ○ means that the setting can be made and × means that the setting is not required.

(2) Precautions

(a) Auto refresh of the CC-Link is available when the Q2ASCPU and the CC-Link module with function version "B" are used.
When either of the Q2ASCPU or the CC-Link module does not indicate function version "B," auto refresh of the CC-Link is not available.

(b) Auto refresh can be set to up to 8 CC-Link modules.
When 9 or more CC-Link modules are used, handle with the FROM/TO instruction of the sequence program for the 9th module or later.

(c) When both the CC-Link module and the master station module for MELSECNET/MINI-S3 are loaded and auto refresh is not set, the default parameter is set to the master station module for MELSECNET/MINI-S3.

(d) The COM instruction or the G(P). ZCOM instruction allows auto refresh to the CC-Link module while performing the sequence program.
However, auto refresh to the CC-Link module cannot be performed with the J(P).ZCOM instruction. Error code "4102" (The network number designated with the dedicated network instruction does not exist) appears.

(e) Refresh operation for the mixture of MELSECNET (/10, /II) and MELSECNET/ MINI-S3.

- Refresh is performed in the order of MELSECNET (/10, /II), CC-Link and MELSECNET/MINI-S3.
  Therefore, the input data specifying the same range is afterward overwritten with the executed data.
- The output data is output to the MELSECNET (/10, /II), CC-Link, and MELSECNET/MINI-S3.

(f) The operation of the Q2ASCPU when the CC-Link module is in the online/offline mode is shown in the table below:

Parameter settings for auto refresh	CC-Link module status	Operation of the Q2ASCPU
Set	Online	The communications with the remote station is performed with the specified parameter for auto refresh.
	Offline	The Q2ASCPU does not generate an error, but does not communicate with the remote station.
Not set	Online	The communications with the remote station is performed by the FROM/TO instruction.
	Offline	The Q2ASCPU does not generate an error, but does not communicate with the remote station.

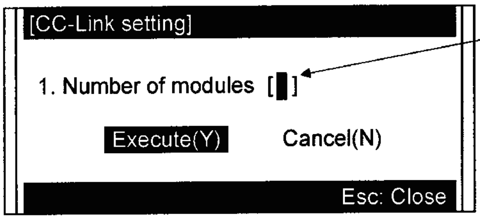
(g) Auto refresh setting to the CC-Link is performed using the following peripheral devices.

- Personal computer:
  GX Developer, SW2IVD-GPPQ type GPP function software package

(3) Setting method

Auto refresh setting to the CC-Link is set with the following procedures.

(a) When the "CC-Link" is selected in the "Parameter", the "CC-Link setting" screen appears.



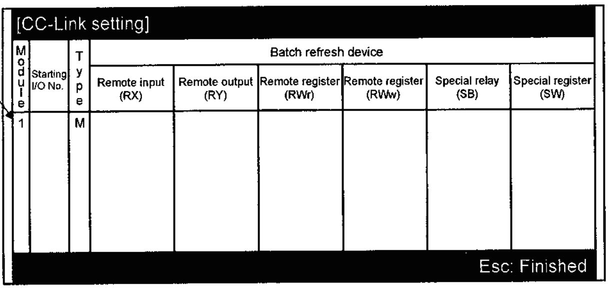
[CC-Link setting]

1. Number of modules [▌]

Set the number of the CC-Link modules (1 to 8).

(b) Set the number of the CC-Link modules loaded on the main base unit and extension base unit for the Q2ASCPU and selct "Execute", then the screen of CC-Link setting list appears.

Module number setting for refresh setting



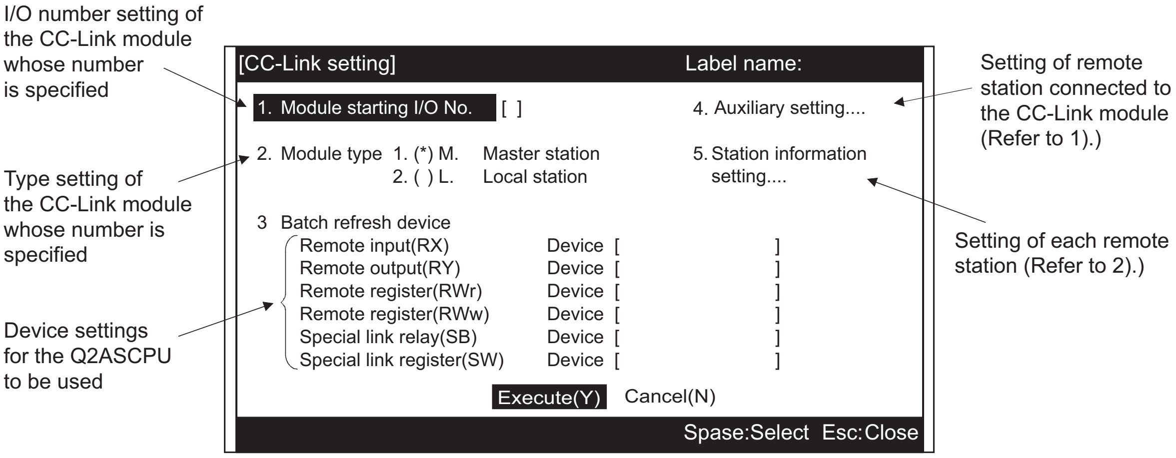
[CC-Link setting]

Module	Starting I/O No.	Type	Batch refresh device					
			Remote input (RX)	Remote output (RY)	Remote register (RWr)	Remote register (RWw)	Special relay (SB)	Special register (SW)
1		M						

Confirmation of settings (Press F3 key for setting.)

Pressing the Esc key registers the set data.

(c) Move the cursor to the module number position for auto refresh setting and press the F3 key (Detail). The "CC-Link setting" screen appears. Select "4. Auxiliary setting...." and "5. Station information setting...." to set detailed data.



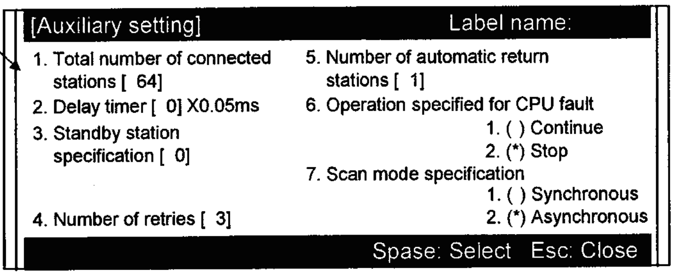
```
[CC-Link setting]                                   Label name:

 1. Module starting I/O No.  [ ]                     4. Auxiliary setting....

 2. Module type  1. (*) M.   Master station          5. Station information
                 2. ( ) L.   Local station              setting....

 3  Batch refresh device
    Remote input(RX)           Device  [                  ]
    Remote output(RY)          Device  [                  ]
    Remote register(RWr)       Device  [                  ]
    Remote register(RWw)       Device  [                  ]
    Special link relay(SB)     Device  [                  ]
    Special link register(SW)  Device  [                  ]

                        Execute(Y)   Cancel(N)
                                          Spase:Select  Esc:Close
```

When pressing the Execute (Y) or the Esc key , the screen returns to the screen of CC-Link setting list.

1) When selecting the "4. Auxiliary setting....", the "Auxiliary setting" screen appears.

   When pressing the Esc key, the screen returns to the CC-Link setting screen of (c).

Set the number of remote stations connected to the CC-Link module.



[Auxiliary setting] Label name:

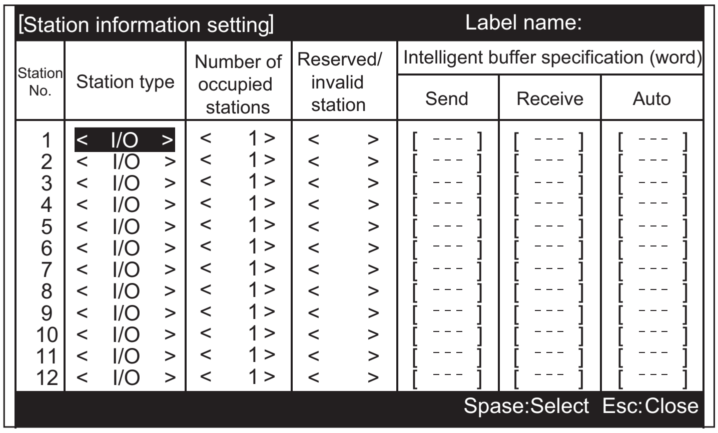
1. Total number of connected stations [64]
2. Delay timer [0] X0.05ms
3. Standby station specification [0]
4. Number of retries [3]
5. Number of automatic return stations [1]
6. Operation specified for CPU fault
   1. ( ) Continue
   2. (*) Stop
7. Scan mode specification
   1. ( ) Synchronous
   2. (*) Asynchronous

Spase: Select Esc: Close

2) When selecting the "5. Station information setting...", the "Station information setting"screen appears.

   When pressing the Esc key, the screen returns to the CC-Link setting screen of (c).

Set the number of stations specified in 2).



[Station information setting] Label name:

Station No.	Station type	Number of occupied stations	Reserved/ invalid station	Intelligent buffer specification (word)		
				Send	Receive	Auto
1	< I/O >	< 1 >	< >	[---]	[---]	[---]
2	< I/O >	< 1 >	< >	[---]	[---]	[---]
3	< I/O >	< 1 >	< >	[---]	[---]	[---]
4	< I/O >	< 1 >	< >	[---]	[---]	[---]
5	< I/O >	< 1 >	< >	[---]	[---]	[---]
6	< I/O >	< 1 >	< >	[---]	[---]	[---]
7	< I/O >	< 1 >	< >	[---]	[---]	[---]
8	< I/O >	< 1 >	< >	[---]	[---]	[---]
9	< I/O >	< 1 >	< >	[---]	[---]	[---]
10	< I/O >	< 1 >	< >	[---]	[---]	[---]
11	< I/O >	< 1 >	< >	[---]	[---]	[---]
12	< I/O >	< 1 >	< >	[---]	[---]	[---]

Spase:Select Esc:Close

8 DEBUGGING FUNCTION

8.1 Function List

Q2ASCPU has a variety of convenient functions when debugging.
The following shows the debugging functions.

Item		Description	Reference
Monitor function		Function that reads CPU programs, device statuses from a peripheral device capable of GPP functions	Section 8.2
Write during RUN		Function that writes a program while the CPU module is running	Section 8.3
Execution time measurement		Functions that displays the processing time of a program being execute	Section 8.4
	Program monitor list	Functions that displays the processing time of a program being executed	Section 8.4.1
	Interrupt program monitor list	Function that displays the number of executions of an interrupt program	Section 8.4.2
	Scan time measurement	Function that measures the execution time of section of a program	Section 8.4.3
Sampling trace function[*1]		Function that continually collects the data of devices in accordance with a timing set at the CPU module	Section 8.5
Status latch function[*1]		Function that collects the device data at the moment to designate	Section 8.6
Step operation		Functions that runs one step or one part of a program, runs a program with a part skipped	Section 8.7
	Step execution	Function that runs a program step by step	Section 8.7.1
	Partial execution	Function that executes a designated part of a program	Section 8.7.2
	Skip execution	Function that executes a program with a designated part skipped	Section 8.7.3
Program trace function[*1]		Function that collects the program execution status	Section 8.8
Simulation function[*2]		Function that simulats execution in isolation from the I/O modules and special function modules	Section 8.9
Debugging by several people		Function that simultaneously debuggs from several peripheral devices capable of GPP functions	Section 8.10
Monitoring trace function		Function that collects device data at a peripheral device capable of GPP functions in accordance with the designated timing	–

For details on the operation for each function, refer to the GPP function Operating Manual.

*1 When executing this function, a memory card is required.
*2 When executing part of this function, a memory card is required.

8.2 Monitor Function

This function reads CPU module programs and device statuses to a peripheral device capable of GPP functions.

Application

This function is used to set monitoring conditions for monitoring the operating statuses of the programmable controller in accordance with a precise timing.
There are three "Monitoring Condition" as follows.

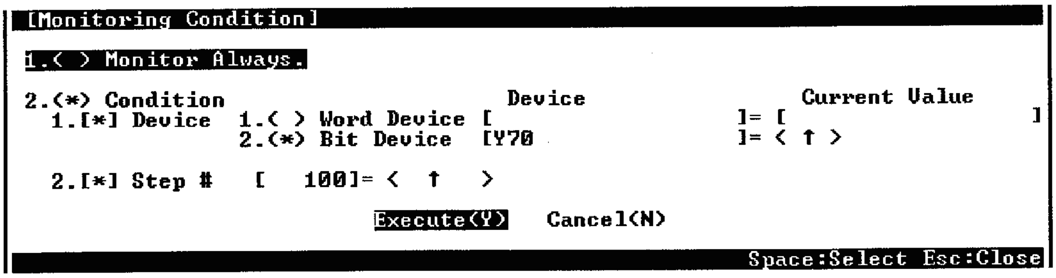
- Executing a monitoring at END processing.
- Setting the step number to be monitored and the step conducting status.
- Setting the device status.

This function is used to retain the monitoring screen by setting "Monitor stop condition setup" in accordance with a precise timing.
When monitoring the CPU module marked Function version B using a peripheral device capable of GPP function, local device monitor test is executed by setting "local device monitor".

8.2.1 Monitoring condition setting

Function Description

(1) This function allows setting of the monitoring condition.
All operations are performed using Monitor/test menu in the ladder mode.
The following shows an example of setting a monitor condition.



```
[Monitoring Condition]

1.( ) Monitor Always.

2.(*) Condition                      Device                Current Value
  1.[*] Device   1.( ) Word Device   [                     ]= [                    ]
                 2.(*) Bit Device    [Y70                  ]= < ↑ >

  2.[*] Step #   [    100]= <  ↑   >

                              Execute(Y)    Cancel(N)

                                                         Space:Select Esc:Close
```

The following shows an explanation of the screen above:
The monitoring condition can be select either "1. ( ) Monitor Always." or "2. ( ) Condition".

(a) When "1. ( ) Monitor Always." is set
The collection timing for monitor data is every scan after END processing at the CPU module.

(b) When "2. ( ) Condition" is set
"1. [ ] Device" and "2. [ ] Step #" can be set.

1) When only "2. [ ] Step #" is set
The monitor data collection timing is the moment when a QnACPU shows designated status immediately before executing the designated step.
The following shows the possible designations to execute.

▪ When switching from OFF to ON:	:< ↑ >
▪ When switching from ON to OFF:	:< ↓ >
▪ All the time only during ON	:< ON >
▪ All the time only during OFF	:<OFF >
▪ All the time in any statuses	:<Always>

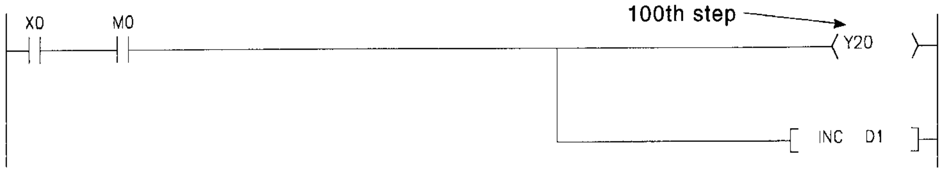
REMARK

1) When Step # [0] is designated, the execution condition must be set as Always.
2) When "1. [ ] Device" only is specified (when "2. [ ] Step #" is not specified), the monitor data collection timing is every scan after END processing of the programmable controller CPU.
   When the data is changed in the same scan, it cannot be detected. (Including the low-speed program)
3) When only "1. [ ] Device" is set, either
   "1. ( ) Word Device" or "2. ( ) Bit Device" can be designated.
   ⓐ When "1. ( ) Word Device" is designated
   The collection timing of the monitor data is the scan END processing when the current value of the specified word device becomes the specified value.
   The following shows the method for designating the current value.
   - Decimal designation: [K Decimal number]
   - Hexadecimal designation: [H Hexadecimal number]

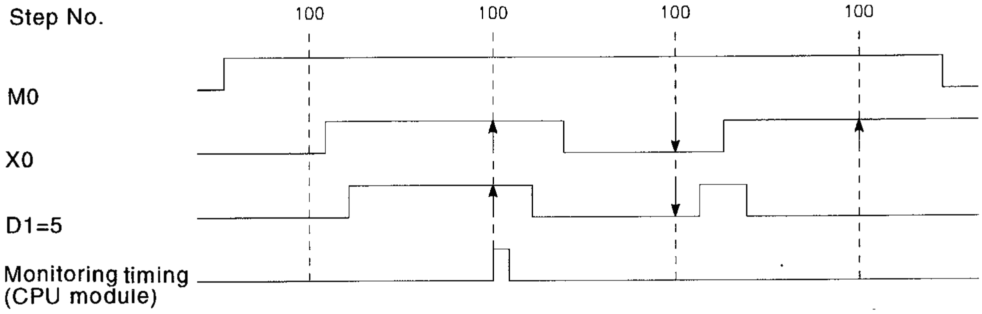
   ⓑ When "2. ( ) Bit Device" is designated
   The collection timing of the monitor data is the scan END processing when the execution status of the specified bit device becomes the specified status.
   The following shows the possible designations for execution status.
   - At leading edge :< ↑ >
   - At trailing edge :< ↓ >
4) When "2. [ ] Step #" and "1. [ ] Device" are designated
   The monitor data collection timing is such that data is collected when the status immediately before execution of the designated step or the execution status (current value) of the designated bit device (word device) attains the designated status.

POINT

In the ladder block shown below, assuming that the detailed condition is set as follows: "Step # [100] = < ↑ >, Word device [D1] = [K5]".



The monitor timing is shown below. However, the monitoring interval at a peripheral device capable of GPP functions depends on the processing speed of that peripheral device. Even if data changes occur faster than the monitor interval, data can be collected only once during the interval.



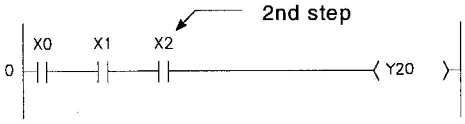
POINT

(1) Assume that "Step # [2] = <ON>" is designated as the detailed condition in the case of the ladders shown below; In this case the monitor execution differs for the two ladders. For (a) it is "X0 and X1 both ON" and for (b) it is "X1 ON (regardless of ON/OFF status of X0)".
If a step part way through an AND/OR block is designated for a monitor condition, the monitor data collection timing is such that data is collected when the status immediately before execution of the step designated from the LD instruction in the block becomes the designated status.

Ladder mode

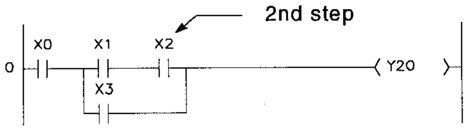
List mode

(a)



```
0   LD    X0
1   AND   X1
2   AND   X2
3   OUT   Y20
```

(b)

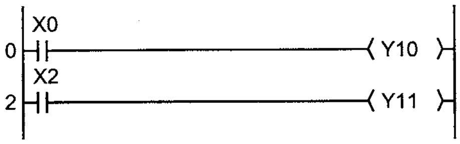


```
0   LD    X0
1   LD    X1
2   AND   X2
3   OR    X3
4   ANB
5   OUT   Y20
```

(2) If the ladder block head other than 0 step is specified to the step number as detailed conditions, the monitor data is collected when the instruction execution status immediately before execution is the specified status.
When "Step # [2] = <ON>" is specified in the ladder below, the monitor data is collected for OUT Y10 ON.

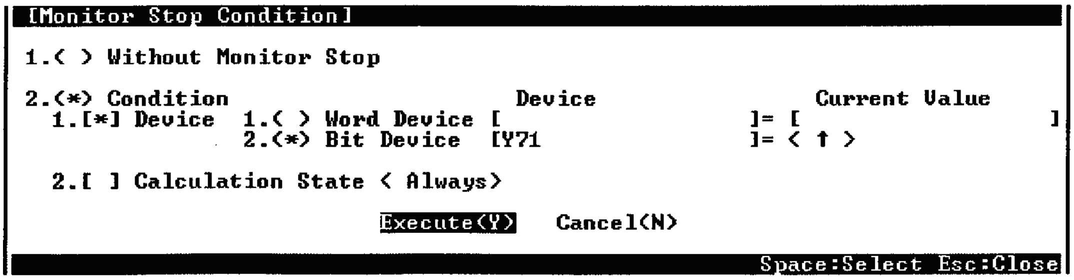
Ladder mode

List mode



```
0   LD    X0
1   OUT   Y10
2   LD    X1
3   OUT   Y11
```

(2) A monitor stop condition can be set.
All operations are performed on the monitor/test screen window in the ladder mode.
The following shows an example of the setting for a monitor stop condition.



```
[Monitor Stop Condition]
1.( ) Without Monitor Stop

2.(*) Condition                        Device                Current Value
  1.[*] Device  1.( ) Word Device [                  ]= [                      ]
                2.(*) Bit Device  [Y71               ]= < ↑ >

  2.[ ] Calculation State < Always>

                          Execute(Y)   Cancel(N)
                                                        Space:Select Esc:Close
```

The following shows an explanation of the screen above:
Either "1. ( ) Without Monitor Stop" or "2. ( ) Condition" can be set for the monitor stop condition.

(a) When "1. ( ) Without Monitor Stop" is set
Monitoring is stopped when Esc key is pressed.

(b) When "2. ( ) Condition" is set
"1. [ ] Device" and "2. [ ] Calculation State" can be set.

1) When "2. [ ] Calculation State" is set
The monitor stop timing is such that monitoring stops when the execution condition of the step designated for the monitor condition attains the designated status.
The following shows the possible designations for execution status.

▪ When switching from OFF to ON	:< ↑ >
▪ When switching from ON to OFF	:< ↓ >
▪ All the time only during ON	:<ON>
▪ All the time only during OFF	:<OFF>
▪ All the time in any statuses	:<Always>

If "2. [ ] Calculation State" isn't set, the timing for monitor stop is such that monitoring is stopped after CPU module END processing.

2) When "1. [ ] Device" is set Either
"1. ( ) Word Device" or "2. ( ) Bit Device" can be set.

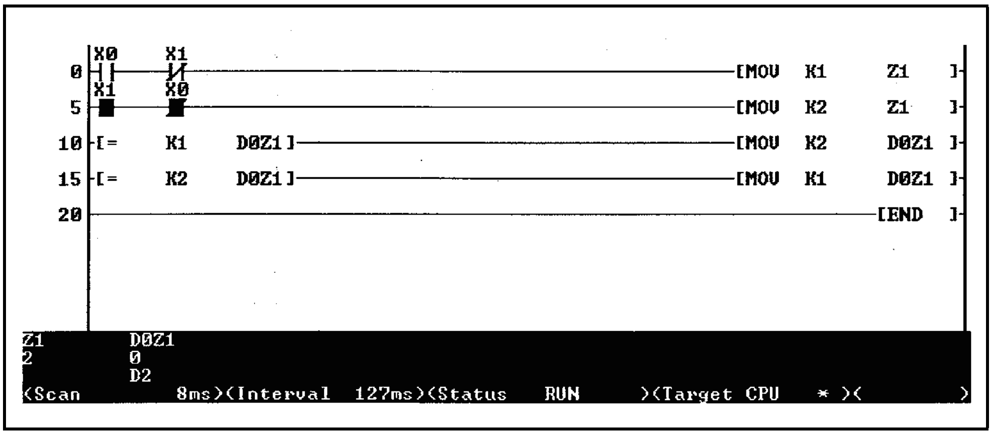
ⓐ When "1. ( ) Word Device" is set
The monitor stop timing is such that monitoring stops when the present value of the designated word device attains the designated value.
The following shows the method for designating the current value.

- For decimal (word) designation : [K Decimal number]
- For hexadecimal (word) designation : [H Hexadecimal number]
- For decimal (double word) designation: [K Decimal number Space L]
- For hexadecimal (double word) designation : [H Hexadecimal number Space L]
- For real number designation : [E Real number]

ⓑ When "2. ( ) Bit Device" is designated
The monitor stop timing is such that monitoring stops when the execution status of the designated bit device becomes the designated status.
The following shows the possible designations for execution status.

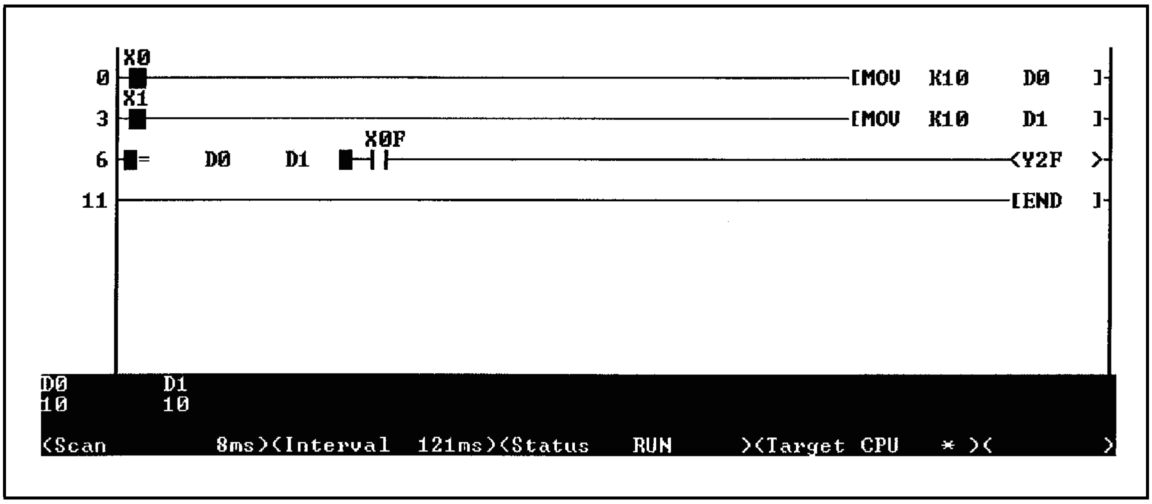
- At leading edge :< ↑ >
- At trailing edge :< ↓ >

(3) In the case of devices for which index qualifications have been made, the index qualified value is monitored.
The following shows an example of this type of monitoring.



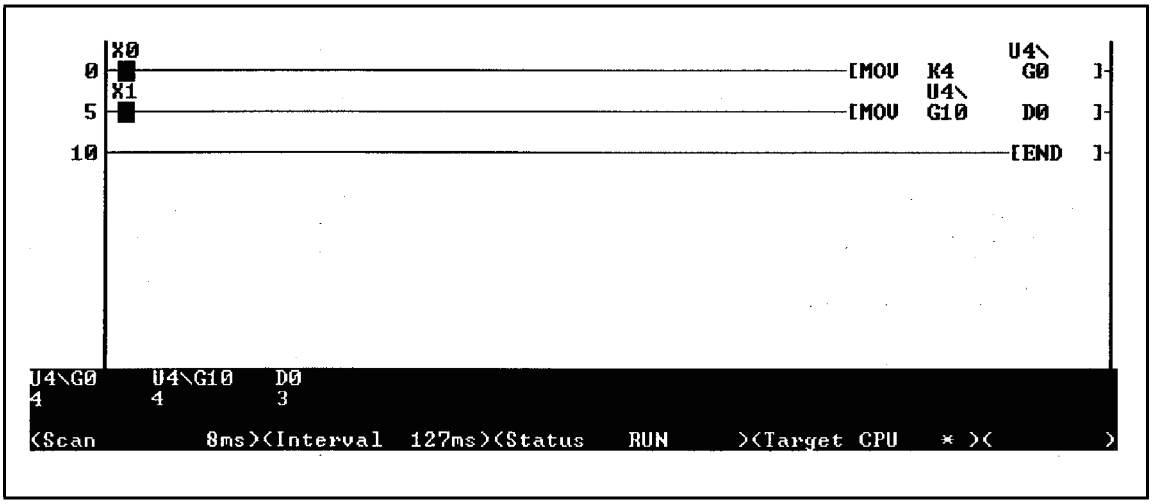
```
Z1          D0Z1
2           0
            D2
<Scan          8ms)(Interval  127ms)(Status    RUN      )(Target CPU    * )(          >
```

(4) The ON status of comparison instructions can be monitored.
The following shows an example of this type of monitoring.



```
    X0
 0 -■-------------------------------------------[MOV  K10   D0   ]-
    X1
 3 -■-------------------------------------------[MOV  K10   D1   ]-
                          X0F
 6 -■=    D0     D1    ■-| |-------------------------------<Y2F  >-
11 ---------------------------------------------------------[END  ]-

D0          D1
10          10

<Scan        8ms><Interval   121ms><Status    RUN      ><Target  CPU    * ><          >
```

(5) The devices of special function modules can be monitored.
The following shows an example of this type of monitoring.

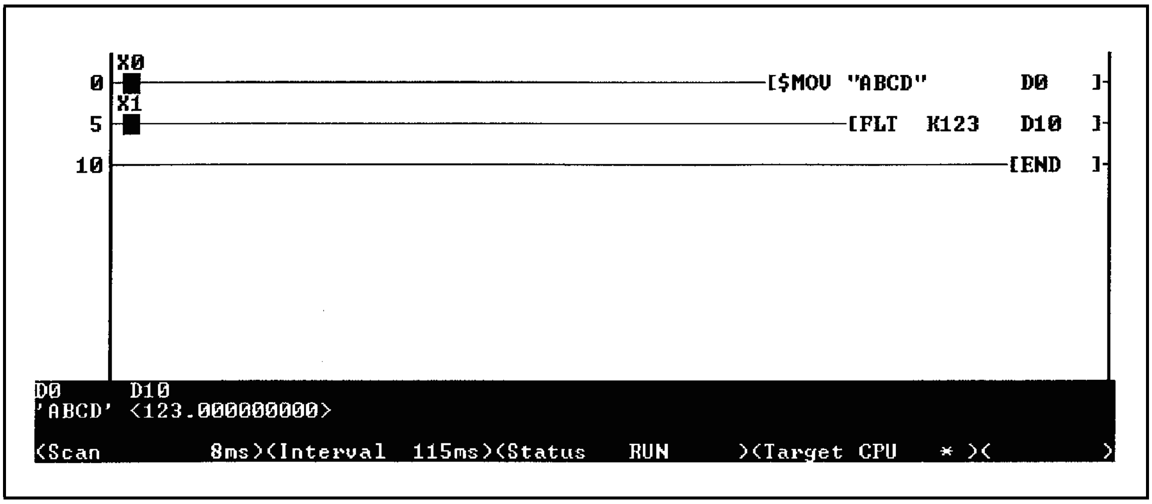


```
    X0                                                         U4\
 0 -■-------------------------------------------[MOV  K4     G0   ]-
    X1                                                  U4\
 5 -■-------------------------------------------[MOV  G10    D0   ]-
10 ---------------------------------------------------------[END  ]-

U4\G0      U4\G10     D0
4          4          3

<Scan        8ms><Interval   127ms><Status    RUN      ><Target  CPU    * ><          >
```

REMARK

To monitor devices of special function modules, set "2. Buffer Memory 1. Monitor" for "5/Monitor Target Setting" under the ladder mode "Option" menu.

(6) Real numbers and character strings can be monitored.
The following shows an example of this type of monitoring.



```
   X0
 0 -■---------------------------------------[$MOV "ABCD"     D0  ]-
   X1
 5 -■--------------------------------------------[FLT  K123  D10 ]-
10 ------------------------------------------------------[END    ]-

D0      D10
'ABCD'  <123.000000000>

<Scan          8ms><Interval  115ms><Status   RUN      ><Target CPU   * ><          >
```

(7) The following shows the devices that can be monitored.

(a) Bit devices : X, FX, DX, Y, FY, DY, M, L, F, SM, V, B, SB, T(Contact), T(Coil), ST(Contact), ST(Coil), C(Contact), C(Coil), J□\X, J□\Y, J□\B, J□\SB, BL□\S

(b) Word device : T(Current value), ST(Current value), C(Current value), D, SD, FD, W, SW, R, Z, ZR, U□\G, J□\W, J□\SW

(8) The following shows the setting device under the detailed condition.

(a) Bit device : X, FX, Y, FY, M, L, F, SM, V, B, SB, T(Contact), ST(Contact), C(Contact), J□\X, J□\Y, J□\B, J□\SB, BL□\S

(b) Word device : T(Current value), ST(Current value), C(Current value), D, SD, FD, W, SW, R, Z, ZR, U□\G, J□\W, J□\SW

The following qualifications are possible with respect to the devices listed above.

- Digit designation for bit devices
- Bit number designation for word devices

NOTE

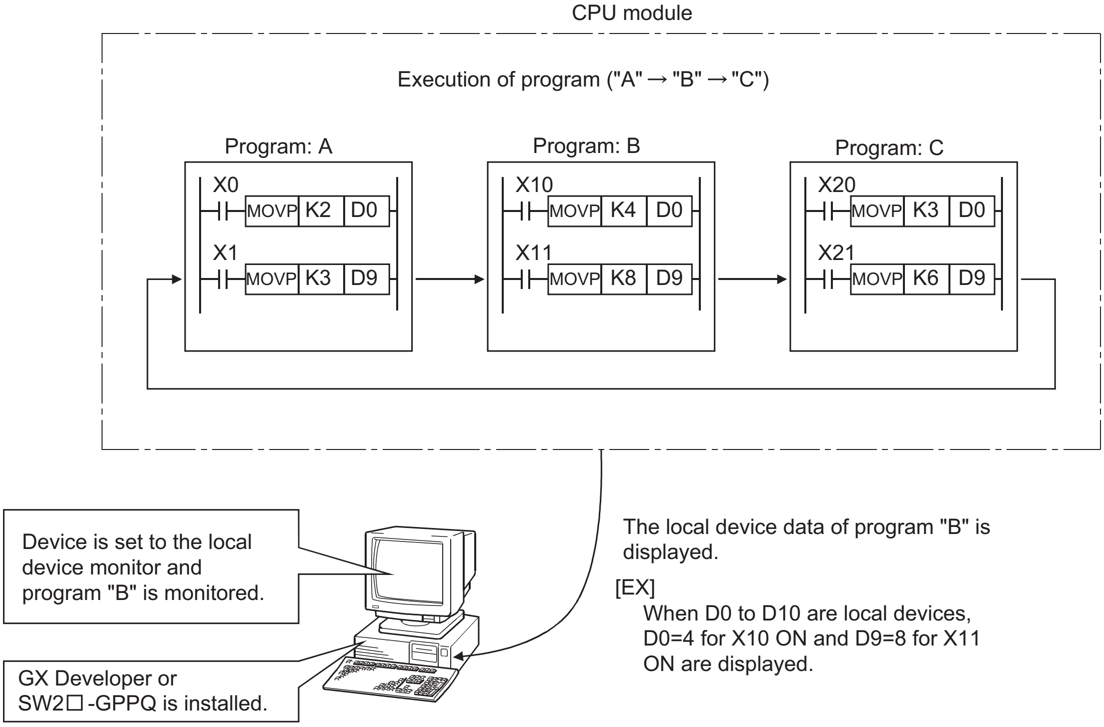
1) When a monitoring is performed with a monitor condition set, the file displayed at the device running GPP function is monitored.
   Make sure that the file name used with GPP function is the same as the file name when monitoring is performed by executing "Newly from PLC".
2) When the buffer memory of a peripheral device is read by designating a direct device, FFFFH is monitored if the peripheral device is faulty or not connected.
3) When monitoring file registers, FFFFH is monitored if no file register designation is made.
4) Before monitoring, make sure that the device assignment of the CPU and GPP function agree.
5) For the local device monitor in each program file, the monitor operation varies depending on presence of the CPU module function version B and the GPP function model.
   [Without function version B]
   - Detailed conditions (step number and device condition) are set for each program file to perform monitoring.

   [With function version B]
   - When the GX Developer and the SW2IVD-GPPQ are used, the local device can be monitored in each program file by setting compatibility with local device.(Refer to Section 8.2.2.)
   - When SW0IVD-GPPQ and SW1IVD-GPPQ are used, the local device can be monitored with the same operation as the operation without function version B.
6) When monitoring the buffer memory of a special function module, the scan time is prolonged in the same way as it is when a FROM/TO instruction is executed.
7) Several people can perform monitoring at the same time.
   The following considerations apply when executing this:
   - High-speed monitoring can be made possible by increasing the system area by 1k steps for each monitor for other station use when formatting the built-in RAM.
     In the monitor for other stations, 15k steps maximum can be set in the system area and the corresponding file space for the user is reduced.
   - Only one person can set the detailed conditions for monitoring.
8) The detailed conditions for monitoring can only be set in ladder monitoring.
9) If the same device is designated for both a monitor condition and monitor stop condition, also designate the "ON" or "OFF" status.
10) When the step number is specified for the monitor condition, monitor conditions are not satisfied for no execution of the applicable step instruction as shown below:
    - Applicable step instructions are skipped by CJ, SCJ and JMP instructions.
    - The applicable instruction is the END instruction and the FEND instruction exists in the program.
11) Do not reset the CPU module while the monitor condition is registered.

8.2.2 Monitor test of local device (function version B or later)

With the "parameter device setting," the device set in the local device can be monitored and tested in the peripheral device.
This function allows debugging while checking details of the local device by peripheral devices.

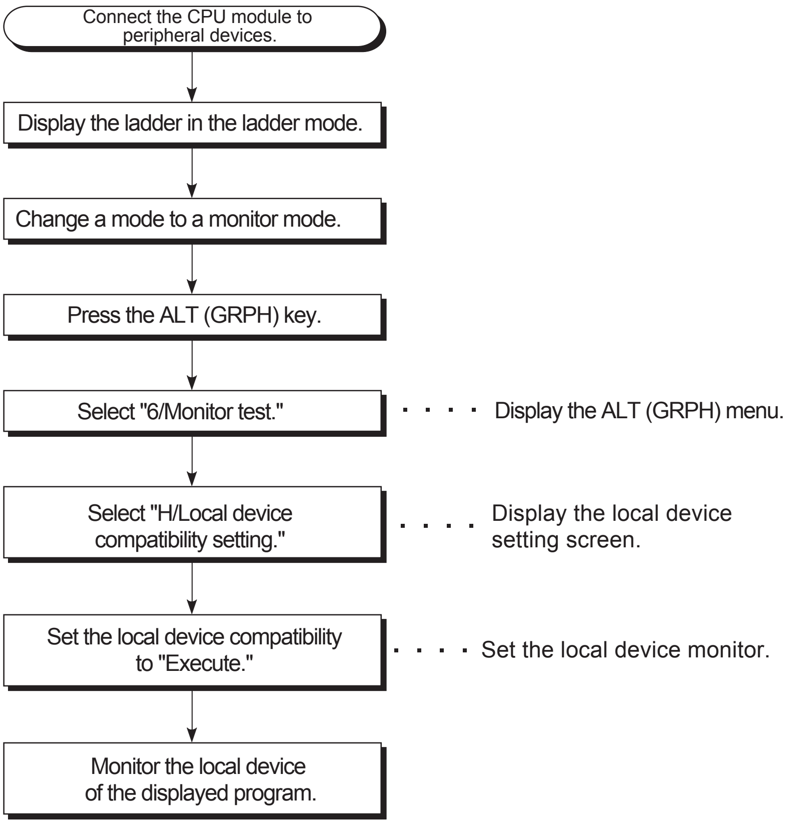
To monitor the local device, set the peripheral device to "local device monitor". The following fig. is an example that monitoring the local device of program B with the CPU module, which is executing programs A, B and C.



(1) Peripheral device
To perform monitor test of the local devices, the following GPP function software packages are required:
- Personal computer
  GX Developer, SW2IVD-GPPQ type GPP function software package

(2) Monitoring procedures of local device

The following shows the procedures to be monitored local devices:



(3) Operation for CPU module and GPP function versions

Table 8.1 shows the operation when the local devices are set to D0 to D99 and when 3 programs with the program names of "A", "B" and "C" are performed in the CPU module.

(The order of the programs is A, B, C, (END processing), A, B...)

Table 8.1 Operation for CPU module and GPPQ function versions

GPPQ Model Name		Monitor Device			
		With Function Version "B"		Without Function Version "B"	
		D0	D100	D0	D100
SW0□-GPPQ SW1□-GPPQ		D0 of program "C" is monitored.	D100 after execution of program "C" is monitored.	D0 of program "C" is monitored.	D100 after execution of program "C" is monitored.
SW2□-GPPQ	When local device is not set	D0 of program "C" is monitored.	D100 after execution of program "C" is monitored.	D0 of program "C" is monitored.	D100 after execution of program "C" is monitored.
	When local device is set	D0 of the displayed program is set.	D100 after execution of the displayed program is set.	An error (error code: 4001) occurs.	

REMARK

GX Developer supports functions of function version B.

(4) Precautions

(a) The local device that can perform the monitor test in one peripheral device is only one program.

Monitor test for multiple program local devices from one peripheral device is not allowed.

(b) The number of programs that allows simultaneous monitor test from multiple peripheral devices is up to 16.

When the local device of the stand-by type program is monitored, the local device data is read/escape. The scan time is extended as follows:

- Q2ASCPU(S1) :560 + 1.3×(Number of words in the local device) [μs]
- Q2ASHCPU(S1) :220 + 0.8×(Number of words in the local device) [μs]

8.3 Write During RUN

This is a function that writes a program to the CPU while the CPU module is in the RUN.

⚠ CAUTION

- ● Read the manual carefully and confirm safety before changing the program during operation.
  An operation error of write during run may result in damage to the machine or accident.

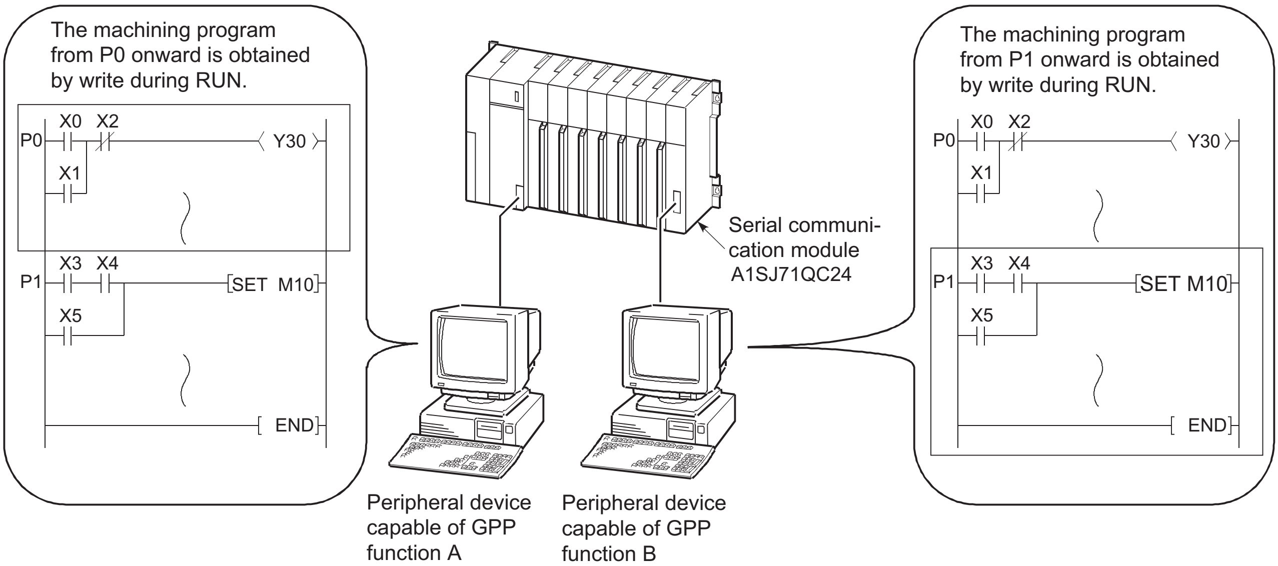
Application

This function is used to change a program without stopping program execution.

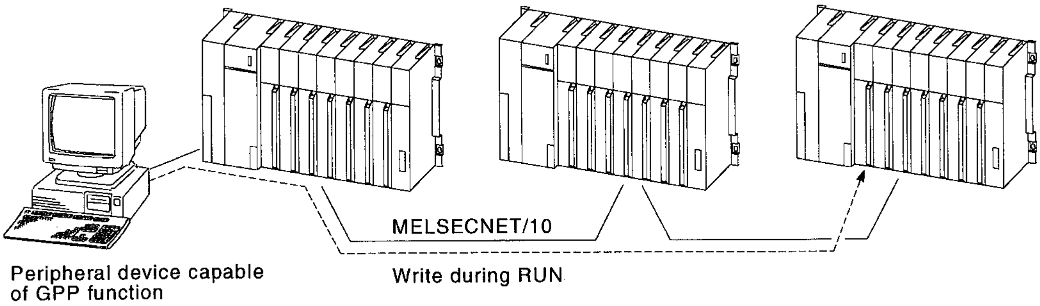
Function Description

(1) Write during RUN is possible from multiple peripheral devices capable of GPP functions with respect to one file.
In order for this, designate the pointer for the programs to be written from the peripheral devices capable of GPP functions in advance. This enables write safely during RUN using peripheral devices capable of GPP functions.

The example below shows a case where peripheral device capable of GPP functions A performs write during RUN from P0, and peripheral device capable of GPP functions B performs write during RUN from P1. The program enclosed in the frame ▭ is the program subject to write during RUN.



(2) It is possible to write programs from peripheral devices capable of GPP functions that are connected to other stations in the network during the RUN.



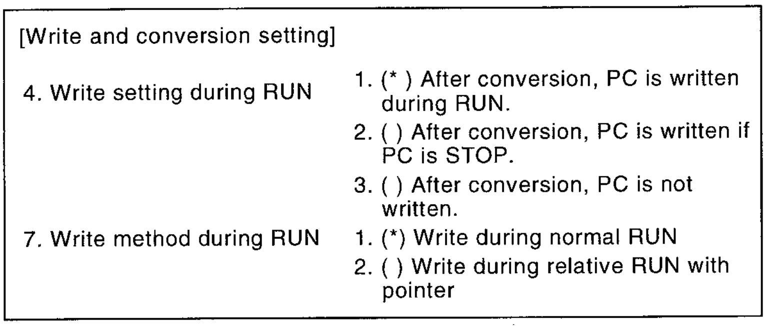
Operation Procedures

To write from the GPP function peripheral devices during RUN, the following two methods are available:

(1) After a ladder is created in the ladder mode, write during RUN is performed by pressing Shift + F4 keys for conversion of the ladder.

(2) With "4. Write/conversion setting" in "8/ Option" menu of the ladder mode, "4. Write setting during RUN" and "7. Write method during RUN" are set.

When the F4 key is pressed for conversion of the ladder after the ladder is created, write during RUN is performed.
The following shows the setting examples:



[Write and conversion setting]

4. Write setting during RUN
1. (*) After conversion, PC is written during RUN.
2. ( ) After conversion, PC is written if PC is STOP.
3. ( ) After conversion, PC is not written.

7. Write method during RUN
1. (*) Write during normal RUN
2. ( ) Write during relative RUN with pointer

(a) In "4. Write setting during RUN," "1. ( ) After conversion, PLC is written during RUN" is set.

(b) In "7. Write method during RUN," "1. ( ) Write during normal RUN" or "2. ( ) Write during relative RUN with pointer" is selected.

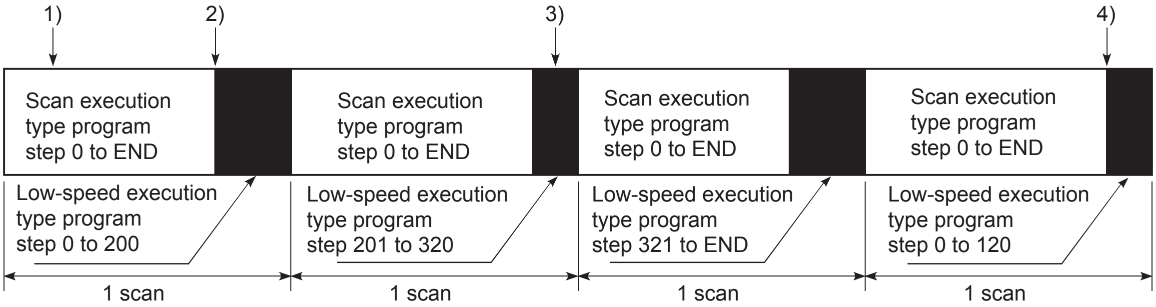
NOTE

The following shows the precautions relating to write during RUN.

(1) The only memory that can be used for write during RUN is the built-in RAM. If write during RUN is performed during a boot operation, also write the program to the memory card at STOP.
When the boot operation is started again without write on the memory card, the program before write during RUN is transferred from the memory card to the built-in RAM for execution.

(2) The maximum number of steps that can be handled in one write during RUN operation is 512.
The number can be changed according to how many steps of write during RUN saved using a peripheral device capable of GPP function. The saved steps of write during RUN can be set during the CPU module OFF. Note that the saved steps of write during RUN decrease every time write during RUN is performed.

(3) During low-speed program execution, write during RUN is started when execution of all low-speed programs is completed. Also, execution of low-speed programs is suspended during write during RUN.



1): Write during RUN command of scan execution type program
2): Write during RUN execution of scan type program
3): Write during RUN command of low-speed execution type program
4): Write during RUN execution of low-speed execution type program

POINT

Write during RUN cannot be performed on the program in step operation.

8.4 Execution Time Measurement

This is a function that displays the processing time of the program being executed.

Application

This function is used to determine the influence of the processing time of each program on the total scan time when making system adjustments.

Function Description

Execution time measurement provides the following three functions. For explanations of each function, refer to Section 8.4.1 through Section 8.4.3.

- Program monitor list
- Interrupt program monitor list
- Scan time measurement

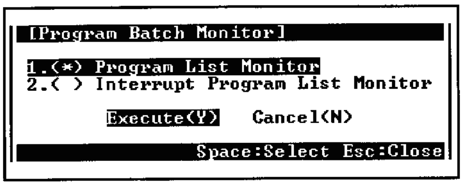
8.4.1 Program monitor list

This is a function that displays the processing time of the program being executed.

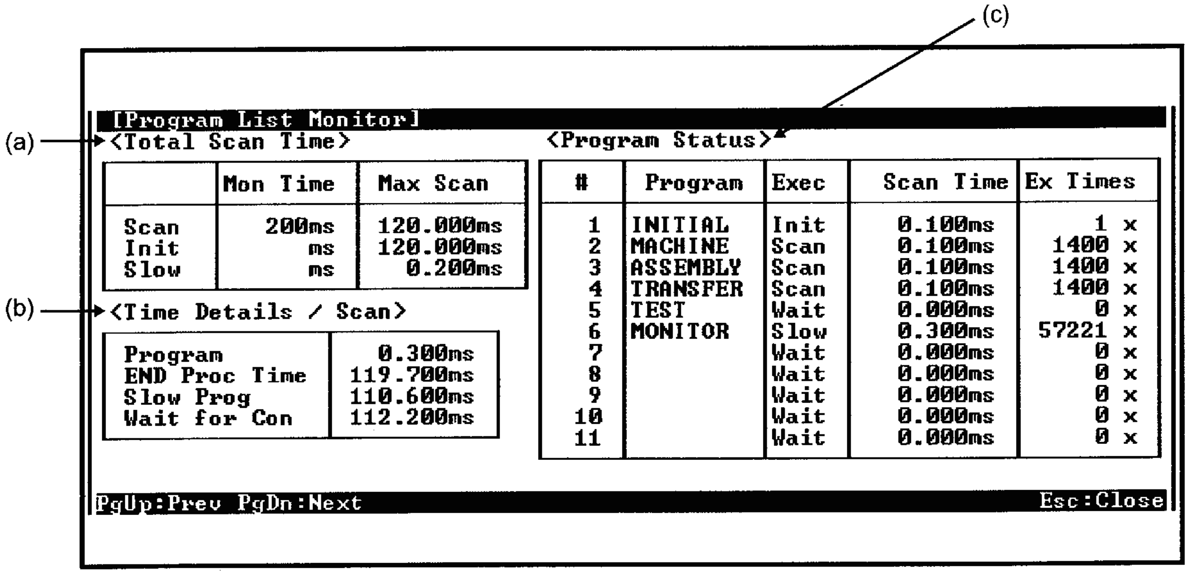
Function Description

The scan time, execution count, and processing time for each item can be displayed for each program.
All operations are performed using Monitor/test menu in the ladder mode.

(1) Select "Program Batch Monitor".



(2) Select "Program List Monitor".
The following shows an example of execution of the program list monitor when a constant scan time of 120ms is set.



[Program List Monitor]

<Total Scan Time>

	Mon Time	Max Scan
Scan	200ms	120.000ms
Init	ms	120.000ms
Slow	ms	0.200ms

<Time Details / Scan>

Program	0.300ms
END Proc Time	119.700ms
Slow Prog	110.600ms
Wait for Con	112.200ms

<Program Status>

#	Program	Exec	Scan Time	Ex Times
1	INITIAL	Init	0.100ms	1 x
2	MACHINE	Scan	0.100ms	1400 x
3	ASSEMBLY	Scan	0.100ms	1400 x
4	TRANSFER	Scan	0.100ms	1400 x
5	TEST	Wait	0.000ms	0 x
6	MONITOR	Slow	0.300ms	57221 x
7		Wait	0.000ms	0 x
8		Wait	0.000ms	0 x
9		Wait	0.000ms	0 x
10		Wait	0.000ms	0 x
11		Wait	0.000ms	0 x

PgUp:Prev PgDn:Next Esc:Close

The following shows an explanation of the screen above:

(a) "Total Scan Time"
The times set in "5.( ) PC RAS Setting" for monitor time and scan time total are displayed here for each program type.

1) "Mon Time"
The monitor times for scan execution type programs, initial execution type programs, and low-speed execution type programs are displayed here.
If the scan time exceeds the time displayed here, a watchdog timer error occurs at the CPU module.

2) "Max Scan"
The total time for the items listed under "Time Details / Scan" is displayed here.

(b) "Time Details / Scan"
The scan time details are displayed here.

1) "Program"
The total execution time of scan execution type programs is displayed here.

2) "END Proc Time"
The END processing time is displayed here.

3) "Slow Prog"
When an execution time for low-speed execution type programs is set, the total execution time for low-speed execution type programs is displayed here.

4) "Wait for Con"
When constant scan is set, the constant scan waiting time is displayed here. However, if an execution time for low-speed execution type programs is also set, 0.000 ms is displayed.

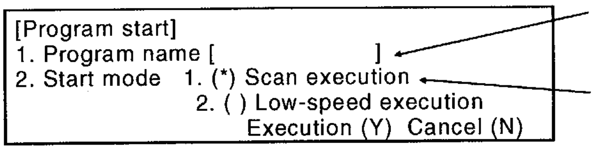
(c) "Program Status"
The times set in "9. ( ) Auxilliary Setting" in the parameter mode are displayed here.

1) "Program"
   The program names are displayed here in the order that the parameters are set.
2) "Exec"
   The types of the programs set in the parameters are displayed here.
3) "Scan Time"
   The actual scan times (current values) are displayed here. In the program stop (stand-by) status, 0.000ms is displayed as the scan time.
4) "Ex Times"
   The number of execution times is displayed here, taking the point at which measurement was started to be 0 in the count (When reaching the maximum times of 65536, the count returns to 0). The number of execution times is retained even when program execution is stopped.
   When the subroutine program/interrupt program of the stand-by program is performed in the subroutine call/interrupt request, the number of the stand-by program is not counted.

(3) Both start (F1) and stop (F2) of the program can be performed on Program monitor list screen.

(a) Start of program (F1)
Either "Scan execution" or "Low-speed execution" can be set for the stand-by program.

When the F1 key is pressed, the window below is displayed.

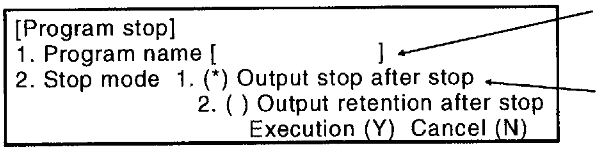


```
[Program start]
1. Program name [               ]
2. Start mode  1. (*) Scan execution
                 2. ( ) Low-speed execution
                     Execution (Y)  Cancel (N)
```

Enter the program name for start with keys.

Select either scan execution or low-speed execution.

(b) Stop of program (F2)



```
[Program stop]
1. Program name [               ]
2. Stop mode  1. (*) Output stop after stop
                2. ( ) Output retention after stop
                    Execution (Y)  Cancel (N)
```

Enter the program name for stop with keys.

Select either output stop after stop or output retention after stop.

1) When the stop mode is set to "Output stop after stop" and stop is executed, the program stops after execution of 1 scan-off. (The operation is the same as the POFF instruction execution.)
   When the above stop opearation is made in the stand-by program, the program stops after 1 scan-off execution.
   Therefore, the "execution count" is added by one.
2) If an error occurs with the RET/IRET instruction during 1 scan OFF execution in the stand-by program, the "execution count" is added by one.
   In this case, the execution type becomes "Scan execution".

8.4.2 Interrupt program monitor list

This function displays the number of executions of interrupt programs.

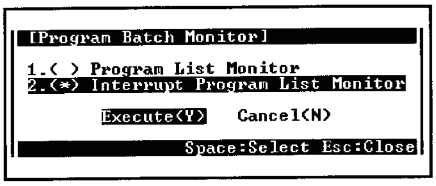
Application

This is used to check the execution status of interrupt programs.

Function Description

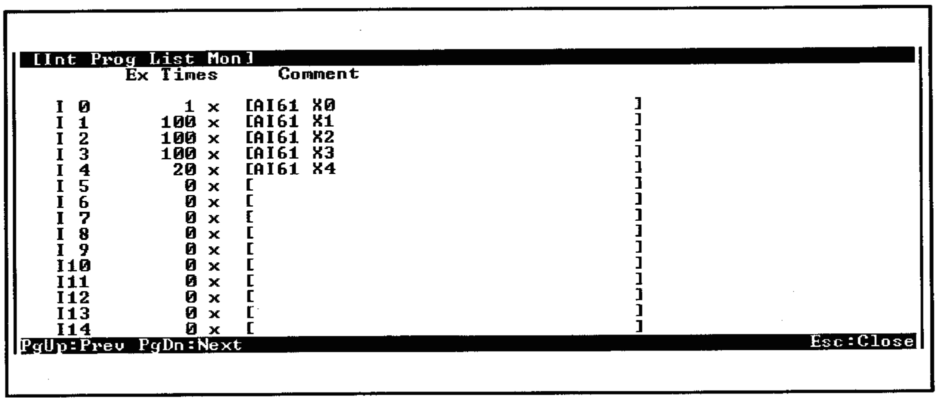
This function allows display of the execution counts of interrupt programs.
All operations are performed using the monitor/test menu in the ladder mode.

(1) Select "Program Batch Monitor".



```
[Program Batch Monitor]
1.( ) Program List Monitor
2.(*) Interrupt Program List Monitor

       Execute(Y)    Cancel(N)

                 Space:Select Esc:Close
```

(2) Select "Interrupt Program List Monitor".
The following shows an example of the display when the interrupt program monitor list function is run.



```
 [Int Prog List Mon]
          Ex Times      Comment

  I 0        1 x   [AI61  X0                          ]
  I 1      100 x   [AI61  X1                          ]
  I 2      100 x   [AI61  X2                          ]
  I 3      100 x   [AI61  X3                          ]
  I 4       20 x   [AI61  X4                          ]
  I 5        0 x   [                                  ]
  I 6        0 x   [                                  ]
  I 7        0 x   [                                  ]
  I 8        0 x   [                                  ]
  I 9        0 x   [                                  ]
  I10        0 x   [                                  ]
  I11        0 x   [                                  ]
  I12        0 x   [                                  ]
  I13        0 x   [                                  ]
  I14        0 x   [                                  ]
PgUp:Prev PgDn:Next                                          Esc:Close
```

The following shows an explanation of the screen above:

(a) "Ex Times"
The number of times the program has been executed, taking the point when monitoring started as 0 in the count, is displayed here.
(When reaching the maximum times of 65536, the count returns to 0.)
The count is cleared to zero when switching to RUN.

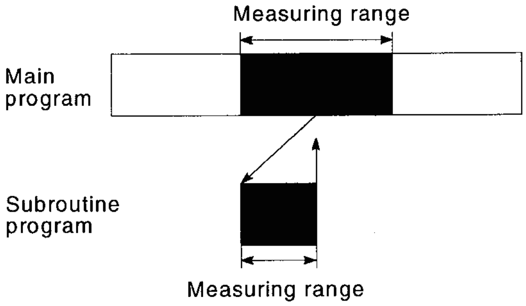
(b) "Comment"
The comments set in the documentation mode are displayed here.

8.4.3 Scan time measurement

This function displays the processing time for section of a program.

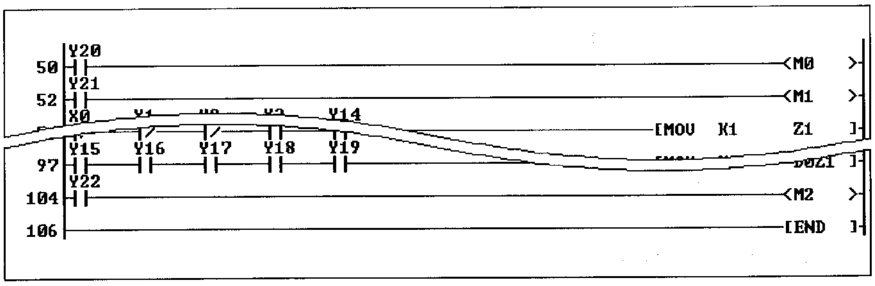
Function Description

This function allows measurement of the execution time of section of the program in a program file.
The function can also be used to measure times within subroutine programs and interrupt programs.
If there is an interrupt program in the monitored section, the processing is added to the total measurement time.

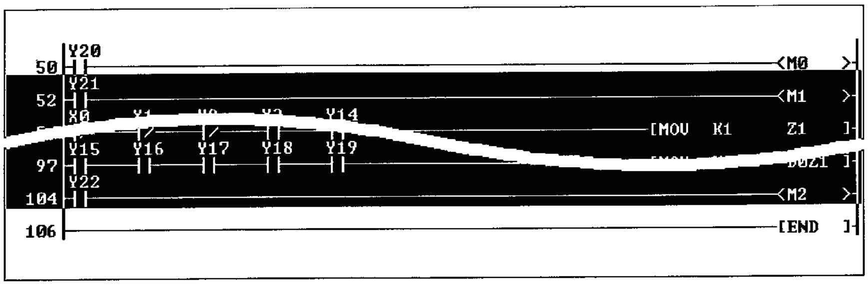


All operations are performed using "Monitor/test" menu in the ladder mode.

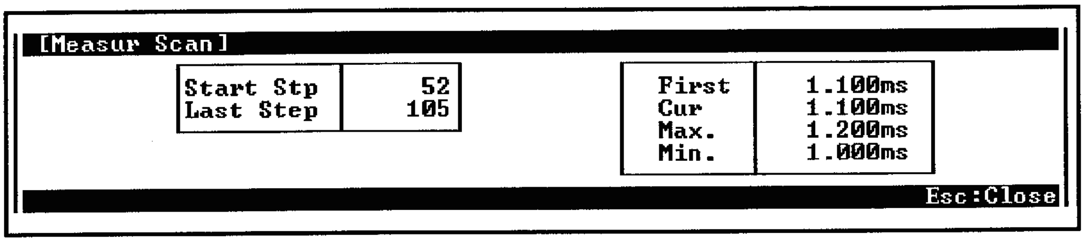
(1) Select "Measure Scan".



(2) Designate the scan time measurement range (The designated part is highlighted).



(3) The scan time measurement results are displayed.



[Measur Scan]

Start Stp	52
Last Step	105

First	1.100ms
Cur	1.100ms
Max.	1.200ms
Min.	1.000ms

NOTE

1) Make sure that the start step is lower than the end step in the setting.
2) Times that span different program files cannot be measured.
3) If the measured time is less than 0.100ms, 0.000ms is displayed.
4) END processing time is not included in the measuring time, being included in the measurement range.

8.5 Sampling Trace Function

The function that collects devices continuously on the CPU module with the specified timing.

POINT
When executing the sampling trace function, a memory card is required.

Application

This allows checking the changes in the contents of the devices used in a program in accordance with a designated timing during debugging.
This enables debugging time to be shortened.

Function Description

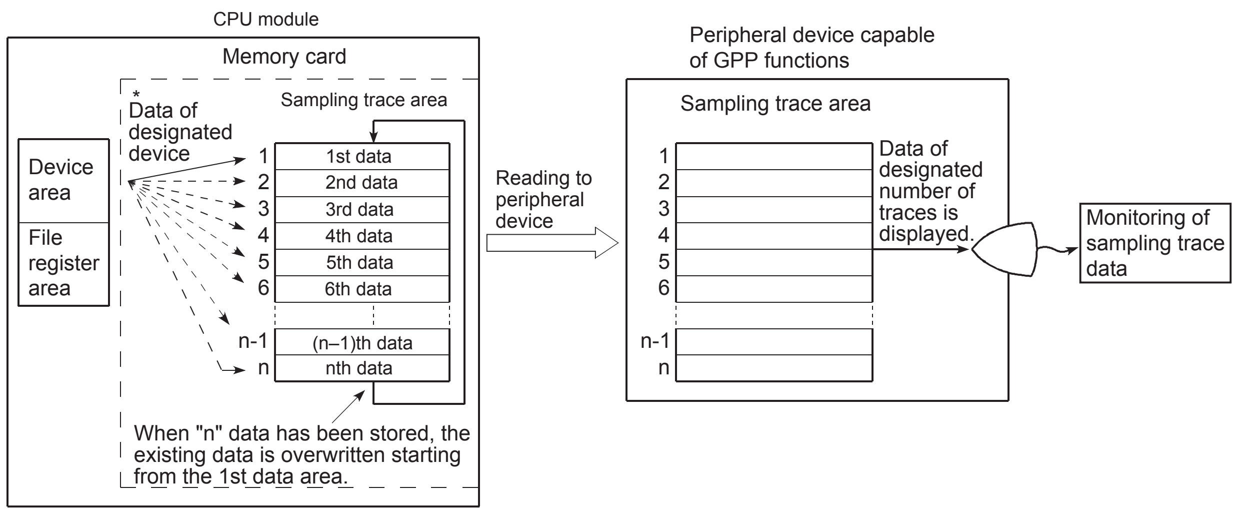
(1) Function

(a) The sampling trace function samples the contents of a designated device in a constant time interval (the sampling cycle) and stores the trace results in a sampling trace file in a memory card.

(b) The devices that can be traced are listed below.

1) Bit device:	X, FX, DX, Y, FY, DY, M, L, F, SM, V, B, SB, T (Contact), T (Coil), ST (Contact), ST(Coil), C (Contact), C (Coil), J☐\X, J☐\Y, J☐\B, J☐\SB, BL☐\S......................Max. 50 points
2) Word device:	T (Current value), ST (Current value), C (Current value), D, SD, FD, W, SW, R, Z, ZR, U☐\G, J☐\W, J☐\SW ....Max. 50 points

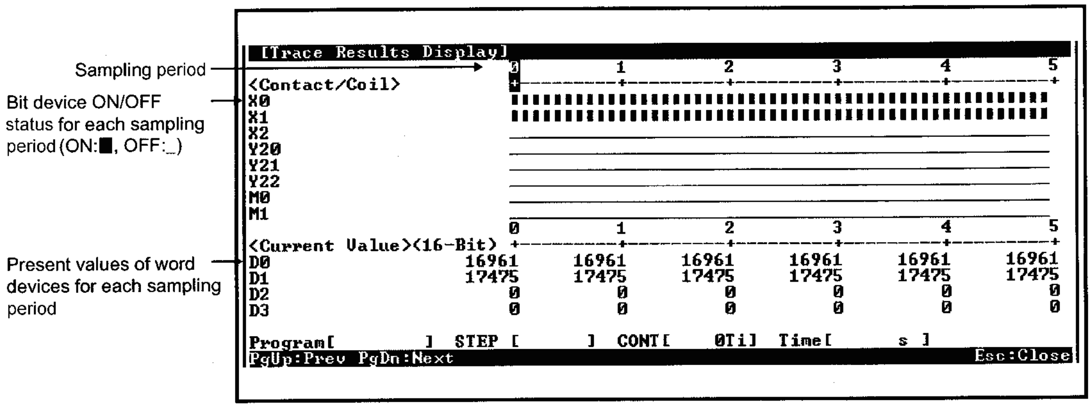
(c) The sampling trace file stores the trace condition data and trace execution data required to execute the sampling trace. Once a GPP function starts tracing, the number of set tracing times are executed.



* When the trigger point is executed, sampling is performed for the designated number of times and then the data of the sampling trace area is latched.

Fig. 8.1 Sampling trace operation

(d) The trace results show the ON/OFF statuses of bit devices, and current values of word devices, for each sampling cycle.



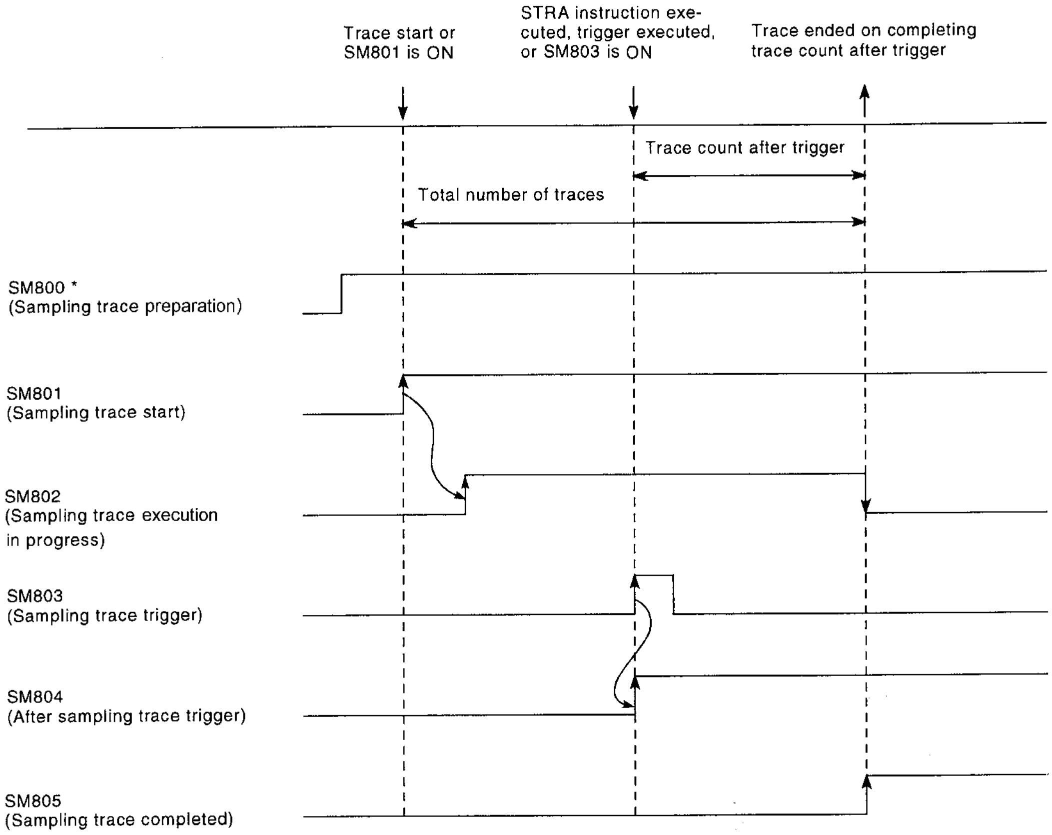
```
[Trace Results Display]
                          0         1         2         3         4         5
<Contact/Coil>            +---------+---------+---------+---------+---------+
X0
X1
X2
Y20
Y21
Y22
M0
M1
                          0         1         2         3         4         5
<Current Value><16-Bit>   +---------+---------+---------+---------+---------+
D0                    16961     16961     16961     16961     16961     16961
D1                    17475     17475     17475     17475     17475     17475
D2                        0         0         0         0         0         0
D3                        0         0         0         0         0         0

Program[        ]  STEP [       ]  CONT[    0Ti]  Time[       s ]
PgUp:Prev PgDn:Next                                            Esc:Close
```

NOTE

While the CPU module is STOP, trace is stopped. The trace result cannot be read.

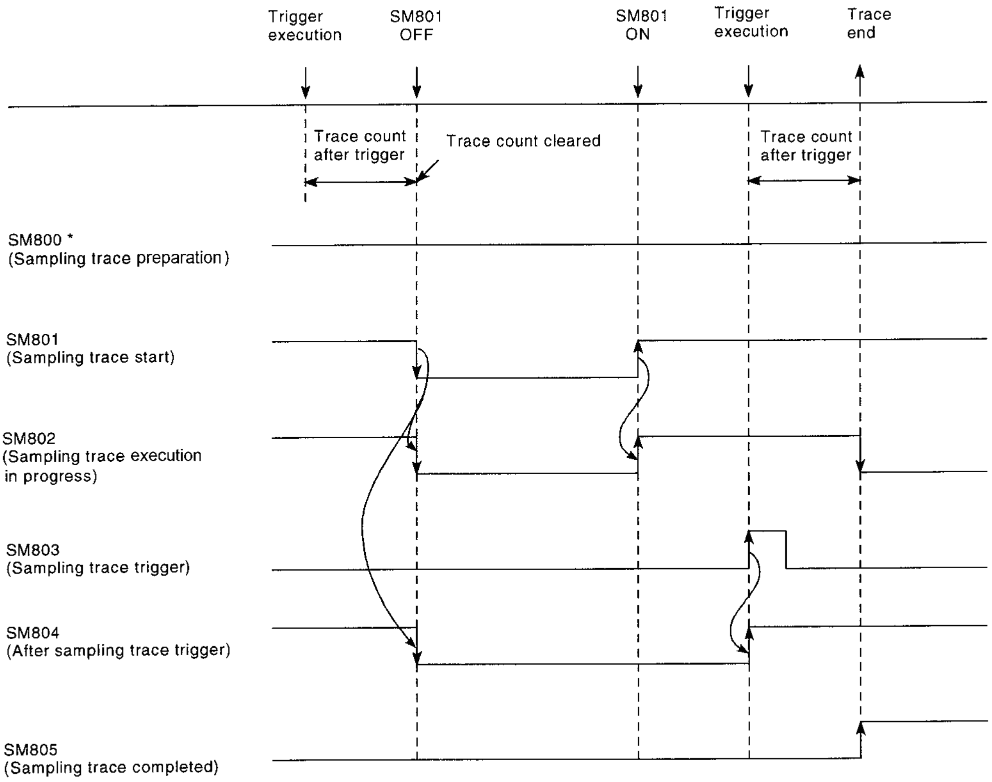
(2) Basic operation
The basic operation for sampling trace is shown below.
The statuses during execution of the sampling trace function can be confirmed by monitoring special relays SM800 to SM805 and SM826.

- Trace execution



* SM800 comes ON automatically when preparation for sampling trace is completed.

- Suspending the trace



* When the trace is suspended from a peripheral device capable of GPP functions, SM800 is turned OFF.

The following shows the operation at error occurrence.
When an error occurs during sampling trace, SM826 (sampling trace error) comes ON, and at the same time, SM801 (sampling trace start) goes OFF.
Start the trace again for turning OFF SM826.

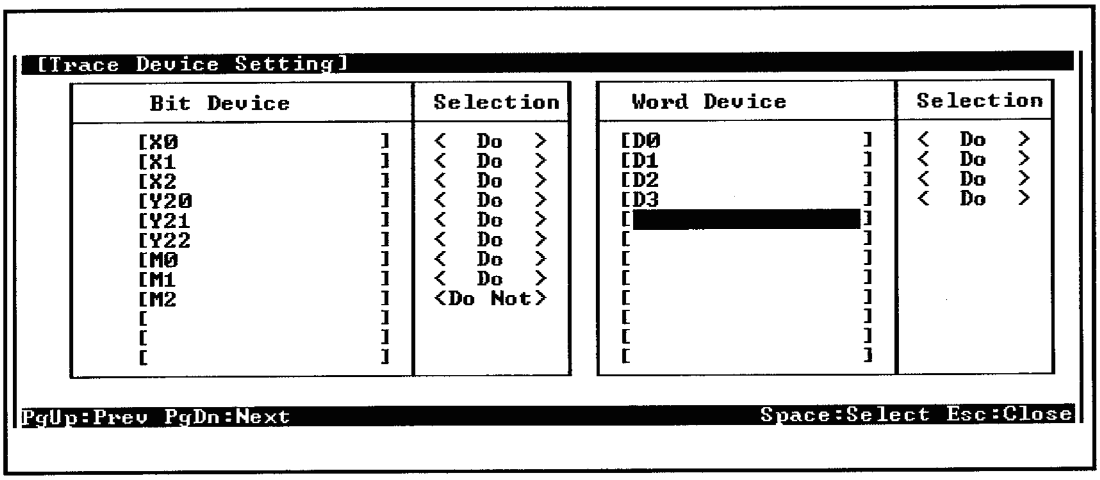
Operation Procedures

The following shows the procedures of sampling trace.
These operations are performed on the "Sampling Trace" screen of the trace menu in the online mode.

(1) Set the trace devices and trace conditions with GPP function.

  (a) Setting the trace devices

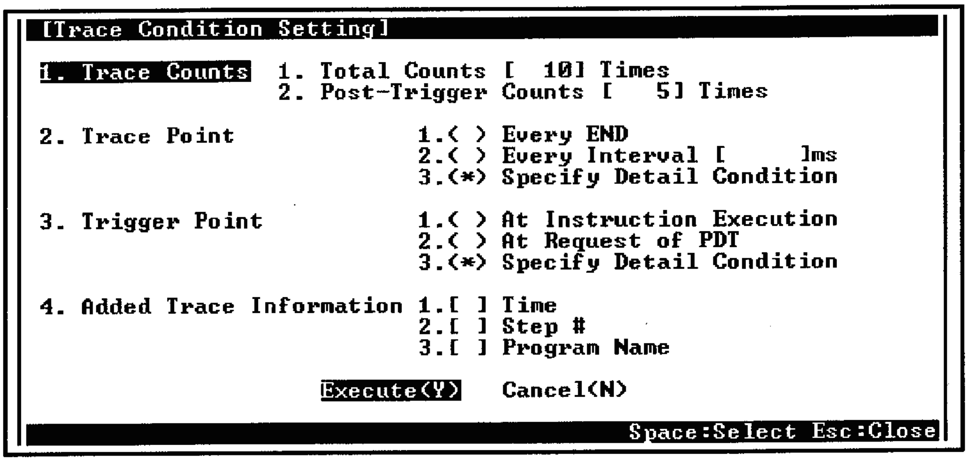
  Set the devices at "Trace Device Setting" on the "Sampling Trace" screen.



```
[Trace Device Setting]
      Bit Device          Selection      Word Device          Selection
  [X0               ]   <  Do   >     [D0               ]   <  Do   >
  [X1               ]   <  Do   >     [D1               ]   <  Do   >
  [X2               ]   <  Do   >     [D2               ]   <  Do   >
  [Y20              ]   <  Do   >     [D3               ]   <  Do   >
  [Y21              ]   <  Do   >     [                 ]
  [Y22              ]   <  Do   >     [                 ]
  [M0               ]   <  Do   >     [                 ]
  [M1               ]   <  Do   >     [                 ]
  [M2               ]   <Do Not>      [                 ]
  [                 ]                 [                 ]
  [                 ]                 [                 ]
  [                 ]                 [                 ]
PgUp:Prev PgDn:Next                            Space:Select Esc:Close
```

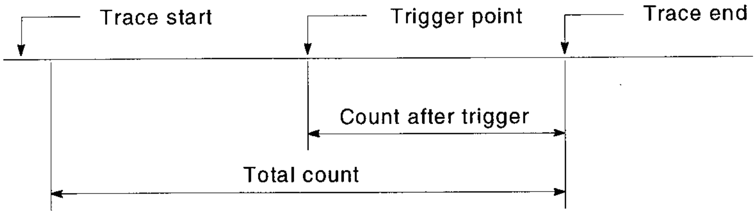
(b) Setting the trace conditions

Set the trace conditions at "Trace Device Setting" on the "Sampling Trace" screen.



```
[Trace Condition Setting]

1. Trace Counts  1. Total Counts [  10] Times
                 2. Post-Trigger Counts [   5] Times

2. Trace Point              1.( ) Every END
                            2.( ) Every Interval [      ]ms
                            3.(*) Specify Detail Condition

3. Trigger Point            1.( ) At Instruction Execution
                            2.( ) At Request of PDT
                            3.(*) Specify Detail Condition

4. Added Trace Information 1.[ ] Time
                            2.[ ] Step #
                            3.[ ] Program Name

                       Execute(Y)   Cancel(N)

                                        Space:Select Esc:Close
```

Sampling the designated number of times (count after trigger) leads completion after the trigger point execution.



The following shows an explanation of the screen above:

One of the following four settings can be made for the trace condition: "1. Trace Counts", "2. Trace Point", "3. Trigger Point", or "4. Added Trace Information".

1) "Trace Counts"

In the case of the total count, set the number of sampling traces executed from start to end of the trace.

In the case of the count after the trigger, set the number of sampling traces executed from the trigger execution to the trace end.

The following shows the formula that sets range for these counts:

Count after trigger≦total count≦8192

2) "Trace Point"

Set the timing for collection of trace data. Select one of the following:

ⓐ Every END : Data collected at END instruction of every scan.

ⓑ Every Interval : Data collected with each designated time. Setting range is 5 to 5000ms in 5ms units.

ⓒ Specify Detail Condition : Set a device and step number. The following shows setting examples: The details on how to make the settings and data collection timing are the same as described in Section 8.2.1 Monitor condition setup in Monitor Function.

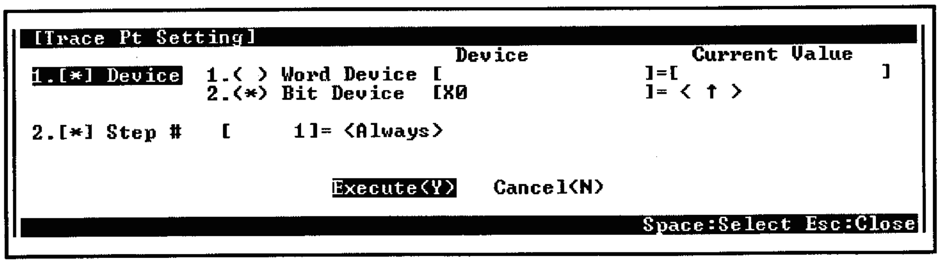
The following shows the setting device under the detailed condition.

ⓐ Bit device : X, FX, Y, FY, M, L, F, SM, V, B, SB, T (Contact), ST (Contact), C (Contact), J□\X, J□\Y, J□\B, J□\SB, BL□\S

ⓑ Word device : T (Current value), ST (Current value), C (Current value), D, SD, FD, W, SW, R, Z, ZR, U□\G, J□\W, J□\SW

The following qualifications are possible with respect to the devices listed above.

- Digit designation for bit devices
- Bit number designation for word devices



```
[Trace Pt Setting]
                                          Device                    Current Value
1.[*] Device    1.( ) Word Device [                        ]=[                     ]
                2.(*) Bit Device  [X0                      ]= < ↑ >

2.[*] Step #    [      1]= <Always>

                            Execute(Y)    Cancel(N)

                                                          Space:Select Esc:Close
```

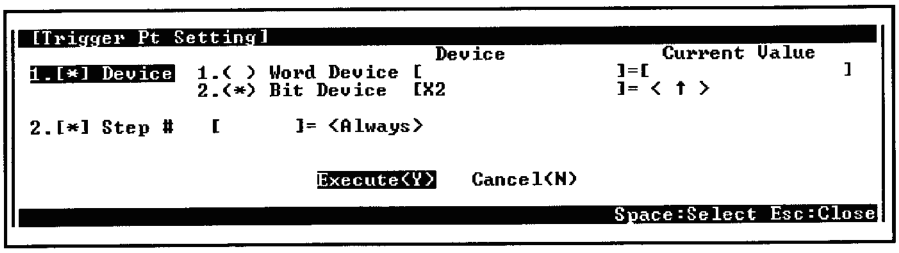
3) "Trigger Point"

The point at which the trigger is executed is set. Select one of the following:

ⓐ At Instruction Execution : When executing STRA instruction

ⓑ At Request of PDT : When operating trigger using GPP functions

ⓒ Specify Detail Condition : Set a device and step number.
The following shows setting examples: The details on how to make the settings and trigger execution timing are the same as described in Section 8.2 Monitor condition setup in Monitor Function.



```
[Trigger Pt Setting]
                                    Device            Current Value
1.[*] Device   1.( ) Word Device [                   ]=[                  ]
               2.(*) Bit Device  [X2                 ]= < ↑ >

2.[*] Step #   [        ]= <Always>

                      Execute<Y>   Cancel<N>

                                               Space:Select Esc:Close
```

4) "Added Trace Information"

Set information to be added at each trace. Select one or multiple item(s) of the following: (Making no selection is possible.)

ⓐ Time : The time at which the trace was executed is stored.

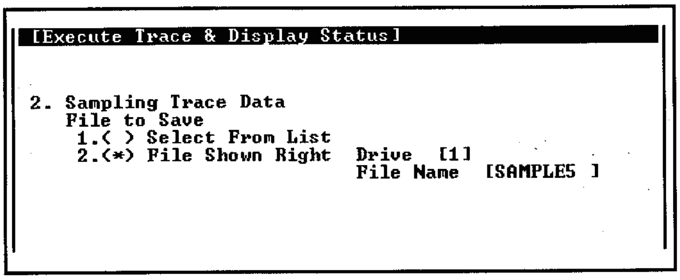
ⓑ Step No. : The step number at which the trace was executed is stored.

ⓒ Program Name : The program name for which the trace was executed is stored.

(2) Write the set trace device and trace condition to the memory card.

(a) Set the trace file and storage destination.

Set the drive number and file name at "1. ( ) Execute Trace & Display Status" on the "Sampling Trace" screen.



```
[Execute Trace & Display Status]

2. Sampling Trace Data
   File to Save
    1.( ) Select From List
    2.(*) File Shown Right  Drive    [1]
                            File Name  [SAMPLE5 ]
```

(b) Write the trace file to the memory card.

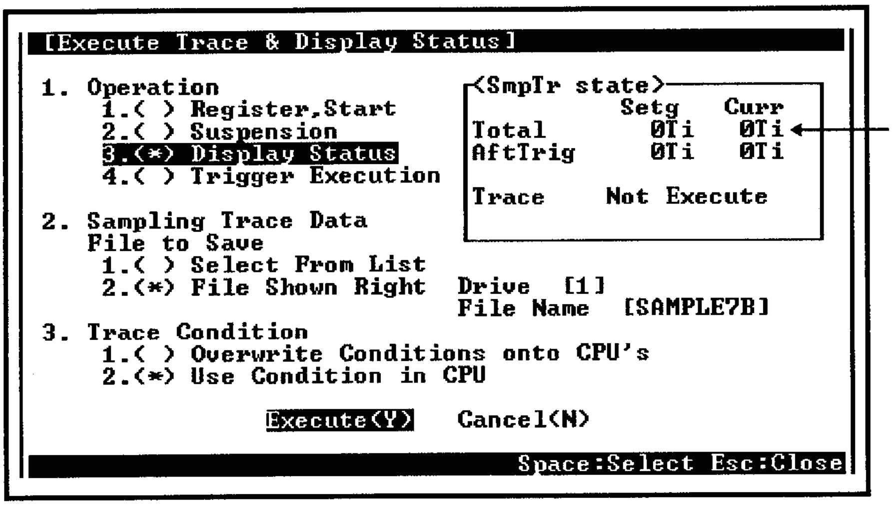
Write the trace file to the memory card by using "9. ( ) Write to PC (Condition)" on the "Sampling Trace" screen.

Since file names are used when writing to the memory card, multiple trace files can be written.

(3) Execute the sampling trace.

Execute the sampling trace by using "1. ( ) Execute Trace & Display Status" on "Sampling Trace" screen.

The following shows a setting example for "1. ( ) Execute Trace & Display Status".



```
[Execute Trace & Display Status]

1. Operation                  <SmpTr state>
   1.( ) Register,Start                 Setg   Curr
   2.( ) Suspension            Total      0Ti    0Ti
   3.(*) Display Status        AftTrig    0Ti    0Ti
   4.( ) Trigger Execution
                               Trace     Not Execute
2. Sampling Trace Data
   File to Save
   1.( ) Select From List
   2.(*) File Shown Right   Drive   [1]
                            File Name   [SAMPLE7B]
3. Trace Condition
   1.( ) Overwrite Conditions onto CPU's
   2.(*) Use Condition in CPU

               Execute(Y)    Cancel(N)

                              Space:Select Esc:Close
```

Displayed only when "Display Status" is selected.

The following shows an explanation of the screen above:

The following settings can be made for "Execute Trace & Display Status" : "1. Operation", "2. Sampling Trace Data", and "3. Trace Condition".

(a) "Operation"

Select one of the following:

1) Register, Start : The trace is registered and started. The trace count is started.

2) Suspension : The trace is suspended. The trace count and the count are cleared after the trigger. (To restart the trace, select "Register, Start" again.)

3) Display Status : The trace statuses are displayed on the same screen.

4) Trigger Execution : The count is started after the trigger.
The trace is ended on reaching the designated count after the trigger.

(b) "Sampling Trace Data"

Select one of the following:

1) Select From List : Data from among the sampling trace files in the memory card are selected.

2) File Shown Right : The drive number and sampling trace file name are set.

(c) "Trace Condition"

Select one of the following:

1) Overwrite Conditions onto CPU's : The trace condition in an existing trace file is overwritten.

2) Use Condition in CPU : Sampling trace under the condition in the trace file designated in "2. Sampling Trace Data" is executed.

(4) Retrieve the trace results from the CPU module and display them.

1) Read the trace results from the CPU module by using "4. ( ) Read from PC (Results)" on the "Sampling Trace" screen.
2) Display the trace results by using "4. ( ) Trace Results Display" on the "Sampling Trace" screen.

POINT

Once the sampling trace has been executed, the second is not executed. To execute the trace again, execute the STRAR instruction to reset sampling trace.

NOTE

1) Set sampling trace files in the RAM area of the memory card.
2) It is possible to execute sampling trace from another station in the network, or from a serial communication module. However, sampling trace cannot be executed from more than one site at the same time. With the Q2ASCPU, sampling trace can be executed from only one site at a time.
3) Since the trace condition registered in the CPU module is latched, the condition data is retained even when the programmable controller power is turned OFF. The data can be cleared by performing a latch clear operation using the RUN/STOP key switch on the Q2ASCPU.
4) The Q2ASCPU must be connected to the peripheral device capable of GPP functions in order to execute sampling trace.

8.6 Status Latch Function

This function collects the data of devices at designated moment.

POINT

When executing status latch function, a memory card is required.

Application

This function is used to retain the statuses of devices used in a program at designated moment during debugging.

Function Description

(1) Function

(a) Status latch stores the device statuses at designated moment in a status latch file of a memory card.

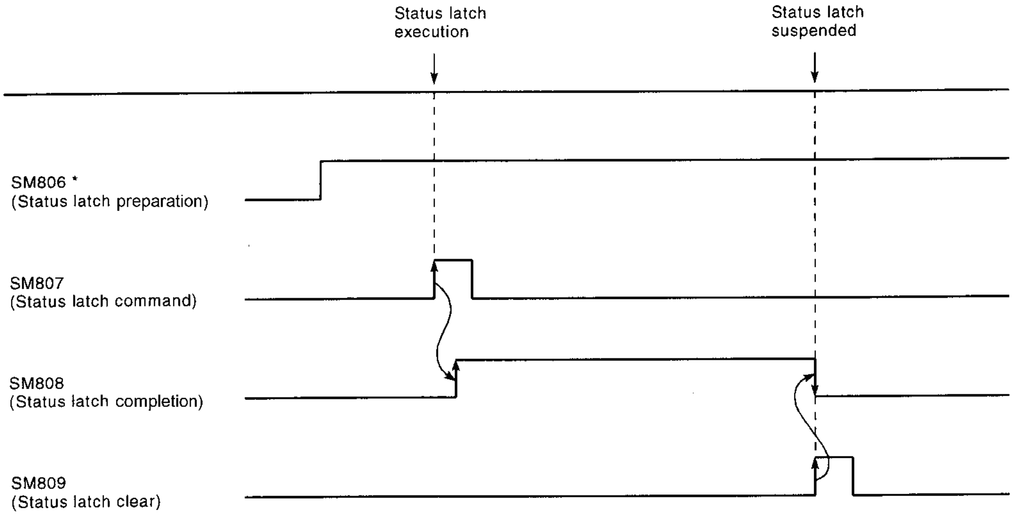
(b) The status latch file stores the status latch condition and status latch execution data for status latch execution.
Saving the device statuses can be executed in the following case.
- When executing SLT instruction in a program
- When specifying a status latch start at GPP functions
- When the conditions of the set devices and step Nos. are met

(c) The status latch results show the bit device ON/OFF statuses and word devices values at designated moment.

(2) Basic operation

The following shows the basic operation for status latch.

The statuses during execution of the status latch function can be checked by monitoring special relays SM806 to SM809 and SM827.



* SM806 comes ON automatically when preparation for status latch is completed.

(3) The following shows the operation at error occurrence.

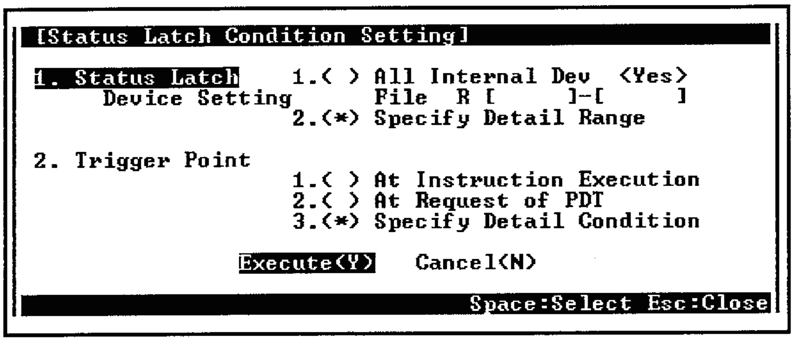
When an error occurs during status latch, SM827 comes ON, and at the same time SM808 (completed) is turned ON.

To turn SM827 OFF, either turn SM809 ON or execute the SLTR instruction.

Operation Procedures

The following shows procedures for status latch.
All operations are performed on the "Status Latch" screen of the trace menu in the online mode.

(1) Setting the status latch condition
Set the status latch condition at "2. ( ) Status Latch Condition Setting" on the "Status Latch" screen.



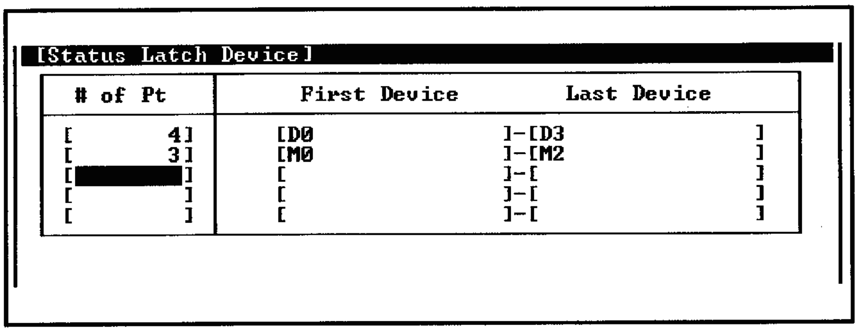
```
[Status Latch Condition Setting]

1. Status Latch     1.( ) All Internal Dev   <Yes>
     Device Setting       File  R [      ]-[      ]
                    2.(*) Specify Detail Range

2. Trigger Point
                    1.( ) At Instruction Execution
                    2.( ) At Request of PDT
                    3.(*) Specify Detail Condition

                Execute(Y)    Cancel(N)

                                  Space:Select Esc:Close
```

The following shows an explanation of the screen above:
Either "1. Status Latch Device Setting" or "2. Trigger Point" can be set for the status latch condition setting.

(a) "Status Latch Device Setting"
Set the devices to execute the status latch. Select one of the following:

1) All Internal Dev : Whether or not QnACPU latches all built-in devices is set.

2) Specify Detail Range : The device types and numbers of points are set. The following shows setting examples:



```
[Status Latch Device]
 # of Pt       First Device          Last Device
[      4]     [D0              ]-[D3              ]
[      3]     [M0              ]-[M2              ]
[       ]     [                ]-[                ]
[       ]     [                ]-[                ]
[       ]     [                ]-[                ]
```

(Applicable devices)

1) Bit device : X, Y, M, L, F, SM, V, B, SB, T (Contact), T (Coil), C (Contact), C (Coil), ST (Contact), ST (Coil), J□\X, J□\Y, J□\B, J□\SB, BL□\S

2) Word device : T (Current value), ST (Current value), C (Current value), D, SD, W, SW, R, ZR, U□\G, J□\W, J□\SW

REMARK

Up to 1000 device ranges can be set including both bit devices and word devices.
The devices listed above cannot be qualified.

(b) "Trigger Point"

Set the condition to execute the status latch. Select one of the following:

1) At Instruction Execution : When executing SLT instruction

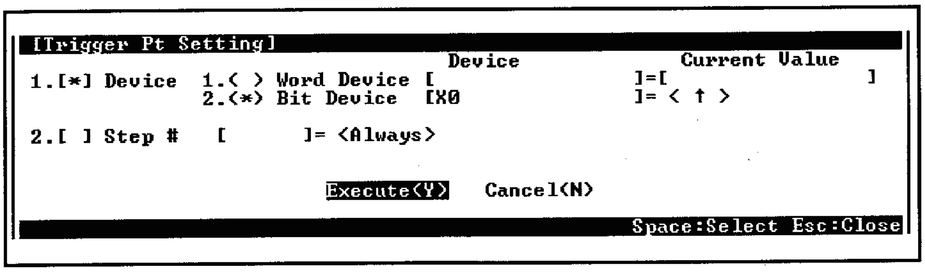
2) At Request of PDT : When operating trigger using the peripheral devices capable of GPP function.

3) Specify Detail Condition : Set a device and step number.

The following shows setting examples: The details on how to make the settings and trigger execution timing are the same as described in Section 8.2.1 Monitor condition setting in Monitor function.

Data collection timing

- In the case that only "Device" is specified, data are collected when the trigger condition is satisfied.
- In the case that "Step No." only is set, data is collected with the END processing when the trigger condition is satisfied.

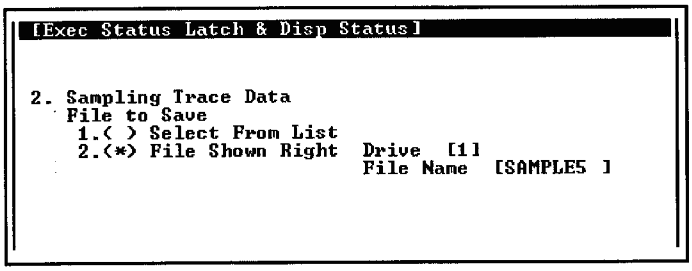


```
[Trigger Pt Setting]
                                    Device              Current Value
1.[*] Device  1.( ) Word Device [                    ]=[                 ]
              2.(*) Bit Device  [X0                  ]= < ↑ >

2.[ ] Step #   [        ]= <Always>

                         Execute<Y>   Cancel<N>
                                          Space:Select Esc:Close
```

(2) Write the created status latch condition to the memory card.

(a) Set the status latch file and storage destination.

Set the status latch condition at "1. ( ) Exec Status Latch & Disp Status" on the "Status Latch" screen.



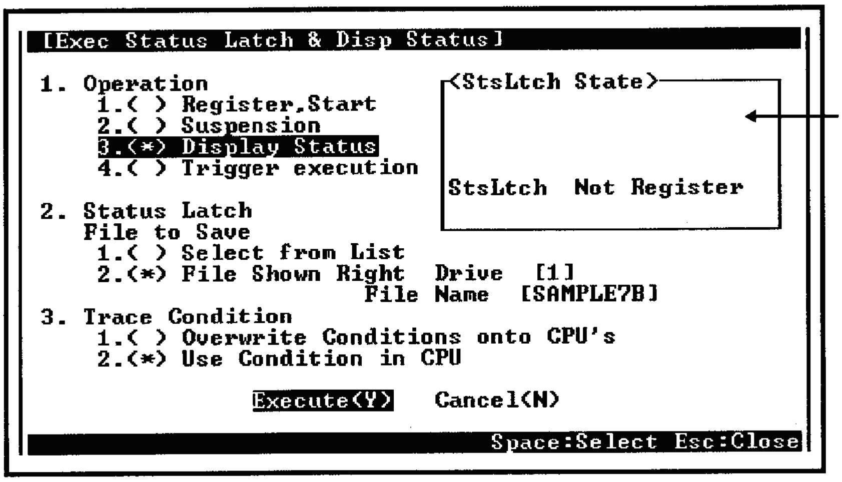
```
[Exec Status Latch & Disp Status]

2. Sampling Trace Data
   File to Save
    1.( ) Select From List
    2.(*) File Shown Right  Drive     [1]
                            File Name [SAMPLE5 ]
```

(b) Write the status latch file to the memory card.

Write the status latch file to the memory card using "7. ( ) Write to PC (Condition)" on "Status Latch" screen.

Since file names are used when writing to the memory card, multiple status latch files can be written.

(3) Execute the status latch.
Execute the status latch by using "1. ( ) Exec Status Latch & Disp Status" on "Status Latch" screen.
The following shows a setting example for "1. ( ) Execute Status Latch & Display Status".



```
[Exec Status Latch & Disp Status]

1. Operation                       <StsLtch State>
    1.( ) Register.Start
    2.( ) Suspension
    3.(*) Display Status
    4.( ) Trigger execution
                                   StsLtch  Not Register
2. Status Latch
   File to Save
    1.( ) Select from List
    2.(*) File Shown Right  Drive   [1]
                            File Name  [SAMPLE7B]
3. Trace Condition
    1.( ) Overwrite Conditions onto CPU's
    2.(*) Use Condition in CPU

              Execute(Y)   Cancel(N)

                               Space:Select Esc:Close
```

Displayed only when "Display Status" is selected.

The following is an explanation of the screen above:
The following settings can be made for "Exec Status Latch & Disp Status": "1 Operation", "2 Status Latch", and "3 Trace Condition".

(a) "Operation"
Select one of the following:

1) Register, Start : The status latch is registered and started. Device data collection is started.
2) Suspension : The status latch statuses are cleared.
3) Display Status : The status latch statuses are displayed on the same screen.
4) Trigger execution : The trigger is executed. (Refer to Precaution 6).)

(b) "Status Latch"
Select one of the following:

1) Select From List : Data from among the status latch files in the memory card are selected.
2) File Shown Right : The drive number and status latch file name are set.

(c) "Trace Condition"
Select one of the following:

1) Overwrite Conditions onto CPU's : The status latch condition in an existing status latch file is overwritten.
2) Use Condition in CPU : Status latch under the condition in the status latch file designated in "2. Status Latch" is executed.

(4) Retrieve the status latch results from the CPU module and display them.

  (a) Read the status latch results from the CPU module by using "8. ( ) Read from PC (Results)" on "Status Latch" screen.

  (b) Display the read trace results by setting "1. ( ) Monitor Target" on the "Monitor Target Setting" screen of "Option" menu in the ladder mode to "3. ( ) Status Latch".

NOTE

1) Set status latch files in the RAM area of the memory card.
2) It is possible to execute status latch from another station in the network, or from a serial communication module. However, sampling trace cannot be executed from more than one site at the same time.
   With Q2ASCPU, sampling trace can be executed from only one site at a time.
3) Since the status latch conditions registered in the CPU module are latched, the status latch data is retained even when the power is turned OFF.
   The data can be cleared by performing a latch clear operation using the RUN/ STOP key switch on the Q2ASCPU.
4) Status latch is performed by connecting the Q2ASCPU with the peripheral devices capable of GPP function.
5) When the monitor destination is set to the "status latch", set values of the timer/ counter are not displayed.
   "0" is displayed for the column of the timer/counter set values.
6) When "device" is specified in the detailed condition for trigger point setting, "device" is specified. When the condition is satisfied before execution of the trigger, trigger cannot be executed.

REMARK

1) When the monitor destination is set to "device memory", the set values of the timer/counter are not displayed.
   "0" is displayed in the set value column of the timer/counter.

8.7 Step Operation

This function runs one step or one part of a program, runs a program with a part skipped.

Application

This function is used to determine the causes of faults during debugging.

Function Description

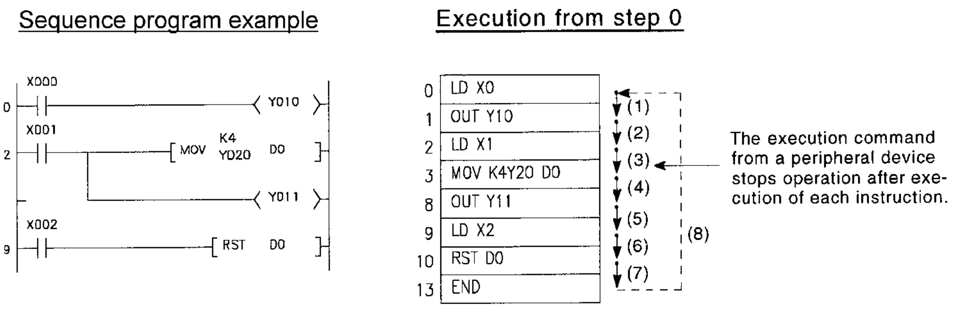
This function can only be used when the CPU module is set to STEP-RUN.
The step operation function provides the following three functions. For explanations of each function, refer to Section 8.7.1 through Section 8.7.3.

- Step execution
- Partial execution
- Skip execution

8.7.1 Step execution

Step execution is a sequence program execution that performs by one step at a time, starting from the designated step.
It allows a sequence program execution while checking an execution status of the sequence program and the contents of each device during debugging.
There are two types of step execution as described below:

(1) Step execution for one instruction
Instructions are executed one for each step starting from the step where program operation is stopped. Program operation is stopped again after execution of each instruction.
This method is used to confirm the status of each devices after execution of one instruction.



The execution command from a peripheral device stops operation after execution of each instruction.

Fig. 8.2 Step execution for each instruction

(2) Step execution with designated loop count
Program execution is repeated for the designated loop count (range: 1 to 32767) beginning with step 0 or the step where program operation was last stopped, and is stopped at the designated step (break point).

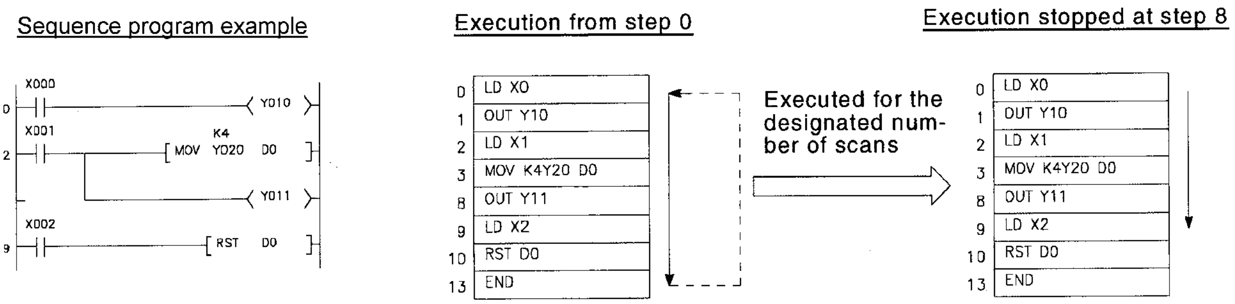
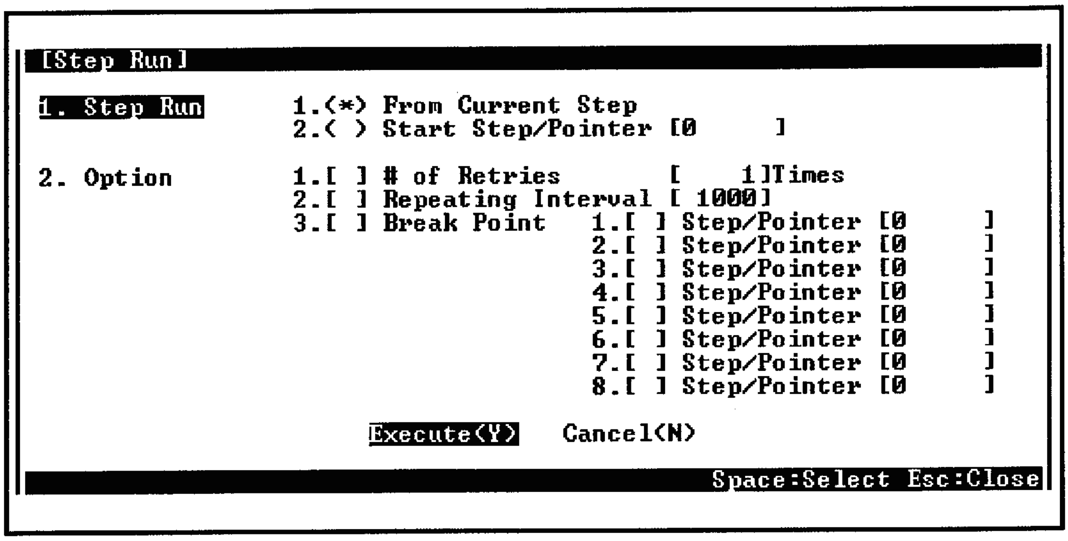


Fig. 8.3 Step execution with designated loop count

Operation Procedures

The following shows the procedures to perform step execution.
All operations are performed on Monitor/test screen in the ladder mode (debugging).

(1) Select "B/Step Run".



```
[Step Run]

1. Step Run      1.(*) From Current Step
                 2.( ) Start Step/Pointer [0     ]

2. Option        1.[ ] # of Retries        [    1]Times
                 2.[ ] Repeating Interval [ 1000]
                 3.[ ] Break Point   1.[ ] Step/Pointer [0     ]
                                     2.[ ] Step/Pointer [0     ]
                                     3.[ ] Step/Pointer [0     ]
                                     4.[ ] Step/Pointer [0     ]
                                     5.[ ] Step/Pointer [0     ]
                                     6.[ ] Step/Pointer [0     ]
                                     7.[ ] Step/Pointer [0     ]
                                     8.[ ] Step/Pointer [0     ]

                     Execute(Y)   Cancel(N)

                                        Space:Select Esc:Close
```

8.7.2 Partial execution

The sequence program is executed from the start step or the step where operation is currently stopped to a designated step (break point).

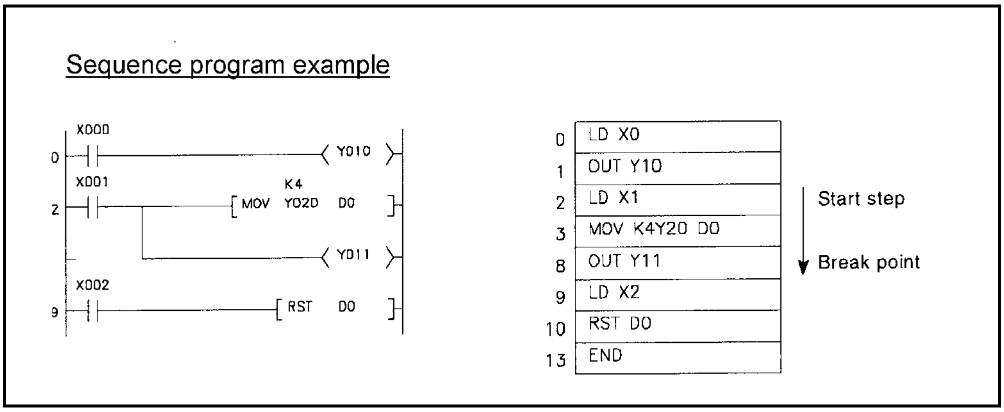


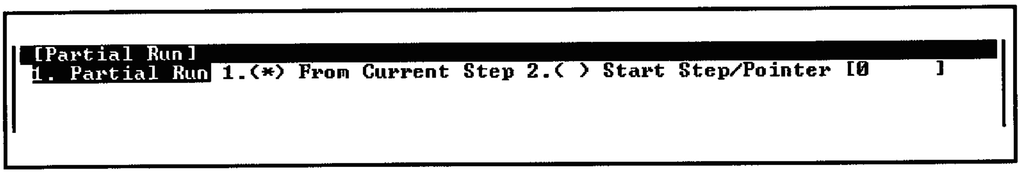
Fig. 8.4 Partial execution

Operation Procedures

The following shows the procedures to perform partial execution.
All operations are performed on Monitor/test screen in the ladder mode (debugging).

(1) Designate the execution start step, break condition, and execution operation with GPP function.

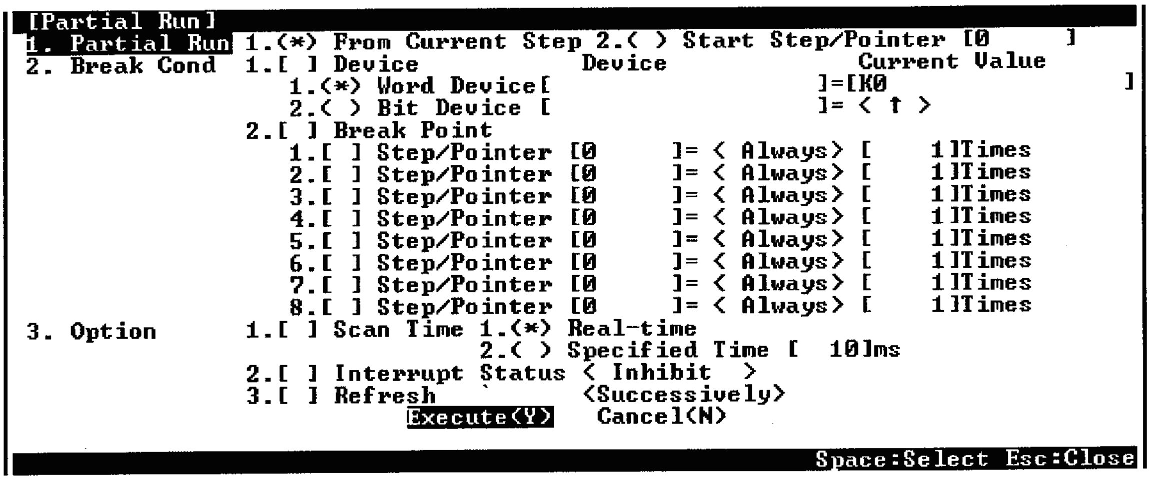
(a) Setting the execution start step
Designate the step at which partial execution is started at "1. Partial Run" on the "Partial Run" screen.



```
[Partial Run]
1. Partial Run 1.(*) From Current Step 2.( ) Start Step/Pointer [0       ]
```

(b) Setting the break condition

Set the device status and break point at "2. Break Cond" on the "Partial Run" screen.



```
[Partial Run]
1. Partial Run 1.(*) From Current Step 2.( ) Start Step/Pointer [0      ]
2. Break Cond   1.[ ] Device           Device           Current Value
                   1.(*) Word Device[                    ]=[K0              ]
                   2.( ) Bit Device [                    ]= < ↑ >
                2.[ ] Break Point
                   1.[ ] Step/Pointer [0      ]= < Always> [     1]Times
                   2.[ ] Step/Pointer [0      ]= < Always> [     1]Times
                   3.[ ] Step/Pointer [0      ]= < Always> [     1]Times
                   4.[ ] Step/Pointer [0      ]= < Always> [     1]Times
                   5.[ ] Step/Pointer [0      ]= < Always> [     1]Times
                   6.[ ] Step/Pointer [0      ]= < Always> [     1]Times
                   7.[ ] Step/Pointer [0      ]= < Always> [     1]Times
                   8.[ ] Step/Pointer [0      ]= < Always> [     1]Times
3. Option       1.[ ] Scan Time 1.(*) Real-time
                                2.( ) Specified Time [  10]ms
                2.[ ] Interrupt Status < Inhibit  >
                3.[ ] Refresh          <Successively>
                          Execute(Y)   Cancel(N)
                                                      Space:Select Esc:Close
```

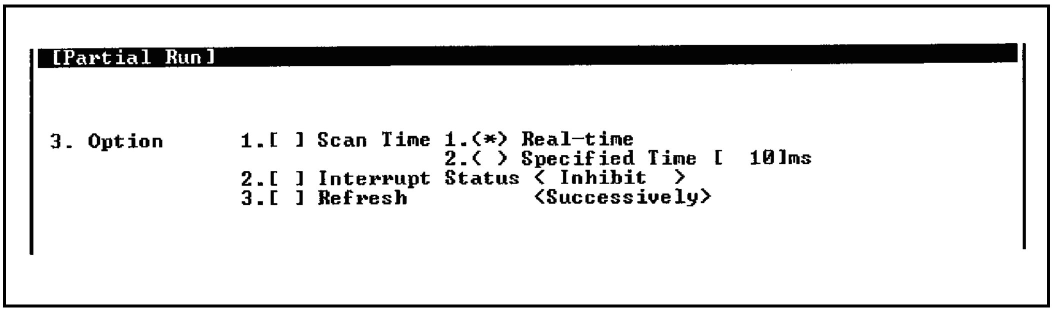
The following shows the devices that can be set.

1) Bit device : X, FX, DX, Y, FY, DY, M, L, F, SM, V, B, SB, T (Contact), T (Coil), ST (Contact), ST (Coil), C (Contact), C (Coil), J□\X, J□\Y, J□\B, J□\SB, BL□\S

2) Word device : T (Current value), ST (Current value), C (Current value), D, SD, FD, W, SW, R, Z, ZR, U□\G, J□\W, J□\SW

(c) Setting the execution operation

Set the scan time, interrupt status, and refresh, at "3. Option" on "Partial Run" screen.



```
[Partial Run]

3. Option      1.[ ] Scan Time 1.<*> Real-time
                               2.< > Specified Time [  10]ms
               2.[ ] Interrupt Status < Inhibit  >
               3.[ ] Refresh          <Successively>
```

*Multiple setting can be made.

The following shows all settings.

Item	Description
Scan time	Designates whether QnACPU executes the scan time by the actual time or by the designated time. (Default: designated time 10ms)
Interrupt status	Designates whether or not interrupts are prohibited during execution. (Default: "Inhibit")
Refresh	Designates whether QnACPU executes I/O refresh whenever program execution is stopped due to satisfaction of a condition, or executes only at END processing. (Default: "Successively")

8.7.3 Skip function

Skip execution or partial execution of a program whereby the program is executed with the designated step(s) skipped.

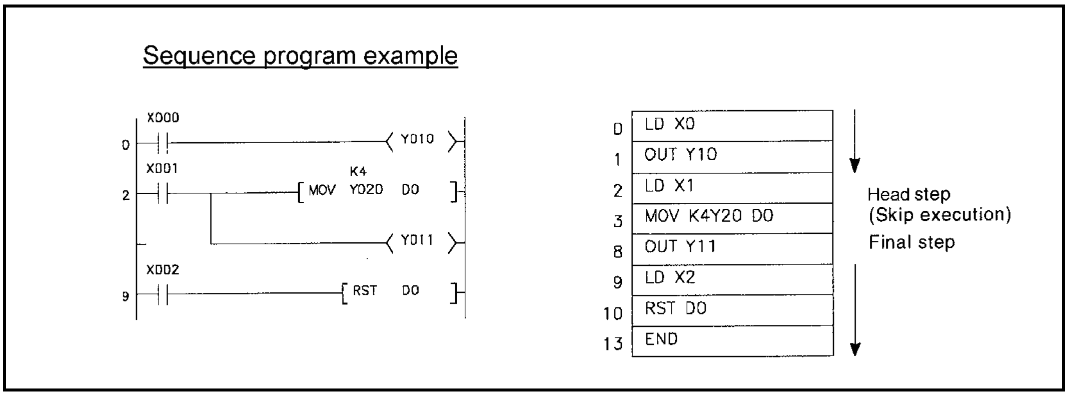
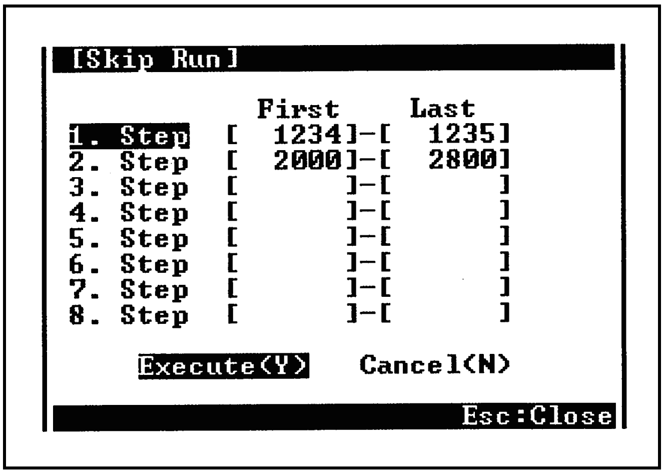


Fig. 8.5 Skip execution

Operation Procedures

The following shows the procedures to perform skip execution.
All operations are performed on Monitor/test screen in the ladder mode (debugging).

(1) Set the program range to be skipped using GPP function.
Designate the step number(s) to be skipped on "D/Skip Run" screen.



```
[Skip Run]

              First      Last
1. Step   [   1234]-[   1235]
2. Step   [   2000]-[   2800]
3. Step   [       ]-[       ]
4. Step   [       ]-[       ]
5. Step   [       ]-[       ]
6. Step   [       ]-[       ]
7. Step   [       ]-[       ]
8. Step   [       ]-[       ]

     Execute(Y)    Cancel(N)

                        Esc:Close
```

8.8 Program Trace Function

This function collects program execution statuses.

POINT
When executing the program trace function, a memory card is required.

Application

This function is used to check the execution status of any step of any program during debugging.
This enables debugging time to shorten.

Function Description

(1) Function

(a) The program trace function collects the execution status of the designated step of the designated program and stores it in a program trace file in the memory card.

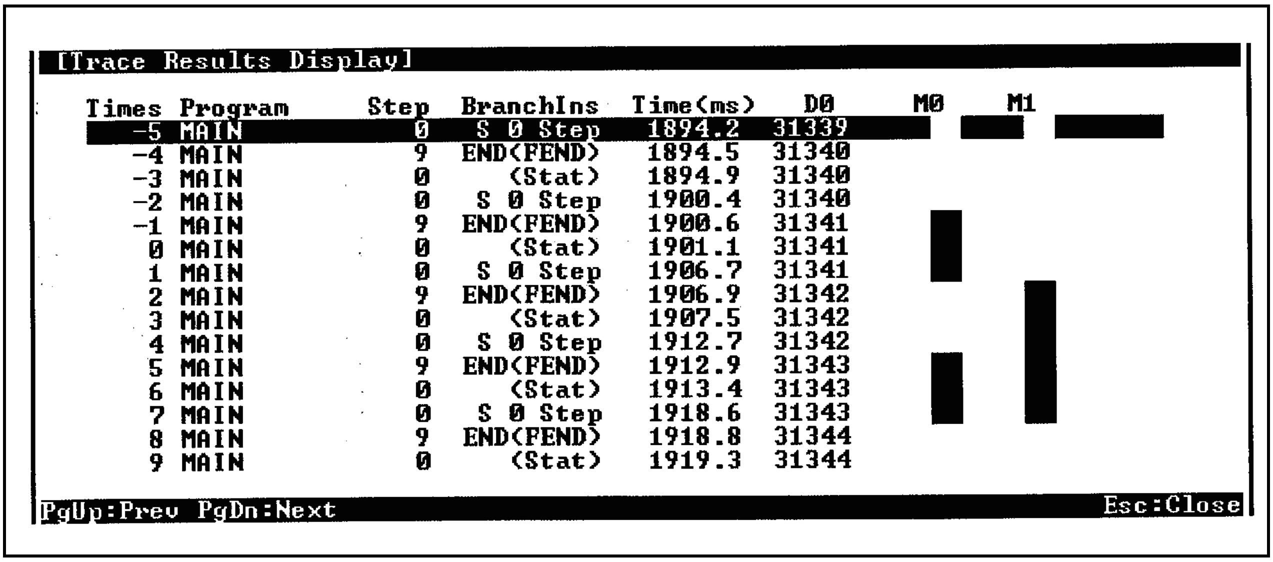
(b) The devices that can be traced are listed below.

1) Bit device : X, FX, DX, Y, FY, DY, M, L, F, SM, V, B, SB, T (Contact), T (Coil), ST (Contact), ST (Coil), C (Contact), C (Coil), J□\X, J□\Y, J□\B, J□\SB, BL□\S........................Max. 50 points

2) Word device : T (Current value), ST (Current value), C (Current value), D, SD, FD, W, SW, R, Z, ZR, U□\G, J□\W, J□\SW ....Max. 50 points

(c) The program trace file stores the trace condition data and trace execution data to execute a program trace. After a trace is started in a peripheral device capable of GPP functions, it is continued until the set number of traces is completed.

(d) The trace results show the program name, step No. device status, etc., for each trace No.



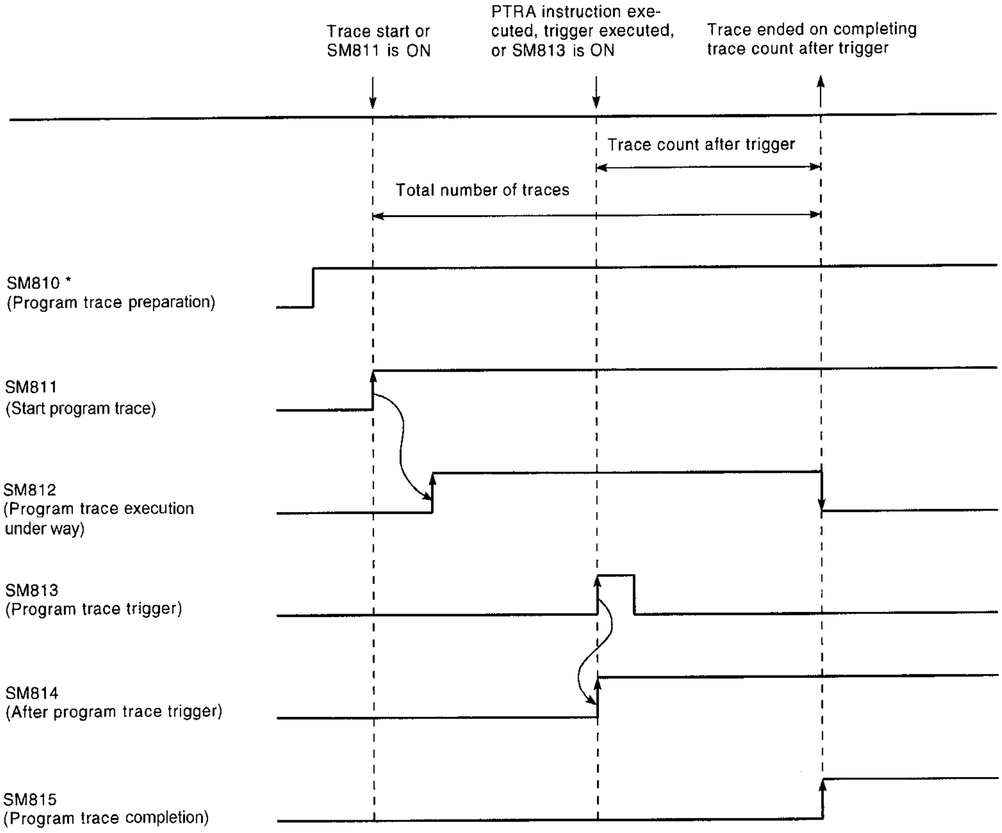
```
[Trace Results Display]

 Times Program        Step  BranchIns   Time(ms)  D0      M0    M1
    -5 MAIN              0   S 0 Step    1894.2  31339
    -4 MAIN              9  END(FEND)    1894.5  31340
    -3 MAIN              0     (Stat)    1894.9  31340
    -2 MAIN              0   S 0 Step    1900.4  31340
    -1 MAIN              9  END(FEND)    1900.6  31341
     0 MAIN              0     (Stat)    1901.1  31341
     1 MAIN              0   S 0 Step    1906.7  31341
     2 MAIN              9  END(FEND)    1906.9  31342
     3 MAIN              0     (Stat)    1907.5  31342
     4 MAIN              0   S 0 Step    1912.7  31342
     5 MAIN              9  END(FEND)    1912.9  31343
     6 MAIN              0     (Stat)    1913.4  31343
     7 MAIN              0   S 0 Step    1918.6  31343
     8 MAIN              9  END(FEND)    1918.8  31344
     9 MAIN              0     (Stat)    1919.3  31344

PgUp:Prev PgDn:Next                                          Esc:Close
```

(2) Basic operation

The following shows the basic operation for program trace.

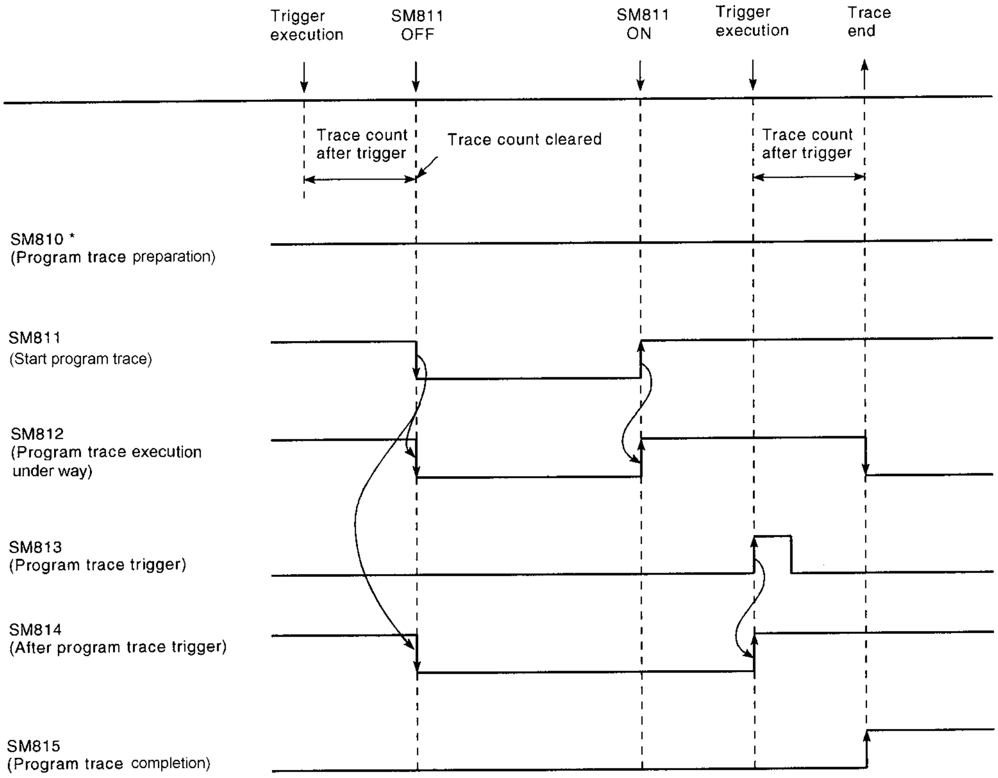
The statuses during execution of the program trace function can be confirmed by monitoring special relays SM810 to SM815 and SM828.

- Without suspension of the trace



* When ready for program trace, SM810 is automatically turns ON.

- With trace suspension



* When the trace is suspended from a peripheral device capable of GPP functions, SM810 is turned OFF.

The following shows an operation at error occurrence.
When an error occurs during program trace, SM828 (program trace error) comes ON, and at the same time, SM811 (program trace start) goes OFF.
To turn SM828 OFF, either turn SM811 ON, or execute the PTRA instruction.

Operation Procedures

The following shows the procedures to perform program trace.

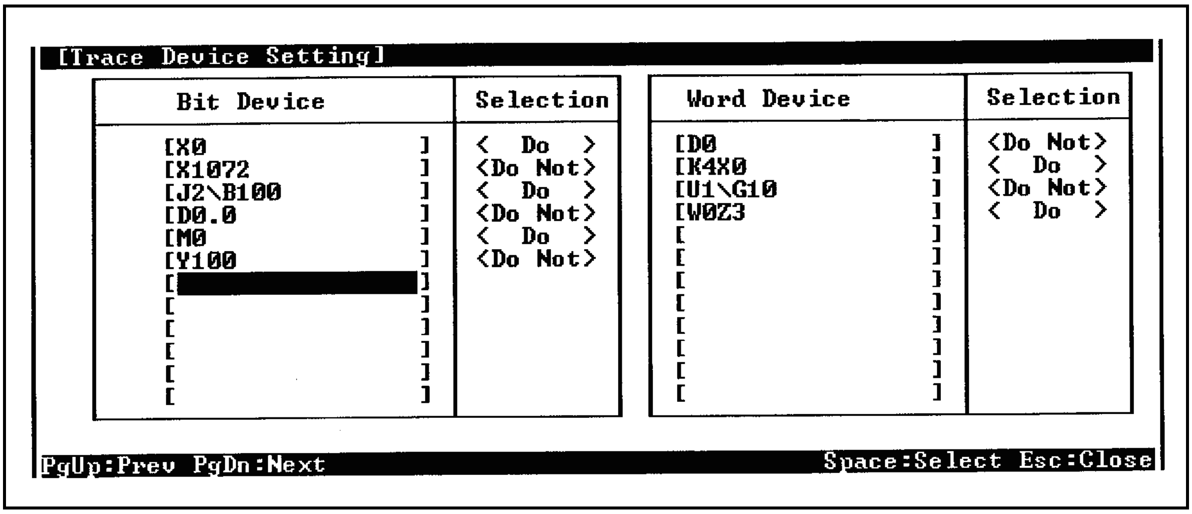
These operations are performed on the "Program Trace" screen of the trace menu in the online mode.

Perform these operations with the CPU module setting to the STEP-RUN. (Refer to Section 8.7.)

(1) Set the trace devices and trace conditions with GPP function.

(a) Setting the trace devices

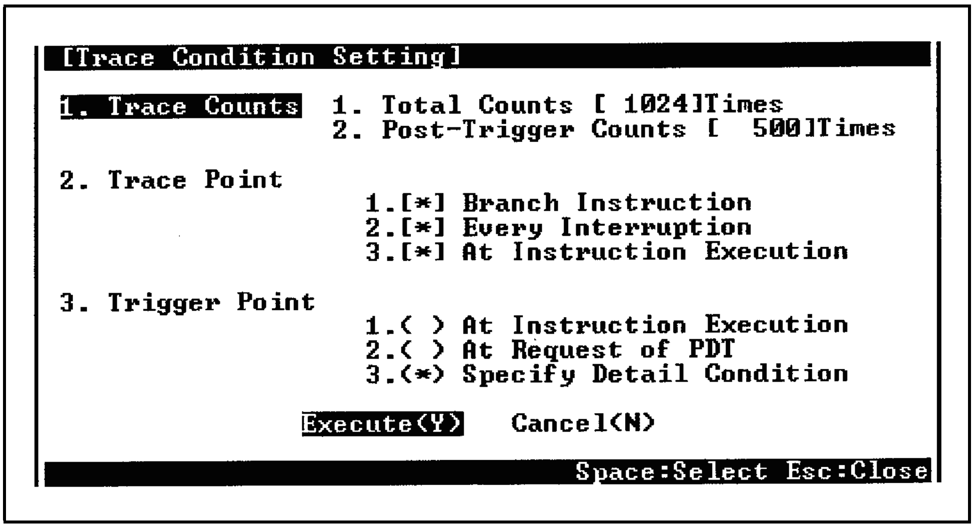
Set the devices at "Trace Device Setting" on the "Program Trace" screen.



```
[Trace Device Setting]
  Bit Device              Selection     Word Device              Selection
  [X0              ]      <  Do   >     [D0              ]       <Do Not>
  [X1072           ]      <Do Not>      [K4X0            ]       <  Do  >
  [J2\B100         ]      <  Do   >     [U1\G10          ]       <Do Not>
  [D0.0            ]      <Do Not>      [W0Z3            ]       <  Do  >
  [M0              ]      <  Do   >     [                ]
  [Y100            ]      <Do Not>      [                ]
  [                ]                    [                ]
  [                ]                    [                ]
  [                ]                    [                ]
  [                ]                    [                ]
  [                ]                    [                ]
  [                ]                    [                ]
PgUp:Prev PgDn:Next                                 Space:Select Esc:Close
```

(b) Setting the trace conditions

Set the trace conditions at "Trace Condition Setting" on the "Program Trace" screen.



```
[Trace Condition Setting]

1. Trace Counts  1. Total Counts [ 1024]Times
                 2. Post-Trigger Counts [  500]Times

2. Trace Point
                   1.[*] Branch Instruction
                   2.[*] Every Interruption
                   3.[*] At Instruction Execution

3. Trigger Point
                   1.( ) At Instruction Execution
                   2.( ) At Request of PDT
                   3.(*) Specify Detail Condition

               Execute(Y)   Cancel(N)

                              Space:Select Esc:Close
```

The following is an explanation of the screen above:

One of the following three settings can be made for the trace condition: "1. Trace Counts", "2. Trace Point", or "3. Trigger Point".

1) "Trace Counts"

For the total count, set the number of program traces executed from the trace start to the trace end.

For the count after the trigger, set the number of program traces executed from execution of the trigger to the trace end.

The following shows the formula that sets range for these counts:

Count after trigger≦total count≦8192

2) "Trace Point"

Set the point at which the trace is to be executed. Select one or multiple item(s) of the following:

ⓐ Branch Instruction :Executed at each CALL, JMP, or other instructions.

ⓑ Every Interruption :Executed at each interrupt program.

ⓒ Upon execution of each instruction :Executed at each PTRAEXE instruction.

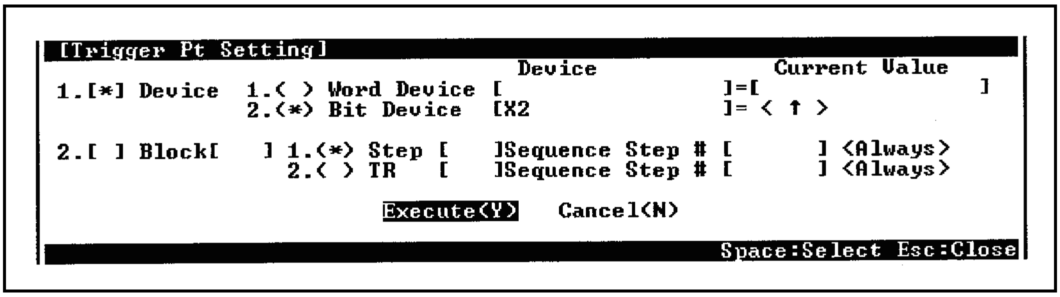
3) "Trigger Point"

Set the point at which the trigger is executed. Select one of the following:

ⓐ Upon execution of each instruction : When executing PTRA instruction

ⓑ At Request of PDI : When operating trigger using the peripheral devices capable of GPP function.

ⓒ Specify Detail Condition : Set a device and step number.
The following shows setting examples: The details on how to make the settings and trigger execution timing are the same as described in Section 8.2 Monitor condition setup in Monitor function.



```
[Trigger Pt Setting]
                                  Device             Current Value
1.[*] Device  1.( ) Word Device [                ]=[                ]
              2.(*) Bit Device  [X2              ]= < ↑ >

2.[ ] Block[   ] 1.(*) Step [   ]Sequence Step # [       ] <Always>
                 2.( ) TR   [   ]Sequence Step # [       ] <Always>

                    Execute(Y)   Cancel(N)
                                              Space:Select Esc:Close
```

The following shows the setting device under the detailed condition.

- Bit device : X, FX, Y, FY, M, L, F, SM, V, B, SB, T (Contact), ST (Contact), C (Contact), J□\X, J□\Y, J□\B, J□\SB, BL□\S
- Word device : T (Current value), ST (Current value), C (Current value), D, SD, FD, W, SW, R, Z, ZR, U□\G, J□\W, J□\SW

The following qualifications are possible with respect to the devices listed above.

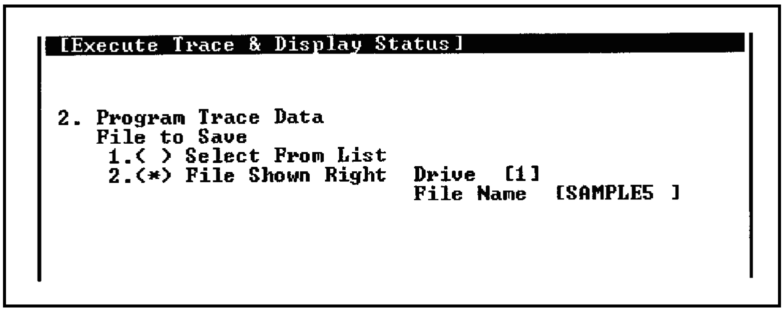
- Digit designation for bit devices
- Bit number designation for word devices

POINT

The trace execution time, program name, step and branch factor are automatically added to the trace results.

(2) Write the set trace device and trace condition to the memory card.

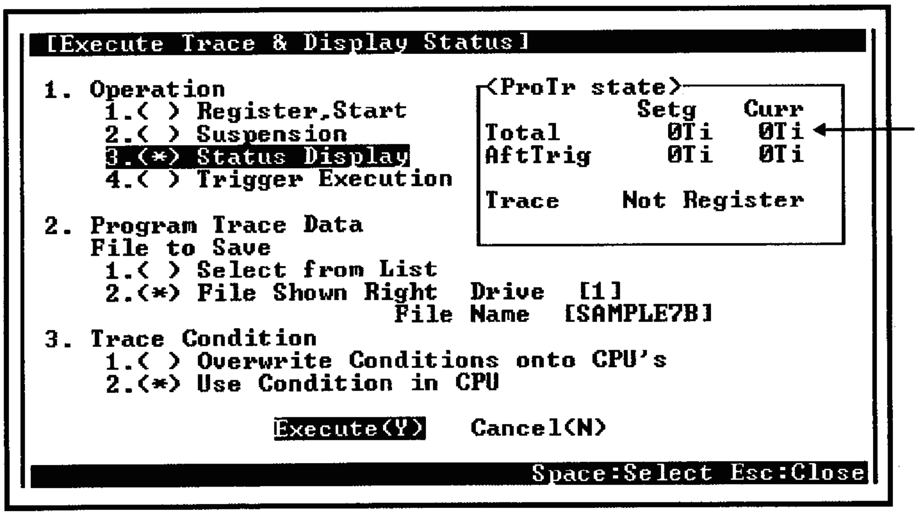
(a) Set the trace file and storage destination.
Set the drive number and file name at "1. ( ) Execute Trace & Display Status" on "Program Trace" screen.



```
[Execute Trace & Display Status]

2. Program Trace Data
   File to Save
    1.( ) Select From List
    2.(*) File Shown Right  Drive   [1]
                            File Name  [SAMPLE5 ]
```

(b) Write the trace file to the memory card.
Write the trace file to the memory card by using "9. ( ) Write to PC (Condition)" on "Program Trace" screen.
Since file names are used when writing to the memory card, multiple trace files can be written.

(3) Execute the program trace.
Write the trace file to the memory card by using "9. ( ) Write to PC (Condition)" on "Program Trace" screen.
The following shows a setting example for "1. ( ) Execute Trace & Display Status".



```
[Execute Trace & Display Status]

1. Operation                      <ProTr state>
    1.( ) Register,Start                     Setg   Curr
    2.( ) Suspension              Total       0Ti    0Ti
    3.(*) Status Display          AftTrig     0Ti    0Ti
    4.( ) Trigger Execution
                                  Trace      Not Register
2. Program Trace Data
   File to Save
    1.( ) Select from List
    2.(*) File Shown Right   Drive   [1]
                        File Name   [SAMPLE7B]
3. Trace Condition
    1.( ) Overwrite Conditions onto CPU's
    2.(*) Use Condition in CPU

              Execute(Y)    Cancel(N)

                             Space:Select Esc:Close
```

Displayed only when "Status Display" is selected.

The following is an explanation of the screen above:
The following settings can be made for "Execute Trace & Display Status": "1. Operation", "2. Program Trace Data", and "3. Trace Condition".

(a) "Operation"
Select one of the following:

1) Register, Start	:	The trace is started. The trace count is started.
2) Suspension	:	The trace is suspended.The trace count and the count are cleared after the trigger. (To restart the trace, select "Register, Start" again.)
3) Display Status	:	The trace statuses are displayed on the same screen.
4) Trigger Execution	:	The count is started after the trigger. The trace is ended on reaching the designated count after the trigger.

(b) "Program Trace Data"
Select one of the following:

1) Select From List	:	Data from among the program trace files in the memory card are selected.
2) File Shown Right	:	The drive number and program trace file name are set.

(c) "Trace Condition"
Select one of the following:

1) Overwrite Conditions onto CPU's	:	The trace condition in an existing trace file is overwritten.
2) Use Condition in CPU	:	Program trace under the condition in the trace file designated in "2. Program Trace Data" is executed.

(4) Retrieve the trace results from the CPU module and display them.
   (a) Read the trace results from the CPU module by using "A. ( ) Read from PC (Results)" on "Program Trace" screen.
   (b) Display the read trace results by using "4. ( ) Trace Results Display" on "Program Trace" screen.

POINT

Once the program trace has been executed, the second is not executed. To execute the trace again, execute the PTRAR instruction to reset program trace.

NOTE

1) The program trace can be performed only for STEP-RUN.
2) Set program trace files in the RAM area of the memory card.
3) It is possible to execute program trace from another station in the network, or from a serial communication module. However, sampling trace cannot be executed from more than one site at the same time. With the Q2ASCPU, sampling trace can be executed from only one site at a time.
4) The program trace is performed by connecting the Q2ASCPU with the peripheral device capable of GPP function.

8.9 Simulation Function

POINT

When the link memory and the buffer memory are simulated in the simulation data file, a memory card is required.

Application

This function simulates execution of a program in step execution or partial execution, with the input module, output module, or special function module isolated from the CPU module.This enables QnACPU to debug a program without any effects on other modules.

Function Description

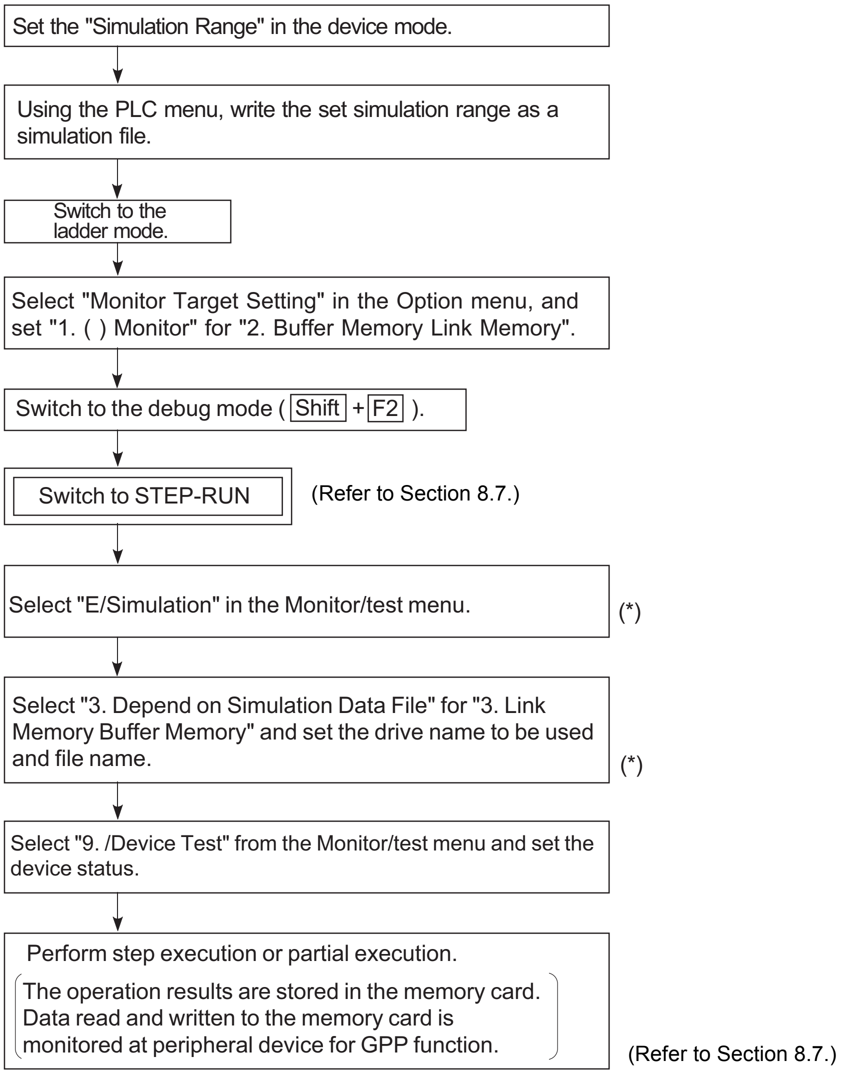
(1) When the program is executed, data chaneges from/to external sources are isolated by setting so that data refreshes for input/output modules are not executed.

(2) Isolation from special function module operations is achieved by setting "Ignore" or "Depend on Simulation Data File" with respect to the buffer memory of the special function module.

Operation Procedures

The following shows the procedures to perform simulation.

☐ indicates a GPP function operation and ▣ indicates an operation at the CPU module.



Set the "Simulation Range" in the device mode.

Using the PLC menu, write the set simulation range as a simulation file.

Switch to the ladder mode.

Select "Monitor Target Setting" in the Option menu, and set "1. ( ) Monitor" for "2. Buffer Memory Link Memory".

Switch to the debug mode (Shift + F2).

Switch to STEP-RUN (Refer to Section 8.7.)

Select "E/Simulation" in the Monitor/test menu. (*)

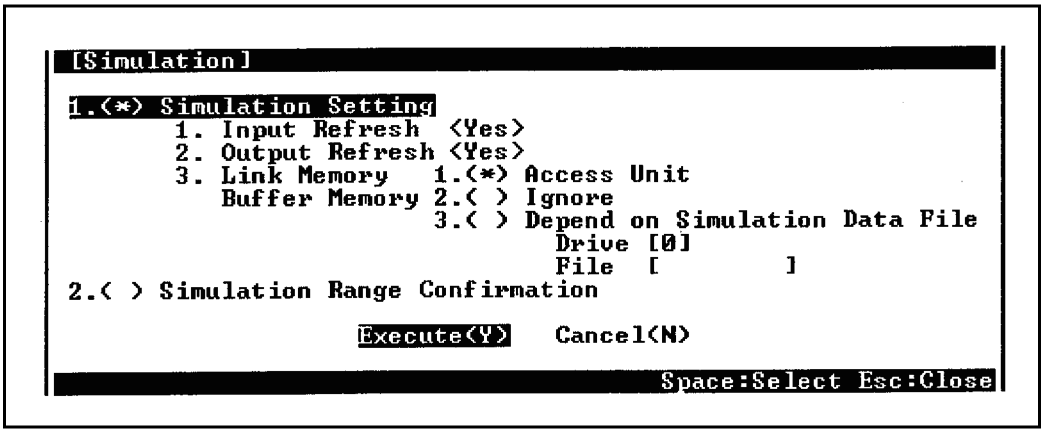
Select "3. Depend on Simulation Data File" for "3. Link Memory Buffer Memory" and set the drive name to be used and file name. (*)

Select "9. /Device Test" from the Monitor/test menu and set the device status.

Perform step execution or partial execution.
(The operation results are stored in the memory card. Data read and written to the memory card is monitored at peripheral device for GPP function.) (Refer to Section 8.7.)

(*) Described in this section.

(1) Make the settings on the simulation setting screen shown below.

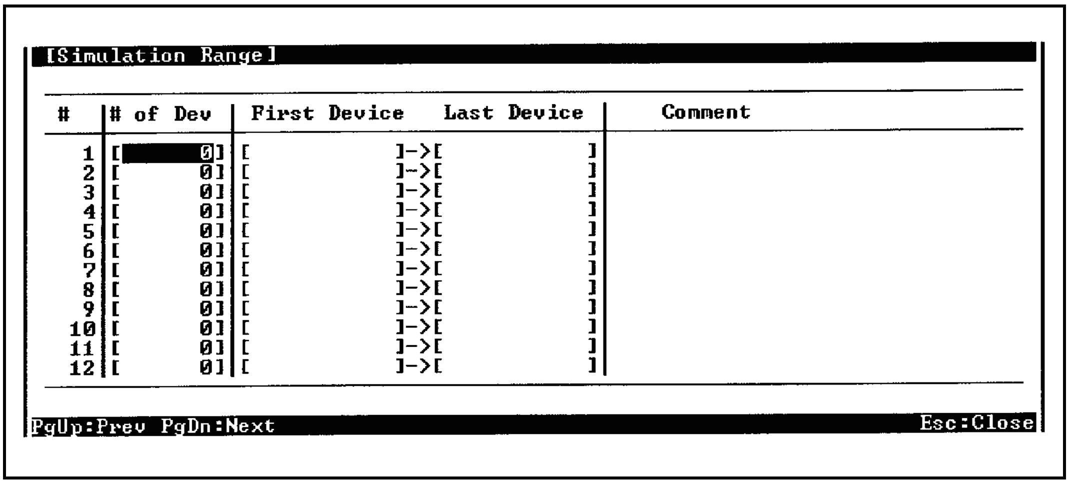


```
[Simulation]

1.(*) Simulation Setting
         1. Input Refresh  <Yes>
         2. Output Refresh <Yes>
         3. Link Memory   1.(*) Access Unit
            Buffer Memory 2.( ) Ignore
                          3.( ) Depend on Simulation Data File
                                  Drive [0]
                                  File  [         ]
2.( ) Simulation Range Confirmation

                     Execute(Y)   Cancel(N)

                                          Space:Select Esc:Close
```

- The following shows details on the settings that can be made for each item:

Setting Item	Setting Option	Description
Input Refresh	Yes/No	Select whether inputs from external sources are input to the CPU module or not.
Output Refresh	Yes/No	Select whether the operation results in the CPU module are output to external destinations or not.
Link Memory/ Buffer Memory	Access Unit Ignore Depend on Simulation Data File	Select the method of accessing each module.

If "Depend on Simulation Data File" is selected for "Link Memory/Buffer Memory", the access range for each module can be checked by checking the simulation range settings.



```
[Simulation Range]
 #  |# of Dev | First Device   Last Device  |      Comment
  1 |[      0]|[             ]->[          ]|
  2 |[      0]|[             ]->[          ]|
  3 |[      0]|[             ]->[          ]|
  4 |[      0]|[             ]->[          ]|
  5 |[      0]|[             ]->[          ]|
  6 |[      0]|[             ]->[          ]|
  7 |[      0]|[             ]->[          ]|
  8 |[      0]|[             ]->[          ]|
  9 |[      0]|[             ]->[          ]|
 10 |[      0]|[             ]->[          ]|
 11 |[      0]|[             ]->[          ]|
 12 |[      0]|[             ]->[          ]|
PgUp:Prev PgDn:Next                                  Esc:Close
```

NOTE

1) Simulation can be performed only for STEP-RUN.
2) A memory card is required to carry out link memory/buffer memory simulation using a simulation data file.
   Set the simulation data file to the RAM area of the memory card.
3) It is possible to carry out simulation from another station in the network, or from a serial communication module. However, simulation cannot be executed from several sites at the same time. With the Q2ASCPU, sampling trace can only be executed from one site at a time.
4) Simulation is performed by connecting the Q2ASCPU and the peripheral devices capable of GPP function.
5) Note the following points when executing simulation:
   - If direct inputs (DX) and direct outputs (DY) are used to handle inputs/outputs directly, the device memory is accessed rather than the actual inputs/outputs.
   - No processing is performed for any special function module instruction.
   - When a "SP.UNIT ERROR" occurs, FFFFH is displayed in the module number area of the common information.
   - If "Ignore" is set for the buffer memory access method, FFFFH is set for access by instruction and the monitor results.

8.10 Debugging by Several People

This function allows simultaneous debugging from several peripheral devices capable of GPP functions.

Application

This function is used to simultaneously debug different files from more than one peripheral device capable of GPP functions.

Function Description

The following shows the combinations of debugging functions that can be used simultaneously by different operators.

Debug function from host	Debug function from other stations						
	Monitor	Write during RUN	Execution Time Measurement	Sampling Trace /Program Trace	Status Latch	Step Operation	Simulation
Monitor	○	×	○	○	○	○	○
Write during RUN	×	○	×	×	×	×	×
Execution time measurement	○	×	×	○	○	○	○
Sampling trace	○	×	○	×	○	○	○
Program trace	○	×	○	×	○	○	○
Status latch	○	×	○	○	×	○	○
Step operation	○	×	○	○	○	×	○
Simulation	○	×	○	○	○	○	×

○: Simultaneous execution possible. (However, the detailed condition setting at only one peripheral device capable of GPP functions is valid; detailed conditions cannot be set at the other peripheral devices capable of GPP functions.)

×: Can only be executed from one peripheral device capable of GPP functions.

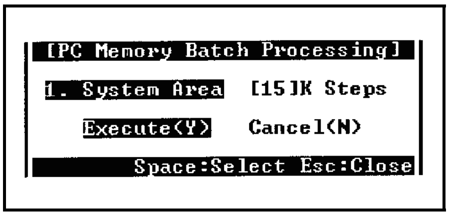
8.10.1 Simultaneous monitoring by several people

The Q2ASCPU allows monitoring for several people. Setting of other station monitor file in the built-in RAM system area allows monitoring at a high-speed from other stations. (Monitor file setting for the host is not required.)

Operation Procedures

The operation for simultaneous monitoring by several people is described below.

(1) Select "5. ( ) Format (with Option)" for "B/PC Memory Batch Processing" in the "2/PC" menu in the online mode, and set a monitor file for another station.
The following shows setting examples:



[PC Memory Batch Processing]

1. System Area [15]K Steps

Execute(Y) Cancel(N)

Space:Select Esc:Close

Up to 15k steps in 1k step units can be set as the system area. The area corresponding to one monitor file for another station is no more than 1k step. Accordingly, a maximum of 15 monitor files can be set.
Since the built-in RAM program file area is in the same area as the monitor file for other stations, the program file area is reduced for the area of the other station monitor file.

(2) After setting, the built-in RAM is formatted.

NOTE

1) The detailed conditions for monitoring can be set from one site only.
2) Monitoring from other stations is possible without setting monitor files for other stations, but in this case, high-speed monitoring is not possible.
3) When simultaneous monitoring from multiple persons is desired, perform this operation before writing the parameter file or the program file in the built-in RAM.
   If this operation is performed after writing the file in the built-in RAM, all files are erased.
4) The number of locations for simultaneous access to one CPU is up to 16.

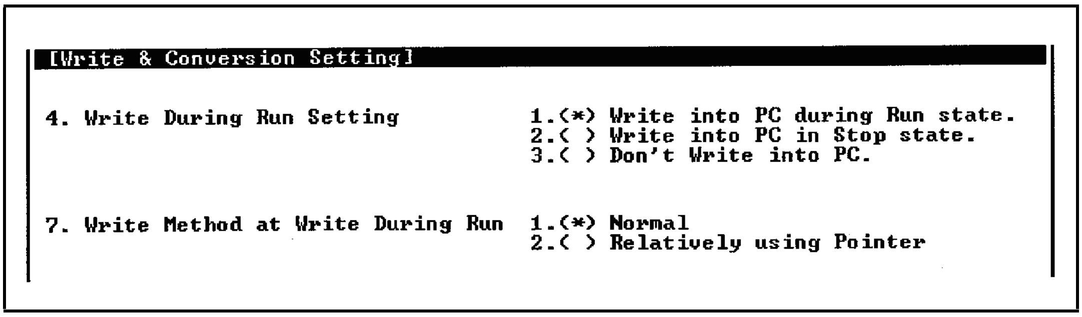
8.10.2 Simultaneous execution of write during RUN by several people

The Q2ASCPU allows simultaneous write during RUN to one file or another file by several people.

Operation Procedures

The following shows the procedures for simultaneous write during RUN executed by several people.

(1) With "4/ Write & Conversion Setting" in "8/ Option" menu of the ladder mode, "4. Write During RUN Setting" and "7. Write Method at Write During RUN" are set.
The following shows setting examples:



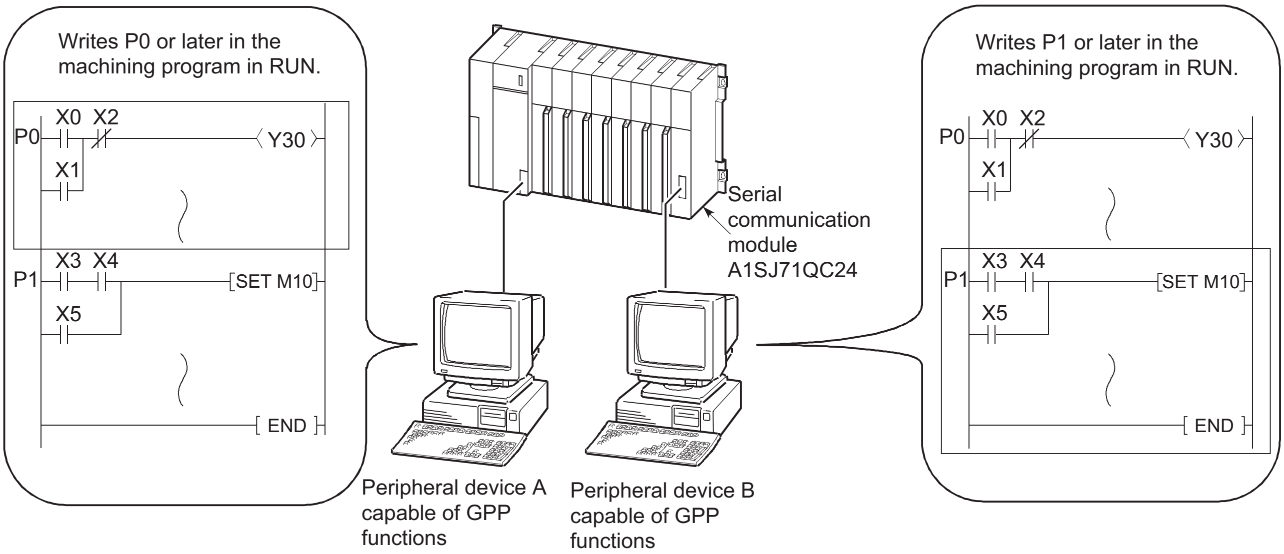
```
[Write & Conversion Setting]

4. Write During Run Setting          1.(*) Write into PC during Run state.
                                     2.( ) Write into PC in Stop state.
                                     3.( ) Don't Write into PC.

7. Write Method at Write During Run  1.(*) Normal
                                     2.( ) Relatively using Pointer
```

(a) Set "1. ( ) Write into PC during Run state" for "4. Write During Run Setting".

(b) Select "1. ( ) Normal" or "2. ( ) Relatively using Pointer" for "7. Write Method at Write During Run".

If more than one person is to perform a write during RUN operation with respect to the same file, set a write during RUN pointer in advance and select "2. ( ) Relatively using Pointer".

The example below shows a case where peripheral device capable of GPP functions A performs write during RUN from P0, and peripheral device capable of GPP functions B performs write during RUN from P1.The program enclosed in the frame ☐ is the program subject to write during RUN.



NOTE

Refer to Section 8.3.

9 MAINTENANCE FUNCTION

9.1 Function List

The following shows the functions for maintenance.

Item		Description	Reference
Watchdog timer		Function that monitors watchdog errors due to CPU module hardware or program errors.	Section 9.2
Self-diagnostics function		Function whereby the Q2ASCPU itself diagnoses whether or not there are any errors.	Section 9.3
Error history		Function that stores the results of diagnosis in memory as a fault history.	Section 9.4
System protect		Function that sets whether reading/writing is enabled or disabled for Q2ASCPU files.	Section 9.5
Keyword Registration		Function that disables GPP function operations with respect to the CPU module.	Section 9.6
System display		Function that allows monitoring of the system configuration by connecting a peripheral device capable of GPP functions.	Section 9.7
LED indication		Function to display the CPU module operation status with the LED located on the front of the CPU module.	Section 9.8
	LED indication	Indicates whether CPU module operation is normal or abnormal.	Section 9.8.1
	Priority setting	Priority for LED indication is set depending on the error.	Section 9.8.2

For details of GPP function operation, refer to the GX Developer Operating Manual or the Type SW☐IVD-GPPQ Software package Operating Manual (Online).

9.2 Watchdog Timer

(1) Watchdog timer (WDT)
The watchdog timer is an internal timer of programmable controller that detects programmable controller CPU hardware errors and sequence program errors.
200ms is set as the default setting for this timer.

REMARK

The time set for the watchdog timer can be changed using "WDT" in PC RAS setting in the GPP function parameter mode.
The setting range is 10ms to 2000ms (in 10ms units).

(2) Resetting the watchdog timer
The Q2ASCPU resets the watchdog timer during END processing.
When the Q2ASCPU is normally operating and executing the END instruction within the setting value of the watchdog timer, the watchdog timer does not give time-out.
WDT times out when the END instruction is not executed within the value set for the watchdog timer due to a Q2ASCPU hardware error or an excessively long sequence program scan time.

REMARK

Scan time is the time taken for the execution of the sequence program, starting from step 0 and ending at step 0.
The scan time is not the same in every scan: it differs according to the execution or non-execution of the instructions used in the program. (Refer to Section 12.1.)

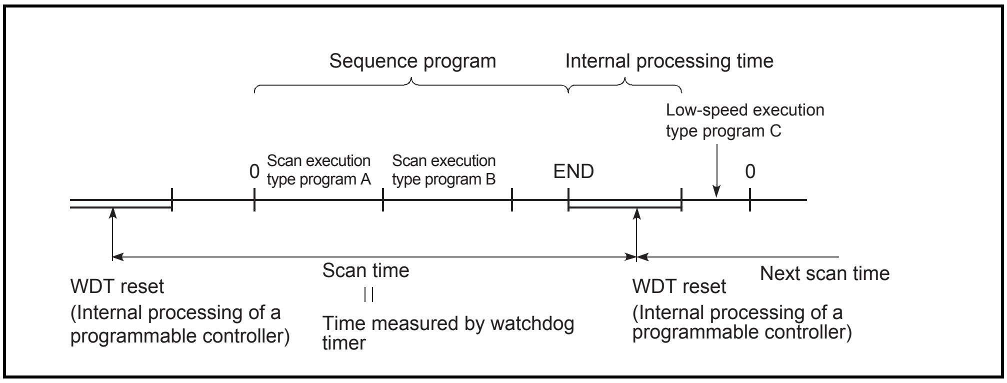


Fig. 9.1 Resetting the watchdog timer

(3) Processing when the watchdog timer times out

When the scan time exceeds the set value of the watchdog timer, a watchdog timer error occurs and the programmable controller operates as follows.

(a) All PLC outputs are turned OFF.

(b) The RUN LED on the front of the CPU module goes off and the ERROR LED flickers.

(c) SM1 turns ON and the error code is stored in SD0.

REMARK

The watchdog timer can be reset by a WDT instruction in the sequence program. However, the scan time value is not reset and scan time is measured up to the END instruction.

POINT

An error occurs within 0ms to 10ms in the measured time for watchdog timer.

9.3 Self-diagnostics Function

The self-diagnosis function is a function whereby the Q2ASCPU diagnoses its own errors.

(1) The self-diagnosis function serves to prevent malfunctions of the programmable controller, and to facilitate preventive maintenance.Self-diagnositics processing is executed if an error occurs at QnACPU power ON or while the programmable controller is in the RUN status, and it involves the display of the error detected by the self-diagnostics function, stopping of programmable controller operation, etc.

(2) The Q2ASCPU stores the error code of the error in the special register SD0, and turns on the ERROR LED and displays a message.
If several errors occur, the error code of the latest error is stored in SD0.

(3) Even if the programmable controller power is turned OFF, the latest 16 errors are recorded with the battery backup.(Refer to 9.4ê?)
The PLC diagnostics mode of the GPP function can check error histories.

(4) When an error is detected by self-diagnosis, CPU module operation complies with one of the following two modes:
- Mode in which programmable controller operation is stopped
  When an error is detected, operation is stopped immediately and all outputs (Y) are turned OFF.
- Mode in which programmable controller operation is continued
  When an error is detected, only the program part affected by the error is not executed; the rest of the program is executed.

In addition, settings can be made in PC RAS setting in the parameter mode to continue operation or stop operation when the following errors occur:

1) Calculation (including SFC programs)
2) Extended Ins
3) Fuse Blown
4) I/O Unit Compare
5) Sp Unit Access
6) IC Card Access
7) IC Card Operate

(The default for all of these in the parameters is "Pause".)

Example: If "Resume" is set for I/O module verify error, operation is continued using I/O address before error occurrence.

When an error is detected, a record of the error occurrence is stored in the special relays (SM0, SM1) and the error contents are stored in a special register (SD0).Use these special relays and this special register in the sequence program to establish programmable controller or mechanical system interlocks.

(5) It is possible to select whether or not the following checks are performed by setting "Yes/No" for error check in PC RAS setting in the parameter mode.

1) Battery Check
2) Fuse Blown Check
3)

(The default for all of these in the parameter settings is "Yes".)

If "No" is set for error check, error detection is not performed for these items, which shortens the processing time for the END instruction.

Even if "Yes" for error check is set in the parameter, 1) thorugh 3), above error check, can be canceled by turning on the special relay SM 1084.

However, if "No" is set in the parameter, turning off SM1084 is ineffective to execute the error check.

Self-diagnostics list

Diagnosis item		Diagnosis timing	Status of the CPU module	LED Status	
				RUN	ERROR
Hardware error	CPU module error	Always	Stop	OFF	Flickers
	END instruction not executed	When executing END instruction	Stop	OFF	Flickers
	RAM check	At power-ON or RESET	Stop	OFF	Flickers
	Operation circuit check	At power-ON or RESET	Stop	OFF	Flickers
	Fuse blown (Default ⋯ stop)*1	When executing END instruction (Default ⋯ check executed)*2	Stop/Continue	OFF/ON	Flickering/ON
	I/O interrupt error	When interruption occurs	Stop	OFF	Flickers
	Special function module error	• At power-ON or RESET • When executing FROM/TO instruction	Stop	OFF	Flickers
	Control bus error	• At power-ON or RESET • When executing FROM/TO instruction	Stop	OFF	Flickers
	Occurrence of momentary power interruption	Always	Continue	ON	OFF
	Low battery	Always (Default ⋯ check executed)*3	Continue	ON	OFF
Handling error	I/O module verification (Default ⋯ stop)*1	When executing END instruction (Default ⋯ check executed)*2	Stop/Continue	OFF/ON	Flickering/ON
	Special function module Special function module allocation error	• When power is ON or RESET • When switching from STOP to RUN	Stop	OFF	Flickers
	Special function module access error (Default ⋯ stop)*1	When executing FROM/TO instruction	Stop/Continue	OFF/ON	Flickering/ON
	No parameters	When power is ON or RESET	Stop	OFF	Flickers
	Boot error	When power is ON or RESET	Stop	OFF	Flickers
	Memory card operation error (Default ⋯ stop)*1	When memory card is inserted/removed	Stop/Continue	OFF/ON	Flickering/ON
	File setting error	When power is ON or RESET	Stop	OFF	Flickers
	File access error (Default ⋯ stop)*1	When executing each instruction	Stop/Continue	OFF/ON	Flickering/ON
	Unable to execute instruction	When power is ON or RESET	Stop	OFF	Flickers
Parameter errors	Parameter setting check	• When power is ON or RESET • When switching from STOP to RUN	Stop	OFF	Flickers
	Link parameter error	• When power is ON or RESET • When switching from STOP to RUN	Stop	OFF	Flickers
	SFC parameter error	When switching from STOP to RUN	Stop	OFF	Flickers

*1 Can be changed to operation continues by GPP function parameter setting.

*2 GPP function parameters can be set so that no check is performed.
Also, checking is not performed when SM251 is on.

*3 GPP function parameters can be set so that no check is performed.

Self-diagnostics list(Continued)

Diagnosis item		Diagnosis timing	Status of the CPU module	LED Status	
				RUN	ERROR
Program error	Instruction code check	• When power is ON or RESET • When switching from STOP to RUN	Stop	OFF	Flickers
	No END instruction	• When power is ON or RESET • When switching from STOP to RUN	Stop	OFF	Flickers
	Pointer setting error	• When power is ON or RESET • When switching from STOP to RUN	Stop	OFF	Flickers
	Pointer setting error	• When power is ON or RESET • When switching from STOP to RUN	Stop	OFF	Flickers
	Operation error (Default ··· stop)*1	When executing each instruction	Stop/Continue	OFF/ON	Flickering/ON
	FOR-NEXT instruction configuration error	When executing each instruction	Stop	OFF	Flickers
	CALL-RET instruction configuration error	When executing each instruction	Stop	OFF	Flickers
	Interrupt program error	When executing each instruction	Stop	OFF	Flickers
	Unable to execute instruction	When executing each instruction	Stop	OFF	Flickers
	Extended instruction error (Default ··· stop)*1	When executing each instruction	Stop/Continue	OFF/ON	Flickering/ON
	SFC program configuration error	When switching from STOP to RUN	Stop	OFF	Flickers
	SFC block configuration error	When switching from STOP to RUN	Stop	OFF	Flickers
	SFC step configuration error	When switching from STOP to RUN	Stop	OFF	Flickers
	SFC syntax error	When switching from STOP to RUN	Stop	OFF	Flickers
	SFC operation check error (Default ··· stop)*1	When executing each instruction	Stop/Continue	OFF/ON	Flickering/ON
	SFC program execution error	When switching from STOP to RUN	Continue	ON	ON
	SFC block execution error	When executing each instruction	Stop	OFF	Flickers
	SFC step execution error	When executing each instruction	Stop	OFF	Flickers
CPU error	Watchdog error supervision	Always	Stop	OFF	Flickers
	Program timeout	Always	Continue	ON	ON
Annunciator check		When executing each instruction	Continue	ON	OFF
CHK instruction check		When executing each instruction	Continue	ON	OFF

*1 Can be changed to operation continues by GPP function parameter setting.

9.3.1 Interruption due to error detection

Q2ASCPU can execute the interrupt program, which is interrupt pointer I32 to I39, at error occurrence.
In the case of errors for which operation can be set to continue or stop in PC RAS setting in the GPP function parameter mode, this function is only executed when "Resume" is set.
If "Pause" is set for the error, a stop error interrupt program (132) is executed.
The following shows the relevant errors.

Interrupt pointer	Corresponding error message
I32	Stop all errors
I33	Vacancy
I34	UNIT VERIFY ERR. FUSE BREAK OFF SP.UNIT ERROR
I35	OPERATION ERROR SFCP OPE.ERROR SFCP EXE.ERROR
I36	ICM.OPE.ERROR FILE.OPE.ERROR
I37	EXTEND INS.ERR.
I38	PRG.TIME OVER
I39	CHK instruction Annunciator detect
I40 to I47	Vacancy

When the error occurs and the system can continue the drive mode. Or it is an error where "continues/stops" can be selected,and "continues" is set.

POINT

Interrupt pointers I32 to I39 are prohibited for execution when the PLC power is ON or when the CPU module is reset.
To use I32 to I39, make the execution allowed with IMASK instruction.

REMARK

1) For details on interrupt pointers, refer to the QnACPU Programming Manual (Fundamentals).
2) For the IMASK instruction, refer to the QCPU (Q mode)/QnACPU Programming Manual (Common Instructions).

9.3.2 LED inidcation due to an error

When an error occurs, the LED located on the front of the CPU module turns on.Refer to Section 9.8 for the details of the LED display.

9.3.3 Resetting error

Q2ASCPU allows error resetting only for the errors that does not block the CPU module operation.

The procedure for resetting an error is as follows.

1) Eliminate the cause of the error.
2) Store the error code to be reset in special register SD50.
3) Turn on special register SM50.
4) The error is reset.

When the CPU module is recovered from canceling the error, the special relay, special register, and LED affected by the error are set to the state before the error occurred.

If the same error occurs again after the error reset, it is recorded in the error history again.

To reset multiple detected annunciators, only the first detected F number is reset.

POINT

When an error is reset by storing its error code in SD50, the last two digits of the error code are ignored.

Example:

If errors with error codes 2100 and 2111 have occurred, and error code 2100 is reset, error code 2111 is also reset.

9.4 Error History

Q2ASCPU can record the results detected by the self-diagnostics function with the detection time in memory as an error history.

POINT

Since the internal clock of the Q2ASCPU is used for setting the detection time, be sure to set the correct time before using the CPU module. (Refer to Section 10.5 for setting method of the clock.)

(1) Storage area

(a) The latest 16 errors are stored in the error history storage memory of the CPU module, which is latched.

(b) In the case of storing more than 16 errors, they can be stored to files in a memory card by making the appropriate setting in the PC RAS settings in the GPP function parameter mode.

(c) If a discrepancy arises between the parameters and memory card error history when executing 1) or 2) below, the contents of the error history files are cleared first, and the 16-point data of the fault history storage memory of the CPU is transferred to the history file.

1) The number of error records in the history file as set in the parameters is changed part way through.

2) A memory card whose capacity does not match the number of error records set in the parameters is installed.

(d) The following shows the storage area for the error history file:

Storage area	File in set memory card
Number of storable error records	Max. 100 (can be changed)[*1]

[*1] When the number of errors that can be stored is exceeded, the oldest error record is cleared and the newest one stored in the same place.

POINT

Even if the error history file set in the parameters does not exist in the memory card, no CPU module error occurs.
The CPU module performs only the processing that stores errors in the error history storage file.

(2) Clearing the error history

The error history is cleared by using the error history clear function in the PLC menu in the PLC diagnosis mode of GPP function.

The error history clear function erases all details in the error history storage memory of the CPU module and in the error history file of the memory card.

9.5 System protect

Q2ASCPU features a number of functions that protect against program changes ("system protect") by restricting general data processing (access processing from GPP functions, serial communication modules, etc.) by third parties other than designers.
The following system protect functions are available.

Target Protection	Valid File for Protection	Description	Method	Valid Timing	Remark
Whole of CPU module	All files	Batch prohibition of write/control to the CPU module.	Turn ON SW1 of system setting switch 1 on the main CPU module . (See Section 15.2.)	Always	Valid for devices
Memory card units	All files	Establishes write protect for the memory card and prohibits writing.	Turn ON the memory card's write protect switch. (Refer to Section 18.5)	Always	
Drive units	Parameter Program	Registers entry codes for the following settings in relation to a specific drive (Example: Built-in RAM): 1) Prohibiting read/write display 2) Prohibiting writing	Register password. (Refer to Section 9.6)	Always	
File units	All files	Changes attributes file for each file as follows: 1) Prohibiting read/write display 2) Prohibiting writing	Change file attributes by password registration. (Refer to Section 9.6)	Always	

* "Control direction", "read/write display" and "writing" in the table above have the following meanings:

Item	Description
Control instruction	CPU module operation instruction by remote operation (Remote RUN, Remote STOP, etc.)
Read/write display	Operations of program read/write
Write	Operations that involve write processing, such as program write and test.

9.6 Password Registration

Passwords serve to prohibit reading and overwriting of data such as programs, comments, etc., in the Q2ASCPU from a peripheral device.

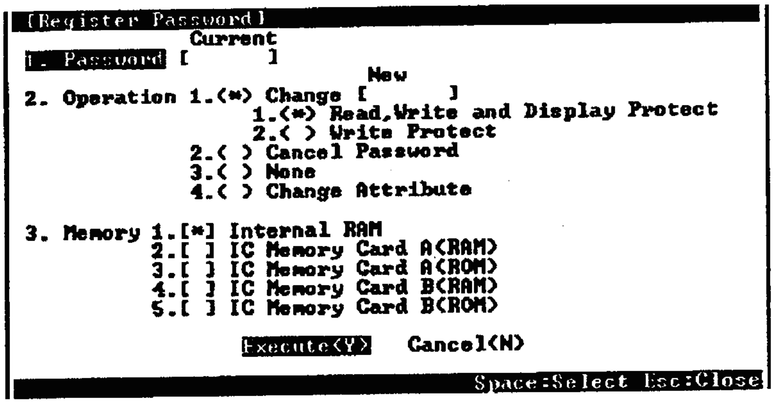
In password registration, the parameter files and program files of a designated memory (built-in RAM, memory card) are made the target of the entry code. There are two types of registration as follows:

- File names are not displayed, and read/write are prohibited.
- File write is prohibited. (Read is possible).

When a password is registered, file operations from a peripheral device are not possible without inputting the entry code registered in the CPU module.

(1) Register Password

Entry codes are registered using the entry code registration function in the PLC menu of the online mode of GPP function.



```
[Register Password]
                  Current
1. Password [        ]
                              New
2. Operation 1.(*) Change [        ]
                  1.(*) Read,Write and Display Protect
                  2.( ) Write Protect
             2.( ) Cancel Password
             3.( ) None
             4.( ) Change Attribute

3. Memory 1.[*] Internal RAM
          2.[ ] IC Memory Card A(RAM)
          3.[ ] IC Memory Card A(ROM)
          4.[ ] IC Memory Card B(RAM)
          5.[ ] IC Memory Card B(ROM)

                   Execute(Y)   Cancel(N)

                                  Space:Select Esc:Close
```

The following shows an explanation of each item in the screen:

(a) Password.... When a password is registered in the CPU module, input the registered password so that file operations are executed.
When an incorrect password is input, file operations are not performed.

(b) Operation.... 1) Change : Register a new password in the CPU module.
Or, if the password matches, change the password.

ⓐRead, Write and Display Protect : File names in the designated memory cannot be displayed or written to.

ⓑWrite Protect : Files in the designated memory cannot be written to.Read is possible.

2) Cancel Password : If the password matches, the registered password is deleted from the CPU module.

3) None : The current password is recorded in the GPP function only and is not registered at the CPU module.

4) Change Attribute : File read/write display or write can be prohibited in file units.
(Operation possible even if no entry code is registered.)

(c) Memory....... Designate the memory for which the password is to be registered.

POINT

(1) Password registration is valid for parameter files and program files only. Invalid for other file types. Other file types can be protected by changing attributes for each file.
(2) The keyword registered in the CPU module cannot be read from the CPU module. If you forget the password, CPU module file operations will be impossible. Keep a record of the password, e.g. on paper, and store it in a safe place.
(3) When a keyword is registered, memory for 1 file is occupied.
(When a keyword is registered in the built-in RAM, 4k bytes are occupied.)

9.7 System Display

The following items can be checked by connecting a peripheral device capable of GPP functions to the Q2ASCPU:

(1) The following information relating to the modules actually mounted on the base unit:
  (a) Type
  (b) No. of Occupied Points
  (c) Head X/Y number

(2) The following module information set in the parameters:
  (a) Type
  (b) No. of Occupied Points
  (c) Type Name

(3) The following information relating to the CPU module:
  (a) Status of the RUN/STOP key switch
  (b) Status of the system setting switches
  (c) LED statuses

These items can be checked using the detail HELP display and CPU module panel items in the display menu of the GPP function PLC diagnositics mode.

9.8 LED indication

The Q2ASCPU module has LEDs on its front face that indicate the operating status of the CPU module.
The following shows the meanings of the LED and LED indications.

9.8.1 LED indication

(1) The following shows the meanings of the indications of each of the LEDs are given.

LED Name	Indication Detail
RUN	Indicates the operating status of the CPU module. ON: Operating with the RUN/STOP key switch set to RUN or STEP RUN. OFF: Operation is stopped, with the RUN/STOP key switch in the STOP, PAUSE, or STEP RUN position. An error that stops operation has been detected. Flickering: The RUN/STOP key switch has been turned from STOP to RUN after writing a program in the STOP status. To light, either turn the RUN/STOP key switch RUN→STOP→RUN, or reset operation using the RUN/STOP key switch.
ERROR	Indicates the CPU module error detection status. ON: A self-diagnostics error that does not stop operation, other than a battery error, has been detected.(The operation mode at error occurrence has been set to "Resume" in PC RAS setting in the parameter mode.) OFF: Normal Flickering: An error that stops operation has been detected.
USER	Indicates the CHK instruction detection status, and annunciator (F) statuses. ON: An error has been detected by the CHK instruction, or an annunciator F has come ON. OFF: Normal Flickering: Executing latch clear.
BAT.ALARM	Indicates the battery statuses of the CPU module itself and the memory card. ON: A battery error has occurred due to low battery voltage. OFF: Normal
BOOT	Indicates status of the boot operation execution. ON: Execution has been completed. OFF: The boot operation has not been executed.

(2) The following shows how to turn OFF an LED that is currently ON.(Excluding the reset operation.)

Method for Turning OFF the LED	LED Name			
	ERROR	USER	BAT. ALARM	BOOT
Resolve the cause of the error, then execute the LEDR instruction.	○	○	○	×
Eliminate the cause of the error, then reset the error using special relay SM50 and special register SD50. (Restricted to error which do not stop operation.)*1	○	○	○	×
Operate the special relay SM202 and special register SD202 to turn off the LED.*1	×	○	×	○

○: Valid ×: Not valid

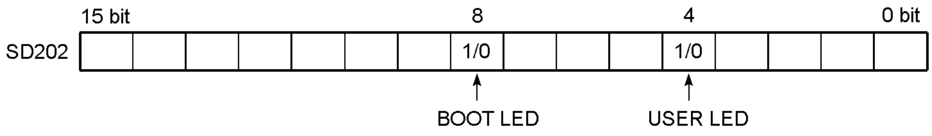
*1 Explanation of special relays and special registers

SM50.......... When turning OFF→ON, resets the error corresponding to the error code stored in SD50.

SD50.......... Stores the error code of the error to be reset.
(For details on error codes, refer to Section 22.3.3.)

SM202........ When turning OFF→ON, turns OFF the LEDs corresponding to each of the bits of SD202.

SD202........ Designates the LED to be turned OFF. (The LEDs that can be turned OFF are the USER LED and the BOOT LED only.)



A bit setting of "1" indicates that the bit is to be turned OFF, "0" indicates that it is not to be turned OFF.
The following shows the setting possibilities (all hexadecimal notation):

- To turn both LEDs OFF: SD202 = 110H
- To turn only the BOOT LED OFF: SD202 = 100H
- To turn only the USER LED OFF: SD202 = 10H

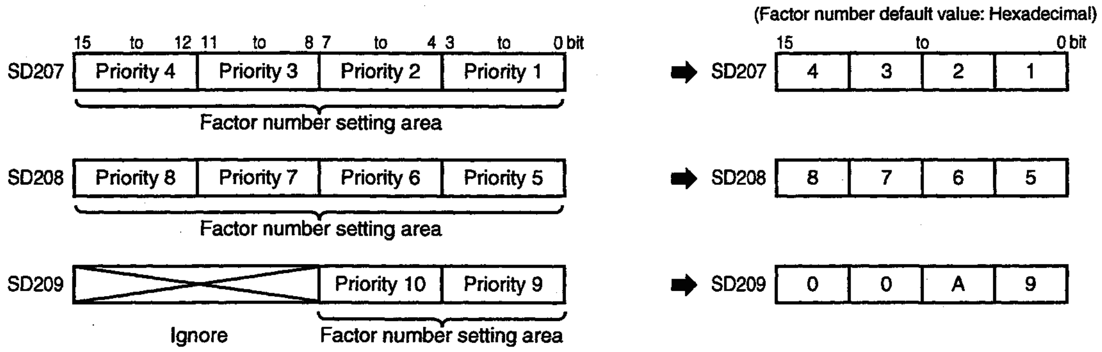
(3) Method for stopping ERROR LED, USER LED, and BAT.ALARM LED indications
ERROR LED, USER LED and BAT. ALARM LED have the same priority order as described for LED indications in Section 9.8.2.
If an error item number is deleted from this order of priority, the LED does not light even if the error corresponding to that error item number occurs.
(For details on the setting method, refer to the POINT in Section 9.8.2.)

9.8.2 Priority setting

If several errors occurred at a time, the indication conforms to the following conditions.

1) Stop error is indicated unconditionally.
2) Operation continue error are indicated in accordance with error item numbers in an order of priority set by default.
   Priorities can be changed. (set with special registers SD207 to SD209)
3) If several errors with the same priority occur, a first detected error is indicated.

The following shows how to set priorities in special registers SD207 to SD209.



The following shows the details of the error item numbers and default for priorities which is set in special registers SD207 to SD209.

Order of priority	Error Item No. (Hex.)	Description	Remark
1	1	AC DOWN	AC power/DC power OFF
2	2	UNIT VERIFY ERR. FUSE BREAK OFF SP.UNIT ERROR	I/O module verification Fuse blown Special function module access error
3	3	OPERATION ERROR LINK PARA.ERROR SFCP OPE.ERROR SFCP EXE.ERROR	Operation error Link parameter error SFC instruction operation error SFC program execution error
4	4	ICM.OPE.ERROR FILE OPE.ERROR EXTEND INST.ERROR	Memory card operation error File access error Extended instruction error
5	5	PRG.TIME OVER	Constant scan setting time over error Low-speed execution monitoring timeout
6	6	CHK instruction	
7	7	Annunciator	
8	8	–	
9	9	BATTERY ERR.	
10	A	Clock data	

POINT

(1) When LED indicator is left OFF for the error occurrence above, set the factor number area to 0, which stores the applicable factor numbers from SD207 to SD209.

Example: To set the ERROR LED to remain OFF when a fuse blown error occurs, set "0" in the item number setting area whose item number is "2".

SD209				SD208				SD207			
0	0	A	9	8	7	6	5	4	3	0	1

Since the item number "2" is not set, the ERROR LED remains OFF even if a fuse blown error is detected.The ERROR LED remains OFF even if another error whose error item number is "2" is detected (I/O module verify error, special function module verify error).

(2) Even if the LED is set to remain OFF, SM0 (the diagnostics error flag) is still turned ON, SM1 (the self-diagnostics error flag) is still turned ON, and the error code is stored in SD0 (CPU diagnosis error register).

10 OTHER FUNCTIONS

10.1 Function List

The following list shows the rest of the functions.

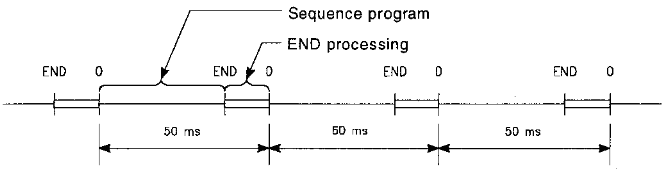
Item		Description	Reference
Constant scan		Performs a program at fixed intervals regardless of the actual program scan time.	Section 10.2
Latch function		Retains the device data when resetting the CPU module while the programmable controller power is OFF.	Section 10.3
Setting of the output status when switching from STOP to RUN		Sets the output (Y) status when the CPU module is switched from STOP to RUN (Re-outputting the outputs before STOP/Outputting the outputs after performing operation).	Section 10.4
Clock function		Runs the internal clock of the CPU module.	Section 10.5
Remote operation		Operates the Q2ASCPU from a remote place.	Section 10.6
	Remote RUN/STOP	Starts or stops the CPU module operation.	Section 10.6.1
	Remote STEP-RUN	Performs a step operation to the CPU module.	Section 10.6.2
	Remote PAUSE	Suspends the CPU module operation.	Section 10.6.3
	Remote RESET	Resets the CPU module.	Section 10.6.4
	Remote latch clear	Clears the CPU module latch data.	Section 10.6.5
	Relationship between remote operation and CPU module RUN/ STOP key switch	Explains the relationship between the CPU module RUN/STOP key switch setting and operation when performing remote operation.	Section 10.6.6
Terminal setting		Uses the Q6PU programming unit's indicator and key input.	Section 10.7
	Message display	Displays messages on the indicator of the Q6PU.	Section 10.7.1
	Key input operation	Reads key input from the Q6PU.	Section 10.7.2
Reading module access time intervals		Monitors the access time intervals (The time between the acceptance of one CPU module access and the acceptance of the next CPU module access) for special function modules, network modules, and peripheral devices.	Section 10.8

For details of GPP function operation, refer to the GX Developer Operating Manual or the Type SW☐IVD-GPPQ GPP Software package Operating Manual (Online).
For details of the Q6PU operation, refer to the Q6PU Operating Manual.

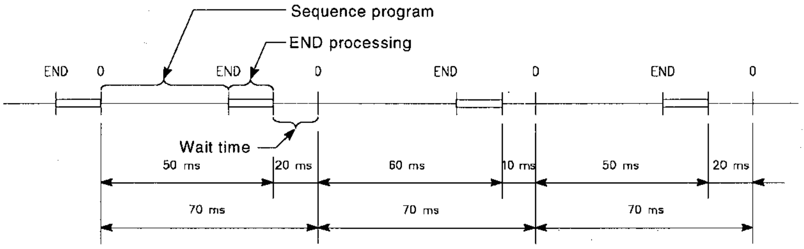
10.2 Constant Scan

(1) Constant scan
In the Q2ASCPU, the scan time varies since the processing time differs depend on the execution status of the instructions used in the sequence program.
Constant scan is a function whereby the sequence program is repeatedly performed while maintaining constant scan time.

Scan time when the constant scan is not used



Scan time when the constant scan time is set to 70ms



Scan time when the constant scan time is set to 100ms while performing multiple programs

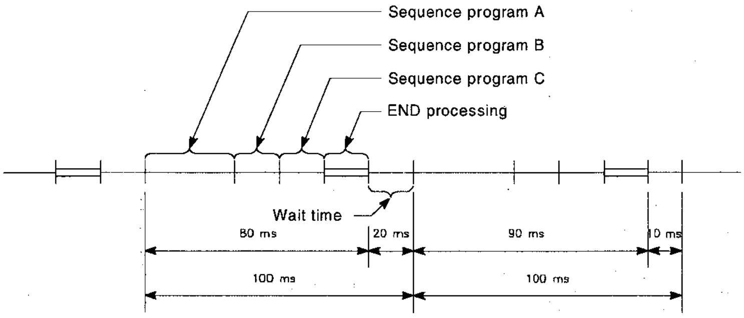


Fig. 10.1 Constant scan operation

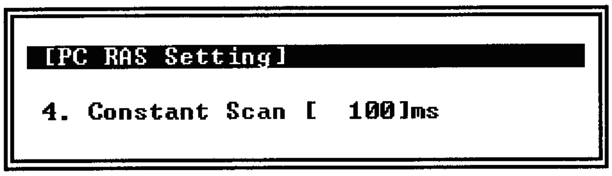
When the low-speed execution type program is used, either this constant scan function or a low-speed program execution time has to be set.
(For details, refer to the QnACPU Programming Manual (Fundamentals).)

(2) Setting the constant scan time

(a) The setting is made in "PC RAS" in the parameter mode of GPP function.

- When performing the constant scan, set the constant scan time.
- When not performing the constant scan, leave the field blank.

Example: When setting 100ms to "Constant scan"



```
[PC RAS Setting]

4. Constant Scan [  100]ms
```

(b) Set constant scan time that is longer than the maximum scan time of the sequence program. If the scan time of the sequence program is longer than the set value for constant scan time, the Q2ASCPU detects an error code (SD0 = 5010), and the sequence program is performed in the own scan time, ignoring the constant scan time setting.
Make sure that the constant scan time setting is shorter than the set time for WDT (Watchdog timer). If it is longer than the set time for WDT, the Q2ASCPU detects a WDT error and the program execution is stopped.

Set the constant scan time within the following range.

Setting time for WDT > Setting time for constant scan > Maximum scan time of sequence program

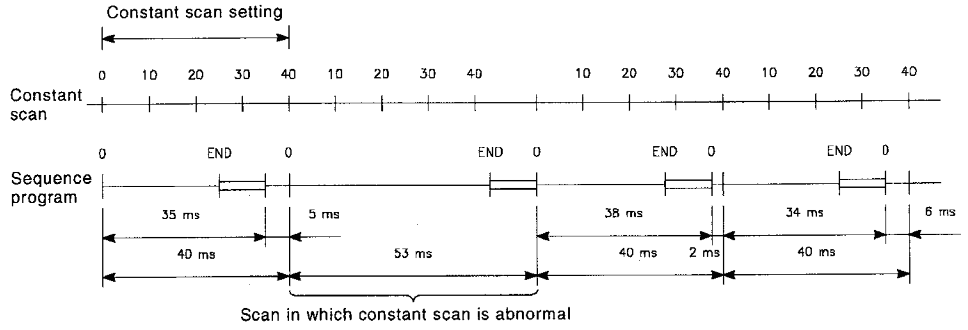


Fig. 10.2 Operation when scan time is longer than constant scan setting time

(c) Sequence program processing is suspended in the wait time between the END processing of the sequence program and the start of the next scan.
However, if an interrupt factor occurs after the execution of END processing, or if there is a low-speed execution type program, the interrupt program or the low-speed execution type program is performed.

(d) Constant scan time error

If there is a low-speed execution type program when performing the constant scan, the constant scan time may be shifted by the time shown below.

$$(\text{Error}) = \left[\begin{array}{l}\text{Maximum processing time}\\ \text{of one instruction in the}\\ \text{low-speed execution type}\\ \text{program}\end{array}\right] + \left[\underline{\underline{\text{low-speed END processing time}}}\right]$$

$$\left(\begin{array}{l}\text{Time taken to execute the END}\\ \text{processing for a low-speed}\\ \text{execution type program}\end{array}\right)$$

The low-speed execution type program is divided and performed within surplus time. Therefore, if one constant scan ends while performing the instruction takes long processing time, the constant scan is completed after finishing the processing of the instruction during execution. The time extended to complete the execution of the instruction is the constant scan error. For details of the instruction processing time, refer to the QCPU (Q mode)/QnACPU Programming Manual (Common Instructions).

10.3 Latch Function

When the programmable controller power is turned ON, the CPU module is reset using the RUN/STOP key switch, or a instantaneous power failure lasting longer than the allowable momentary power interruption time occurs, the all device values in the Q2ASCPU are cleared, and the default values are set in the devices (Bit devices: OFF, word devices: 0).
The latch function retains the data in the devices when performing these operations.
The operations in the program are the same whether or not the latch function is used.

(1) Application of the latch function
The latch function can be used when continuing the control to retain data such as production quantities, numbers of defects, addresses, etc., even if a instantaneous power failure longer than the allowable time occurs.

(2) Devices that can be latched
(a) The following devices can be latched.
1) Latch relay
2) Link relay
3) Annunciator
4) Edge relay
5) Timer
6) Retentive timer
7) Counter
8) Data register
9) Link register

POINT

Even if a latch designation is set for a device, the device will not be latched if a local device designation or device initial value designation is made.

(b) The latch range is set on the "Device" in the parameter mode of GPP function. In latch range setting, it is possible to set a range within which the latch clear key is effective (Latch (1) Start) and a range within which the key is not effective (Latch (2) Start). For details on device latch ranges for each device, refer to the QnACPU Programming Manual (Fundamentals).

POINT

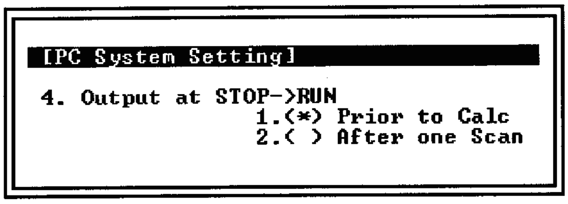
The devices data in the latch range are retained by the battery (A6BAT) installed in the CPU module.
(1) Even if a sequence program is written to a memory card and ROM operation is performed, the battery is required for the latch function.
(2) If the battery connector is disconnected from the CPU module connector while the programmable controller power is OFF, the devices data in the latch range is lost.

(3) Clearing the device data in the latch range
  (a) To clear the devices data in a latch range and set the default values instead, perform "Latch clear". When the latch clear is performed, the devices data in the non-latched range is also cleared.
  However, the devices for which the latch clear key has been set to Disable in the Parameter are not cleared by performing latch clear.
  (b) For the methods of performing latch clear, refer to Section 12.4.

10.4 Setting of the Output (Y) Status When Switching from STOP to RUN

When the RUN or other status is changed to the STOP status, the CPU module stores the output (Y) in the RUN status into the programmable controller and turns all outputs (Y) OFF.

In this function, whether to re-output the outputs (Y) when switching from STOP to RUN or to output them after an operation can be set in the "PC system" in the parameter mode of GPP function .



```
[PC System Setting]

4. Output at STOP->RUN
                1.(*) Prior to Calc
                2.( ) After one Scan
```

(a) Re-output (Prior to Calc).....................The output (Y) status immediately before the STOP status is output, and then the sequence program is calculated.

(b) Output after operation execution
(After one Scan).................................The output is OFF status.
The output (Y) will be output after the sequence program operation is executed.

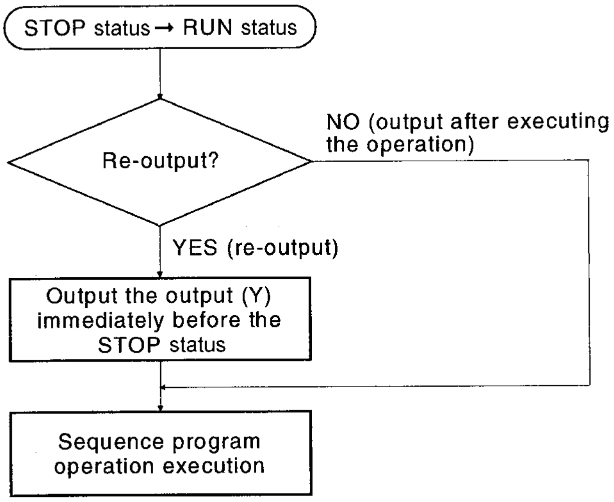


Fig. 10.3 Processing when a programmable controller is switched from STOP to RUN

10.5 Clock Function

The Q2ASCPU has a clock in the CPU module.
Since the clock data can be read in the sequence program, it can be used for time control of the user system.
In addition, the clock data can also be used for time control to the functions performed by the CPU module, such as the breakdown history.
Clock operation by the clock function is continued with the battery when the programmable controller power is turned OFF or a instantaneous power failure lasting longer than the allowable momentary power interruption time occurs.

POINT

The CPU module system uses the clock data for a breakdown history.When using a CPU module, be sure to set the correct time first.

(1) Clock data
The clock data is composed of the year, month, day, hour, minute, second, and day of the week used by the clock element in the programmable controller CPU, as shown below.

Data name	Description	
Year	Last two digits of the year	
Mon	1 to 12	
Sun	1 to 31(Leap year, automatic identification)	
Hour	0 to 23 (24-hour system)	
Minute	0 to 59	
Second	0 to 59	
Day of the week	0	Sunday
	1	Monday
	2	Tuesday
	3	Wednesday
	4	Thursday
	5	Friday
	6	Saturday

(2) Accuracy
The accuracy of the clock function depends on the ambient temperature, as shown below.

Ambient Temperature	Accuracy (daily variance)
0℃	-1.7 to + 4.9s (TYP. +1.7S)
+ 25℃	-1.0 to + 5.2s(TYP.+2.2S)
+ 55℃	-7.3 to + 2.5s(TYP.-1.9S)

(3) Writing clock data to the clock elements

(a) Use the following procedure to write clock data to the clock elements.

1) Writing from a peripheral device

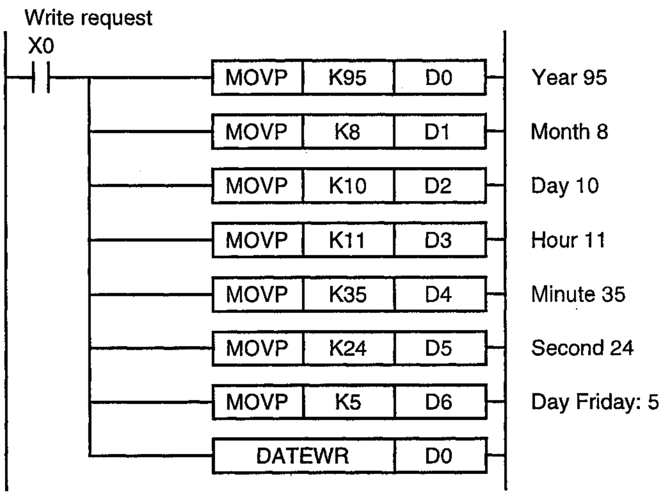
- When using GPP function, clock data can be written to the clock elements using the "Set clock" of the PLC menu in PLC diagnostics mode.
- When using the Q6PU, clock data can be written to the clock elements by using the clock monitor option in the monitor functions of the PLC system in the other mode.

(For details on the operation for each peripheral device, refer to the Operating Manual for each.)

2) Writing from a program

Clock data is written to the clock elements using the clock instruction (DATEWR).

The following shows the example of the program.



For details on the DATEWR instruction, refer to the QCPU (Q mode)/ QnACPU Programming Manual (Common Instructions).

POINT

(1) Clock data is not written to clock elements in advance.Write clock data to the clock elements before using the Q2ASCPU.

(2) Even if partly changing the clock data, rewrite all data to the clock elements.

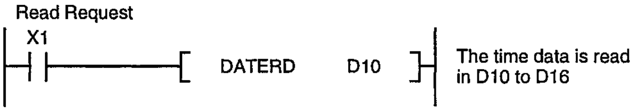
(3) If the nonexistent time is written to the clock elements, normal clock operation is impossible.

Example

Setting "13" to the month.

(4) Clock data read

(a) To read clock data to data registers, use the clock data read instruction (DATERD) in the program.
An example of a program using the instruction is shown below.



For details on the DATERD instruction, refer to the QCPU (Q mode)/QnACPU Programming Manual (Common Instructions).

(b) To read the clock data to SD210 to SD213, turn SM213 ON from a sequence program or a peripheral device.

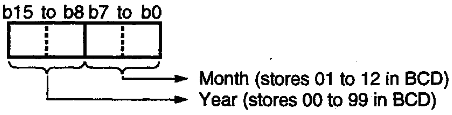
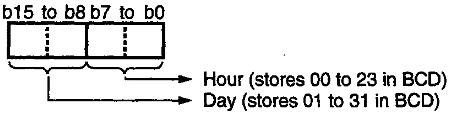
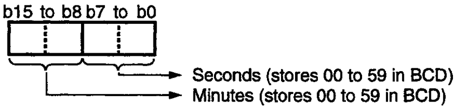
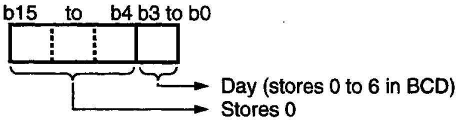
(5) Special relays and special registers for reading/writing clock data

The section explains the special relays and special registers used for setting data and reading clock data for clock operation.

(a) Special relays used for the clock function

Device	Name	Description
SM210	Clock data set request	• Writes clock data to the special registers (SD210 to SD213) and performs clock operation. • Writes the clock data stored in SD210 to SD213 to the clock elements after execution of the END instruction in the scan in which SM210 turns from OFF to ON.
SM211	Clock data error	• Used to determine whether or not there are any errors when the clock data is set. • Turns ON if any data is not a BCD cord.
SM213	Clock data read request	• Reads the clock data stored in special registers SD210 to SD213. • When SM213 is ON, the clock data is read to SD210 to SD213 after execution of the END instruction.

(b) Special registers used for clock data

<table>
<tr><th>Device</th><th>Name</th><th>Description</th></tr>
<tr><td>SD210</td><td>Clock data
(year, month)</td><td>• The year and month are recorded as follows.The year data is the last two digits of the year.
 </td></tr>
<tr><td>SD211</td><td>Clock data
(day, hour)</td><td>• The day and hour are recorded as follows.
 </td></tr>
<tr><td>SD212</td><td>Clock data
(minute, second)</td><td>• The minute and second are recorded as follows.
 </td></tr>
<tr><td>SD213</td><td>Clock data
(day of the week)</td><td>• The day of the week is recorded as follows.

• The settings for the day of the week are as follows:
<table><tr><td>Day of the week</td><td>Sun</td><td>Mon</td><td>Tue</td><td>Wed</td><td>Thu</td><td>Fri</td><td>Sat</td></tr><tr><td>Storage data</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td></tr></table></td></tr>
</table>

10.6 Remote Operation

With the Q2ASCPU, the operating status of the CPU module can be controlled from external sources (GPP function, intelligent special function module, remote contact, etc.).

REMARK

In this chapter, a serial communication module is used as an example of an intelligent special function module.

10.6.1 Remote RUN/STOP

Remote RUN/STOP refers to the function that sets the Q2ASCPU to RUN or STOP from an external source while the CPU module RUN/STOP key switch is set to the RUN position.

(1) Application of remote RUN/STOP

Remote RUN/STOP operation from remote location is useful in the following cases.

(a) When the CPU module is installed in an inaccessible location

(b) When setting the CPU module in a control panel to RUN/STOP from an external source

(2) Operation for remote RUN/STOP

The following shows the program operations to which remote RUN/STOP is performed.

(a) Remote STOP......... The program is performed up to the END instruction, then STOP state is established.

(b) Remote RUN........... When remote RUN is performed with the CPU in STOP set by remote STOP, the program will be in RUN state again and be performed from step 0.

(3) Method for performing remote RUN/STOP

The following two methods are available for performing remote RUN/STOP.

(a) Method using a remote RUN contact

The remote RUN contact is set in the PLC system in the parameter mode of GPP function.

The settable device range is from input X0 to 1FFF.

Remote RUN/STOP can be performed by switching the remote RUN contact ON/OFF.

1) When the remote RUN contact is OFF, the CPU module is in RUN state.

2) When the remote RUN contact is ON, the CPU module is in STOP state.

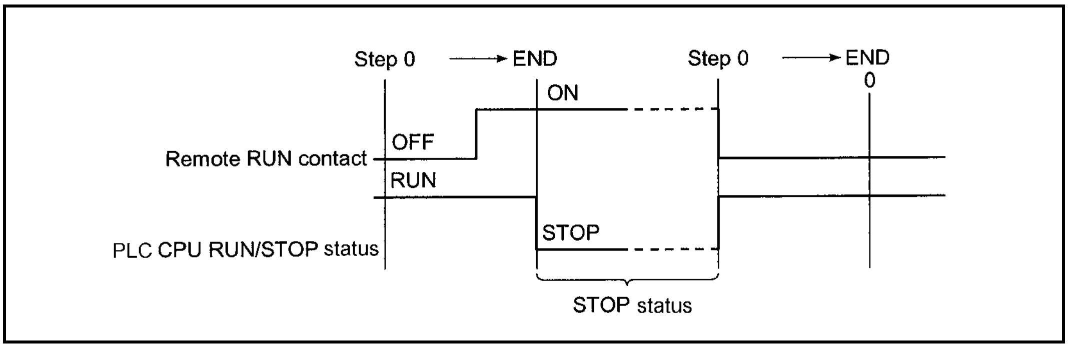


Fig. 10.4 Time chart for RUN/STOP switching with remote RUN contact

(b) Method using GPP function, serial communication module, etc.
The CPU module can be set to RUN or STOP by remote RUN/STOP operation from GPP function, or a serial communication module, etc.
The operation using GPP function can be performed in the Remote operation of the PLC menu in any mode.

The control using a serial communication module is performed with the commands in the dedicated protocol.
For details on serial communication module control, refer to the Serial Communication Module User's Manual.

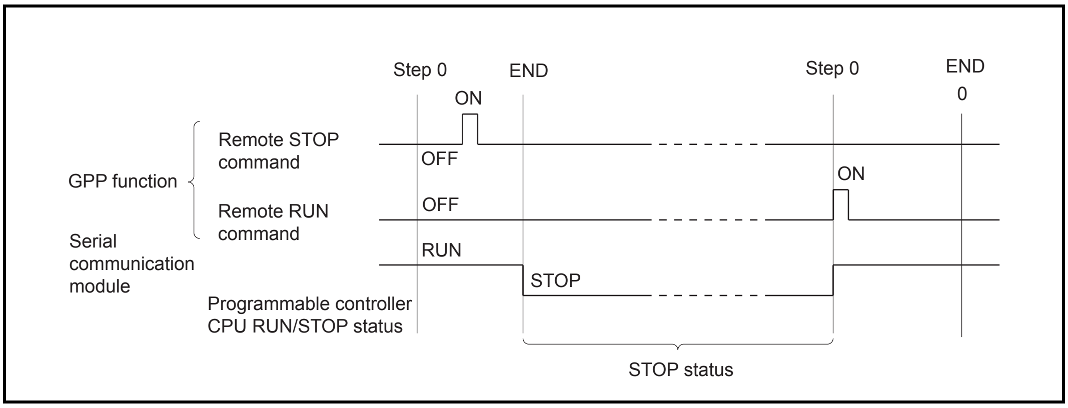


Fig. 10.5 Time chart for remote RUN/STOP switching with GPP function or a serial communication module

(4) Precautions
(a) Since the STOP state has a priority in the Q2ASCPU, pay attention to the following points.
1) In the Q2ASCPU, if remote STOP is performed from any one of remote RUN contact, GPP function, serial communication module, etc., the QnACPU will be STOP.
2) In order to set the Q2ASCPU to RUN again after it has been set to STOP by remote STOP, all external factors which set remote STOP (Remote RUN contact, GPP function, serial communication module, etc.) have to be set to RUN.

REMARK

The RUN/STOP status is defined as follows.

・RUN status..............	Status in which the sequence program is repeatedly performed from step 0 to the END instruction.
・STOP status...........	Status in which the sequence program operation is stopped and all outputs (Y) are OFF.

10.6.2 Remote STEP-RUN

Remote STEP-RUN refers to the function whereby the step run of the Q2ASCPU is performed from GPP function while the RUN/STOP key switch of the module is in RUN position.
"Step run" is program execution that operates by one step at a time, starting from the designated step.
For details on step run, refer to Section 8.7.

(1) Application of remote STEP-RUN
When debugging the system, for example, the program can be performed while checking its execution and the contents of each device.

(2) Method for performing remote STEP-RUN
The procedure for remote STEP-RUN is as follows.
1) Set the RUN/STOP key switch of the CPU module to RUN position.
2) Perform STEP-RUN operation with GPP function.

10.6.3 Remote PAUSE

Remote PAUSE refers to the function that performs PAUSE function to the Q2ASCPU from an external source while the CPU module RUN/STOP key switch is set to the RUN position.
The PAUSE function stops a CPU module operation while retaining the ON/OFF status of all outputs (Y).

(1) Application of remote PAUSE
This function can be used to retain the output (Y) with ON status even if the CPU module is in STOP due to process control.

(2) Methods for remote PAUSE
The following two methods are available for performing remote PAUSE.
(a) Method using a remote PAUSE contact
The remote PAUSE contact is set in the PLC system in the parameter mode of GPP function.

The settable device range is from input X0 to 1FFF.

1) When the scan END processing is performed with both the remote PAUSE contact and the PAUSE enable flag (SM206) are ON, the PAUSE status contact (SM204) turns ON.
When performing up to the END instruction of the scan following the scan in which the PAUSE status contact turned ON, the PAUSE state is established and operation is stopped.
2) When turning the remote PAUSE contact OFF or turning SM206 OFF with GPP function, the PAUSE status is reset and the sequence program operation is again performed from step 0.

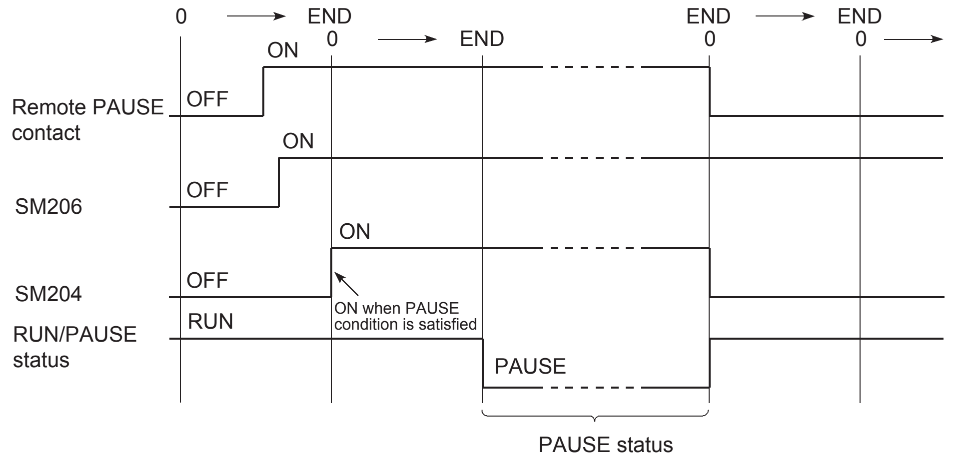


Fig. 10.6 Time chart for PAUSE with remote PAUSE contact

REMARK

When the remote RUN contact is made same as the remote PAUSE contact, the remote PAUSE contact will be invalid.

(b) Methods using GPP function or a serial communication module
The remote PAUSE operation can be performed from GPP function or from a serial communication module.
The operation using GPP function can be performed in the Remote operation of the PLC menu in any mode.

The control using a serial communication module is performed with the commands in the dedicated protocol.
For details on serial communication module control, refer to the Serial Communication Module User's Manual.

1) When the END processing of the scan in which the remote PAUSE command has received from GPP function is performed, the PAUSE status contact (SM204) turns ON.
When performing up to the END instruction of the scan following the scan in which the PAUSE status contact turned ON, the PAUSE status is established and the operation is stopped.
2) When the remote RUN command is received from GPP function, the sequence program operation is again performed from step 0.

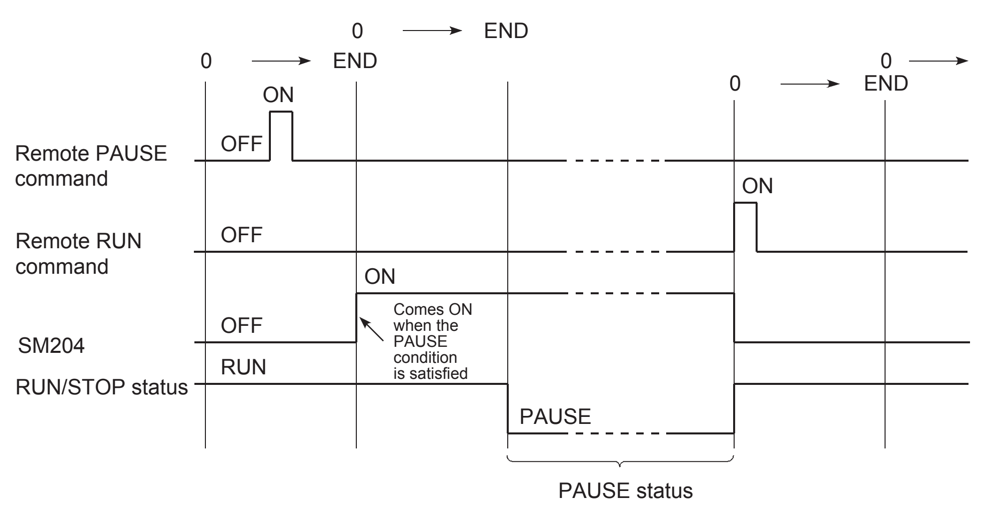
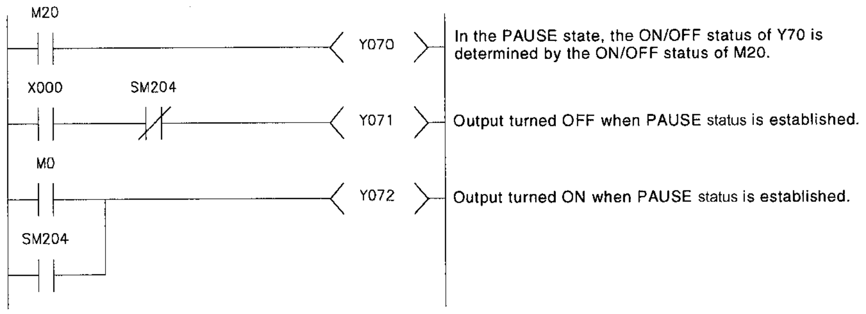


Fig. 10.7 Time chart for PAUSE with GPP function

POINT

To arrange for outputs (Y) to be turned ON or OFF when the PAUSE status is established, provide an interlock using the PAUSE state contact (SM204).



In the PAUSE state, the ON/OFF status of Y70 is determined by the ON/OFF status of M20.

Output turned OFF when PAUSE status is established.

Output turned ON when PAUSE status is established.

10.6.4 Remote RESET

Remote RESET is a function for resetting the Q2ASCPU by operation from an external device while the CPU module is in STOP.
Resetting is also possible even when the RUN/STOP key switch on the CPU module is set to the RUN position if the CPU module is stopped by an error detectable by the self-diagnostics function.

POINT

Remote RESET cannot be performed when the CPU module is in RUN.

(1) Application of remote RESET
Remote RESET can be used to reset the CPU module by remote operation when an error has occurred in the place from where the CPU module cannot be directly operated.

(2) Methods for remote RESET
Remote RESETcan only be performed by operation from GPP function or a serial communication module.

(a) Regardless of whether reset is performed from GPP function or a serial communication module, the setting to enable remote RESET has to be made in the Parameter before performing the reset operation.

The remote RESET "Enable/Disable" setting is made in the PLC system in the parameter mode of GPP function.

(b) When the parameter is set to "Allow" in the "Remote reset" and written to the CPU module, resetting is performed with remote operation.
- When using GPP function, perform the reset in the PLC menu in any mode.
- When using a serial communication module, perform the reset with dedicated protocol commands.

For details on serial communication module control, refer to the Serial Communication Module User's Manual.

10.6.5 Remote latch clear

Remote latch clear is a function for resetting the latched device data of the Q2ASCPU while the CPU module is in STOP by using such as a GPP function.

POINT

Remote latch clear cannot be performed when the CPU module is in RUN.

(1) Application of remote latch clear

Remote latch clear is useful for latch clear operation when the CPU module is at the locations below: In this case, the function is used in combination with the remote RUN/STOP function.

- When the CPU module is installed in an inaccessible location
- When performing latch clear to the CPU module in a control panel from an external source

(2) Methods for remote latch clear

Remote latch clear can only be performed by operation from GPP function or a serial communication module.

- The operation using GPP function can be performed in the Remote operation of the PLC menu in any mode.
- The control using a serial communication module is performed with the commands in the dedicated protocol.

For details on serial communication module control, refer to the Serial Communication Module User's Manual.

POINT

1. According to the device latch ranges set in "Device" in parameter mode, there are ranges within which latch clear is valid and ranges within which it is not valid.Remote latch clear is only valid for devices set in the range for which "Latch clear valid" is set.
2. When remote latch clear is performed, devices that are not latched are also cleared.

10.6.6 Relationship between remote operation and CPU module RUN/STOP key switch

Using the combination of the remote operation and the RUN/STOP key switch of the CPU module explained in Section 10.6.1 through Section 10.6.5, the operating status of the Q2ASCPU is determined as follows.

Key switch	Remote Operation					
	RUN[*1]	STEP-RUN	STOP	PAUSE[*2]	RESET[*3]	Latch Clear
RUN	RUN	STEP-RUN	STOP	PAUSE	Operation is not possible.[*4]	Operation is not possible.[*4]
STOP	STOP	STOP	STOP	STOP	RESET	Latch Clear

*1 If performed using a remote RUN contact, beforehand set "RUN-PAUSE contacts" in the PLC system in parameter mode.

*2 If performed using a remote PAUSE contact, beforehand set "RUN-PAUSE contacts" in the PLC system in parameter mode.Furthermore, the remote PAUSE enable coil (SM206) has to be turned ON in advance.

*3 "Remote reset" field in the PLC system has to be set to "Allow"in parameter mode.

*4 The operation status can be RESET if the CPU module is stopped by remote operation.

When the RUN/STOP key switch is set to RUN and multiple remote operation requests are received, the CPU module first performs the operation with the highest priority.

Remote operation	RUN	STEP-RUN	STOP	PAUSE	RESET	Latch Clear
Order of priority	4)	3)	1)	2)	-	-

The order of priority increases from (4) to (1).

10.7 Terminal Operation

This function sets the Q6PU programming unit in the terminal mode and performs the data communications shown below by using the instructions for peripheral devices of theQ2ASCPU.

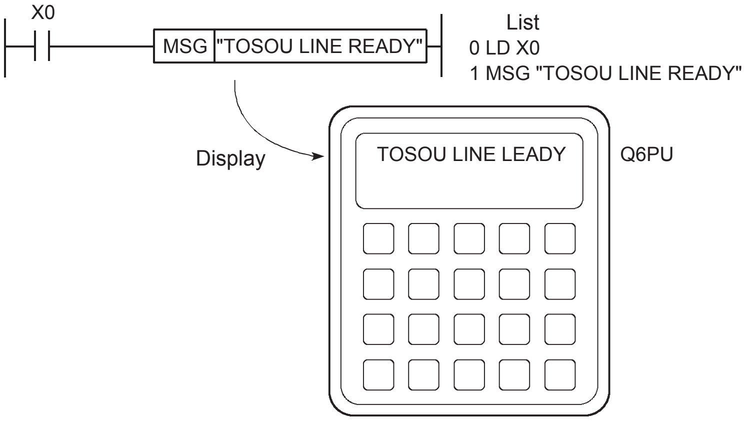
1) Display of messages from the Q2ASCPU on the display of the Q6PU.
2) Storage of the Q6PU key input data in the devices of the Q2ASCPU.

In this way, the Q6PU can be used as a terminal of the Q2ASCPU.
These functions are explained from the next section.
However, for details on the instructions for peripheral devices, refer to the QCPU (Q mode)/QnACPU Programming Manual (Common Instructions).

10.7.1 Operation for message display

Specified character strings can be displayed on the Q6PU using the MSG instruction for peripheral devices.
Furthermore, character strings can be displayed with GPP function by using the CPU messages of the Display menu in the PLC diagnostics mode.

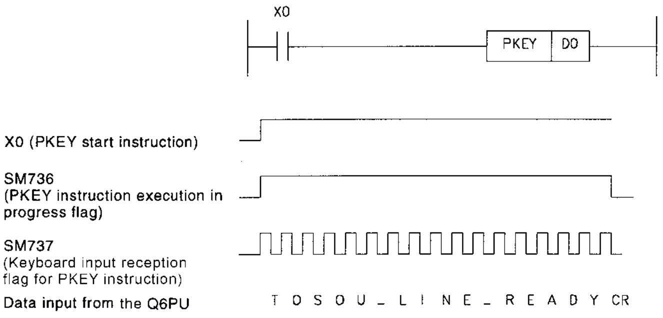
Example: Program to display "TOSOU LINE READY" as a message No.1 on the Q6PU when X0 is turned ON.



10.7.2 Key input operation

Character string data input at the Q6PU can be stored as ASCII data without change in specified devices by using the PKEY instruction for peripheral device.Data input ends when a CR code is input or when the 32nd character is input.

Example: Program to input "TOSOU LINE READY" on the Q6PU when X0 is turned ON.



List
```
0 LD X0
1 PKEY D0
```

Data registers

D0	17	
D1	4FH	54H
D2	4FH	53H
	20H	55H
	49H	4CH
	45H	4EH
	4CH	20H
	41H	45H
	59H	44H
D9	00	0dH

Number of key input data

Data is input in this way.

10.8 Reading Module Access Time Intervals

The Q2ASCPU can monitor the access interval time (The time between one access reception and the next access reception) for intelligent special function modules, network modules, data link modules, or GPP function. This enables to grasp the frequency of accesses to the CPU module from external sources.

The operation for reading the module access interval time involves the following special relay and special registers.

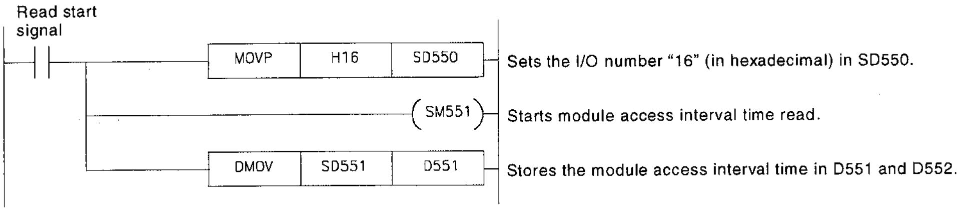
(1) Special relay

Number	Name	Description
SM551	Read module service interval	When this relay turns from OFF to ON, the module access interval time for the special function module specified in special register SD550 is read into special registers SD551 to SD552. ON : Read OFF : Ignored

(2) Special register

Number	Name	Description
SD550	Service interval measurement module	Set the I/O number of the module whose access interval time is to be measured. Set the I/O number of the peripheral device connected to the RS-422 interface of the CPU module to FFFFH. Also set the I/O number of the upper 2 digits in the 3-digit representation.
SD551 to SD552	Service interval time	When SM551 is turned ON, it stores the interval time for access from the module specified at SD550. SD551: 1ms units (Range: 0 to 65535) SD552: 1μs units (Range: 0 to 900, stored every 100μs) Example: When the module access interval time is 123.4ms: SD551=123, SD552=400

Program example:
Program for reading the module access interval time of the special function module at X/Y160.



POINT

To read the access interval time for access from GPP function at another station in the network, set the I/O number of the network module.

REMARK

- The module access interval includes a transient request interval such as a monitor, a test and a program read/write.
  The access interval via cyclic communication from a network module or a data link module is not stored.

11 COMMENTS THAT CAN BE STORED IN Q2ASCPU

11.1 Function List

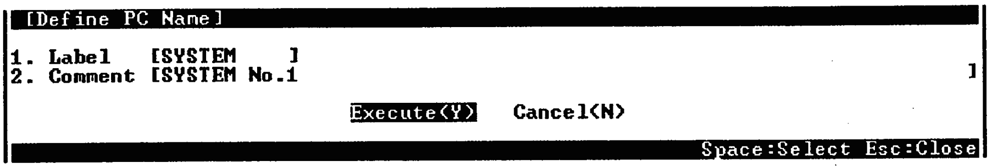
The Q2ASCPU can store various types of comments. This has improved the CPU module operability, allowing users other than programmers to read programs easily.
The types of comments that can be stored in the Q2ASCPU are listed in the table below.

Item	Function	Refer to
PLC name	Naming the CPU module to be used.	Section 11.2
Drive title	Assigning a title to each drive.	Section 11.3
File title	Assigning a title to each file.	Section 11.4
Device comment	Assigning comments and/or labels to devices used in a program.	Section 11.5
Statements/notes	Assigning comments to each program step number or P or I pointer.	Section 11.6
Initial device value comment	Assigning a comment to the initial device value file.	Section 11.7

For details on the setting method for each function, refer to the GX Developer Operating Manual or SW☐IVD-GPPQ Operating Manual (Offline).

11.2 PLC name

PLC name appends a comment to a CPU module to make it easier to confirm the CPU module when accessing the Q2ASCPU by GPP function.
Two types of PLC names can be set: labels and comments. The settings are made on the "Define PC name" screen in the parameter mode of GPP function.



```
[Define PC Name]
1. Label   [SYSTEM    ]
2. Comment [SYSTEM No.1                                                    ]
                      Execute(Y)   Cancel(N)
                                          Space:Select Esc:Close
```

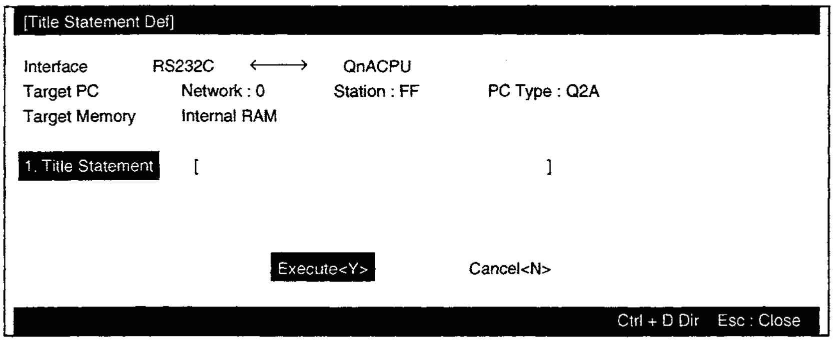
The setting details are indicated in the table below.

Item	Setting	Setting range	Default value
Label	Set a label for the CPU module.	Up to 10 characters	No setting
Comment	Set a comment for the CPU module.	Up to 64 characters	

11.3 Drive Title

The drive title function assigns a title to a drive to allow users to easily identify what file is stored in the built-in RAM or memory card.

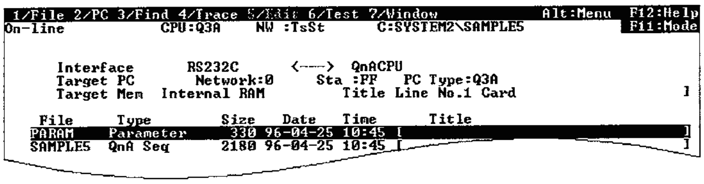
Drive titles are created on the "Title Statement Def" screen under the PC menu in the online mode of GPP function.



Interface RS232C ⟵⟶ QnACPU
Target PC Network : 0 Station : FF PC Type : Q2A
Target Memory Internal RAM

1. Title Statement [ ]

A created title is displayed on the screen as shown below.



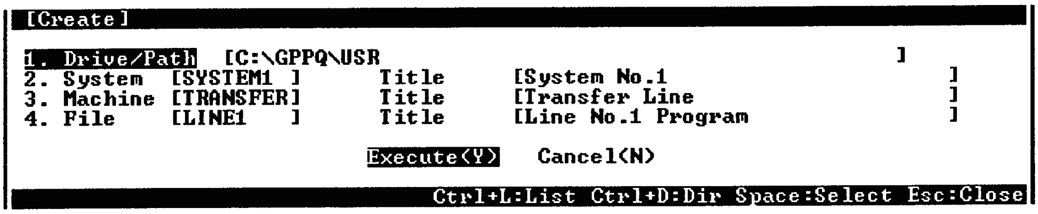
```
On-line          CPU:Q3A     NW :TsSt       C:SYSTEM2\SAMPLES5

    Interface      RS232C       <---> QnACPU
    Target PC        Network:0     Sta :FF   PC Type:Q3A
    Target Mem   Internal RAM        Title Line No.1 Card              ]

  File      Type           Size   Date     Time        Title
 PARAM     Parameter        330 96-04-25 10:45 [                       ]
 SAMPLE5   QnA Seq         2180 96-04-25 10:45 [                       ]
```

POINT
Note that creating a drive title uses an area equivalent to one file in each memory.

11.4 File Title

The file title function allows file titles to be assigned to files so that the contents of the files can be figured out.

File titles are set in file setting performed when starting GPP function, or in PLC writing from the PLC menu in any mode. Up to 32 characters can be used.



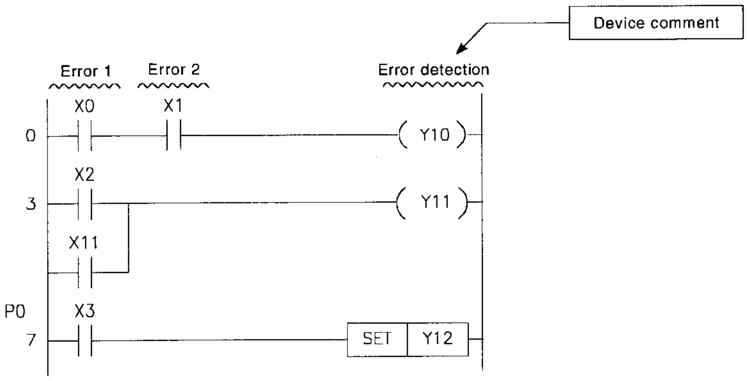
```
[Create]

1. Drive/Path  [C:\GPPQ\USR                                     ]
2. System   [SYSTEM1 ]      Title      [System No.1                 ]
3. Machine  [TRANSFER]      Title      [Transfer Line               ]
4. File     [LINE1    ]     Title      [Line No.1 Program           ]

                       Execute(Y)   Cancel(N)

                    Ctrl+L:List Ctrl+D:Dir Space:Select Esc:Close
```

File titles are stored in the corresponding created files.

Note that they are not stored in any files for file registers.

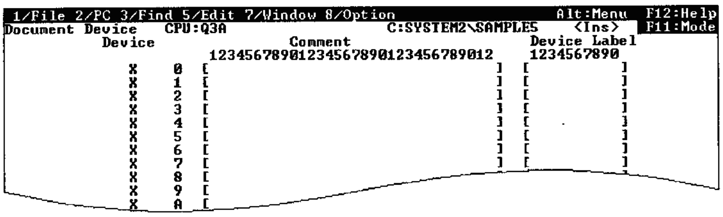
11.5 Device Comment

The device comment function displays comments assigned to respective devices so that programs can be read easily.
In addition, by setting "Xtype" for the CPU type with GPP function, programs can be created using labels instead of devices.



POINT
A memory card is required to create device comments and store a device comment file in a CPU module.

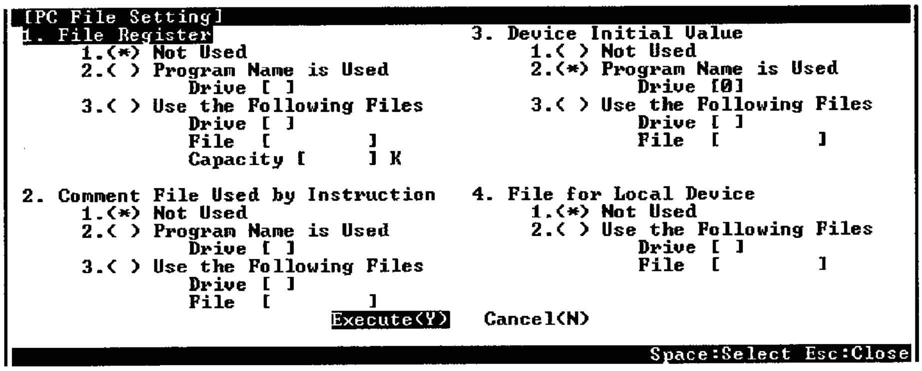
(1) Device comments are set in the documentation mode of GPP function.
Up to 32 characters are used for each comment and up to 10 characters for each label (device label name).



```
 1/File 2/PC 3/Find 5/Edit 7/Window 8/Option              Alt:Menu  F12:Help
Document Device    CPU:Q3A                 C:SYSTEM2\SAMPLES     <Ins>   F11:Mode
             Device                Comment                      Device Label
                        12345678901234567890123456789012        1234567890
               X    0  [                                ]  [          ]
               X    1  [                                ]  [          ]
               X    2  [                                ]  [          ]
               X    3  [                                ]  [          ]
               X    4  [                                ]  [     .    ]
               X    5  [                                ]  [          ]
               X    6  [                                ]  [          ]
               X    7  [                                ]  [          ]
               X    8  [
               X    9  [
               X    A  [
```

- Comments and labels can be assigned to the following devices.
  Device name:X, Y, M, L, F, SM, B, SB, V, T (present value), C (present value), ST (present value), D, SD, W, SW, R, ZR, P, I, U□\G□,
  J□\X, J□\Y, J□\B, J□\SB, J□\W, J□\SW, BL□\S, BL□\TR
  (When P or I comments are used as pointers for programs such as subroutine or interrupt programs, they are not displayed. To display these comments, make them displayed as pointer statements. (Refer to Section 11.6))

(2) When using comments with application instructions (LEDC, PRC, etc.), if a device comment file has been written to the CPU module, enable one of the options in the parameter setting for the device comment file.
This setting is made at "2. Comment file used in a command" on the "PLC file" screen in the parameter mode of GPP function.



```
[PC File Setting]
1. File Register                          3. Device Initial Value
     1.(*) Not Used                            1.( ) Not Used
     2.( ) Program Name is Used                2.(*) Program Name is Used
              Drive [ ]                                 Drive [0]
     3.( ) Use the Following Files             3.( ) Use the Following Files
              Drive [ ]                                 Drive [ ]
              File  [         ]                         File  [         ]
              Capacity [      ] K

2. Comment File Used by Instruction       4. File for Local Device
     1.(*) Not Used                            1.(*) Not Used
     2.( ) Program Name is Used                2.( ) Use the Following Files
              Drive [ ]                                 Drive [ ]
     3.( ) Use the Following Files                      File  [         ]
              Drive [ ]
              File  [         ]
                          Execute(Y)     Cancel(N)

                                                  Space:Select Esc:Close
```

The setting details are as follows:

1. "Not Used":
   No setting is made for the comment file to be used. To use the comment file, use the QCDSET instruction. (For details on the QCDSET instruction, refer to the QCPU (Q mode)/QnACPU Programming Manual (Common Instructions).)
2. "Program Name is Used":
   Use the comment file with the same file name as the program that exists in the specified drive and is currently being executed.
   When the program is changed, the comment file is also changed.
3. Using the designated device comment file:
   Use the name of the file that is stored in the drive specified by the parameter.

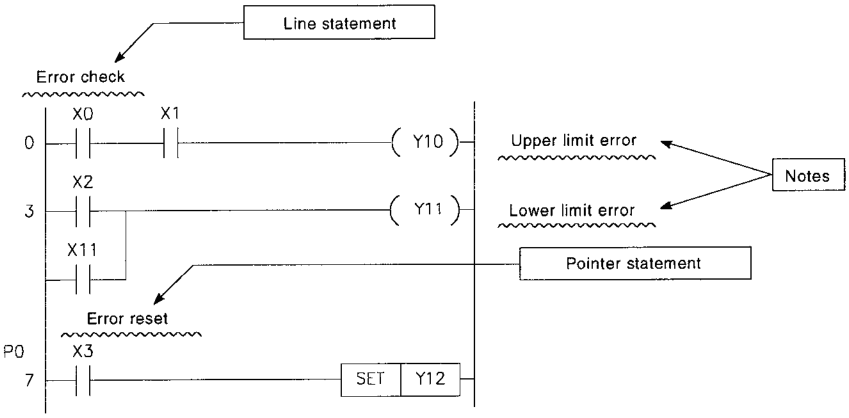
POINT

(1) When using the QCDSET instruction, note the following points.
  (a) When the above 1) or 2) has been set, the file set with the QCDSET instruction is valid for all program files.
  (b) When 3) is set, the file set with the QCDSET instruction is valid only for the program file for which the QCDSET instruction is executed.

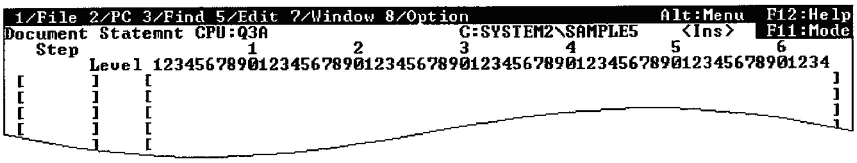
(2) Even if the file set with the parameter does not exist in the specified drive, no CPU module error is generated. Since no file exists, however, the CPU module does not display any comments.

11.6 Statements/Notes

Statements and notes are assigned to each program step, or to P or I pointers, in order to facilitate program reading.



(1) Statements or notes are set on the "Pointer statement", "Statement", or "Note" screen displayed from the edit menu in the documentation mode of GPP function.



```
 1/File 2/PC 3/Find 5/Edit 7/Window 8/Option                    Alt:Menu  F12:Help
Document Statemnt CPU:Q3A                       C:SYSTEM2\SAMPLE5    <Ins>  F11:Mode
   Step                 1         2         3         4         5         6
          Level 1234567890123456789012345678901234567890123456789012345678901234
 [       ]   [                                                                    ]
 [       ]   [                                                                    ]
 [       ]   [                                                                    ]
 [       ]   [                                                                    ]
         ]   [
```

(2) The details of each comment are as follows:

(a) Statement (Line statement)

A comment can be appended to a ladder block provided for individual function to explain the meaning and usage of the function.

(b) Pointer statement

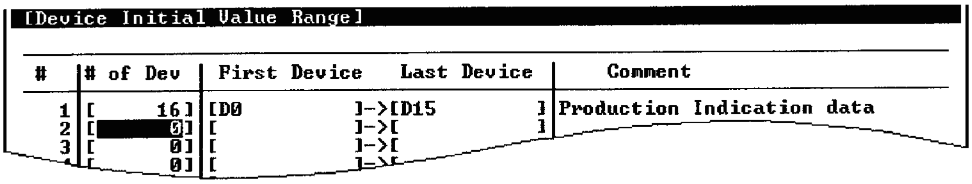
A comment can be appended to a pointer placed in the head of a subroutine or interrupt program to explain the meanng and usage of each program.

(c) Note

A comment can be appended to individual ladder blocks to explain the meaning and usage of the function.

11.7 Initial Device Value Comment

Initial device value comments are assigned to initial device value files so that individual file contents can be figured out.
Initial device value comments are stored in an initial device value file.
They are set on the "Device Initial Value Range" screen displayed from the edit menu in the device mode of GPP function.

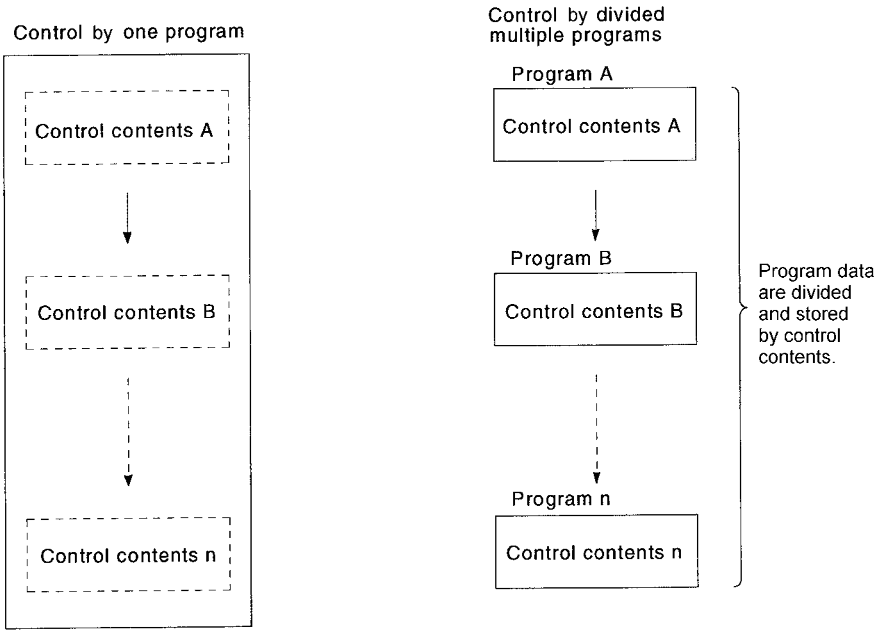


```
[Device Initial Value Range]

#   |# of Dev  | First Device    Last Device  |     Comment
  1 |[      16]|[D0          ]->[D15         ]|Production Indication data
  2 |[       0]|[            ]->[            ]|
  3 |[       0]|[            ]->[
    |[       0]|[            ]->[
```

12 OVERVIEW OF PROCESSING PERFORMED BY THE Q2ASCPU

12.1 Program Execution Types

Programs to be executed by the Q2ASCPU are stored in the built-in RAM of the CPU module or in a memory card.
While all of the data can be stored as one program in the built-in RAM or a memory card, they can be also divided into several programs based on control units and stored.
When programming is undertaken by more than one designer, all the programming process can be divided into several parts based on the processing units for each designer and all of the programming data can be stored in the built-in RAM of a CPU module or a memory card.

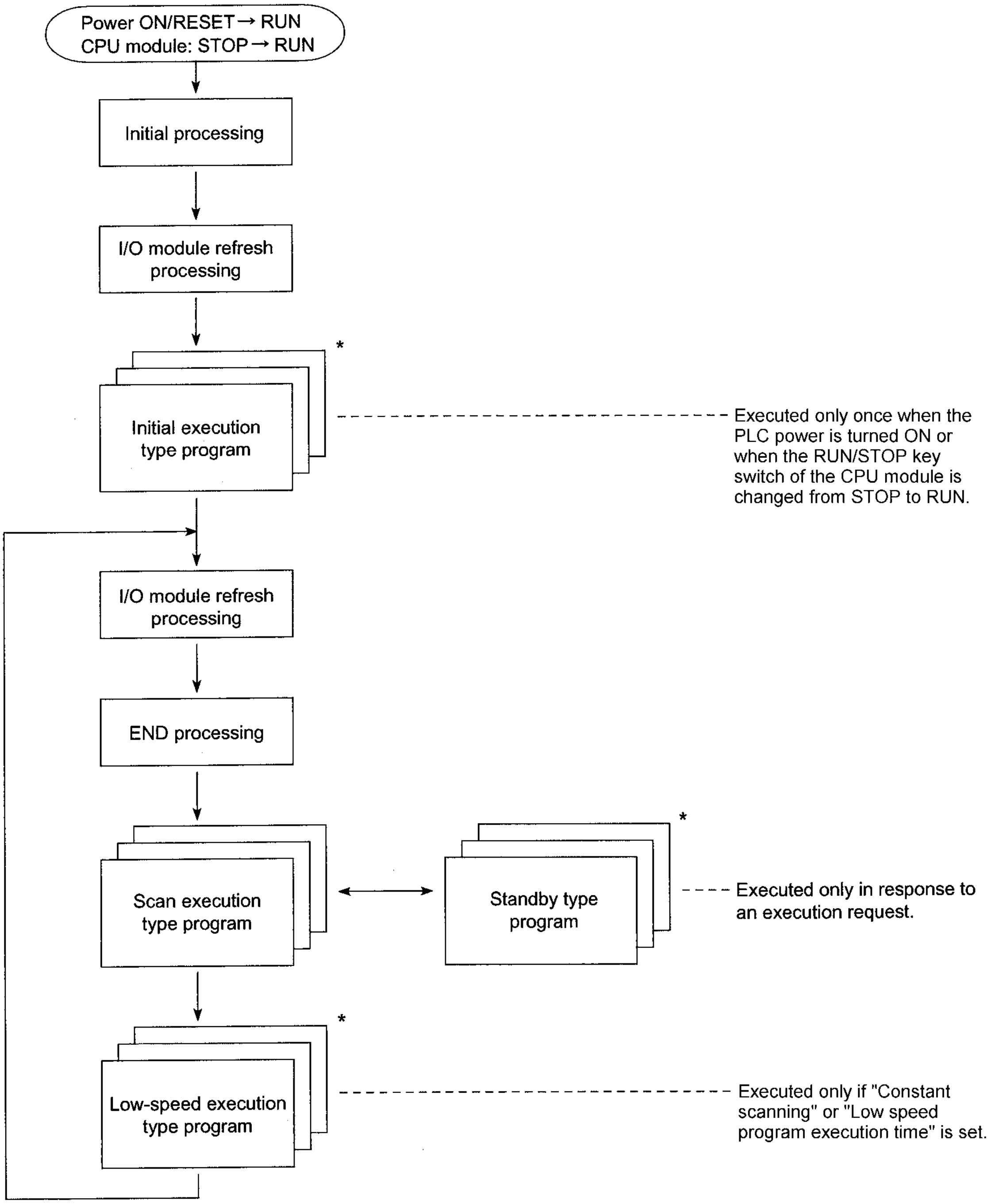


When dividing a program data into multiple programs, set "execution type" for each program in program setting in the parameter mode of GPP function.
The Q2ASCPU executes each execution type program in order of setting.
There are four executions types: "Initial execution type", "Scan execution type", "Low-speed execution type", and "Standby type".

- Initial execution type : Program executed only once when a programmable controller is powered ON, when a CPU module is reset, or when the RUN/STOP key switch of the CPU module is switched from STOP to RUN.
(Refer to Section 12.1.1)
- Scan execution type : Program that is executed once per scan, starting from the next scan after execution of the initial execution type program.
(Refer to Section 12.1.2)

- Low-speed execution type : Program that is executed only in the surplus scan time after execution of a scan execution type program in the constant scan setting, or only when the low-speed type program execution time is set.
(Refer to Section 12.1.3)
- Standby type : Program that is only executed when an execution request is made for it.
(Refer to Section 12.1.4)

The following shows the flow of operation processing when a programmable controller is powered ON, when a CPU module is reset, or when the RUN/STOP key switch of a CPU module is switched from STOP to RUN.



POINT

For the Q2ASCPU, all execution types need not be set.
Use initial execution type, low-speed execution type, and standby type programs marked with asterisks if required.

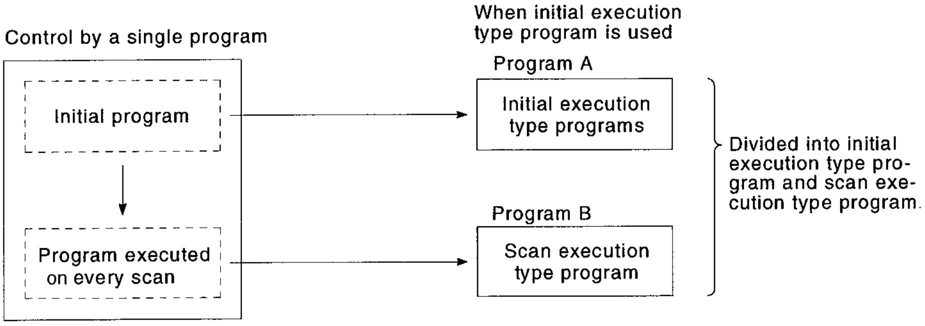
12.1.1 Initial execution type programs

(1) Definition

(a) The initial execution type program is a program executed only once when a programmable controller is powered ON, when a CPU module is reset, or when the RUN/STOP key switch of the CPU module is switched from STOP to RUN.

(b) The execution type is set to "Init" in program setting in the parameter mode of GPP function.

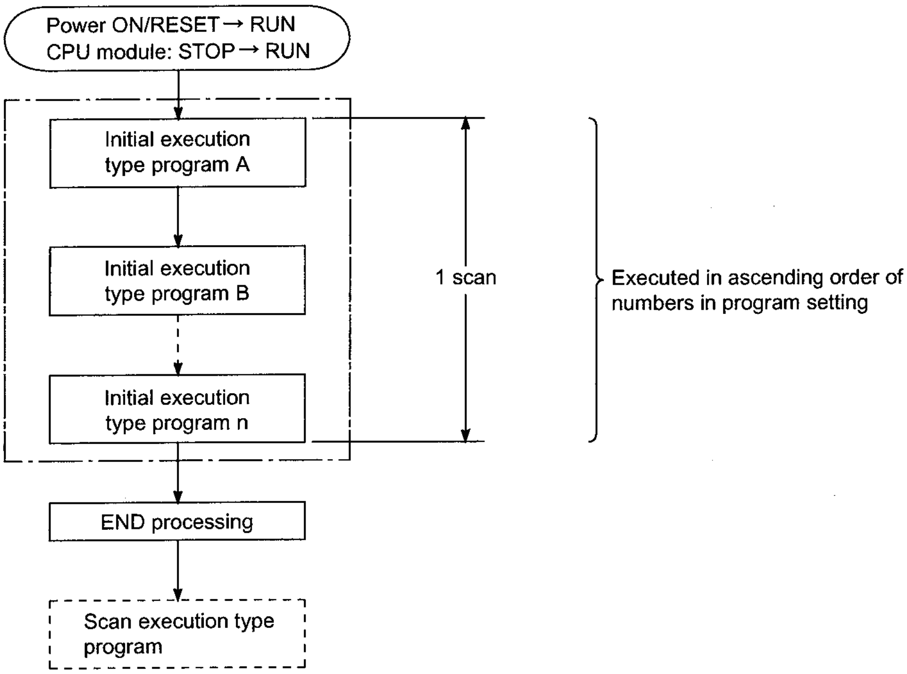
(c) Initial execution type programs can be used for applications such as the initial processing for a special function module, where once the program has been executed, it need not be executed from the next scan.*



(2) Execution of multiple initial execution type programs

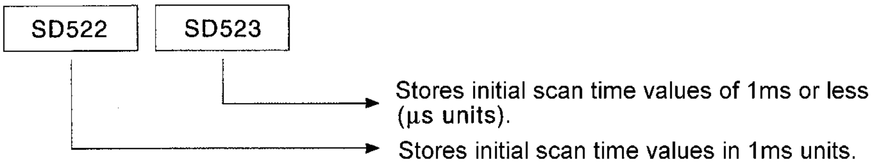
If there are more than one initial execution type program, they are executed in ascending order of the program numbers set in the parameter mode.

(3) END processing
When execution of all initial execution type programs is completed, END processing is performed and a scan execution type program is executed from the next scan.



POINT
* Instructions that contain a completion device cannot be used in initial execution type programs.

(4) Initial scan time
   (a) This is the execution time of an initial execution type program.
   If multiple initial execution type programs are to be executed, it is the time required to complete execution of all these programs.
   (b) The Q2ASCPU measures the initial scan time and stores it in special registers SD522 and SD523.[*1]
   The initial scan time can be checked by monitoring SD522 and SD523.



Example:
If "3" and "400" are stored in SD522 and SD523 respectively, the initial scan time is 3.4ms.

[*1] The accuracy of each scan time stored in the special registers is ±0.1ms. Note that, even if a watchdog timer (WDT) reset instruction is executed in the sequence program, measurement of the initial scan time is continued.

(5) Initial execution monitoring time
This is a timer for monitoring the execution time of initial execution type programs; no default value is set.
To monitor the execution time of an initial execution type program, a value can be set within the range of 10ms to 2000ms in "PLC RAS" in the parameter mode. (Unit: 10 ms)
If the initial scan time exceeds the set initial execution monitoring time, a "WDT ERROR" occurs and the Q2ASCPU stops its operation.

POINT

An error may be generated in the range of 0 to 10ms in measurement of the initial execution monitoring time.
Because of this, if the initial execution monitoring time (t) is set as 10ms, a WDT ERROR will occur when the initial scan time exceeds the limit within the range of $10\text{ms} \leqq t < 20\text{ms}$.

12.1.2 Scan execution type program

(1) Definition

(a) The scan execution type program is a program that is executed once for every scan, starting from the next scan after execution of the initial execution type program.

(b) The execution type is set to "Scan" in program setting in the parameter mode of GPP function.

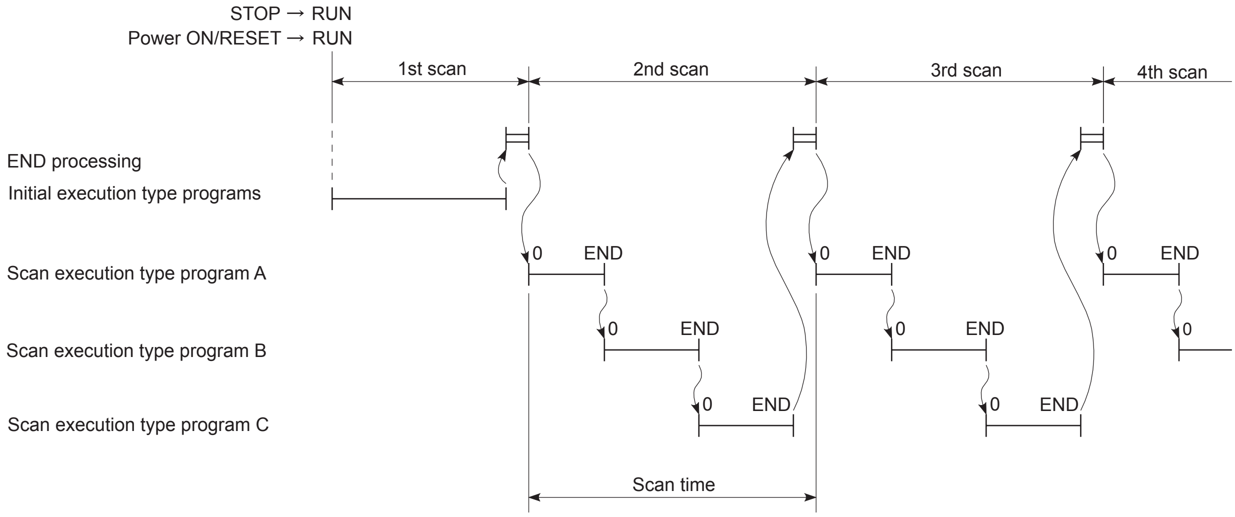
(2) Execution of multiple scan execution type programs

If there are more than one scan execution type program, they are executed in ascending order of the program numbers set in the parameter mode.

(3) END processing

When all the scan execution type programs have been executed, END processing is performed and then the first scan execution type program is executed again.

By inserting a COM instruction at the end of a scan execution type program, END processing (general data processing, link refresh) can be executed for each program.



(4) When constant scan time is set[*1]

When constant scan is set, the scan execution type program is executed once for every preset constant scan time.

REMARK

[*1] Constant scan is a function whereby a scan execution type program is repeatedly executed at fixed intervals.
See Section 10.2.

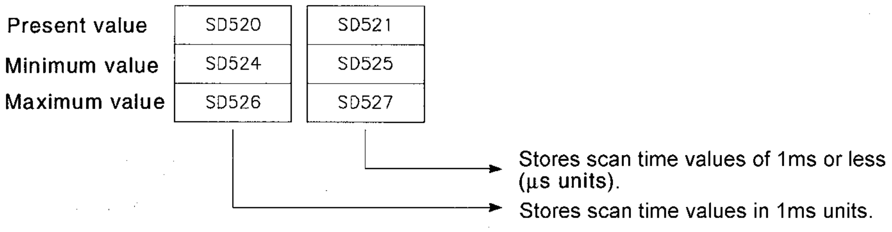
POINT

(1) For the index register processing in the case where an interrupt program is executed during execution of a scan execution type program, refer to the QnACPU Programming Manual (Fundamentals).

(5) Scan time

- The scan time is a total of the scan execution type program execution time, the END processing time, and either the low-speed program execution time or the constant scan waiting time.[*1]
  When more than one scan execution type program is executed, "the execution time of the scan execution type program" is the time required for completing execution of all these programs.

  *1 Refer to Section 12.1.3.
- The Q2ASCPU measures the present, minimum, and maximum values for the scan time and stores them in special registers SD520, SD521, and SD524 to SD527.[*2]
  The scan time can be checked by monitoring these special registers.



Example: If "3" and "400" are stored in SD520 and SD521 respectively, the scan time is 3.4ms.

*2 The accuracy of each scan time stored in the special registers is ±0.1ms. Note that, even if the watchdog timer (WDT) reset instruction is executed in the sequence program, measurement of each scan time is continued.

(6) WDT (watchdog timer)

This is a timer that monitors the scan time; and 200ms is set as a default value.
WDT is set within the range of 10ms to 2000ms in "PLC RAS" in the parameter mode. (Unit: 10ms).
When using a low-speed execution type program(s), make sure that the specified WDT value is greater than the sum of the scan time and the low-speed execution type program execution time.
If the scan time (total of execution times for scan execution type programs and low-speed execution type programs, END processing time, and low-speed END processing time) exceeds the time set for WDT, a "WDT ERROR" occurs and the Q2ASCPU stops its operation.

POINT

The WDT measurement error is 0 to 10ms.
Because of this, when WDT (t) is set to 10ms, a WDT ERROR may not occur even if the scan time exceeds the limit within the range of 10ms≦t＜20ms.

12.1.3 Low-speed execution type program

(1) Definition

(a) The low-speed execution type program is a program that is executed only in the surplus time of constant scan operation or in the preset low-speed execution program execution time.

- When using a fixed scan time to give priority to control accuracy, set the constant scan time in "PLC RAS" in the GPP function's parameter mode.
  (Setting range: 5 to 2000ms; Unit: 5ms)
- To ensure the execution time for low-speed execution type programs in each scan and to make these programs operate properly, set the low-speed program execution time in "PLC RAS" in the parameter mode.
  (Setting range: 1 to 2000ms; Unit: 1ms)
- In order to execute low-speed execution type programs, either the constant scan time or the low-speed program execution time must be set.

(b) Set "Slow" as the execution type in program setting in the parameter mode.

(c) This execution type is used for programs that do not have to be executed every scan, such as a program for printer output.

(2) Execution of multiple low-speed execution type programs

If there are more than one low-speed execution type program, they are executed in ascending order of the program numbers set in the parameter mode.

(3) Execution time for low-speed execution type program executed in one scan

(a) When operation of all low-speed execution type programs is completed within one scan and there is surplus time, the subsequent processing varies depending on the setting status of special relay SM330 and the execution condition for the low-speed execution type programs.

• Non-synchronization method (SM330 = OFF)	: Operation of a low-speed execution type program is continuously executed within surplus time.
• Synchronization method (SM330 = ON)	: Even if there is surplus time, operation of a low-speed execution type program is not executed and another operation starts from the next scan.

Operation method of low-speed execution type program	Setting status of SM330	Execution condition of low-speed execution type program	
		Constant scan setting	Low-speed program execution time setting
Non-synchronization method	OFF	Re-executes low-speed execution type program.[*1]	Re-executes low-speed execution type program.[*2]
Synchronization method	ON	Constant scan wait time occurred[*3]	Starts scan execution type program operation.[*4]

*1 When the constant scan time is set, the low-speed execution type program is repeatedly executed for the surplus time of the constant scan.
Accordingly, the execution time of the low-speed execution type program is different at each scan. If surplus time in constant scan is less than 2ms, the low-speed execution type program is not executable.
When using a low-speed execution type program, set a proper constant scan time so that surplus time will be 2ms or longer.

*2 When the low-speed program execution time is set, a low-speed execution type program is repeatedly executed for the set time duration.
Accordingly, the scan time is different at each scan.

*3 When the constant scan time is set, surplus time after completion of the low-speed END processing is used as wait time. When the set constant scan time is reached, the scan execution type program is executed.
Wait time for constant scan
= (Set constant scan time) - (Scan time) - (Low-speed scan time)

Therefore, the scan time for each scan is constant.
If surplus time in constant scan is less than 2ms, the low-speed execution type program is not executable. When using a low-speed execution type program, set a proper constant scan time so that surplus time will be 2ms or longer.

*4 When the low-speed program execution time is set, operation of the scan execution type program is started ignoring the surplus time after completion of the low-speed END processing.
Surplus time in low-speed program execution time
= (Set low-speed program execution time) - (Low-speed scan time)

Accordingly, the scan time is different at each scan.

(b) If the low-speed execution type program is not processed within surplus time of the constant scan time or within the low-speed program execution time, the program execution is interrupted and is resumed in the next scan.

POINT

(1) For the index register processing in the case where a scan execution type program is switched to a low-speed execution type program, refer to the QnACPU Programming Manual (Fundamentals).

(2) For the index register processing in the case where an interrupt program is executed during execution of a low-speed execution type program, refer to the QnACPU Programming Manual (Fundamentals).

(3) Set a proper low-speed program execution time so that the value obtained by adding it to the scan time is smaller than the set WDT value.

(4) The COM instruction cannot be used in the low-speed program.

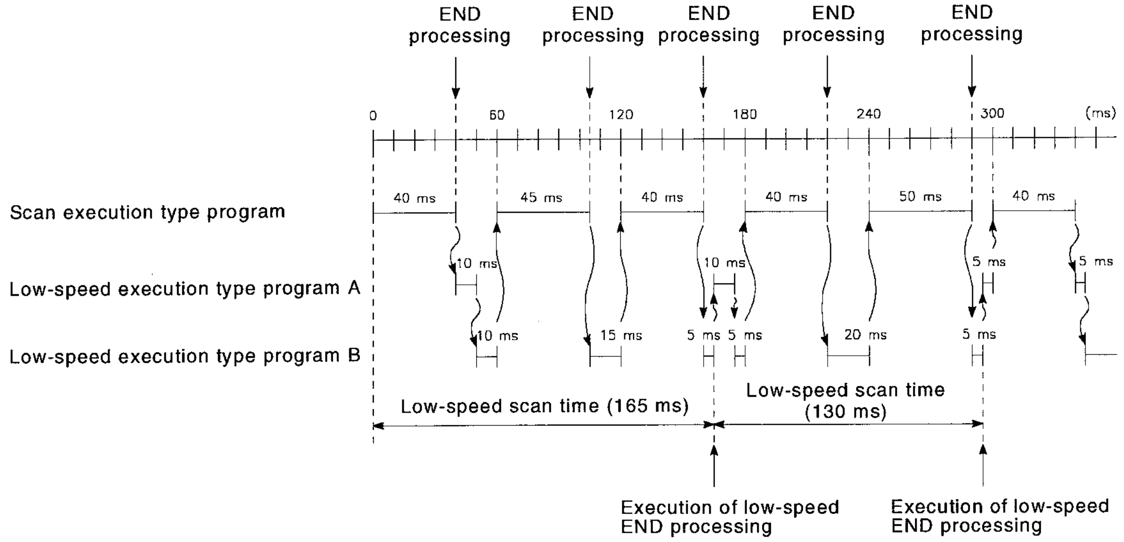
(5) When "Constant scan time" and "Low-speed program execution time" are set, PRG. TIME OVER (Error code: 5010) occurs in the case of (Surplus time of constant scan) ＜ (Low-speed program execution time)
Execute the low-speed execution type program either in the constant scan time or in low-speed program execution time.

1: For non-synchronous method:

(1) When "Constant scanning" is set

The operation when a low-speed execution program is executed under the following conditions is shown below.

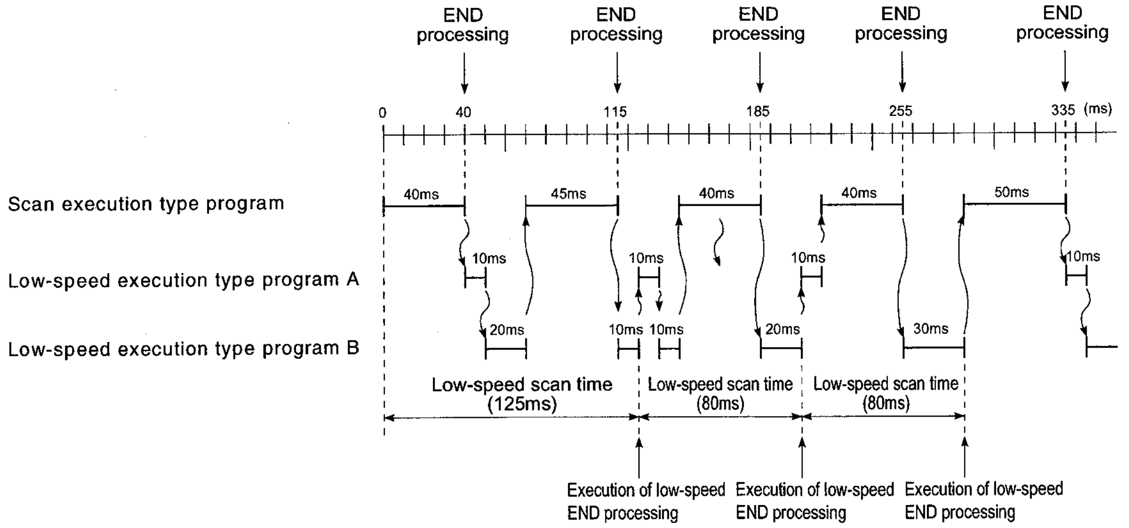
- Constant scan time: 60 ms
- Total for scan execution type programs: 40 ms to 50 ms
- Execution time for low-speed execution type program A: 10 ms
- Execution time for low-speed execution type program B: 30 ms
- END processing: 0 ms (assuming 0 ms here to make the explanation easy)
- Low-speed END processing: 0 ms (assuming 0 ms here to make the explanation easy)



(2) When a low-speed program execution time is set

The operation when a low-speed execution program is executed under the following conditions is shown below.

- Low-speed program execution time: 30 ms
- Total for scan execution type programs: 40 ms to 50 ms
- Execution time for low-speed execution type program A: 10 ms
- Execution time for low-speed execution type program B: 30 ms
- END processing: 0 ms (assuming 0 ms here to make the explanation easy)
- Low-speed END processing: 0 ms (assuming 0 ms here to make the explanation easy)

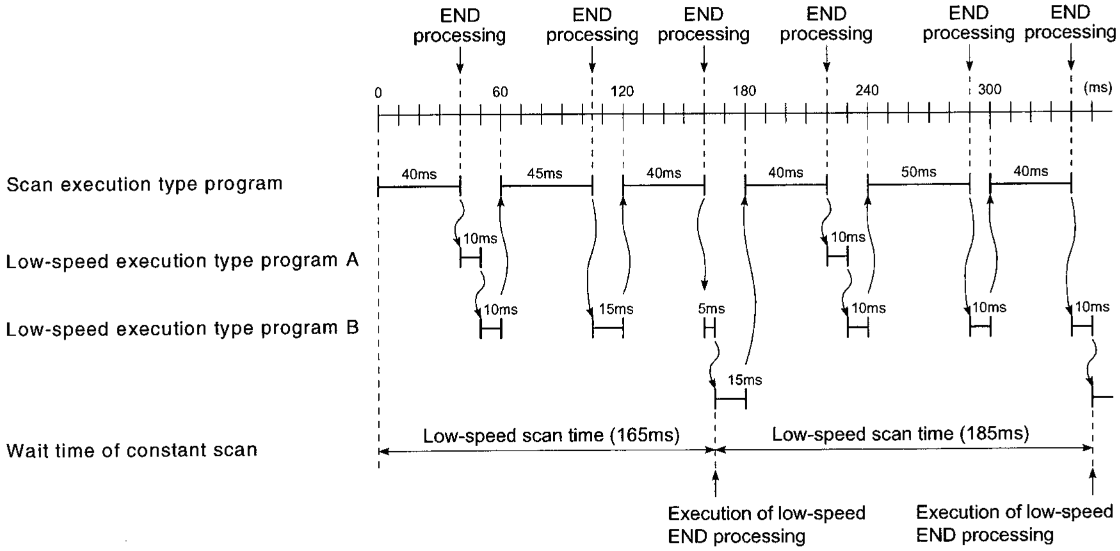


2: For synchronous method:

(1) When "Constant scanning" is set

The operation when a low-speed execution program is executed under the following conditions is shown below.

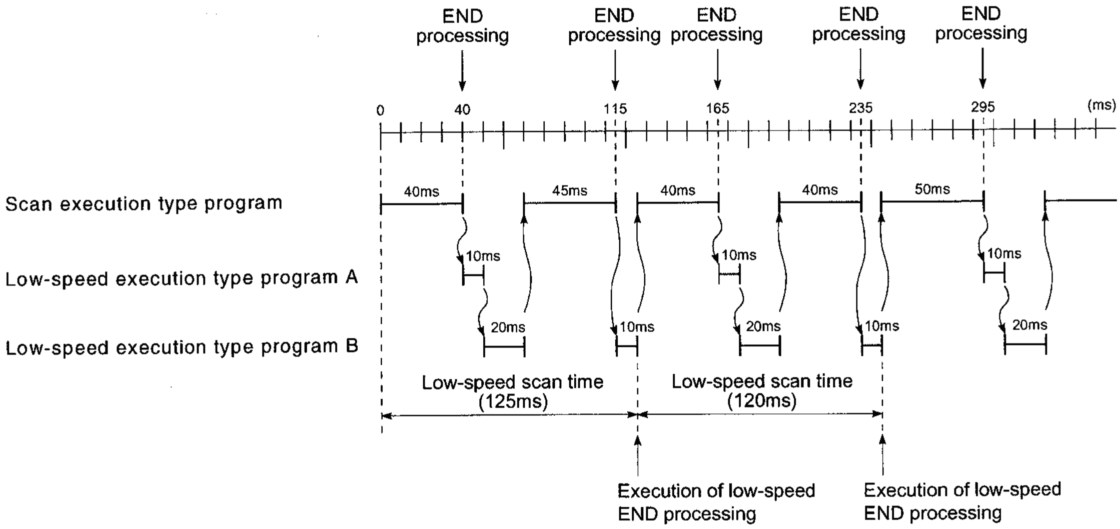
- Constant scan time: 60 ms
- Total for scan execution type programs: 40 ms to 50 ms
- Execution time for low-speed execution type program A: 10 ms
- Execution time for low-speed execution type program B: 30 ms
- END processing: 0 ms (assuming 0 ms here to make the explanation easy)
- Low-speed END processing: 0 ms (assuming 0 ms here to make the explanation easy)



(2) When a low-speed program execution time is set

The operation when a low-speed execution program is executed under the following conditions is shown below.

- Low-speed program execution time: 30 ms
- Total for scan execution type programs: 40 ms to 50 ms
- Execution time for low-speed execution type program A: 10 ms
- Execution time for low-speed execution type program B: 30 ms
- END processing: 0 ms (assuming 0 ms here to make the explanation easy)
- Low-speed END processing: 0 ms (assuming 0 ms here to make the explanation easy)



(4) END processing

When all of the low-speed execution type program has been completed, low-speed END processing is executed.

The following processing is performed in low-speed END processing:

- Setting of special relays/special registers for the low-speed execution type program
- Writing the low-speed execution type program during RUN
- Measurement of the low-speed scan time
- Resetting the watchdog timer for the low-speed execution type program

When low-speed END processing is completed, the low-speed execution type program is executed again from the beginning.

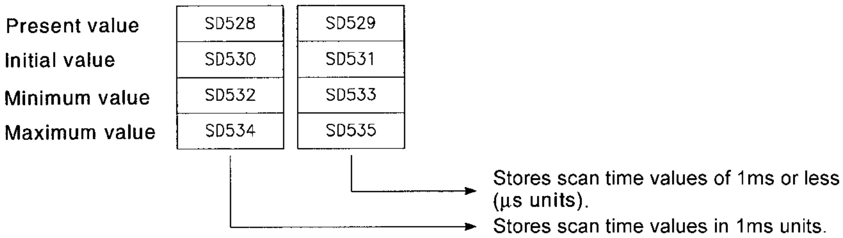
POINT

In execution of a low-speed execution type program, the constant scan time may be extended by a time equivalent to the maximum processing time for the instructions executed plus the low-speed END processing time.

(5) Low-speed scan time

(a) The low-speed scan time is a total time of the time required for completion of the low-speed execution program and the low-speed END processing time.
If multiple low-speed execution type programs are to be executed, it is the total time of the time required for completion of all low-speed execution type programs and the low speed END processing time.

(b) The Q2ASCPU measures the low-speed scan time and stores it in special registers SD528 to SD535.[*1]
The low-speed execution scan time can be checked by monitoring these registers.



Present value	SD528	SD529
Initial value	SD530	SD531
Minimum value	SD532	SD533
Maximum value	SD534	SD535

Example:

If "3" and "400" are stored in SD528 and SD529 respectively, the scan time is 3.4ms.

[*1] The accuracy of each scan time stored in the special registers is ±0.1ms. Note that, even if a watchdog timer (WDT) reset instruction is executed in the sequence program, measurement of each scan time is continued.

(6) Low-speed execution monitoring time

This is a timer for monitoring the execution time of low-speed execution type programs; no default value is set.

To monitor the execution time of an low-speed execution type program, a value can be set within the range of 10ms to 2000ms in "PLC RAS" in the parameter mode. (Setting units: 10ms).

If the low-speed scan time exceeds the set low-speed execution monitoring time, a "PRG TIME OVER" error occurs. The Q2ASCPU however continues its operation.

POINT

The low-speed execution monitoring time is measured in low-speed END processing.

Because of this, when the low-speed execution monitoring time (t) is set to 100ms, a "PRG TIME OVER" error occurs if the low-speed scan time measured in low-speed END processing exceeds 100ms.

12.1.4 Standby type program

(1) Definition

(a) The standby type program is a program that is executed only in response to an execution request.

(b) The standby type program has the following applications:

1) Program library

Subroutine programs and interrupt programs are set as standby type programs and controlled separately from the main program.

2) Set-up of programs

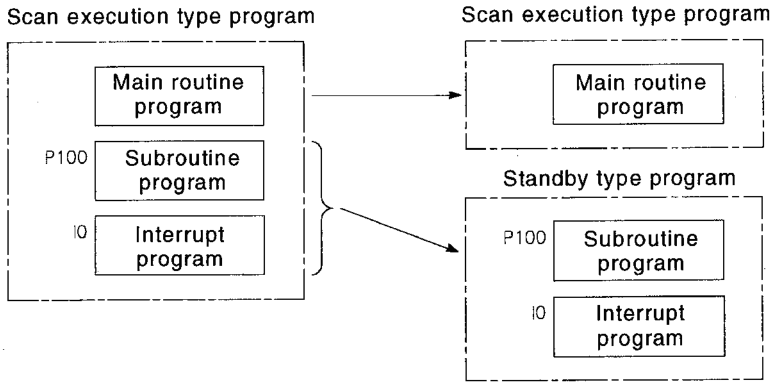
The main routine program is registered to the standby type program and programs required for control are changed to the scan execution type programs. Programs not used for control are changed to the standby type programs.

(2) Program library

(a) Library creation of program

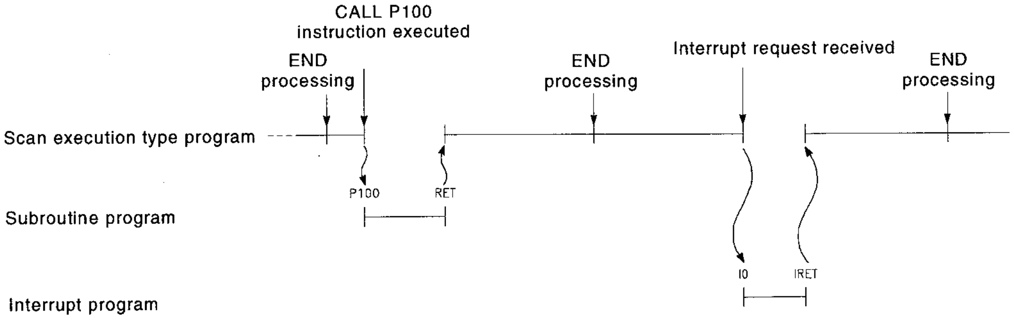
1) Program library is used to control subroutine programs and interrupt programs separately from the main routine program.

It is possible to create multiple subroutine programs and interrupt programs as one standby type program.



2) When a standby type program execution is completed, control returns to the program that was being executed before execution of the standby type program.

The following shows the operation performed when a subroutine program and an interrupt program in a standby type program are executed.



POINT

(1) Timers are not to be used in standby type programs because they update present values and turn ON/OFF the contacts when the OUT T☐ instruction is executed.

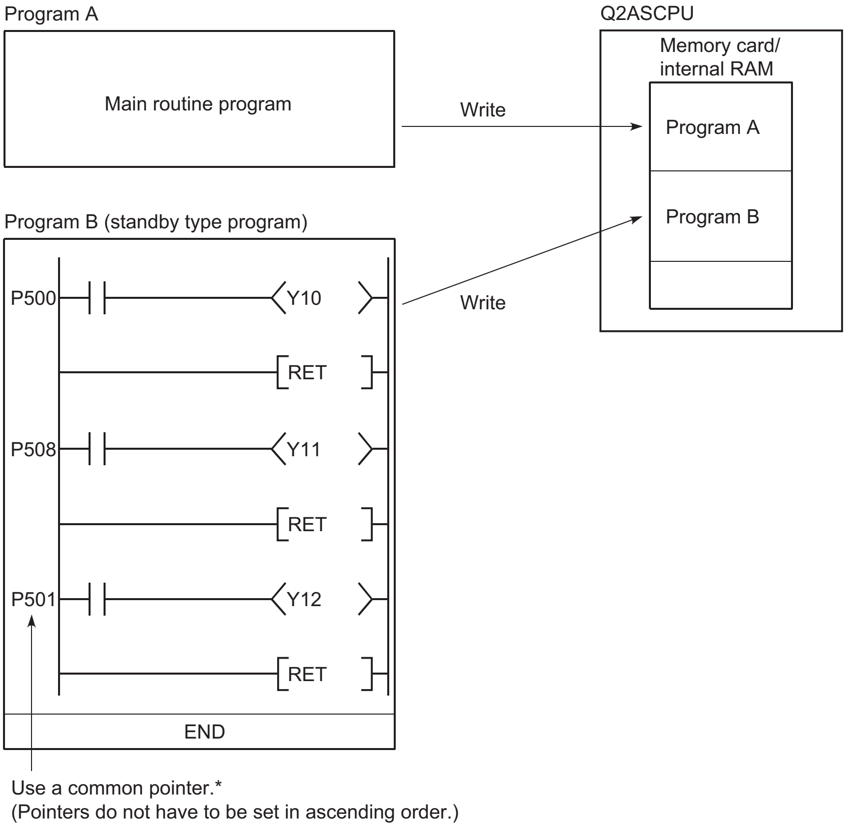
(2) When setting a subroutine program as a standby type program, use a common pointer.
Standby type programs that use local pointers are not executable.
For details on common and local pointers, refer to the QnACPU Programming Manual (Fundamentals).

(b) When grouping several subroutine programs into one

1) Create subroutine programs in order starting from step 0 in the standby type program.
An END instruction is required at the end of the subroutine programs.

2) Since there are no restrictions on the order of creation of subroutine programs, there is no need to arrange pointers in ascending order of pointer numbers when creating multiple subroutine programs.

3) Use common pointers. *
Subroutine programs using common pointers can be called from all the programs that are being executed by the Q2ASCPU.



Use a common pointer.*
(Pointers do not have to be set in ascending order.)

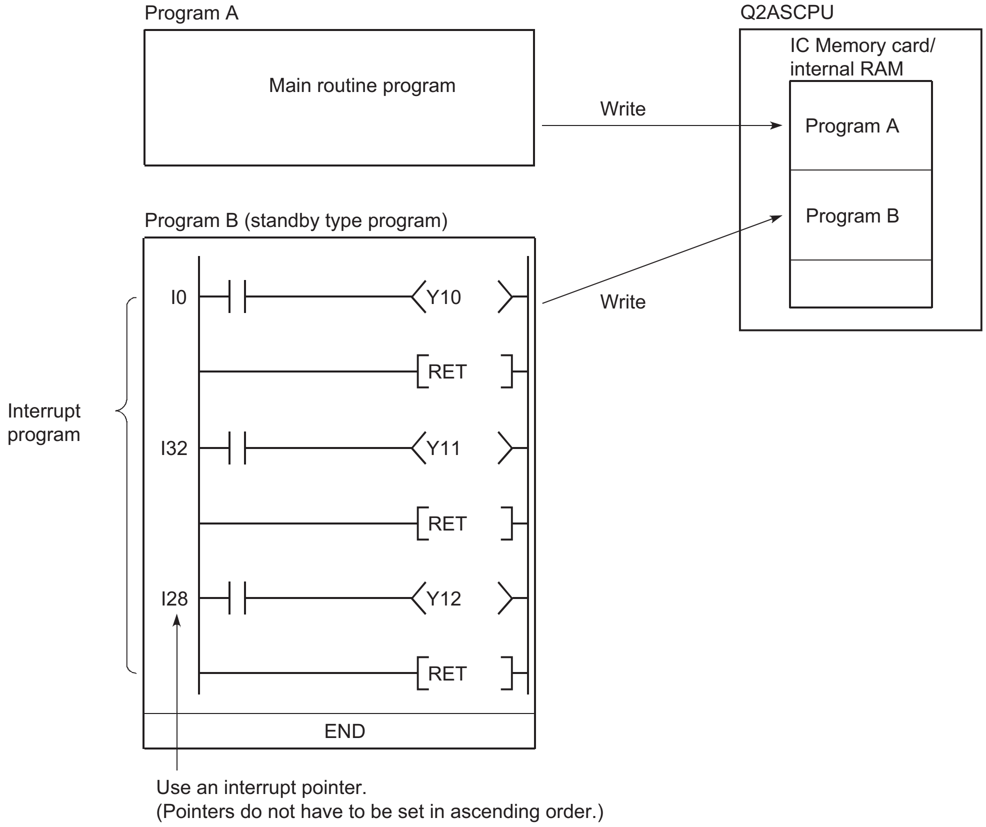
REMARK

* For details on common pointers, refer to the QnACPU Programming Manual (Fundamentals).

(c) When grouping several interrupt programs into one

1) Create interrupt programs in order starting from step 0 in the standby type program.
An END instruction is required at the end of the interrupt programs.

2) Since there are no restrictions on the order of creation of interrupt programs, there is no need to arrange pointers in ascending order of pointer numbers when creating multiple interrupt programs.



REMARK

For details on interrupt pointers, refer to the QnACPU Programming Manual (Fundamentals).

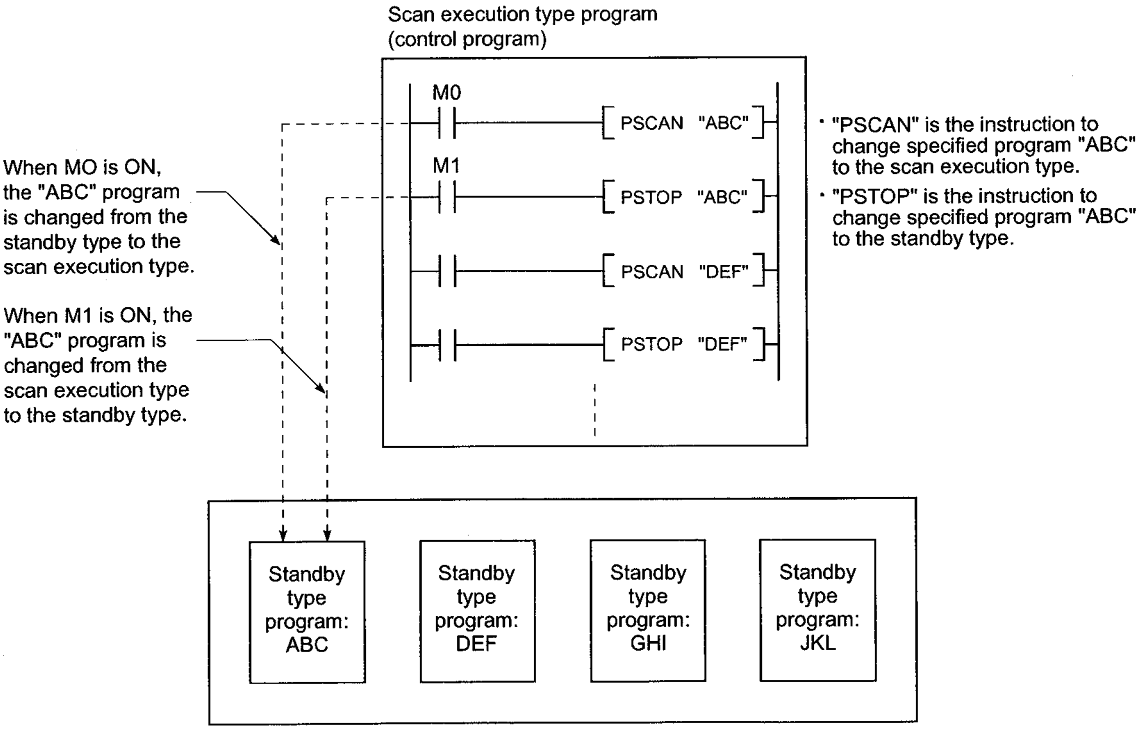
(3) Set-up of programs

(a) Programs corresponding to all of the systems can be created in advance, and thereby necessary programs only can be executed.
Programs set as the standby type with parameters can be changed to the scan type programs in the sequence program for execution.
Use the following instructions to change the execution type in the Q2ASCPU:

1) PSCAN instruction : Changes the program type from the standby type to the scan execution type.
2) PLOW instruction : Changes the program type from the standby type to the low-speed execution type.
3) PSTOP instruction : Changes the program type from the scan execution/low-speed execution type to the standby type.

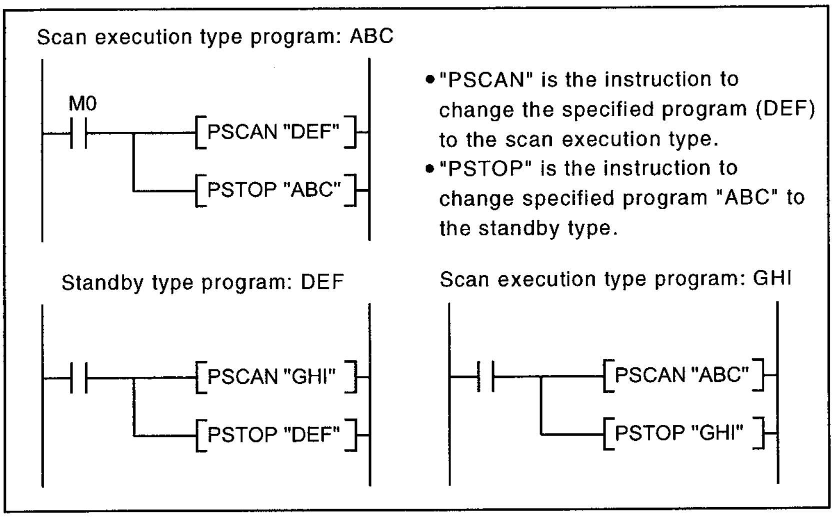
(b) The following methods are available to switch programs for execution:

1) When selecting programs to be executed in a control program:
   - Defining the scan execution type program as the control program, the QnACPU switches between the standby type program and the scan execution type program according to the set conditions to control the program to be executed.
   - The following shows how the excution types of standby programs, "ABC," "DEF," "GHI" and "JKL" are changed in the control program.



2) When changing the execution type of another program from the scan execution type program:
   - In the scan execution type program in execution, the type of the program to be executed next is changed from the standby type to the scan execution type.
   - The following shows the operation that the QnACPU switches the standby type program "DEF" to the scan execution type, and the scan execution program "ABC" to the standby type program when M0 in program "ABC" turns on.

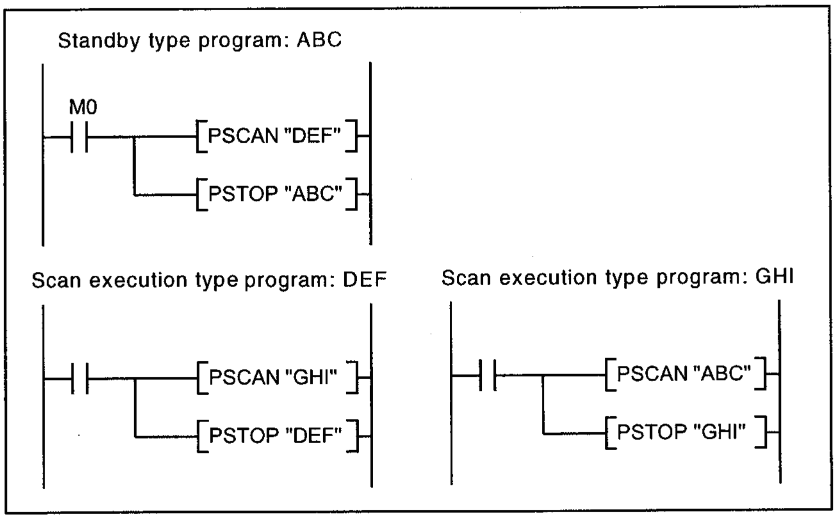
[Before execution of PSCAN and PSTOP instructions]



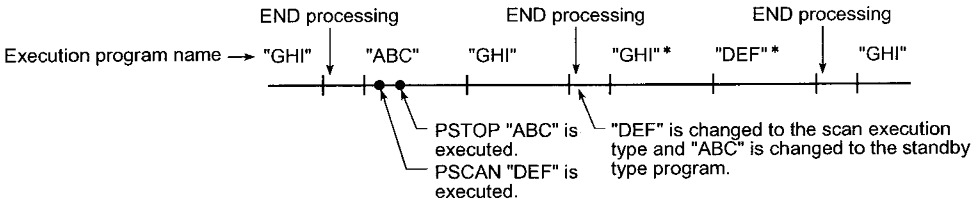
- "PSCAN" is the instruction to change the specified program (DEF) to the scan execution type.
- "PSTOP" is the instruction to change specified program "ABC" to the standby type.

When MO is ON

[After execution of PSCAN and PSTOP instructions]



(c) The program execution type is changed by the PSCAN or PSTOP instruction in the END processing.
Therefore, it is not changed during program execution.



REMARK

* The "GHI" and "DEF" programs are executed in the sequence set with parameters in the program setting.

12.1.5 Initial processing

Initial processing is the pre-processing for executing sequence operations.
The QnACPU executes it only once in the case of the CPU module status described in the following table.
Once the initial processing is completed, the CPU module is placed into the operation status set by the RUN/STOP switch.

Item	Status of the CPU module		
	At Power ON	At RESET	STOP → RUN Status*1
Initialization of I/O module	○	○	×
Initialization of devices outside latched range (Bit device: OFF, Word device: 0)	○	○	×
Self-diagnostics	○	○	○
Auto allocation of module I/O No.	○	○	○
Setting MELSECNET/10 network info and MELSECNET(II)/B data link info	○	○	×
Setting CC-Link info and MELSECNET/MINI-S3 info	○	○	×
Setting initial device values	○	○	○
Booting from memory card	○	○	×

○: Executed ×: Not executed

*1 Indicates the case that the CPU enters RUN status without being reset after changing a parameter or program in STOP status.
(The RUN/STOP key switch is operated as follows: STOP→RUN→(RUN LED is flickering.)→STOP→RUN.)
Note that the instructions for conversion into pulse (PLS, ☐P) may not function properly since the previous information may not be retained depending on the program change (write during RUN in STOP status, or write to PLC).

12.1.6 Refresh processing of I/O module

Refresh processing of I/O modules is executed.
(Refer to the QnACPU Programming Manual (Fundamentals).)

12.1.7 END processing

This is a post-process to finish one cycle of operation processing of the sequence program and to return the execution of the sequence program to step 0.

(a) Self-diagnostic checks are performed for fuse blown, module verify, or low battery. (Refer to Section 9.3)

(b) When data read/write is requested from a peripheral device or an intelligent special function module (computer link module, serial communication module, Ethernet module, etc.), data are exchanged between the programmable controller CPU and the peripheral device or intelligent special function module.

(c) Refresh processing is performed when a refresh request is issued from a network module or a link module.

(d) When the trace point for sampling trace is set to each scan (after execution of END instruction), the status of the set device is stopred into the sampling trace area.

(e) Refresh processing based on the MELSECNET/MINI-S3 automatic refresh function is performed. (Refer to Chapter 7))

POINT

(1) If the constant scan function (see Section 10.2) is set, the END processing time result is retained during the period between completion of END processing and start of the next scan.

(2) If a low-speed execution type program (see Section 12.1.3) is executed, low-speed END processing is performed separately from normal END processing. In low-speed END processing, the special relays and special registers for low-speed execution programs are set.

12.2 Operation Processing of RUN, STOP, PAUSE, and STEP-RUN

The Q2ASCPU has four kinds of operation statuses: RUN, STOP, PAUSE, and step operation (STEP-RUN) statuses.
Operation processing of programmable controller CPU in each operation status is explained here.

(1) RUN status operation processing
   (a) The RUN status represents a status in which sequence program operation is repeated in the order from step 0 → END (FEND) instruction → step 0.
   (b) When entering the RUN status, the CPU outputs the output status data saved in STOP status according to the output mode setting parameter for STOP → RUN.
   (c) Processing time from switching STOP → RUN to the start of the sequence program operation is usually one to three seconds, although it may vary depending on the system configuration.
   Note that it may be longer than this depending on the conditions.

(2) STOP status operation processing
   (a) The STOP status is a status in which sequence program operation is stopped by the RUN/STOP key switch or due to remote STOP (see Section 10.6.1).
   (b) When entering the STOP status, the CPU saves the output status data and turns all output points to OFF. Data memories except for output (Y) are retained.

(3) PAUSE status operation processing
   (a) The PAUSE status represents a status in which operation of sequence program is suspended with the output and data memory statuses retained. (Refer to Section 10.6.3)

(4) Step operation (STEP-RUN) operation processing
   (a) STEP operation is an operation mode in which operation processing of a sequence program can be paused/continued by each instruction using GPP function. (Refer to Section 8.7)
   (b) Since an operation processing is paused while retaining the output and data memories, the execution condition can be confirmed.

(5) Operation processing of Q2ASCPU when RUN/STOP key switch is operated

RUN/STOP state	Q2ASCPU Operation Processing				
	Operation processing of sequence program	External output	Data memory		Remark
			M, L, S, T, C, D	Y	
RUN→STOP	Executes up to the END instruction, then stops.	OS saves the output status, and sets all the output points to OFF.	Retains the condition immediately before entering the STOP status.	OS saves the output status, and sets all the output points to OFF.	
STOP→RUN	Starts from step 0.	Depends on the output mode set by the parameter for STOP→RUN.	Starts operations from the condition immediately before entering the STOP status.	Depends on the output mode set by the parameter for STOP→RUN.	

POINT

The Q2ASCPU executes the following processing in any of RUN state, STOP state, or PAUSE status.

- Refresh processing of I/O modules
- Data communication with peripheral devices, computer link modules, and/or serial communication modules.
- Link refresh processing.

Thus, even in the STOP state or PAUSE state, I/O monitoring and test operations using a peripheral device, reading/writing from computer link modules or serial communication modules, and communication with other stations via MELSECNET can be performed.

12.3 Operation Processing for Instantaneous Power Failure

The Q2ASCPU detects a momentary power failure when the input power voltage supplied to the power supply module becomes lower than the specified range.
When the Q2ASCPU detects an instantaneous power failure, the following operation processing is performed.

(1) When an instantaneous power failure shorter than the allowable momentary power failure period occurred:
  (a) When an instantaneous power failure occurs, the output statuses are held and the operation processing is suspended after the name of the currently accessing file and error history have been stored.
  (The timer count continues.)
  (b) If there is an SFC continuous operation designation, system save processing is executed.
  (c) When power is restored, the operation processing will be continued.
  (d) While the operation is interrupted due to an instantaneous power failure, measurement of the watchdog timer (WDT) continues. For example, if 200ms is set for the WDT parameter setting, power failure of 15ms in the scan time of 190ms will cause a watchdog timer error.

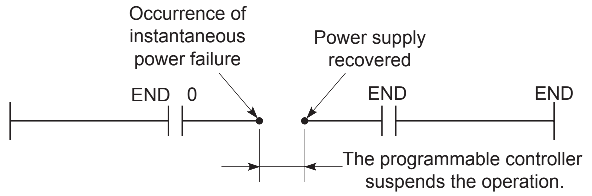


Fig. 12.1 Operation Processing for Instantaneous Power Failure

(2) When power failure longer than the allowable momentary power failure period occurred:
The Q2ASCPU starts from the first.
The operation processing is the same as that performed at programmable controller power-up or at CPU module reset by the RUN/STOP key switch.

12.4 Data Clear Processing

The Q2ASCPU clears data other than the following by turning the RUN/STOP key switch to RESET or by resetting the programmable controller power (ON, OFF and ON):

(a) Data in the built-in RAM (except data specified for memory clear in the boot specification)[*1]
(b) Data in the memory card
(c) Data of latch-specified devices(Latch clear key enabled)
(d) Data of latch-specified devices(Latch clear key disabled)
(e) File register data
(f) Local device data
(g) Fault history data

[*1] For the boot specification, refer to the QnACPU Programming Manual (Fundamentals).

Data given in (c) and (g) are cleared by latch clear operation using the RUN/STOP key switch (Refer to Section 15.3.) or by remote latch clear operation from GPP function (Refer to Section 10.6.5.)

The latch range is specified for each device on the "Device" screen in the parameter mode of GPP function. There are the following two latch range setting options.

1) Latch clear key enabled : Used to set a latch range which can be cleared by the latch clear operation using the RUN/STOP key switch.
2) Latch clear key disabled: Used to set a latch range which cannot be cleared by the latch clear operation using the RUN/STOP key switch.

Devices for which the latch clear key is disabled can be cleared by an instruction or by the clear operation of GPP function.

1) Clearing by an instruction : Reset by RST instruction , or transfer K0 with MOV instruction .
2) Clearing by GPP function : Execute device memory all clear from the PLC menu in the online mode.

For details on device latch ranges, refer to the QnACPU Programming Manual (Fundamentals).
For details on the operation method of GPP function, refer to the GX Developer Operating Manual or Type SW☐IVD-GPPQ Software Package Operating Manual (Online)/(Offline).

POINT

To clear file registers or local devices, reset them with the RST instruction or transfer KO with the MOV instruction.

MEMO

13 PARAMETER LIST

The parameters set for the Q2ASCPU are listed in the table below.
For details on each parameter, refer to the section or reference manual indicated.

Item			Parameter No.	Description
PLC name			–	Set labels and/or comments for peripheral devices on the CPU module. This setting does not affect CPU module operation.
	Label		0000H	Set a label for the CPU module.
	Comment		0001H	Set a comment for the CPU module.
PLC system			–	Make various settings that are required for the CPU module system.
	Timer limit setting	Low-speed timer	1000H	Set the low-speed or high-speed timer limit.
		High-speed timer		
	RUN-PAUSE contacts		1000H	Set the contact to control RUN/PAUSE of the CPU module.
	Remote reset		1002H	Enable or disable the remote reset operation.
	Output at STOP→RUN		1003H	Set the output mode for switching from STOP to RUN.
	Common pointer No.		1005H	Set the first common pointer number.
	General data processing		1006H	Set the number of modules that are processed in one general data processing.
	Points occupied by empty slot		1007H	Set the number of points occupied by empty slots.
	System interrupt	Interrupt counter	1008H	Set the first interrupt counter number, and the fixed scan interval for the interrupt pointer.
		Fixed scan interval		
PLC file			–	Set various kinds of files used by the CPU module.
	File register		1100H	Set a file register file to be used.
	Comment file used in a command		1101H	Set a comment file used in an instruction.
	Initial device value		1102H	Set a file for initial device values to be used.
	File for local device		1103H	Set a file for local devices to be used.

Setting		Reference Section/Reference Manual
Default value	Setting range	
–	–	Section 11.2
No setting	Up to 10 characters	
No setting	Up to 64 characters	
–	–	–
100ms	10ms to 1000ms (in 10ms units)	QnACPU Programming Manual (Fundamentals)
10ms	1ms to 100ms (in 1ms units)	
No setting	X0 to X1FFF	Section 10.6.1, Section 10.6.3
Disabled	Enabled/disabled	Section 10.6.4
Before operation	Before operation/After 1 scan	Section 10.4
No setting	P0 to P4095	QnACPU Programming Manual (Fundamentals)
1 module	1 to 6 modules	Section 6.3
16 points	0 point to 64 points (in 16-point units)	Section 5.3
No setting	C0 to C65535	QnACPU Programming Manual (Fundamentals)
I28→100ms I29→40ms I30→20ms I31→10ms	5ms to 1000ms (in 5ms units)	
–	–	–
Not used	• Not used • Use the same file name as the program. • Use the following file.	QnACPU Programming Manual (Fundamentals)
Not used	• Not used • Use the same file name as the program. • Use the following file.	Section 11.5
Use the same file name as the program.	• Not used • Use the same file name as the program. • Use the following file.	QnACPU Programming Manual (Fundamentals)
Not used	• Not used • Use the following file.	QnACPU Programming Manual (Fundamentals)

Item			Parameter No.	Description	
Device			–	Set the number of points, latch range, etc., for each device.	
	Device points		2000H	Set the number of device points used.	
	Latch (1) start (Enable C/L key)		2001H	Set the latch range for which latch clear key operation is enabled.	
	Latch (2) start (Disable C/L key)		2002H	Set the latch range for which latch clear key operation is disabled.	
	Local device		2003H	Set the range of devices to be set as local devices.	
PLC RAS			–	Set various kinds of settings for the RAS function.	
	WDT setup	WDT setting	3000H	Set the watchdog timer for the CPU module.	
		Initial execution monitoring time			
		Low speed execution monitoring time			
	Error check		3001H	Set whether to detect the specified errors or not.	
	Operating mode when there is an error		3002H	Set the operation mode in which the CPU module enters when an error is detected.	
	Constant scan		3003H	Set the constant scan time.	
	Annunciator display mode	Display F No.	3004H	Set the display mode that is activated when an annunciator comes ON.	
		Comment display			
		Time of occurrence			
	Breakdown history		3005H	Set where the CPU module breakdown history is stored.	
	Low speed program execution time		3006H	Set the time required for execution of low-speed execution type programs.	

Setting		Reference Section/Reference Manual
Default value	Setting range	
–	–	–
X→8k points Y→8k points M→8k points L→8k points B→8k points F→2k points SB→2k points V→2k points S→8k points T→2k points ST→0k point C→1k point D→12k points W→8k points SW→2k points	Fixed to X (8k points), Y (8k points), S (8k points), SB (2k points), SW (2k points). Up to 32k points per device within a range of 28.8k words, including the above points However, the total for bit devices is 64k points.	QnACPU Programming Manual (Fundamentals)
No setting	1 range only for each device	Section 10.3
No setting	1 range only for each device	Section 10.3
No setting	1 range only for each device	QnACPU Programming Manual (Fundamentals)
–	–	—
200ms	10ms to 2000ms (in 10ms units)	Section 9.2
No setting	10ms to 2000ms (in 10ms units)	Section 12.1.1
No setting	10ms to 2000ms (in 10ms units)	Section 12.1.3
Checked	Error checked	Section 9.3
Stop	Stop/Continue	Section 9.3
No setting	5ms to 2000ms (in 5ms units)	Section 10.2
Displayed	Displayed/Not displayed	Section 9.8.2
Not displayed	Displayed/Not displayed	
Not displayed	Displayed/Not displayed	
Stored in built-in RAM	Stored in built-in RAM/specified history file	Section 9.4
No setting	1ms to 2000ms (in 1ms units)	QnACPU Programming Manual (Fundamentals)

Item			Parameter No.	Description
I/O Assign			–	Set the mounting status of each module.
	Slot setting	Classification	4000H	Set the module type, number of points, head I/O No., etc.
		Number of points		
		Start XY		
		Model Name		
	Base setting	Power model name	4001H	Set model names of a power supply module and/or extension cables. This setting does not affect CPU module operation.
		Extension cable		
MELSECNET/Ethernet setting			–	Set link parameters for the MELSECNET (II) data link system, network parameters for the MELSECNET/10 network system or Ethernet parameters.
		Unit count	5000H	
		Valid module for access to other station	5001H	
	MELSECNET (II) and MELSECNET/ 10 network setting	Inter-device transfer parameters	5002H	
		Routing parameter	5003H	
		Network setting	5NM0H	
		Network refresh parameter	5NM1H	
		Common parameter	5NM2H	
		Station inherent parameter	5NM3H	
		I/O assignment	5NM4H	
	Ethernet network setting	Group No.	9N00H	
		IP address		

Setting		Reference Section/Reference Manual
Default value	Setting range	
–	–	–
No setting	Empty/Input/Output/Special	Section 5.3
No setting	0 to 64 points (in 16-point units)	
No setting	0 to 1FFFH (in 10H units, hexadecimal)	
No setting	Up to 16 characters	
No setting	Up to 16 characters	Section 5.3
–	For QnA/Q4AR MELSECNET/10 Network System Reference Manual	For QnA/Q4AR MELSECNET/10 Network System Reference Manual MELSECNET, MELSECNET/B Data Link System Reference Manual

* N and M indicate the following:
N: Number of the module counted from the first.
M: Network type

M	Network type	M	Network type	M	Network type
0H	MELSECNET/10 (Default)	7H	MELSECNET (Local station)	DH	MELSECNET/10 (Multiple remote submaster, No remote master in the host CPU module)
1H	MELSECNET/10 (Control station)	8H	MELSECNET II mixed (Local station)		
2H	MELSECNET/10 (Normal station)	9H	MELSECNET II (Local station)	EH	MELSECNET/10 (Multiple remote submaster, There is a remote master in the host CPU module.)
3H	MELSECNET/10 (Remote master station)	AH	MELSECNET/10 (Standby station)		
4H	MELSECNET (Master station)	BH	MELSECNET/10 (Multiple remote master)	FH	MELSECNET/10 (Parallel remote submaster)
5H	MELSECNET II mixed (Master station)	CH	MELSECNET/10 (Parallel remote master)		
6H	MELSECNET II (Master station)				

Item			Parameter No.	Description	
MELSECNET/MINI setting			–	Make the settings for automatic refresh of the MELSECNET/MINI system.	
Number of master modules			6000_H	Set the number of MELSECNET/MINI master modules to be used.	
	MELSECNET/ MINI detailed settings	Master module head I/O No.	$600N_H$*	Make the detailed setting required for automatic refresh of the MELSECNET/MINI system.	
		Model name & number of stations			
		Receive data batch refresh			
		Send data batch refresh			
		Retry count for communication errors			
		FROM/TO instruction access priority			
		Receive data clear at communication error			
		Faulty station detection bit data			
		Error No.			
		MINI link operation when CPU stopped			
		Circuit error check			
Supplementary settings			7000_H	Perform various settings required when multiple programs are used.	
	Program setting			Set programs to be executed among multiple programs.	
	Boot file setting			Set the file for boot operation and other settings.	
SFC			–	Perform various settings required for SFC programs.	
	SFC program start mode		8002_H		
	Start condition		8003_H		
	Output mode when the block is stopped		8005_H		
Acknowledge XY assignment			–	Allows confirmation of the settings made in I/O assignment. This setting does not affect CPU module operation.	

* N means the number of the master module counting from the first. (N: 1 to 8)

Setting		Reference Section/Reference Manual
Default value	Setting range	
–	–	Chapter 7
0	0 to 8	
No setting	Number of I/O points of CPU module	
MINIS3	MINIS3/MINI( ) stations	
X1000 to 200H	X, M, L, B, T, ST, C, D, W, R, ZR, none (Bit device: multiples of 16)	
Y1000 to 200H	Y, M, L, B, T, ST, C, D, W, R, ZR, none (Bit device: multiples of 16)	
5 times	0 to 32 times	
CPU	CPU/Link	
Clear	Clear/Hold	
No setting	M, L, B, T, ST, C, D, W, R, ZR, none	
No setting	D, W, T, ST, C, R, ZR	
Stop	Continue/Stop	
Latch data	Test message/OFF data/Latch data	
–	–	QnACPU Programming Manual (Fundamentals)
No setting	Program name/Scan/Low-speed/Initial/Standby	
No setting	File name/Type/Transfer source drive/ Transfer destination drive	
–	QCPU (Q mode)/QnACPU Programming Manual (SFC)	QCPU (Q mode)/QnACPU Programming Manual (SFC)
–	–	GX Developer Operating Manual SW☐IVD-GPPQ Software Package Operating Manual (Offline)

Item			Parameter No	Description
Network parametars Setting the CC-Link			-	Make the settings for automatic refresh of the CC-Link system.
Number of CC-Link			C000H	Set the number of CC-Link master modules to be used.
	CC-Link detailed settings	Master module head I/O No	CNM2H	Make the detailed setting required for automatic refresh of the CC-Linkl system.
		Module type	CNM1H	
		Receiving data batch refresh bit device (Input data)		
		Transmission data batch refresh bit device (Output data)		
		Receiving data batch refresh word device (Remote device: RWr)		
		Transmission data batch refresh device (Remote device: RWw)		
		Batch refresh device for special relay		
		Batch refresh device for special register		
		Number of retries	CNM2H	
		Number of automatic return stations		
		Standby master staiton No.		
		PLC down select		
		Scan mode setting		
		Delay timer		
	Station information setting	Station type		
		Number of occupied stations		
		Specification of reserved station/Specification of invalid station		

Setting		Reference Section/Reference Manual
Default value	Setting range	
-	-	Chapter 7
-	1 to 8	
-	0000H to 0FE0H	
-	M: Master station/L: Local station/ T: Stand-by station	
-	X,M,L,B,T,ST,C,D,W,R,ZR	
-	Y,M,L,B,T,ST,C,D,W,R,ZR	
-	M,L,B,T,ST,C,D,W,R,ZR	
-	M,L,B,T,ST,C,D,W,R,ZR	
-	M,L,B,T,ST,C,D,W,R,ZR	
-	T,ST,C,D,W,R,ZR	
-	1 to 7	
-	1 to 10	
-	Not reserved/Reserved	
-	Continue/Stop	
-	Synchronization/Non-synchronization	
-	1 to 100 (0 is invalid.)	
-	Remote I/O station/Remote device station /Intelligent device station	
-	1 station/2 station/3 station/4 station	
-	Specification of reserved station/Specification of invalid station	

* N and M indicate the following:
N:Number of the module counted from the first.
M:Network type M:Network type

M	Network type	M	Network type
0H	Master station)	4H	MELSECNET (Master station)
1H	Local station)	5H	MELSECNET II mixed (Master station)
2H	Standby master station	6H	MELSECNET II (Master station)
3H	MELSECNET/10 (Remote master station)		

14 SELECTING MEMORY CARD MODELS

Since the Q2ASCPU has a built-in RAM as a standard feature to store parameters and programs, programs can be executed without installing a memory card.
Each CPU model has a built-in RAM of the following program capacity.

Q2ASCPU....................	28k steps	(112k bytes)
Q2ASCPU-S1.............	60k steps	(240k bytes)
Q2ASHCPU.................	28k steps	(112k bytes)
Q2ASHCPU-S1............	60k steps	(240k bytes)

14.1 Applications of Memory Cards

A memory card is required in the following cases:

(1) To perform a boot operation
Parameters, programs, initial device values, comments, and boot files are stored in a memory card, and they are loaded to the built-inRAM at the time of program execution.

(2) To use file registers.[*1]

(3) To use local devices.[*2]

(4) To use a simulation data file with the simulation function.[*2]

(5) To use the sampling trace function.[*2]

(6) To use the status latch function.[*2]

(7) To use the program trace function.[*2]

(8) To store the breakdown history data in a file.[*2]

(9) To execute programs of the maximum number of steps available for the Q2ASCPU.
When a program of the maximum capacity is stored in the built-in RAM, the parameter files and initial device values must be stored in a memory card.

(10) To use the SFC trace function.[*2]

*1 They will be read-only in programs if they are set in the ROM area of the memory card.

*2 Can only be set in the RAM area of the memory card.

14.2 Selecting Memory Card Capacity

Select a memory card capacity according to the types and sizes of files to be stored in the memory card. The sizes of files are calculated using the formulas presented below.

Function	Approximate File Capacity (Unit: Bytes)
Drive title	64
Keyword	72
Parameters[3]	MELSECNET, NET/10 None → 330 When MELSECNET (II, /B) set → Max. 4096 per module
Boot file	(Number of files × 18) + 67
Sequence program[3]	(Number of steps × 4) + 122
Device comments[3]	(Total commend data size of each device) + 74 • Setting with GX Developer The comment data size of 1 device is as follows: 10250 × a + 40 × b + 10 (Quotient of (No. of devices / 256) is substituted for a and the remainder for b.) • Setting with SW□IVD-GPPQ Although the size varies depending on EMS capacity, it is equivalent to or less than the size obtained in the above DX Developer case.
Initial device value[3]	(Number of device points × 2) + (device types[1] × 44) + 66
File register	Number of points for file registers × 2 bytes
Local device	$(72 + (6 \times \text{No. of Setting range}^{*4}) + (2 \times \text{No. of word devices}) + \underset{\text{Round-up}}{\underline{(\text{No. of bit devices / 8})}} \times \text{No. of program files used}$
Simulation data	$(\text{Number of word device points} \times 2) + \underset{\text{Rounded up}}{\underline{(\text{number of bit device points / 16})}} \times 2 + (\text{device ranges}^{*2} \times 44) + 66$
Sampling trace data	362 + (No. of word device points + No. of bit device points) × 12 + (N1 + N2 + N3 + No. of word device points × 2 + (No. of bit device points/16) × 2) × trace count (total count)[5] • According to the items set in the added trace information on the trace device setting screen, the following values are added forN1 to N3. (Refer to Section 8.5 (2) (b)) N1: When setting time, "4" is added. N2: When setting step No., "10" is added. N3: When setting the program name, "8" is added.
Status latch data	For all devices : 58576 For detailed devices : (Number of word device points × 2) $+ \underset{\text{Rounded up}}{\underline{(\text{number of bit device points / 16})}} \times 2 + (\text{device types} \times 8) + 352$
Program Trace Data	Same as sampling trace
Breakdown history data	54 × number of faults stored + 72 bytes
SFC trace data	Max. 48k (in 1 kbyte units)

[1] "Device types" represents the number of registered device names.
For example, if D, W, and T are registered, it is 3.

[2] "Device ranges" represents the number of registered range settings.

[3] These files can be transferred from the memory card to the built-in RAM in the boot operation.

[4] The total number of setting ranges is the total number of types of the devices that are set as local devices.

[5] Decimal fraction of "number of bit device points/16" is rounded up.

POINT

Note that the capacity may be rounded up as follws depending on the memory area used for storage:

Built-in RAM.................. 4096 bytes (1k step) units
Memory card................. 512 bytes units

Note that, when a file is transferred from the memory card to the built-in RAM in boot operation, the reserved capacity is changed after transfer.

15 HARDWARE SPECIFICATIONS OF CPU MODULES

15.1 SPECIFICATIONS

The general specification common to various modules is shown.

Specifications

Item	Specifications					
Operating ambient temperature	0 to 55℃					
Storage ambient temperature	-20 to 75℃					
Operating ambient humidity	10 to 90 % RH, No-condensing					
Storage ambient humidity	10 to 90 % RH, No-condensing					
Vibration resistance	Conforming to JIS B 3502, IEC 61131-2		Frequency	Acceleration	Amplitude	Sweep count
		Under intermittent vibration	10 to 57Hz	–	0.075mm (0.003 inch)	10 times each in X, Y, Z directions
			57 to 150Hz	9.8m/s^2	–	
		Under continuous vibration	10 to 57Hz	–	0.035mm (0.001inch)	–
			57 to 150Hz	4.9m/s^2	–	
Shock resistance	Conforming to JIS B 3502, IEC 61131-2 (147m/s^2 , 3 times in each of 3 directions XYZ)					
Operation ambiance	No corrosive gasses					
Operating elevation*3	2000m (6562 ft.) or less					
Installation location	Control panel					
Overvoltage category*1	II or lower					
Pollution degree*2	2 or lower					
Equipment category	Class I					

*1 This indicates that the equipment is assumed to be connected to which power distributer in the area from the public electrical power distribution network to machinery in the premises.
Category II applies to equipment to which electrical power is supplied from fixed facilities.
The surge voltage withstand level for up to the rated voltage of 300V is 2500V.

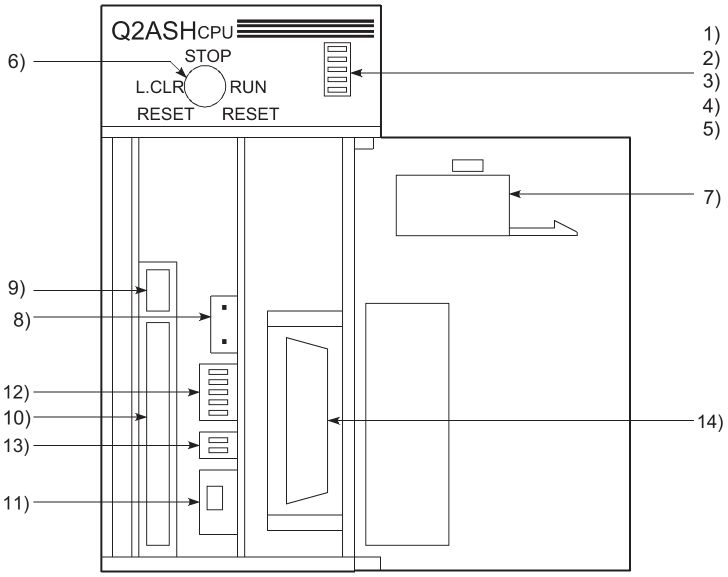
*2 This index indicates the degree of conductive material generation in the environment where the equipment is used.
In Pollution degree 2, only non-conductive pollution occurs. Occasionally, however, temporary conductivity caused by condensation can be expected.

*3 Do not use or store the programmable controller in the environment where the pressure is higher than the atmospheric pressure at sea level. Otherwise, malfunction may result.
To use the programmable controller in high-pressure environment, please contact your local Mitsubishi representative.

15.2 Part Names

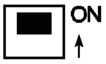
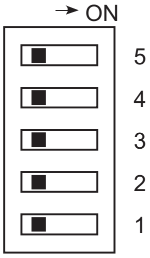
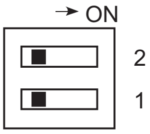
The names of module parts and their settings are described here.

Q2ASCPU, Q2ASCPU-S1, Q2ASHCPU, Q2ASHCPU-S1



Viewed with the front cover open

No.	Name	Application
1)	RUN LED	Indicates the operating status of the CPU module. ON: Operating with the RUN/STOP key switch set to RUN or STEP RUN. OFF: Stopped with the RUN/STOP key switch set to STOP, PAUSE, or STEP RUN. Or, an error that stops operation was detected. Flickering: The RUN/STOP key switch was shifted from STOP to RUN after writing a program in the STOP state. The CPU module is not in the RUN state.To actually put the CPU module in the RUN state, either move the switch one more time from "RUN" → "STOP" → "RUN", or use the RUN/STOP key switch to perform reset operation.Alternatively, reset it with the RUN/STOP key switch.
2)	ERROR LED	ON: A self-diagnostics error that does not stop operation, other than a battery error, has been detected. (When the parameter setting is made for operation to continue when an error occurs.) OFF: Normal Flickering: An error that stops operation has been detected.
3)	USER LED	ON: An error has been detected by the CHK instruction, or an annunciator F has come ON. OFF: Normal Flickering: When latch clear is performed.
4)	BAT.ALARM LED	ON: A battery error occurred due to low battery voltage in the CPU module or memory card. OFF: Normal
5)	BOOT LED	ON: Execution of the boot operation is completed. OFF: No boot operation has been executed.
6)	RUN/STOP key switch	RUN/STOP: Starts/stops sequence program operation. L.CLR: Clears all data in the latch area (to "OFF" or "0") which is set with parameters. Clears sampling trace and status latch registrations. RESET: Resets the hardware. Resets and initializes operation when an operation error occurred.

<table>
<tr><th>No.</th><th>Name</th><th colspan="5">Application</th></tr>
<tr><td>7)</td><td>Battery (A6BAT)</td><td colspan="5">Backup battery for the built-in RAM and the power failure compensation function.</td></tr>
<tr><td>8)</td><td>Battery connector pin</td><td colspan="5">Used for connection of the battery lead wire.
(To prevent battery drain, the battery lead wire is disconnected from the connector before shipment. See Section 18.6.)</td></tr>
<tr><td>9)</td><td>Memory card EJECT button</td><td colspan="5">Used to eject the memory card from the CPU module.(Refer to Section 18.7)</td></tr>
<tr><td>10)</td><td>Memory card installing connector</td><td colspan="5">Connector for installing the memory card in the CPU module.</td></tr>
<tr><td>11)</td><td>Memory card in/out switch
(with built-in LED)
 </td><td colspan="5">Used to enable/disable memory card installation or removal while the power is ON.Factory-set to OFF.
ON: Cannot be removed (LED lit)
OFF: Can be removed (LED unlit)
Refer to Section 15.3 (3) and (4) for installation or removal of a memory card.</td></tr>
<tr><td rowspan="9">12)</td><td rowspan="9">System setting switch 1
 </td><td colspan="5">Settings required to operate the CPU module are made.All switches are set to OFF before shipping.</td></tr>
<tr><td colspan="5">SW5: Boot file settingSetting of the memory used for operation.
ON: Boot operation
OFF: Boot operation is not performed.</td></tr>
<tr><td colspan="5">SW2 to 4: Parameter areaSetting of the memory in which parameters are written.</td></tr>
<tr><td rowspan="2">System setting switch 1</td><td rowspan="2">Built-in RAM</td><td colspan="2">Memory card</td><td rowspan="5">* SW2 to 4 are valid even if SW5 is OFF.</td></tr>
<tr><td>RAM</td><td>ROM</td></tr>
<tr><td>SW4</td><td>OFF</td><td>ON</td><td>OFF</td></tr>
<tr><td>SW3</td><td>OFF</td><td>OFF</td><td>ON</td></tr>
<tr><td>SW2</td><td>OFF</td><td>OFF</td><td>OFF</td></tr>
<tr><td colspan="5">SW1: System protectProhibition of all writing and control directions to the CPU module.
ON: System protection enabled
OFF: System protection disabled</td></tr>
<tr><td rowspan="3">13)</td><td rowspan="3">System setting switch 2
 </td><td colspan="5">Settings required to operate the CPU module are made.All switches are set to OFF before shipping.</td></tr>
<tr><td colspan="5">SW2: Not used. (Fixed to OFF.)</td></tr>
<tr><td colspan="5">SW1: Peripheral protocol.Select the type of the peripheral device connected to the peripheral interface of the CPU module.
(When accessing an ACPU on another station from a peripheral device for ACPU, set this switch to “ON”. The setting becomes valid immediately after switching.)
ON: Peripheral device for ACPU</td></tr>
<tr><td>14)</td><td>RS-422 connector</td><td colspan="5">Connector for connecting to a peripheral device.</td></tr>
</table>

15.3 Relationship between Switch Operations and LEDs/LED Display

(1) Writing programs with the CPU module in STOP state

To write a program to the CPU module while it is in the STOP state, use the following procedure.

1) Set the RUN/STOP key switch to STOP
   RUN LED :OFF
2) Set the RUN/STOP key switch to RESET
   RUN LED :OFF ......CPU module in RESET state
3) Set the RUN/STOP key switch to STOP → RUN
   RUN LED :ON ......CPU module in RUN state

POINT

(1) For the Q2ASCPU, after writing a program (except for writing to PLC during RUN), set the CPU module to RESET and then to RUN.

(2) If the key switch is set to RUN without resetting, the CPU module will remain in STOP state displaying as follows:

RUN LED :Flickers *1

After this occurs, the CPU can be placed into RUN state by setting the RUN/STOP key switch to RESET.

In this case, internal CPU module data such device data are cleared.

(3) To prevent the internal CPU module information from being cleared, switch the RUN/STOP key switch STOP→RUN again without resetting.

*1 If Remote STOP→RUN is performed for the CPU module, the CPU will be in RUN status, not in "PROG.CHECK" status.

(2) Performing latch clear

To perform latch clear, operate the RUN/STOP key switch as follows.

1) Move the "RUN/STOP" key switch of the CPU module from the "STOP" to the "L.CLR" position several times to flicker the "USER LED" on the CPU module front.
Normally, the LED flickers after the switch is moved several (three or four) times. When the "USER LED" flickers, it indicates that latch clear is ready.
2) After the "USER LED" has flickered, moving the key switch from the "STOP" to the "L.CLR" position again executes latch clear and lights up the "USER LED". When the "USER LED" is lit for 2 seconds and then goes off, it indicates normal completion of latch clear.
3) To cancel latch clear midway, move the key switch to the "RUN" position to put the CPU module in a RUN status or to the "RESET" position to reset.

POINT

(1) The latch clear operation can be set enabled or disabled for each device in the device setting in the parameter mode.
(2) Remote latch clear executed by the GPP function is an alternative method other than using the RUN/STOP key switch.(Refer to Section 10.6.5)

(3) Removing a memory card while the programmable controller power is ON:

When removing a memory card with the programmable controller power ON, operate the memory card in/out switch as follows:

1) In/out switch: ON.
LED in the switch :ON .... Memory card removal prohibited
2) In/out switch: OFF
LED in the switch :OFF .... Memory card removal permitted
Removal of memory card

POINT

(1) The LED in the in/out switch may not come OFF if the memory card is being used for a CPU module system function (sampling trace, status latch, etc.) or by a program. In such a case, stop the function or program using the memory card. After aborting it, confirm that the LED in the in/out switch has gone OFF, then remove the memory card.
(2) When a file register, local device or breakdown history set with parameters is present, the memory card cannot be removed.
Even if the memory card in/out switch is turned OFF, its built-in LED does not turn OFF. When the file register is set to "Not used" with the QDRSET (P) instruction, the memory card can be removed.
(3) After removing the memory card, do not turn on the memory card insertion/ disconnection switch for preventing an error.

(4) Installing a memory card while the programmable controller power is ON:
When installing a memory card with the programmable controller power ON, operate the memory card in/out switch as follows:

1) Install the memory card.
2) In/out switch: ON
   LED in the switch :ON .... Memory card removal prohibited

POINT

(1) After installing the memory card, set the memory card in/out switch to ON. If it is not set to ON, the memory card cannot be used.
(2) During one scan after the memory card installation, mounting processing is performed again. Note that the scan time may be increased by 10ms at maximum.

16 POWER SUPPLY MODULE

This section describes the specifications and selection of power supply modules.

16.1 Specifications

16.1.1 Power supply module specifications

(1) Standard power supply module

Power supply module specifications

Item		Performance specifications		
		A1S61PN	A1S62PN	A1S63P
Slot position		Power supply module slot		
Input power supply		100 to 240VAC$^{+10\%}_{-15\%}$ (85 to 264VAC)		24VDC$^{+30\%}_{-35\%}$ (15.6 to 31.2VAC)
Input frequency		50/60Hz± 5		–
Input voltage distortion		Within 5% (See Section 19.8)		–
Max. input apparent power		105VA		41W
Inrush current		20A, 8ms or less *4		81A, 1ms or less
Rated output current	5VDC	5A	3A	5A
	24VDC	–	0.6A	–
Overcurrent protection *1	5VDC	5.5A or higher	3.3A or higher	5.5A or higher
	24VDC	–	0.66A or higher	–
Overvoltage protection*2	5VDC	5.5 to 6.5V		
	24VDC	–		
Efficiency		65% or higher		
Allowable momentary power failure period *3		20ms or less		10ms or lower (24VDC or higher)
Dielectric withstand voltage	Between primary and 5VDC	AC across input/LG and output/FG, 2830VAC rms/3 cycles (altitude 2000m (6562ft.))		500VAC
	Between primary and 24VDC			–
Insulation resistance		AC across input/LG and output/FG 10MΩ or higher, measures with a 500VDC insulation resistance tester (10MΩ or above by insulation resistance tester)		
Noise durability		• Checked by noise simulator of noise voltage 1500Vp-p, noise width 1 μ, and noise frequency 25 to 60Hz • Checked by noise simulator of noise voltage IEC801-4, 2kV, 1500Vp-p, noise width 1 μs, and noise frequency 25 to 60Hz		• Checked by noise simulator of noise voltage 1500Vp-p, noise width 1 μ, and noise frequency 25 to 60Hz
Power indicator		Power LED indication (light at the time of output of 5VDC)		
Fuse		Built-in (User cannot change.)		

Item	Performance specifications		
	A1S61PN	A1S62PN	A1S63P
Terminal screw size	M3.5×7		
Applicable wire size	0.75 to 2mm^2		
Applicable solderless terminal	RAV1.25 to 3.5, RAV2 to 3.5		
Applicable tightening torque	59 to 88N・cm		
External dimensions	130mm×55mm×93.6mm		
Weight	0.60kg	0.60kg	0.50kg

REMARK

1) The number of occupied slots for the A66P is 1.

POINT

*1 Overcurrent protection

(a) The overcurrent proctector shuts off the 5VDC and/or 24VDC circuit(s) and stops the system if the current exceeding the specified value flows in the circuit(s).
As this results in voltage drop, the power supply module LED turns OFF or is dimly lit.

(b) When this device is activated, eliminate probable causes such as insufficient current capacity or short circuit, and then start the system. When the current has reached the normal value, the system will start from the first.

*2 Overvoltage protection

The overvoltage protector shuts off the 5VDC circuit and stops the system if overvoltage of 5.5 to 6.5V is applied to the circuit.
The power supply module LED turns OFF. When restarting the system, switch the input power OFF, then back ON. The system is started up with an initial start. If the system is not booted and the LED remains off, this means that the power supply module has to be replaced.

*3 Allowable momentary power failure period

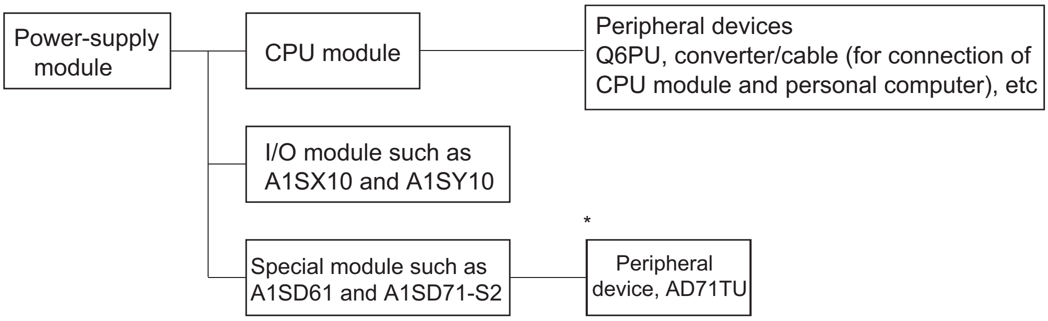
The allowable momentary power failure period of programmable controller CPUs varies depending on the power supply module used.
In the system using the A1S63P, it is the time from when the primary side of the stabilized power supply supplying 24VDC to the A1S63P turns OFF until the voltage (secondary side) has dropped from 24VDC to the specified value (15.6VDC) or less.

*4 Inrush current

If power is reapplied immediately after power OFF (within 5 seconds), an inrush current exceeding the specified value may flow (for 2ms or less). Therefore, before reapplying power, make sure that 5 seconds have elapsed after power off.
When selecting a fuse or breaker for an external circuit, consider the above as well as meltdown and detection characteristics.

16.1.2 Power supply module selection

A power supply module is selected based on to the total current consumption of I/O modules, special function modules and peripheral devices to which power is supplied by the power supply module. Remember that when an extension base module such as A1S52B(S1), A1S55B(S1), A1S58B(S1), A52B, A55B or A58B are used, power is supplied by the main base.
For 5VDC current consumption of I/O modules, special function modules and peripheral devices, refer to Section 3.3.



* Select the module in consideration of the current consumption of the peripheral device that will be connected to the special-function module. For example, when AD71TU is connected to A1SD71-S2, the current consumption of AD71TU must also be considered.

(2) Selection of power-supply module when extension base modules such as A1S52B(S1), A1S55B(S1), A1S58B(S1), A52B, A55B and A58B are used
When extension base modules such as A1S52B(S1), A1S55B(S1), A1S58B(S1), A52B, A55B and A58B are used, the 5 VDC power supply is supplied through the extension cable from the power-supply module of the main base module. Thus, when one of these units is used, pay attantion to the following:

(a) When mounting a power supply module on the main base unit, select a model that can cover 5VDC current consumed by modules mounted on the A1S52B(S1), A1S55B(S1), A1S58B(S1), A52B, A55B, and/or A58B.
[Example] When the 5 VDC current consumption on the main base module is 3A and the 5 VDC current consumption on A1S55B is 1A, the power-supply module that is loaded into the main base module must be A1S61PN(DC5V 5A).

(b) Since power to the A1S52B(S1), A1S55B(S1), A1S58B(S1), A52B, A55B or A58B is supplied via an extension cable, a voltage drop occurs through the cable. It is necessary to select a power supply module and cables with proper length so that 4.75VDC or more is available on the receiving end. Refer to the usage standard of the Extension Base Module in Section 17.3 for details on voltage drops, etc.

16.2 Precautions for Handling

The following explains the handling precautions for unpacking to mounting of the power supply module.

(1) Do not drop the power supply module or give it hard shock since its case, terminal block connectors and pin connectors are made of resin.

(2) Tighten the module mounting screws (unnecessary in normal operating status), terminal screws, etc. in the following ranges.

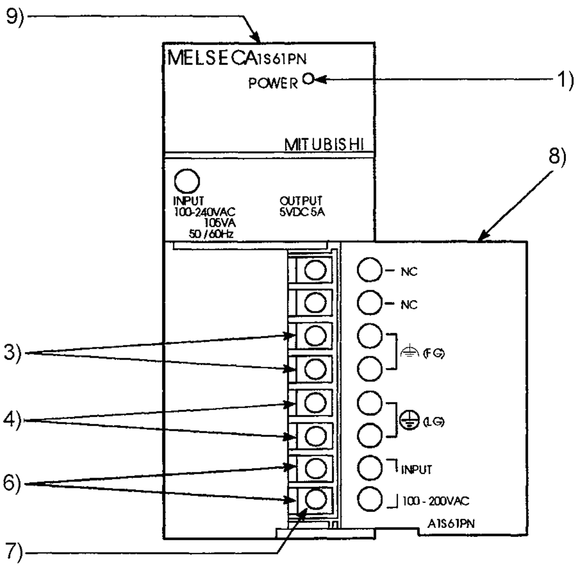
Screw	Tightening torque range
Power supply module terminal block terminal screw (M3 screw)	39 to 59N・cm
Power supply module terminal block terminal screw (M4 screw)	98 to 137N・cm
Module mounting screws (Optional) (M4 screw)	78 to 118N・cm

(3) When installing the module to the base unit, press the module completely so that its hook is locked into the base. When dismounting the module, press the hooks until they come off the base completely, and then pull the module toward you. (See Section 19.5.)

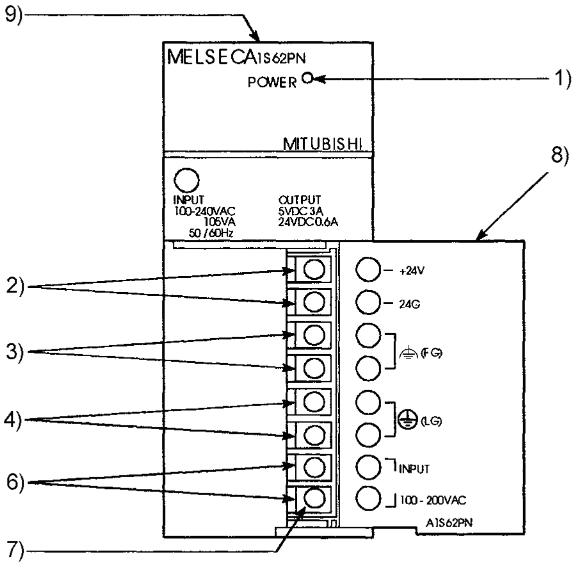
16.3 Part Names

The following gives the names and description of the parts of the power supply modules :

(1) A1S61PN

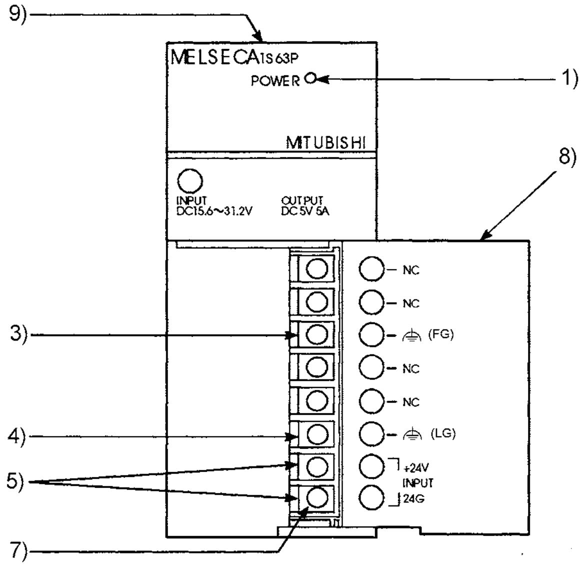


(2) A1S62PN



No.	Name	Application
1)	POWER LED	The indicator LED for the 5 V DC power.
2)	24 V and 24 G terminals	Used to supply 24 V DC to inside the output module (using external wiring).
3)	FG terminal	The grounding terminal connected to the shield pattern of the printed circuit board.
4)	LG terminal	Grounding for the power supply filter. The potential of A1S61P or A1S62P terminal is 1/2 of the input voltage.

(3) A1S63P



No.	Name	Application
5)	Power supply input terminals	Used to connect a 24VDC power supply.
6)	Power supply input terminals	Used to connect 100VAC to 200VAC power supply.
7)	Terminal screw	M3.5×7
8)	Terminal cover	The protective cover of the terminal block.
9)	Module mounting screw	Used to fix the module to the base unit. (M4 screw, tightening torque : 59 to 88 N・cm)

POINT

(1) Do not cable to the unused terminals such as FG and LG on the terminal block (terminals whose name is not printed on the terminal cover).
(2) Be sure to ground the terminal LG to the protective ground conductor.

17 BASE UNIT AND EXTENSION CABLE

This section explains the specifications of the base units (the main and extension base units) and extension cables available for the systems, and the application standards for use of extension base units.

17.1 Base Unit Specifications

(1) Main base unit specifications

Table 17.1 Main base unit specifications

Item	A1S32B	A1S33B	A1S35B	A1S38B	A1S38HB
I/O module installation range	2 modules can be installed.	3 modules can be installed.	5 modules can be installed.	8 modules can be installed.	8 modules can be installed.
Extension possibility	Extendable				
Installation hole size	φ6 bell-shaped holes (for M5 screws)				
External dimensions	220mm (3.3inch) ×130mm (2.1inch) ×28mm (0.1inch)	255mm (3.3inch) ×130mm (2.1inch) ×28mm (0.1inch)	325mm (3.3inch) ×130mm (2.1inch) ×28mm (0.1inch)	430mm (3.3inch) ×130mm (2.1inch) ×28mm (0.1inch)	
Weight	0.52kg	0.65kg	0.75kg	0.97kg	1.0kg
Accessory	Attaching screws: M5×25 4 screws				

(2) Extension base unit specifications

Table 17.2 Extension base unit specifications

Item	A1S65B	A1S65B-S1	A1S68B	A1S68B-S1	A1S52B	A1S52B-S1	A1S55B	A1S55B-S1	A1S58B	A1S58B-S1
I/O module installation range	5 modules can be installed.		8 modules can be installed.		2 modules can be installed.		5 modules can be installed.		8 modules can be installed.	
Power supply module loading necessity	Power supply module required				Unnecessary (Refer to POINT below)					
Installation hole size	φ6 bell-shaped holes (for M5 screws)									
Terminal screw size	–		–		M4×6(FG terminal)					
Applicable wire size	–		–		0.75 to 2mm^2					
Applicable solderless terminal	–		–		(V) 1.25-4 (V) 1.25-YS4(V)2-YS4A (Applicable tightening torque: 78 to 118N・cm					
External dimensions	315mm (3.3inch) ×130mm (2.1inch) ×28mm (0.1inch)		420mm (3.3inch) ×130mm (2.1inch) ×28mm (0.1inch)		135mm (3.3inch) ×130mm (2.1inch) ×28mm (0.1inch)		260mm (3.3inch) ×130mm (2.1inch) ×28mm (0.1inch)		365mm (3.3inch) ×130mm (2.1inch) ×28mm (0.1inch)	
Weight	0.71kg		0.95kg		0.38kg		0.61kg		0.87kg	
Accessory	Attaching screws: M5×25 4 screws				[*1] Dustproof cover (for I/O module): 1 pc. Attaching screws: M5×25 4 screws					

*1 1 For the attachment of the dustproof cover, refer to Section 19.6.

POINT

(1) 5VDC power for the A1S52B(S1), A1S55B(S1) or A1S58B(S1) is supplied from the power supply module mounted to the main base unit.
(2) Refer to Section 16.1.2, "Selecting the Power Supply Module" or Section 17.3, "Application Standards of Extension Base Unit" when using A1S52B(S1), A1S55B(S1) and A1S58B(S1).

17.1.1 Main base unit for high-speed access (A1S38HB/A1S38HBEU)

The main base units, (A1S38HB/A1S38HBEU) for high-speed access have been improved in the speed of access to the buffer memory of the special function module mounted on A1S38HB/A1S38HBEU.

POINT

(1) The A1S38HB/A1S38HBEU can perform high-speed access to the buffer memories of special function modules only. I/O devices of I/O modules are not accessed at high speed but at the same access speed as that of a conventional main base unit.

(2) When an extension base unit is connected to the A1S38HB/A1S38HBEU, the buffer memories of the special function modules on the extension base unit are not accessed at high speed. The access speed is the same as the one in the case of connecting to a conventional main base unit.

REMARK

(1) The A1S38HB/A1S38HBEU base unit is dedicated to the Q2ASCPU and cannot be used with the AnSCPU.

(2) When using the simulation module A6SIM-X64Y64, set its base unit specification to "1" or later.
If "0" is set, the A6SIM-X64Y64 does not operate normally.
When "0" is to be set for the base unit specification of the A6SIM-X64Y64, replace the base unit with the A38B.

17.2 Extension Cable Specification List

The specifications of the extension cables used for the Q2ASCPU system are shown below:

Item	A1SC01B	A1SC03B	A1SC07B	A1SC12B	A1SC30B	A1SC60B	A1SC05NB	A1SC07NB	A1SC30NB	A1SC50NB
Cable length	0.055m	0.33m	0.7 (2.30)	1.2 (3.94)	3.0 (9.84)	6.0 (19.69)	0.45 (1.48)	0.7 (2.30)	3.0 (9.84)	5 (16.43)
Resistance value of 5VDC supply line (at 55℃)	0.02Ω	0.02Ω	0.04Ω	0.06Ω	0.12Ω	0.18Ω	0.04Ω	0.05Ω	0.12Ω	0.18Ω
Application	Connection between a main base and A1S5□B(S1)/A1S6□B(S1)						Connection between a main base and A5□B/A6□B			
Weight	0.025	0.10	0.14	0.20	0.40	0.65	0.20	0.22	0.40	0.56

⚠ **CAUTION**

- Connect the extension cable to the connector of the base unit or module.
  After that, check for incomplete insertion.
  Poor electrical contact may cause incorrect inputs and/or outputs.
- When using extension cables, keep them away from the main circuit cables (high voltage, large current).

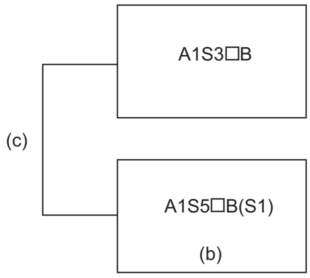
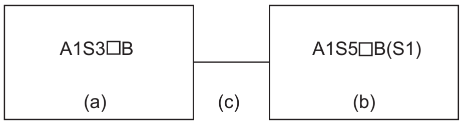
17.3 Application Standards of Extension Base Unit (A1S52B(S1), A1S55B(S1), A1S58B(S1), A52B, A55B, A58B)

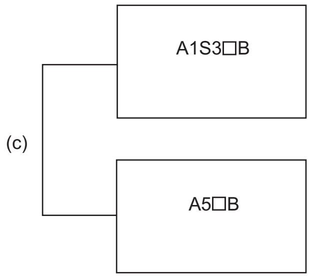
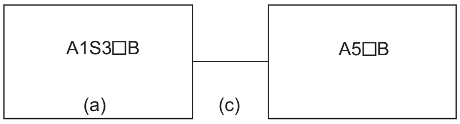
To the A1S52B(S1), A1S55B(S1), A1S58B(S1), A52B, A55B and A58B extension base units, 5VDC is supplied from the power supply module on the main base unit. (Power is not supplied from any power supply module on the A62B, A65B and A68B.)
Therefore, if a voltage drop occurs on an extension cable, the specified voltage may not supplied to the receiving end, resulting in erroneous inputs and outputs.
It is recommended to connect the A1S52B(S1), A1S55B(S1), A1S58B(S1), A52B, A55B and/or A58B after a main base unit to minimize a voltage drop.
Determine applicability of the A1S52B(S1), A1S55B(S1), A1S58B(S1), A52B, A55B and A58B by the following calculation method.

(1) Selection condition
The voltage received by the module installed in the last slot of an extension base unit A1S52B(S1), A1S55B(S1), A1S58B(S1), A52B, A55B or A58B must be 4.75 V or above.
Since the output voltage of the power supply module is set at 5.1 V or above, the voltage drop must be 0.35 V or less.

(2) Classification of voltage drop
Voltage drop is classified into (a), (b), and (c) as follows according to the connecting method and type of extension base units.

(a) Voltage drop of a main base unit
(b) Voltage drop of an extension base unit
(c) Voltage drop over an extension cable

Extension base unit used	Extension cable connected to the left side of main base unit (serial)	Extension cable connected to the right side of main base unit (Parallel)
A1S52B(S1), A1S55B(S1) or A1S58B(S1) extension base unit is used	 Voltage drop of the main base unit can be ignored.	

Extension base unit used	Extension cable connected to the left side of main base unit (serial)	Extension cable connected to the right side of main base unit (Parallel)
A52B, A55B or A58B extension base unit is used	 Voltage drop of the main and the extension base units can be ignored.	 Voltage drop of the extension base units can be ignored.

17.4 Handling Precautions

The handling precautions to be taken from unpacking to mounting a base unit are described below.

The terminal connectors and pin connectors of the base unit are made of resin. Do not drop them or apply heavy impact to them.

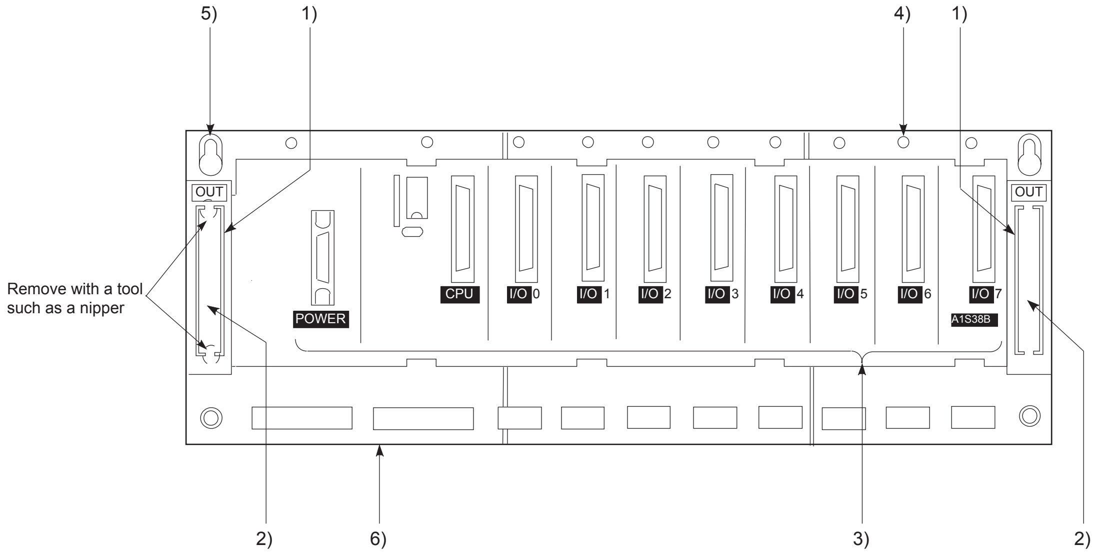
⚠ CAUTION

- Do not remove the printed-circuit board from the base unit.
  Doing so may cause failure, malfunctions, personal injuries and/or a fire.
- Use caution to prevent foreign matter, such as dust or wire chips, from entering the base unit during wiring.
  Failure to do so may cause a failure, malfunction or fire.

17.5 Part Names

Part names of the base unit are shown here.

(1) Main base module (A1S32B, A1S33B, A1S35B, A1S38B, A1S38HB, A1S38HBEU)



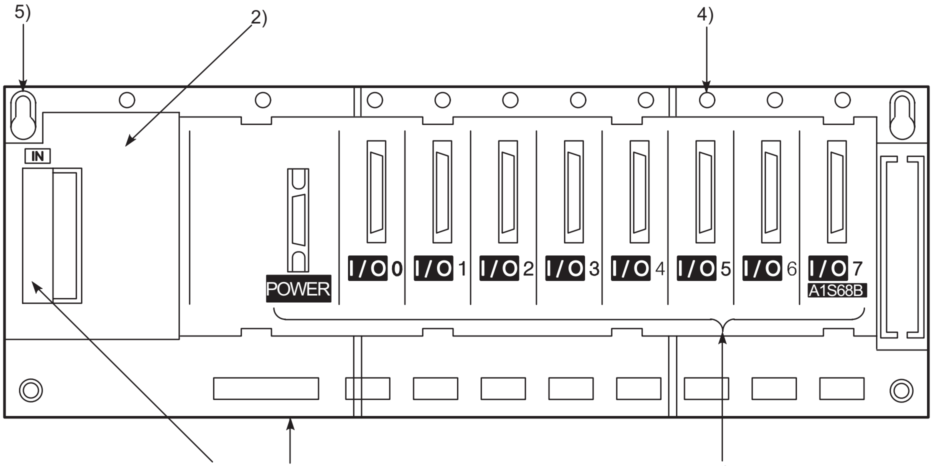
No.	Name	Application
1)	Extension cable connector	Connects an extension cable using a signal-communication connector for the extension base module.
2)	Base cover	Protective cover for the extension cable connector.To expand it, the area surrounded by grooves located below the word "OUT" on the base cover should be removed with a nipper, etc.
3)	Module connector	Connector to load the power supply module, CPU module, I/O module and specialfunction module. To prevent dust accumulation, load the attached connector cover, blank cover (A1SG60) or a dummy module (A1SG62) to reserved space connector not loaded a module.
4)	Module mounting screw	Screws to attach the module to the bases. Screw size: for M4 screw
5)	Base mounting hole	Mounting hole to attach the base module to the panel of the control panels, etc. (For M5 screw)
6)	DIN rail hook	Attachment hook for DIN rail One piece each for A1S32B and A1S33B. Two pieces each for A1S35B, A1S38B, A1S38HB and A1S38HBEU.

IMPORTANT

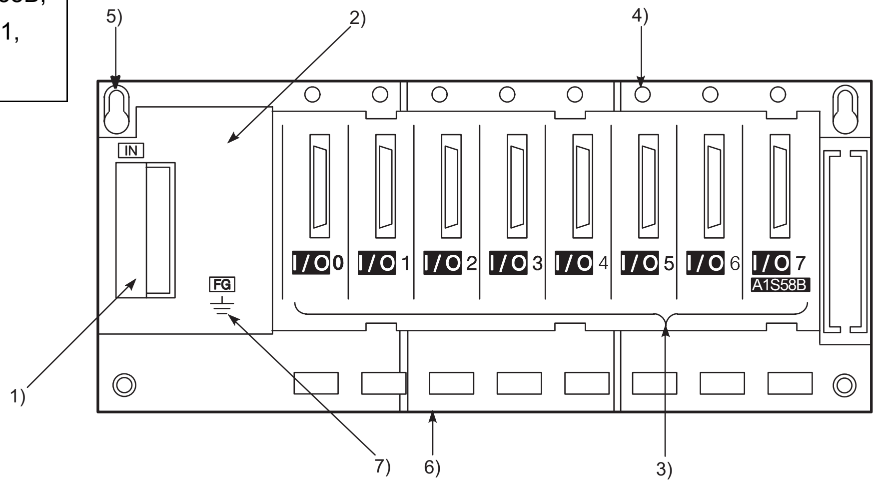
Only one extension base module can be connected to a main base module. Connecting two extension connectors of the main base module to extension base modules may result in input and/or output errors.

(2) Extension base module (A1S52B, A1S55B, A1S58B, A1S52B-S1, A1S55B-S1, A1S58B-S1, A1S65B, A1S68B, A1S65B-S1, A1S68B-S1)

A1S65B, A1S68B,
A1S65B-S1,
A1S68B-S1



A1S52B, A1S55B, A1S58B,
A1S52B-S1, A1S55B-S1,
A1S58B-S1



No.	Name	Application
1)	Extension cable connector	Connects an extension cable using a signal-communication connector for the extension base module. Connect the extension cable after removing the connector cover.
2)	Base cover	Protective cover for the extension cable connector.
3)	Module connector	Connector to load power-supply module, CPU module, I/O module and specialfunction module.To prevent dust accumulation, load the attached connector cover, blank cover (A1SG60) or a dummy module (A1SG62) to reserved space connector not loaded a module.
4)	Module mounting screw	Screws to attach the module to the bases. Screw size: for M4 screw
5)	Base mounting hole	Mounting hole to attach the base module to the panel of the control panels, etc. (For M5 screw)

No.	Name	Application
6)	DIN rail hook	Attachment hook for DIN rail One piece each for A1S52B, A1S55B, A1S52B-S1 and A1S55B-S1. Two pieces each for A1S65B, A1S68B, A1S58B, A1S65B-S1, A1S68B-S1 and A1S58B-S1.
7)	FG terminal	The grounding terminal connected to the shield pattern of the printed circuit board.

18 MEMORY CARDS AND BATTERIES

This section describes the specifications and handling of the memory cards and batteries that can be used with the Q2ASCPU.

18.1 Memory Card Specifications

The specifications of the memory cards that can be used with Q2ASCPU conform to JEIDA Ver. 4.0.

(1) SRAM type memory cards

Item	Model Name					
	Q1MEM-64S	Q1MEM-128S	Q1MEM-256S	Q1MEM-512S	Q1MEM-1MS	Q1MEM-2MS
SRAM memory capacity before formatting	64k bytes	128k bytes	256k bytes	512k bytes	1M bytes	2M bytes
SRAM memory capacity after formatting	59k bytes	123k bytes	250.5k bytes	506k bytes	1016.5k bytes	2036k bytes
Number of storable files	118	128				256
Insertion/removal limit	5000 times					
External dimensions	85.6mm (3.3inch) × 54mm (2.1inch) × 3.3mm (0.1inch)					
Weight	0.04kg					

(2) SRAM + E^2PROM type memory cards

Item		Model Name				
		Q1MEM-64SE	Q1MEM-128SE	Q1MEM-256SE	Q1MEM-512SE	Q1MEM-1MSE
Memory capacity before formatting	SRAM	32k bytes	64k bytes	128k bytes	256k bytes	512k bytes
	E^2PROM	32k bytes	64k bytes	128k bytes	256k bytes	512k bytes
Memory capacity after formatting	SRAM	28.5k bytes	58.5k bytes	122.5k bytes	250k bytes	505.5k bytes
	E^2PROM	29k bytes	59k bytes	123k bytes	250.5k bytes	506k bytes
Number of storable files	SRAM	57	117	128		
	E^2PROM	58	118	128		
Maximum number of writes to E^2PROM		10,000 times				

(2) SRAM + E^2PROM type memory cards

Item	Model Name				
	Q1MEM-64SE	Q1MEM-128SE	Q1MEM-256SE	Q1MEM-512SE	Q1MEM-1MSE
Insertion/removal limit	5000 times				
External dimensions	85.6mm (3.3inch) × 54mm (2.1inch) × 3.3mm (0.1inch)				
Weight	0.04kg				

18.2 Handling Memory Cards

(1) Formatting memory cards
All memory cards used with Q2ASCPU must be formatted.
The purchased memory card is not formatted. Use the memory card after formatting with the GPP function.

(a) SRAM+E^{2}PROM type memory card
Format both RAM and ROM.
If installed with only one of them formatted, the Q2ASCPU detects an error (ICM.OPE.ERROR).

For information on how to format SRAM and E^{2}PROM, see the following manual.

- GX Developer Operating Manual
- Type SW□IVD-GPPQ Software package Operating Manual (Online)

(2) Installing the battery in the memory card
The memory card is packaged with a RAM memory bakup battery.To use the RAM memory of the memory card, this battery must be installed first.

POINT

The battery installed in the CPU module does not back up RAM memories of memory cards.
Also, a battery installed in a memory card does not back up the internal RAM of a CPU module.

(3) Switch setting when using a memory card
When using a memory card, turn ON the memory card in/out switch which is close to the connector.If it is set to OFF, the memory card cannot be used.

18.3 Battery Specifications (CPU Module and Memory Card Batteries)

(1) CPU module batteries

Item	Model Name
	A6BAT
Type	Thionyl chloride lithium battery
Initial voltage	3.6VDC
Battery life when stored	5 years
Battery life when used	Refer to Section 21.3.1
Lithium content	0.48g
Application	Built-in RAM memory backup and power failure compensation
External dimensions	ϕ 16 × 30mm [0.6 × 1.2 inch]

REMARK

- For the battery directive in EU member states, refer to Appendix 11.

(2) Memory card batteries

Item	Model Name
	BR2325 or equivalent
Type	BR-type coin cell lithium battery
Initial voltage	3.0VDC
Battery life when stored	5 years
Battery life when used	Refer to Section 21.3.1
Lithium content	0.05g
Application	Card memory backup and power failure compensation

18.4 Handling Precautions

Handling precautions on memory cards and batteries from unpacking to mounting are listed below.

(1) Memory card
- (a) Do not drop, bend or apply any strong impact to the memory card.
- (b) Do not expose the memory card to water.
- (c) Do not expose the memory card to direct sunlight or leave it near a heat source.
- (d) Be careful to prevent dust from entering the connector.
- (e) Do not store the memory card in high temperature or high humidity areas.
- (f) To protect the memory card from static electricity, always enclose it in a plastic case before transporting or storing.
- (g) Do not touch the terminals of the memory card.

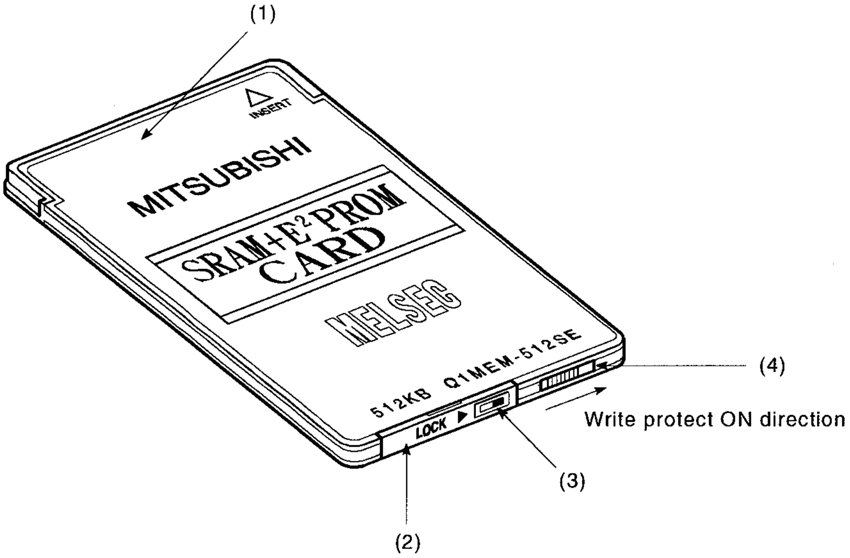
⚠CAUTION

● Insert the memory card and fully press it to the memory card connector.Check for incomplete connection after installing it.
Poor electrical contact may cause malfunctions.

(2) Battery
- (a) Do not short the battery.
- (b) Do not disassemble the battery.
- (c) Do not put it into a fire.
- (d) Do not heat it.
- (e) Do not apply solder to the battery poles.

18.5 Part Names of Memory Card

Part names of the memory card are shown below.

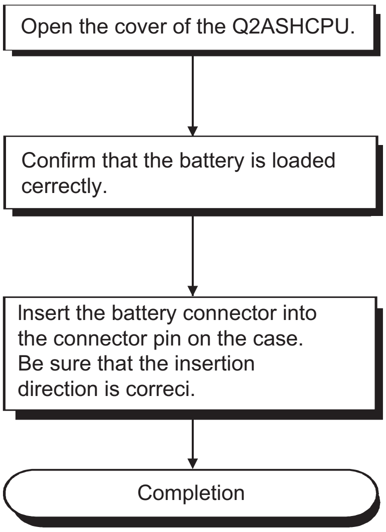


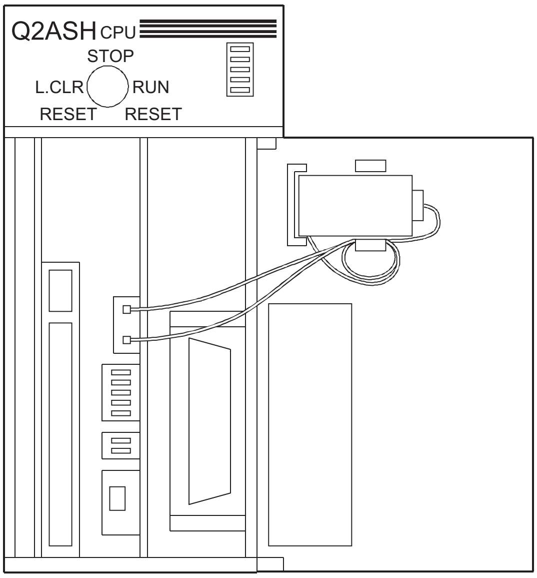
No.	Name	Description	Remark
(1)	Connector	Connects the memory card to the CPU module.	
(2)	Battery holder	Holds a lithium battery that is used to backup RAM data.	*
(3)	Battery holder locking switch	Locks the battery holder to the memory card. (Locked in "LOCK" position.)	
(4)	Write protect switch	Enables or disables writing to the memory.Factory-set to OFF. ON : Data writing disabled OFF : Data writing enabled	*

* Must be set before writing a program and starting operation.

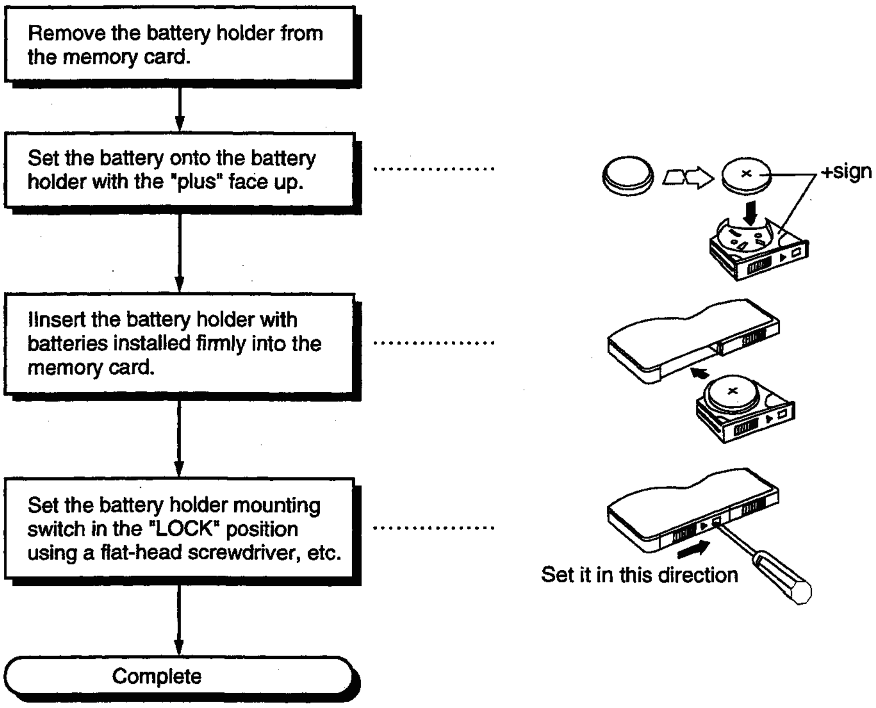
18.6 Installing Batteries (CPU Module and Memory Card Batteries)

(1) Since the CPU module battery is shipped with its battery connector disconnected, connect the connector according to the procedure indicated below.





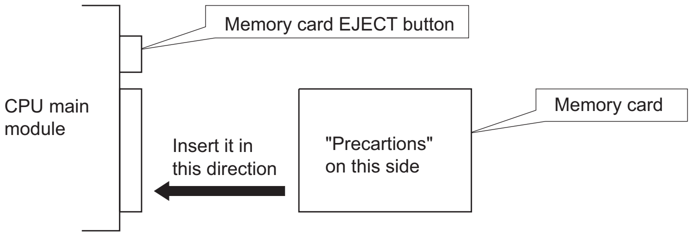
(2) Since the memory card battery is removed from the battery holder before shipping, set it in the battery holder before use of the RAM.



POINT
Firmly push the battery connector all the way in to the connector pin.

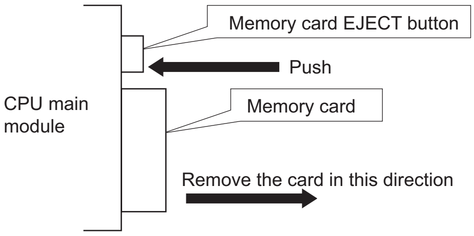
18.7 Installing/Removing A Memory Card

(1) Installing a memory card
When installing a memory card into the CPU module with its power ON, make sure that the orientation of the memory card is correct, then insert it fully until it's edge is flush with the face of the EJECT button.
After installing it, set the memory card in/out switch to "ON".
The memory card is operable after the LED on the memory card in/out switch turns ON.



⚠CAUTION ● Insert the memory card and fully press it to the memory card connector. After that, check for incomplete insertion. Poor electrical contact may cause malfunctions.

(2) Removing the memory card
Before removing the memory card from the CPU module with its power ON, set the memory card in/out switch to "OFF".
Verify that the LED on the switch has gone OFF. Then, press the memory card EJECT button and remove the memory card.



POINT

(1) When a memory card is installed, the scan time will increase by 10ms at maximum. The scan time increases only in 1 scan during which the Q2ASCPU performs mount processing.
(2) If the memory card in/out switch is turned OFF while the system or a program is using the memory card, it may take a while for the LED on the switch to go OFF.
(3) Installing or removing a memory card with the memory card in/out switch set ON while the power is ON will destroy the contents of the memory card.

(3) Memory card remove/insert prohibit flag (special relays SM605)
Instead of operating the memory card in/out switch, turning ON/OFF special relays SM605 (memory card) can be also used as the card remove/insert prohibit flag. Once removal/insertion is prohibited with the remove/install prohibit flag, it is still disabled even if the memory card in/out switch is set to ON.
The relationship between the memory card in/out switch and the memory card remove/insert prohibit flag is shown in the table below.

Memory card remove/insert prohibit flag	Memory card in/out switch	
	ON (Removal/Insertion Prohibited)	OFF (Removal/Insertion Permitted)
ON (removal/insertion prohibited)	Removal/insertion prohibited	Removal/insertion prohibited
OFF (removal/insertion permitted)	Removal/insertion prohibited	Removal/insertion permitted

19 LOADING AND INSTALLATION

This chapter describes the loading and installation procedures and precautions to obtain the maximum system reliability and performance.

19.1 Fail-Safe Circuit Concept

When the programmable controller is powered ON or OFF, improper outputs may be generated temporarily depending on the delay time and start-up time differences between the programmable controller power supply and the external power supply for the control target (especially, DC).

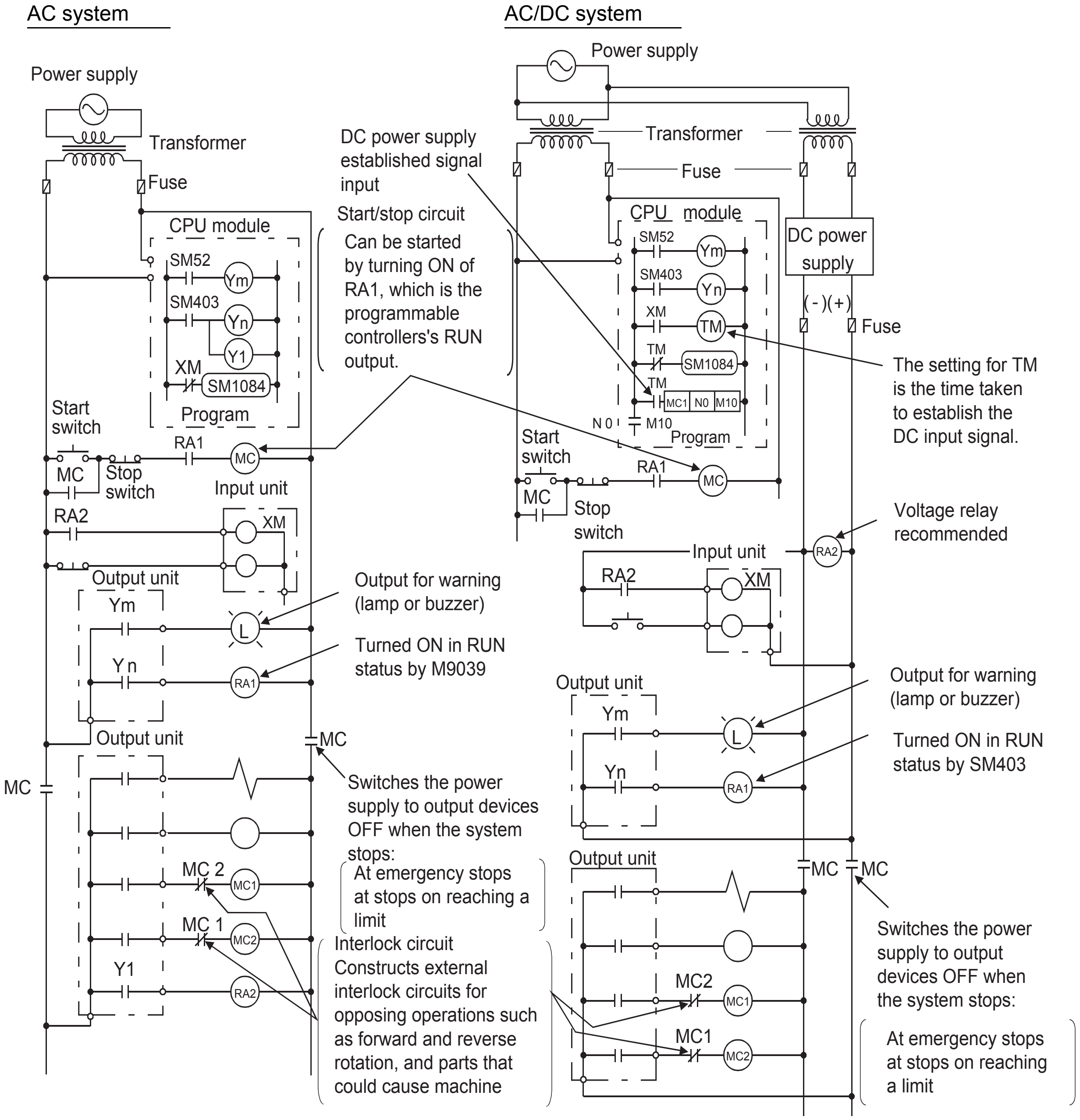
For example, if the external power supply for a DC output module is powered ON and then the programmable controller is powered ON, the DC output module may generate incorrect outputs temporarily upon the programmable controller power-ON. To prevent this, it is required to build a circuit by which the programmable controller is powered on first.

Also, an external power failure or programmable controller failure may lead to erroneous operation.

In order to eliminate the possibility of an system error and to ensure fail-safe operation, create a circuit (emergency stop circuit, protection circuit, interlock circuit, etc.) outside the programmable controller for the parts whose faulty operation could cause mechanical damage and/or accidents.

A system design circuit example based on the above is provided later.

(1) System design circuit example



AC system

DC power supply established signal input

Start/stop circuit
Can be started by turning ON of RA1, which is the programmable controllers's RUN output.

Output for warning (lamp or buzzer)

Turned ON in RUN status by M9039

Switches the power supply to output devices OFF when the system stops:
At emergency stops at stops on reaching a limit

Interlock circuit
Constructs external interlock circuits for opposing operations such as forward and reverse rotation, and parts that could cause machine

AC/DC system

The setting for TM is the time taken to establish the DC input signal.

Voltage relay recommended

Output for warning (lamp or buzzer)

Turned ON in RUN status by SM403

Switches the power supply to output devices OFF when the system stops:
At emergency stops at stops on reaching a limit

The procedures used to switch on the power supply are indicated below.

AC system

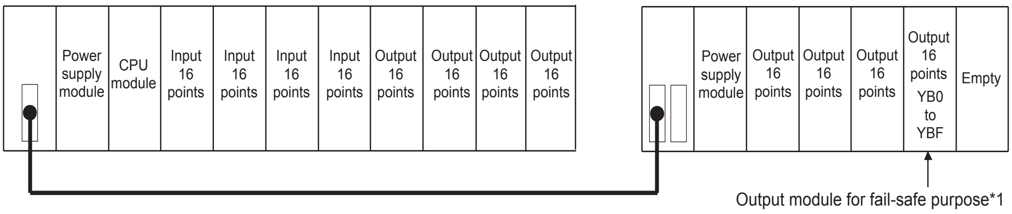
1) Switch the power supply ON.
2) Set the CPU module to RUN.
3) Switch the start switch ON.
4) The output devices are driven in accordance with the program when the magnetic contactor (MC) comes ON.

AC/DC system

1) Switch the power supply ON.
2) Set the CPU module to RUN.
3) Switch RA2 ON when the DC power supply starts.
4) Set the timer (TM) to "ON" upon 100% establishment of DC power supply.
   (The set value for TM shall be the period from turning "ON"RA2 to 100% establishment of DC power supply. Set 0.5 seconds for it.)
5) Switch the start switch ON.
6) The output devices are driven in accordance with the program when the magnetic contactor (MC) comes ON.
   (When a voltage relay is used for RA2, the timer in the program (TM) is not necessary.)

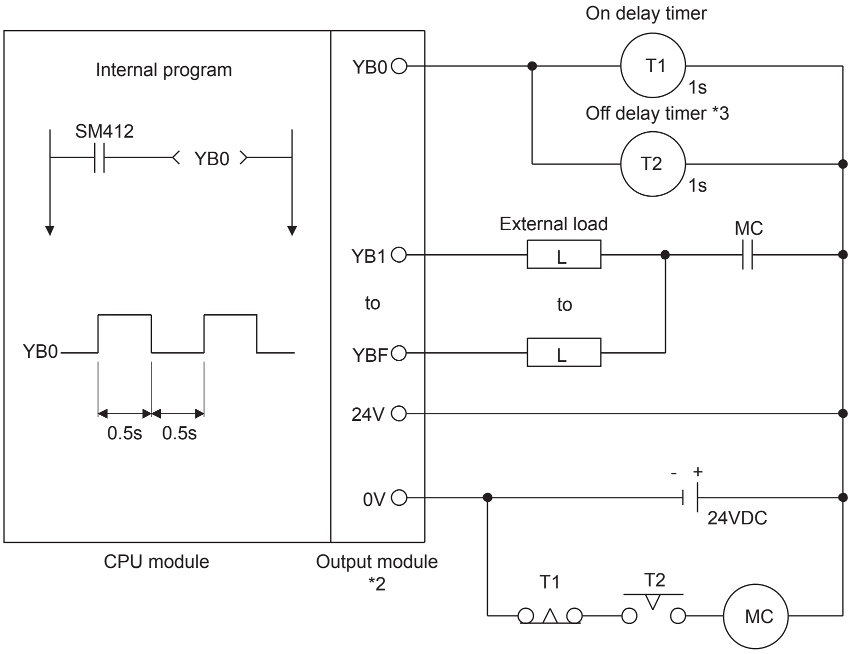
(2) Fail-safe measures for programmable controller failure
While failure of a CPU module and its memory can be detected by the self-diagnostics function, an error occurred in I/O control area may not be detected by the CPU module.
In such a case, depending on the condition of the failure, all device points could turn ON or OFF resulting in a situation where normal operations of the control target and safety cannot be ensured.
Though Mitsubishi programmable controllers are manufactured under strict quality control, create a fail-safe circuit outside the programmable controller to prevent mechanical damage and accidents in the case of a programmable controller failure occurred due to any cause.
Examples of a system and its fail-safe circuitry are described below:

＜System example＞



*1 The output module for fail safe purpose should be mounted on the last slot of the system.(YB0 to YBF in the above system.)

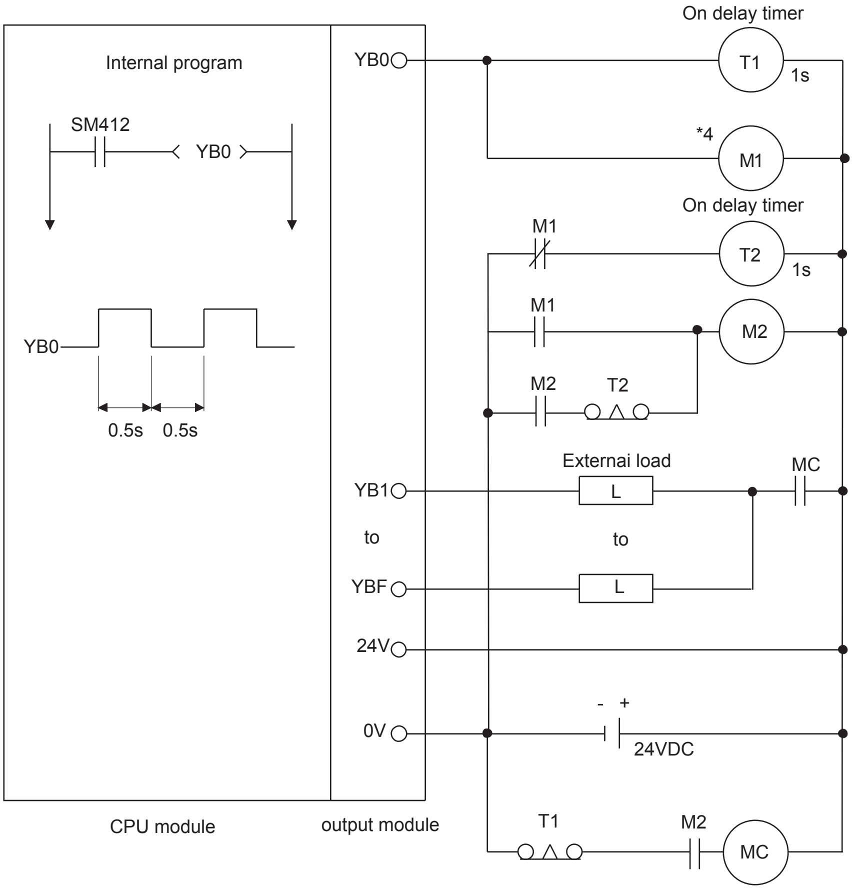
＜Fail safe circuit example＞



*2 Since YB0 turns ON and OFF alternatively at 0.5 second intervals, use a contactless output module (a transistor is used in the above example).

*3 If an off delay timer (especially miniature timer) is not available, construct a fail safe circuit using an on delay timer shown on the next page.

When constructing a fail safe circuit using on delay timers only



*4 Use a solid state relay for the M1 relay.

19.2 Installation Environment

Avoid the following environment when installing a programmable controller system:

(1) The ambient temperature may fall outside the range of 0 to 55°C .

(2) The ambient humidity may fall outside the range of 10 to 90%RH.

(3) Condensation may occur due to drastic changes in temperature.

(4) Corrosive gas or flammable gas exists.

(5) A lot of conductive powdery substance such as dust or iron powder, oil mist, salt, or organic solvent exists.

(6) A location exposed to direct sunlight.

(7) Strong electric or magnetic fields may be generated.

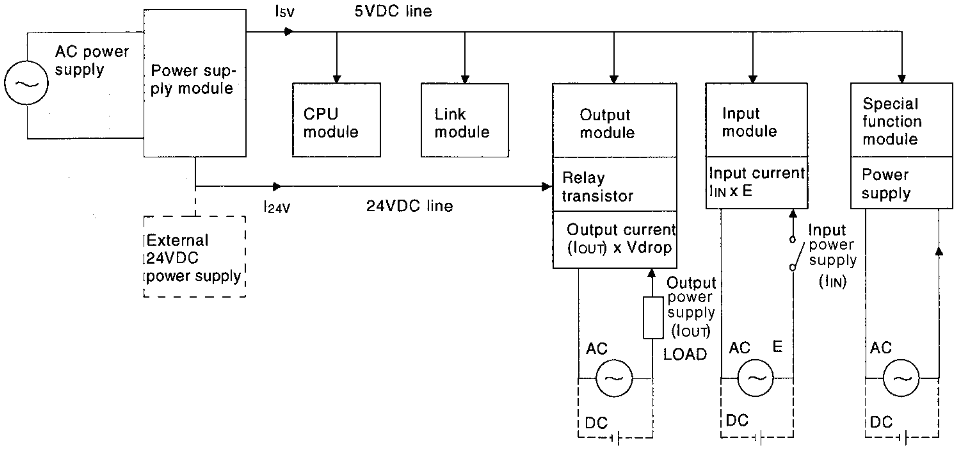
(8) Vibrations and shocks are transmitted directly to the system.

19.3 Calculation of Heat Generated by the programmable controller

The operating ambient temperature in the panel where the programmable controller is stored must be kept 55°C or less.For heat dissipation design of the panel, it is necessary to know the average power consumption (heat generation) of the devices and machinery stored inside.In this section, a method to obtain the average power consumption of the programmable controller system is explained.
Calculate the temperature rise inside the panel from the power consumption.

Average Power Consumption

The power consuming parts of the programmable controller may be roughly classified into the following blocks:



(1) Power consumption by power supply module
The power conversion efficiency of the power supply module is about 70%, and 30% is consumed as heat generated, thus, 3/7 of the output power is the power consumption.Therefore, the calculation formula is:

$$W_{pw}=\frac{3}{7}\{(I_{5V}\times 5)+(I_{15V}\times 15)+(I_{24V}\times 24)\}\ (W)$$

I_{5V} : Current consumption of 5VDC logic ladder circuit of each module

I_{15V} : Current consumption of 15VDC external power supply part of special function module

I_{24V} : Average current consumption of 24VDC power supply for output module's internal consumption
(Current consumption equivalent to the points simultaneously ON)
...... Not applicable to a system where 24VDC is supplied externally and a power supply module with no 24VDC output is used.

(2) Total power consumption of 5VDC logic circuits of modules
The 5VDC output circuit power of the power supply module is regarded as the power consumption of each module.

$$W_{5V}=I_{5V}\times 5\ (W)$$

(3) Total 24VDC average power consumption of the output module (power consumption equivalent to the points simultaneously ON)
The average 24VDC output circuit power of the power supply module is regarded as the total power consumption of each module.

$W_{24V}=I_{24V} \times 24$ (W)

(4) Average power consumption due to output voltage drop of the output modules (power consumption equivalent to the points simultaneously ON)

$W_{OUT}=I_{OUT} \times Vdrop \times$ Output points $\times$ that are simultaneously ON (W)

I_{OUT} : Output current (actual operating current) (A)
Vdrop : Voltage dropped across each output load (V)

(5) Average input power consumption of the input modules (power consumption equivalent to the points simultaneously ON)

$W_{IN}=I_{IN} \times E \times$ Input points $\times$ that are simultaneously ON (W)

I_{IN} : Input current (effective value for AC) (A)
E : Input voltage (actual operating voltage) (V)

(6) Power consumption of the external power supply part of the special function module

$W_S=I_{+15V} \times 15+I_{-15V} \times 15+I_{24V} \times 24$ (W)
The total of the power consumption values obtained for each block is power consumption of the entire programmable controller system.

$W=W_{pw}+W_{5V}+W_{24V}+W_{OUT}+W_{IN}+W_S$ (W)

Using this value (W), calculate the amount of heat generation and temperature rise inside the panel.

The calculation formula to obtain the temperature rise inside a panel is shown as:

$$T=\frac{W}{UA} \, [°C]$$

W : Power consumption of the entire programmable controller system (the value obtained above)
A : Surface area inside the panel (m^2)
U : When temperature inside panel is kept constant by a fan, etc. 6
When air inside panel is not circulated 4

POINT

If the temperature inside the panel can exceed the specified range, it is recommended to install a heat exchanger to the panel to lower the inside temperature.
If a ordinary ventilation fan is used, it sucks dust together with the outside air and it may affect the performance of the programmable controller.

19.4 Installing the Base Units

Precautions on installation of the main base unit and extension base unit are described here.

19.4.1 Installation precautions

Precautions for installing a programmable controller to a panel, etc. are explained below.

(1) To improve the ventilation and to facilitate the exchange of the module, provide at least 30mm (1.18in.) of distance between the top part of the module and any structure or part.
However, when A52B, A55B, A58B, A62B, A65B or A68B extension base unit is used, provide at least 80mm (3.15in.) of distance between the top of the unit and any structural part.

(2) Do not install the programmable controller vertically or horizontally, because it may affect the ventilation.

(3) If the base unit is installed to the surface which is not flat or is distorted, an excessive force is applied to the printed-circuit board and it may cause a fault. Be sure to install it to a flat surface.

(4) Avoid sharing the same panel with any source of vibration such as a large-sized magnetic contactor or no-fuse breaker, and install to a separate panel or away from such devices.

(5) Provide wiring ducts as necessary.
However, when the clearance from the top or bottom of the programmable controller is less than that in Fig. 19.1 and Fig. 19.2, pay attention to the following:
   (a) When installing to the top of programmable controller, to improve the ventilation, keep the height of the duct to 50mm (1.97in.) or below.
   In addition, the distance from the top of the programmable controller should be sufficient for tightening and loosening works for the mounting screws on the top of the module.
   The module cannot be replaced if the screws on the top of the module cannot be loosened or tightened.
   (b) When placing a duct under the programmable controller, take into account the use of optical fiber cables or coaxial cables as well as the minimum bending radius of the cables.

(6) If any device is installed in front of the programmable controller (i.e. installed in the back of the door), position it to secure at least 100mm (3.94inch) of distance to avoid the effects of radiated noise and heat.
Also, place the base unit at least 50mm (1.97inch) away from any other equipment on the right or left.

(7) When installing the base unit to DIN rail in an environment with large vibration, use a vibration-proofing bracket (A1S-PLT-D). Mounting the vibration-proofing bracket (A1S-PLT-D) enhances the resistance to vibration.
Depending on the environment to set up the base unit, it is also recommended to fix the base unit to the control panel directly.

19.4.2 Installation

Installation location of the main base unit and the extension base unit is shown below.

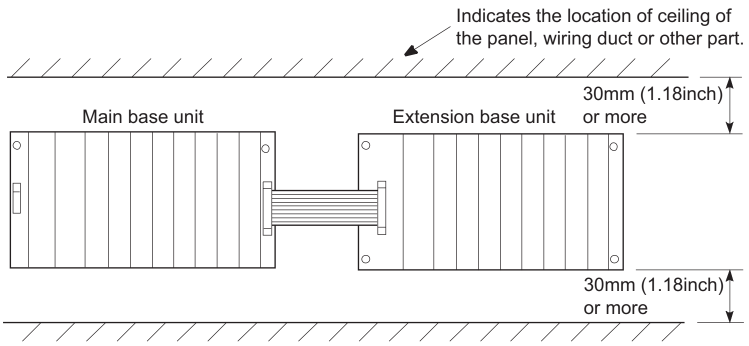


Fig19.1 Parallel Installation

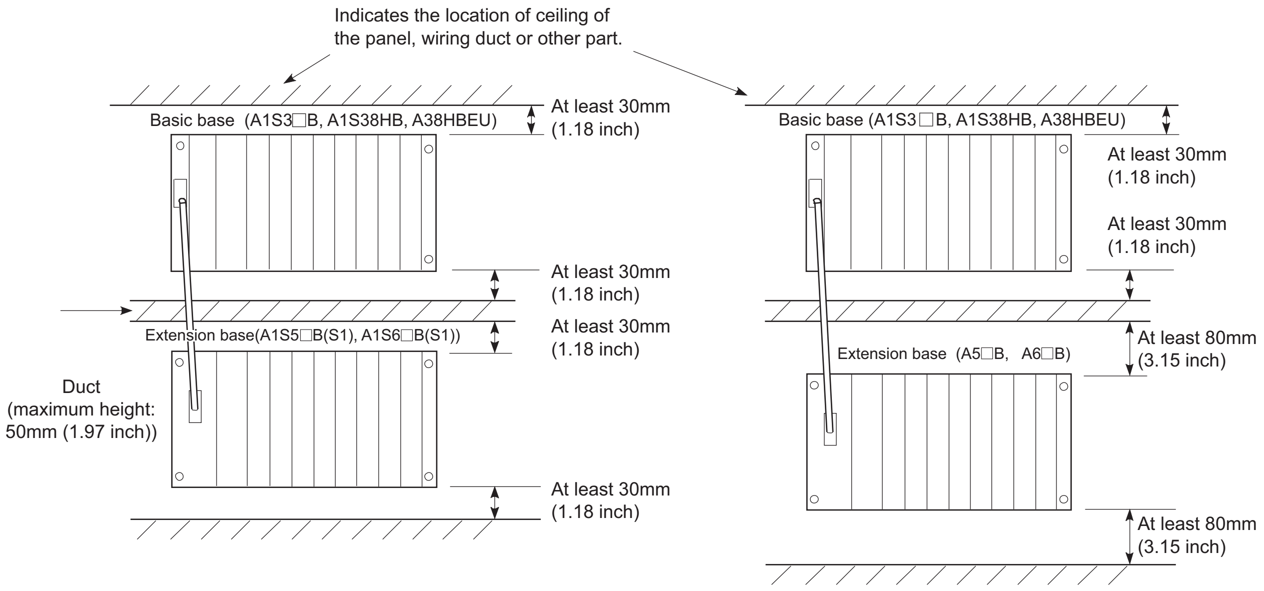


Fig. 19.2 Series Installation

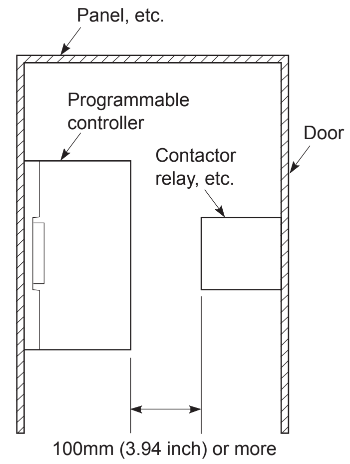


Fig. 19.3 Distance from Front Device

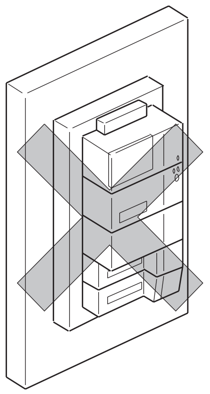


Fig. 19.4 Vertical Mounting (not allowed)

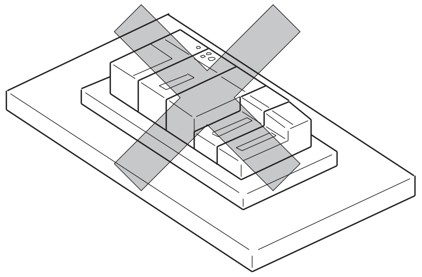
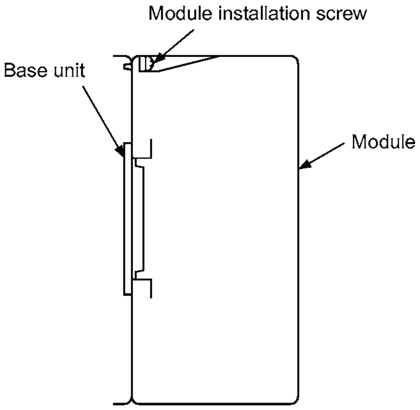


Fig. 19.5 Horizontal Mounting (not allowed)

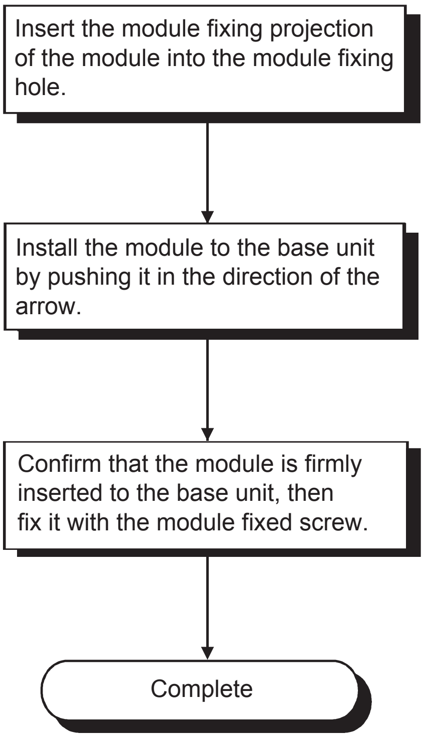
19.5 Installation and Removal of Modules

This section explains how to install or remove the power supply module, CPU module, I/O module and special function module, etc. to or from the base unit.



(1) Module installation

The procedure for mounting a module is described below.

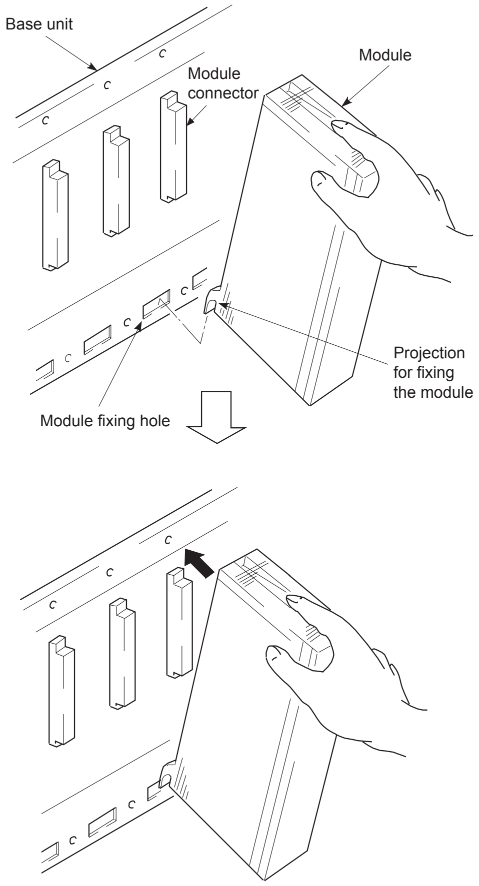


Insert the module fixing projection of the module into the module fixing hole.

Install the module to the base unit by pushing it in the direction of the arrow.

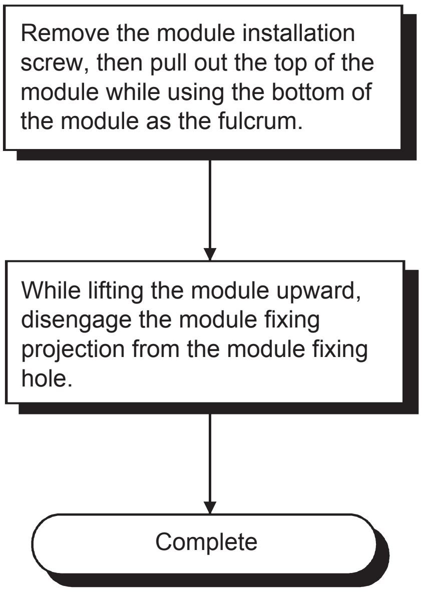
Confirm that the module is firmly inserted to the base unit, then fix it with the module fixed screw.

Complete



(2) Removing a module

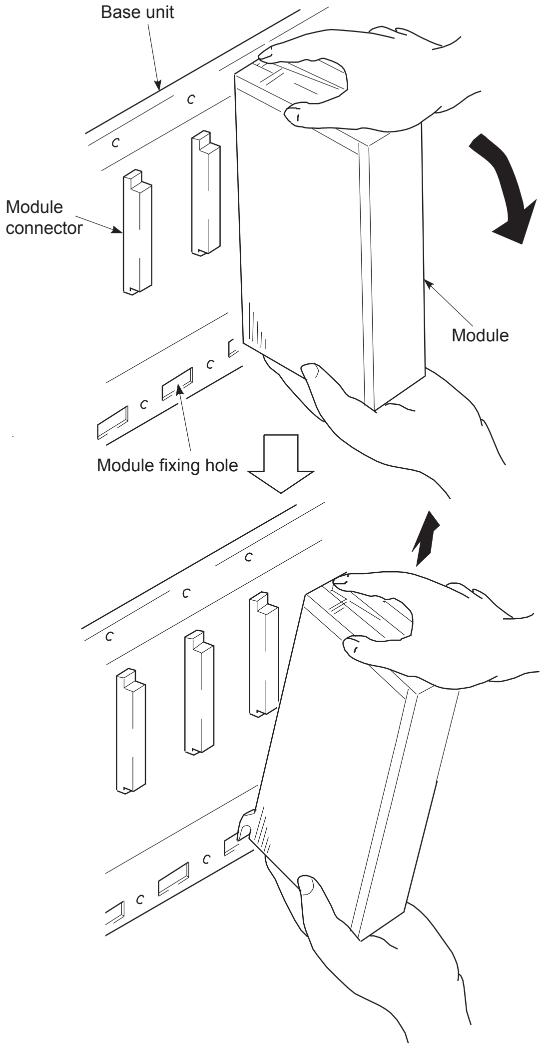
The procedure for removing a module is explained here.



Remove the module installation screw, then pull out the top of the module while using the bottom of the module as the fulcrum.

While lifting the module upward, disengage the module fixing projection from the module fixing hole.

Complete

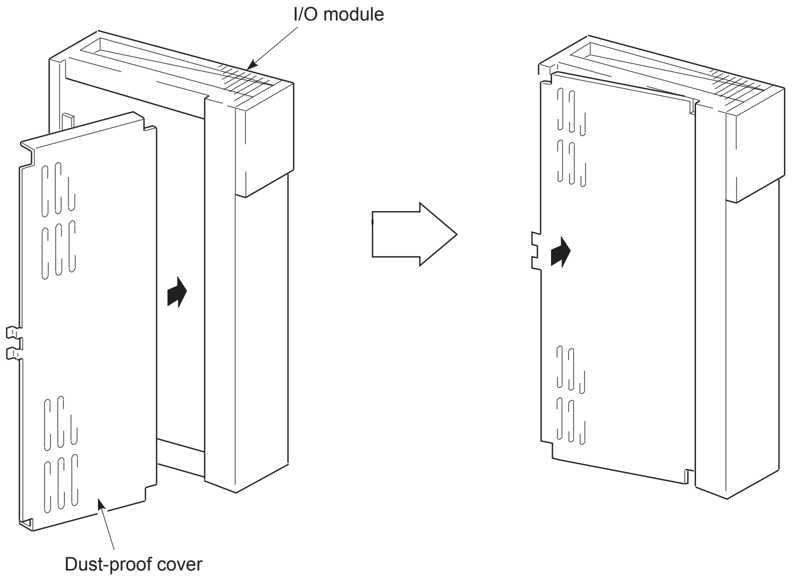


POINT
To dismount the module, be sure to disengage the hook from the module fixing hole and then remove the module fixing projection from the module fixing hole.If the module is forcibly removed, the hook or module fixing projection will be damaged.

19.6 Installation and Removal of the Dustproof Cover

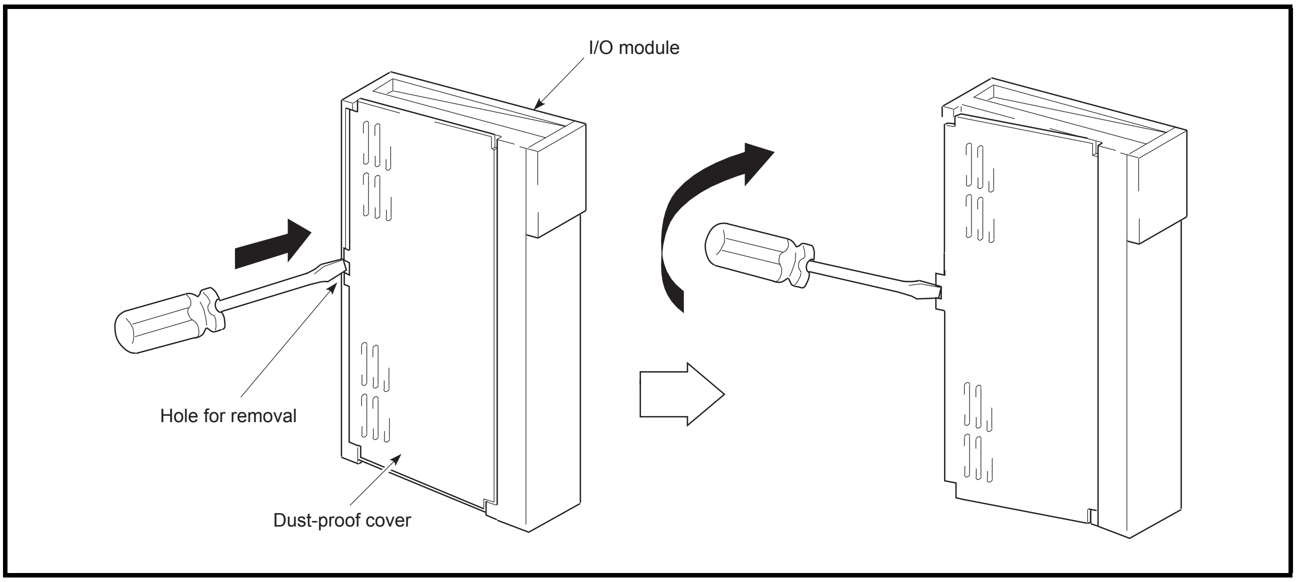
When using the A1S52B(S1), A1S55B(S1) or A1S58B(S1), attach the dustproof cover supplied with the extention base unit to the I/O module on the left end.If no dustproof cover is attached, foreign matter will enter the I/O module, causing a failure.
Procedures for installing and removing the dustproof cover are described below.

(1) Installation



Insert the dustproof cover into the connector- or terminal-side groove of the I/O module first as shown in the figure, and then push the dustproof cover.

(2) Removal



To remove the dustproof cover from the I/O module, insert the tip of a flat-head screwdriver into the hole as shown in the figure, then pry the tab of the cover out from the hole using the screwdriver.

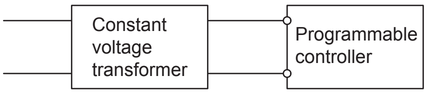
19.7 Wiring

19.7.1 Wiring instructions

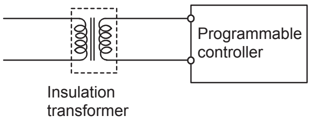
Instructions for wiring of power cables or I/O cables are given in this section.

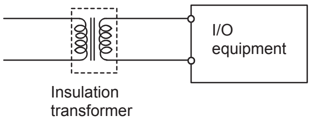
(1) Wiring the power supply

(a) When voltage fluctuates outside the specified value range, connect a constant-voltage transformer.



(b) Use a power supply which generates minimal noise between wires and between the programmable controller and ground.
If excessive noise is generated, connect an isolating transformer.





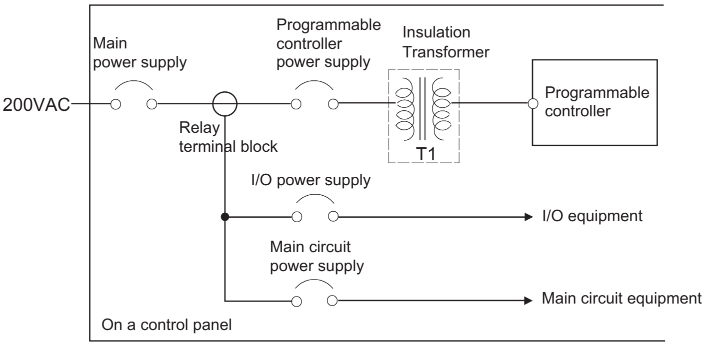
(c) When using a power transformer or an isolating transformer to reduce the voltage from 200VAC to 100VAC, its capacity must be equal to or greater than the corresponding value shown in the following table.

Power Supply Module	Transformer Capacity
A1S61PN	110VA× n
A1S62PN	110VA× n

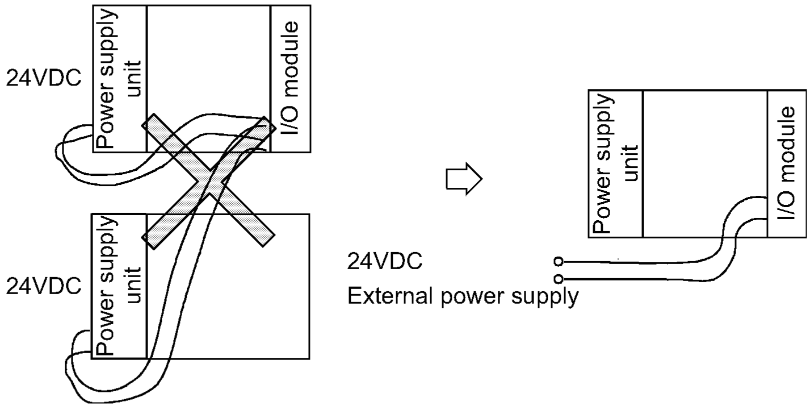
n: Stands for the number of power supply modules.

(d) Separate the programmable controller's power supply line from the lines for I/O devices and power devices as shown below.
When there is much noise, connect an isolating transformer.

(e) Taking rated current or inrush current into consideration when wiring the power supply, be sure to connect a breaker or an external fuse that have proper blown and detection.
When using a single programmable controller, a 10A breaker or an external fuse are recommended for wiring protection.



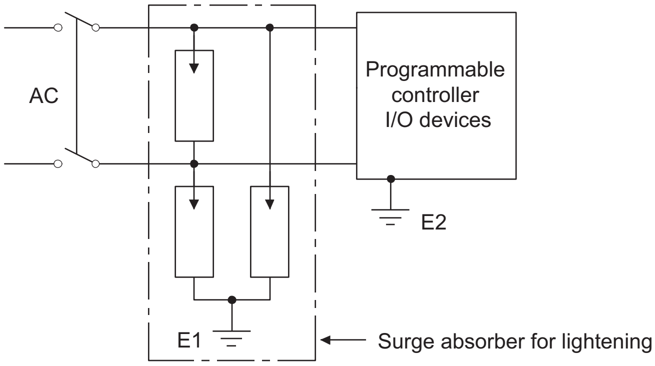
(f) Note on using 24VDC output of the A1S62PN power supply module.
If the 24VDC output power from a single power supply module is insufficient, supply it from the external 24VDC power supply.



(g) Twist the 100VAC, 200VAC or 24VDC wires as tightly as possible, and use the minimum length to make connection between modules.
Also, use a thick wire (max. 2 mm^2) to minimize voltage drop.

(h) Do not install 100VAC and 24VDC wires together with main circuit wires (high voltage and large current) or I/O signal lines (including common line). Provide a distance of 100mm (3.94inch) or more between them if possible.

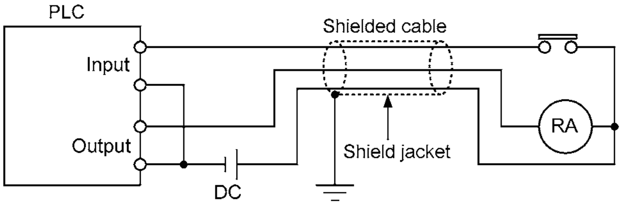
(i) As a measure against lightning surges, connect a lightning surge absorber as shown below.



POINT

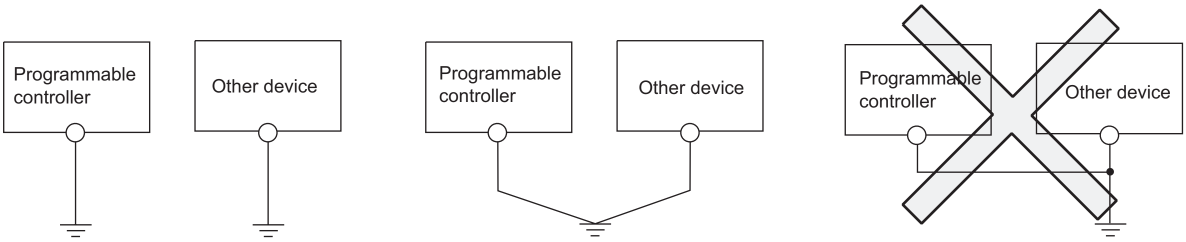
(1) Ground the lightning surge absorber (E1) and the programmable controller (E2) separately from each other.
(2) Select a lightning surge absorber whose voltage does not exceed the maximum allowable circuit voltage even when line voltage reaches the maximum.

(2) Wiring I/O equipment
   - (a) The applicable wire size for a terminal block connector is 0.75 to 2mm^2. It is recommended to use wire of 0.75mm^2 for easy use.
   - (b) Run the input line and output line away from each other.
   - (c) Separate the I/O signal lines (including common line) at least 100mm (3.94inch) away from the main circuit line carrying high voltage and large current.
   - (d) If it is not possible, use a batch shielding cable and ground it on the programmable controller side.
     However, ground it on the opposite side in some cases.



   - (e) When ducts are used for wiring, securely ground them.
   - (f) Separate the 24VDC I/O cables from the 100VAC and 200VAC cables.
   - (g) In a long distance wiring of 200m (656.2ft.) or longer, leak current due to capacitance may cause failure.
   - (h) As protective measures against lightning surges, separate the AC wiring from the DC wiring and connect a lightning surge absorber as shown in (1) (i).
     Failure to do so increases the risk of I/O equipment failure due to lightning.

(3) Grounding
   - (a) Carry out the independent grounding if possible.
   - (b)

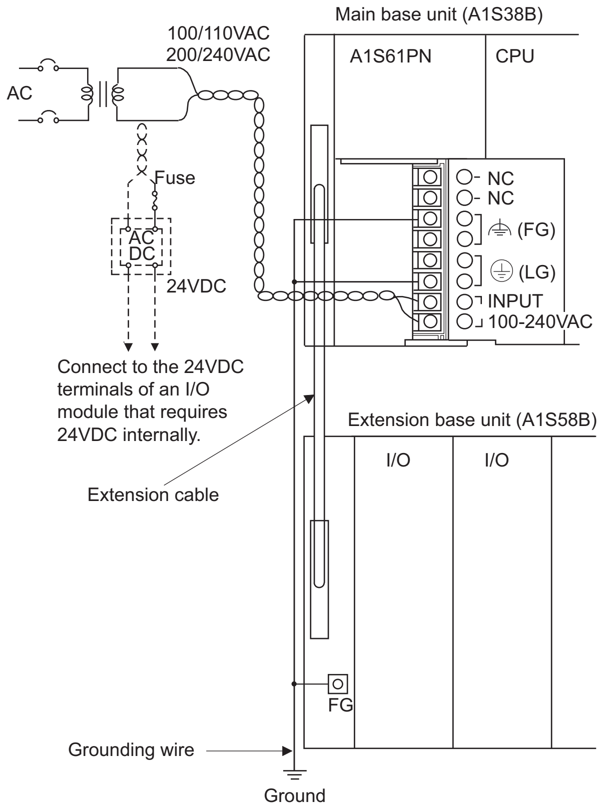


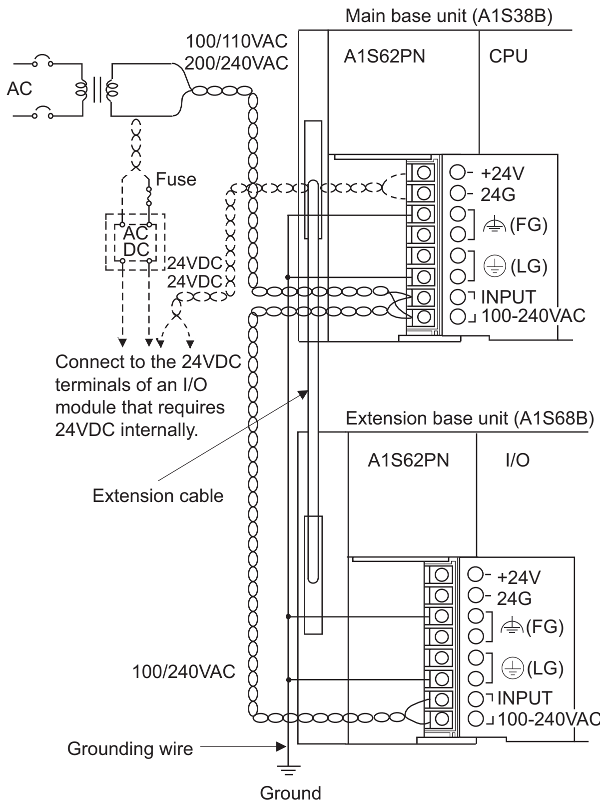
(1) Independent grounding .... Best (2) Shared grounding .... Good (3) Common grounding .... Not allowed

   - (c) Use the cable of 2mm^2 (0.0031in.^{2}) or more for grounding.
     Set the grounding point closer to the programmable controller to make the grounding cable short as possible.
   - (d) If any malfunction occurs due to grounding, disconnect either or both of the LG and FG terminals of the base unit from the ground.

19.7.2 Wiring to module terminals

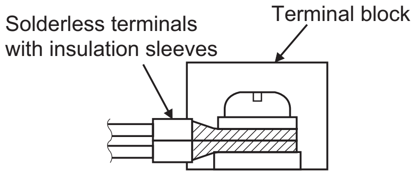
This section provides an example for wiring power cables and ground wires to the main and extension bases.





POINT

(1) Use the thickest possible (max. 2 mm^2 (14 AWG)) wires for the 100/200 VAC and 24 VDC power cables. Be sure to twist these wires starting at the connection terminals. For wiring a terminal block, be sure to use a solderless terminal. To prevent short-circuit due to loosening screws, use the solderless terminals with insulation sleeves of 0.8 mm (0.03 inch) or less thick. The number of the solderless terminals to be connected for one terminal block are limited to 2



(2) When connection is made between the LG and FG terminals, be sure to ground the wire. Failure to observe this instruction after connecting the LG and FG terminals will make the line susceptible to noise. Note that each LG terminal has half the potential of the input voltage; you might get an electric shock if you touch it.

(3) A1S61PN and A1S62PN do not need to be switched as the are 100 to 240VAC wide-range.

19.8 Precautions When Connecting Uninterruptible Power Supply Module (UPS)

When onnecting a programmable controller system to an uninterruptible power supply (UPS), pay attention to the following.

When connecting an uninterruptible power supply (UPS) to the programmable controller system, use an online UPS or line-interactive UPS with a voltage distortion rate of 5% or less.
When connecting a standby UPS, use a Mitsubishi FREQUPS FW-F series UPS (hereinafter FW-F series UPS)[*1]. (Example: FW-F10-0.3K/0.5K)
Do not use any standby UPS other than the FW-F series UPS.

[*1] The FW-F series UPS whose serial number starts with the letter "P" or later, or ends with the letters "HE" is applicable.

SERIAL : Q00000000
Starts with "P" or later

SERIAL : B00000000 HE
Ends with "HE"

20 EMC AND LOW VOLTAGE DIRECTIVES

The products sold in the European countries have been required by law to comply with the EMC and Low Voltage Directives of the EU Directives since 1996 and 1997, respectively. The manufacturers must confirm by self-declaration that their products meet the requirements of these directives, and put the CE mark on the products.

(1) Authorized representative in Europe
Authorized representative in Europe is shown below.
Name: Mitsubishi Electric Europe BV
Address: Gothaer Strase 8, 40880 Ratingen, Germany

20.1 Requirements for Compliance with EMC Directives

The EMC Directives specifies emission and immunity criteria and requires the products to meet both of them, i.e., not to emit excessive electromagnetic interference (emission): to be immune to electromagnetic interference outside (immunity). Guidelines for complying the machinery including MELSEC-QnA series programmable controller with the EMC Directives are provided in Section 20.1.1 to Section 20.1.9 below.
The guidelines are created based on the requirements of the regulations and relevant standards, however, they do not guarantee that the machinery constructed according to them will comply with the Directives. Therefore, manufacturers must finally determine how to make it comply and how it is compliant with the EMC Directives.

20.1.1 EMC Directive related standards

(1) Regulations regarding emission

Standard	Test item	Test description	Value specified in standard
EN61131-2:2007	CISPR16-2-3 Radiated emission [*2]	Radio waves from the product are measured.	• 30M-230MHz QP: 40dBμV/m (10m in measurement range) [*1] • 230M-1000MHz QP: 47dBμV/m (10m in measurement range)
	CISPR16-2-1, CISPR16-1-2 Conducted emission [*2]	Noise from the product to the power line is measured.	• 150k-500kHz QP: 79dB, Mean: 66dB [*1] • 500k-30MHz QP: 73dB, Mean: 60dB

[*1] QP: Quasi-peak value, Mean: Average value

[*2] Programmable controllers are open-type devices (devices designed to be housed inside other equipment) and must be installed inside a conductive control panel. The corresponding tests were conducted with the programmable controller installed inside a control panel.

(2) Regulations regarding immunity

Standard	Test item	Test description	Value specified in standard
EN61131-2:2007	EN61000-4-2 Electrostatic discharge immunity [*1]	Immunity test in which electrostatic is applied to the cabinet of the equipment.	• 8kV Air discharge • 4kV Contact discharge
	EN61000-4-3 Radiated, radio-frequency, electromagnetic field immunity [*1]	Immunity test in which electric fields are irradiated to the product.	80% AM modulation@1kHz • 80M-1000MHz: 10V/m • 1.4G-2.0GHz: 3V/m • 2.0G-2.7GHz: 1V/m
	EN61000-4-4 Electrical fast transient/burst immunity [*1]	Immunity test in which burst noise is applied to the power line and signal line.	• AC/DC main power, I/O power, AC I/O (unshielded): 2kV • DC I/O, analog, communication: 1kV
	EN61000-4-5 Surge immunity [*1]	Immunity test in which lightning surge is applied to the power line and signal line.	• AC power line, AC I/O power, AC I/O (unshielded): 2kV CM, 1kV DM • DC power line, DC I/O power: 0.5kV CM, DM • DC I/O, AC I/O (shielded),analog, communication: 1kV CM
	EN61000-4-6 Immunity to conducted disturbances, induced by radio-frequency fields [*1]	Immunity test in which high frequency noise is applied to the power line and signal line.	0.15M-80MHz, 80% AM modulation@1kHz, 10Vrms
	EN61000-4-8 Power-frequency magnetic field immunity [*1]	Immunity test in which the product is installed in inductive magnetic field.	50Hz/60Hz, 30A/m
	EN61000-4-11 Voltage dips and interruption immunity [*1]	Immunity test in which power supply voltage is momentarily interrupted.	• Apply at 0%, 0.5 cycles and zero-cross point • 0%, 250/300 cycles (50/60Hz) • 40%, 10/12 cycles (50/60Hz) • 70%, 25/30 cycles (50/60Hz)

[*1] Programmable controllers are open-type devices (devices designed to be housed inside other equipment) and must be installed inside a conductive control panel. The corresponding tests were conducted with the programmable controller installed inside a control panel.

20.1.2 Installation inside the control panel

The programmable controller is open equipment and must be installed within a control panel for use.* This is effective not only for ensuring safety but also for shielding electromagnetic noise generated from the programmable controller.

* Each network remote station also needs to be installed inside the control panel. However, waterproof type remote stations can be installed outside the control panel.

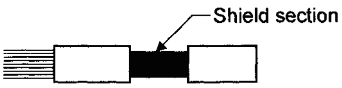
(1) Control panel
  (a) Use a conductive control panel.
  (b) When attaching the control panel's top plate or base plate, mask painting and weld so that good surface contact can be made between the panel and the bolt.
  (c) To ensure an electrical contact with the control panel, mask the bolt areas of the inner plates when painting to allow conductivity over the widest possible area.
  (d) Ground the control panel with a thick wire so that a low impedance can be ensured even at high frequencies.
  (e) Holes made in the control panel must be 10 cm (3.94 inch) diameter or less. If the diameter is more than 10cm (3.94 inch), radio waves can be leaked.
  (f) Lock the control panel so that only those who are trained and have acquiredenough knowledge of electric facilities can open the control panel.

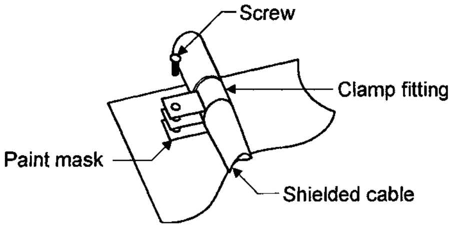
(2) Connection of power cable and ground wires
Handle the power cables and ground wires as described below.
  (a) Provide a grounding point near the power supply module. Ground the power supply module's LG and FG terminals (LG : Line Ground, FG : Frame Ground) with the thickest and shortest wire possible. (The wire length must be 30 cm (11.18 inch) or shorter.) As the LG and FG terminals release the noise generated in the programmable controller to the ground, the lowest possible impedance must be ensured.
  The ground wires also need to be short as they are used to release noise. Because the wire itself carries large noise, short wiring prevents it from acting as an antenna.
  (b) Twist the ground wire led from the grounding point with the power cable. By doing this, noise from the power cable can be released to the ground. If a filter is attached to the power cable, however, this twisting may not be needed.

20.1.3 Cables

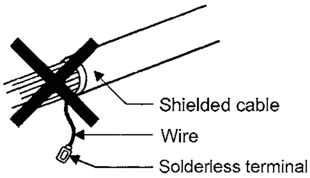
The cables extracted from the control panel contain a high frequency noise component. On the outside of the control panel, therefore, they serve as antennas to emit noise. To prevent noise emission, use shielded cables for the cables which are connected to the I/O modules and intelligent function modules and may be extracted to the outside of the control panel.
The use of a shielded cable also increases noise resistance.
The signal lines (including common line) of the programmable controller, which are connected to I/O modules, intelligent function modules and/or extension cables, have noise durability in the condition of grounding their shields by using the shielded cables. If a shielded cable is not used or not grounded correctly, the noise resistance will not meet the specified requirements.

(1) Grounding of shielded cables
  (a) Shielding must be done close to the control panel. Otherwise, electromagnetic induction from the cable after the grounding point will generate high frequency noise.
  (b) Partly remove the outer sheath of the shielded cable so that it can be contact with the widest possible area of the control panel. A clamp may also be used as shown in the figure below. In this case, cover the control panel's inner surface which will come in contact with the clamp when painting.



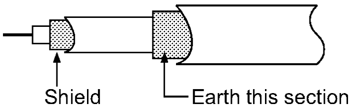


Note) Grounding a shield cable by soldering a wire to the shield section as illustrated below is not recommended. The high frequency impedance will increase and the shield will be ineffective.



(2) MELSECNET (II) and MELSECNET/10 modules

(a) Use double-shielded coaxial cables (MITSUBISHI CABLE INDUSTRIES, LTD.: 5C-2V-CCY) for the MELSECNET modules (such as A1SJ71AR21, A1SJ71QLR21, A1SJ71QBR11) which uses coaxial cables.Noise in the range of 30 MHz or higher in radiated noise can be suppressed by the use of double-shielded coaxial cables. Ground the double-shielded coaxial cable by connecting its outer shield to the ground.



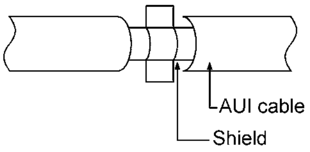
Refer to (1) for the grounding of the shield.

(b) Always attach a ferrite core to the double-shielded coaxial cable connected to the MELSECNET module. In addition, position the ferrite core on each cable near the outlet of the control panel. The ferrite core manufactured by TDK Corporation, ZCAT3035-1330, is recommended.

(3) Ethernet module

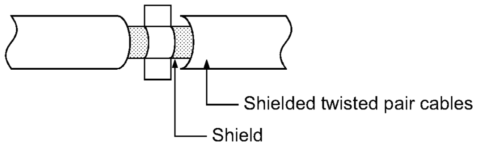
Precautions to be followed when AUI cables*1, twisted pair cables and coaxial cables are used are described below.

(a) Be sure to ground the AUI cables[*1] connected to the 10BASE5 connectors. Because the AUI cable is of the shielded type, as shown in the figure below, partly remove the outer sheath, and ground the exposed shield section to the widest possible surface.



Refer to (1) for the grounding of the shield.

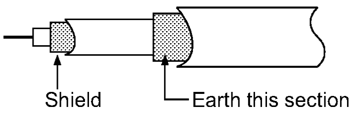
(b) Use shielded twisted pair cables as the twisted pair cables[*1] connected to the 10BASE-T connectors. Partly strip the outer sheath of the shielded twisted pair cable, and ground the exposed shield section to the widest possible area as shown below.



Refer to (1) for the grounding of the shield.

*1 Make sure to attach a ferrite core to the cable.
The ferrite core manufactured by TDK Corporation, ZCAT2032-0930, is recommended.

(c) Always use double-shielded coaxial cables as the coaxial cables[*2] connected to the 10BASE2 connectors. Ground the double-shielded coaxial cable by connecting its outer shield to the ground.



Refer to (1) for the grounding of the shield.

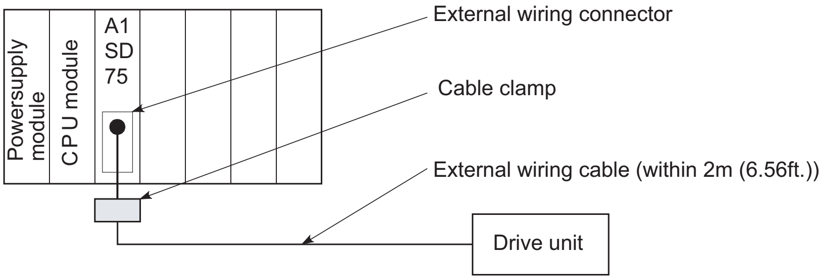
[*2] Make sure to attach a ferrite core to the cable.
The ferrite core manufactured by TDK Corporation, ZCAT3035-1330, is recommended.

Ethernet is the registered trademark of XEROX, Co.,LTD

(4) I/O signal cables and other communication cables
Always ground the I/O signal lines (including common line) and other communication cables (RS-232, RS-422, etc.) in the same manner as described in (1) if they are brought out of the control panel.

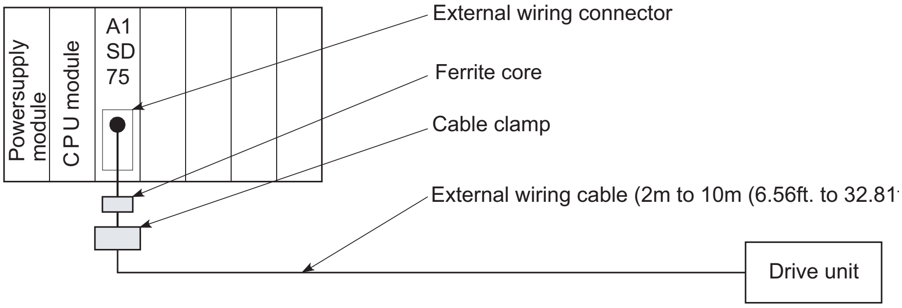
(5) Positioning modules
Precautions for configuring machinery compliant with the EMC Directives using the A1SD75P□-S3 are described below.

(a) When using a cable of 2m (6.56ft.) or less
- Ground the shield section of the external wiring cable with a cable clamp. (Ground the shield at the closest location to the A1SD75□-S3 external wiring connector.)
- Connect the external wiring cable to a drive unit or an external device in the shortest distance.
- Install the drive unit in the same panel.



(b) When connecting a cable longer than 2m (6.56ft.), but not exceeding 10m (32.81ft.)

- Ground the shield section of the external wiring cable with a cable clamp. (Ground the shield at the closest location to the A1SD75☐-S3 external wiring connector.)
- Install a ferrite core.
- Connect the external wiring cable to a drive unit or an external device in the shortest distance.

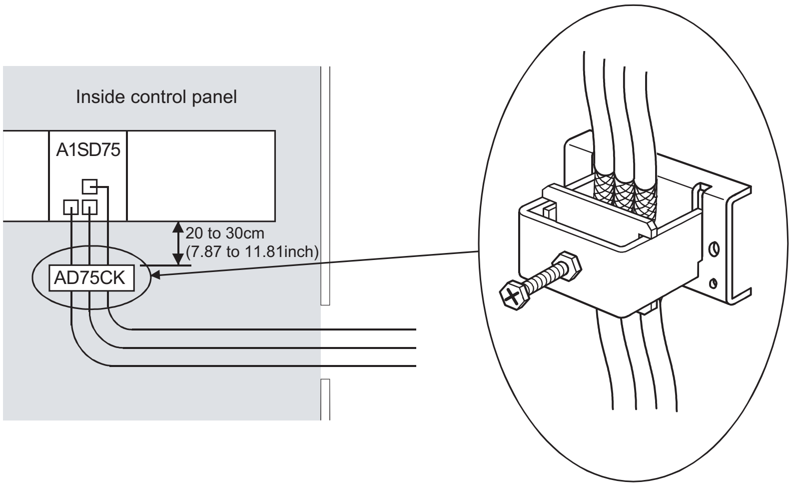


(c) Models and required quantities of the ferrite core and cable clamp

- Cable clamp
  Model: AD75CK (Manufactured by Mitsubishi Electric)
- Ferrite core
  Model: ZCAT3035-1330 (TDK ferrite core)
  Contact: TDK Corporation
- Required quantity

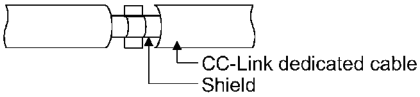
Cable length	Optional part	Required quantity		
		1 axis	2 axes	3 axes
Within 2m (6.56ft.)	AD75CK	1	1	1
2m (6.56ft.) to 10m (32.81ft.)	AD75CK	1	1	1
	ZCAT3035-1330	1	2	3

(d) Cable clamp position



(6) CC-Link module

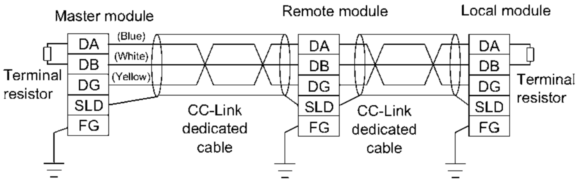
(a) Be sure to ground the shield of the cable that is connected to a CC-Link module close to the exit of the control panel or to any of CC-Link stations within 30cm (11.81inch) from the module or stations.
The CC-Link dedicated cables are shielded cables. As shown in the illustration below, remove a part of the outer sheath and ground it to the widest possible area.



(b) Always use the specified CC-Link dedicated cable.

(c) Connect the CC-Link module and each CC-Link station to the FG line inside the control panel with the FG terminals as shown below.

[Simplified diagram]

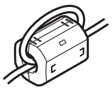


(d) Each power line connecting to the external power supply terminal or module power supply terminal must be 30m (98.43 ft) or less.

(e) Install a noise filter to the external power supply. Use a noise filter with an attenuation characteristic equivalent to that of the MA1206 (TDK-Lambda Corporation). Note that a noise filter is not required when the module is used in Zone A defined in EN61131-2.

(f) Keep the length of signal cables connected to the analog input terminals of the following modules to 30m or less.
Wire cables connected to the external power supply and module power supply terminal in the control panel where the module is installed.
- AJ65BT-64RD3
- AJ65BT-64RD4
- AJ65BT-68TD

(g) For the cable connected to the power supply terminal of the AJ65SBT-RPS or AJ65BT-68TD, attach a ferrite core with an attenuation characteristic equivalent to that of the ZCAT3035-1330 from TDK Corporation. Twist the cable around the ferrite core by one as shown below.



(7) CC-Link/LT module
To supply the CL2DA2-B and CL2AD4-B with 24VDC power using the CL1PAD1, keep the length of the power cable from the CL1PAD1 to the 24VDC power supply to 30m or less.

(8) Measures against static electricity
When using an insulation displacement connector without connector cover, a connected cable for the connector is thin in applicable wire size and coating.
Therefore, note that the module may cause an electric discharge failure.
As measures against the failure, using pressure-displacement type connector whose applicable wire size is thick or soldering type connector is recommended.

20.1.4 Power supply module

The precautions required for each power supply module are described below.Always observe the items noted as precautions.

Model Name	Precautions[*2]
A1S61PN, A1S62PN	Make sure to short the LG and FG terminals with a cable of 6 to 7cm and ground the cable.
A1S63P[*1]	Use a CE-compliant 24VDC power supply in the control panel.
A1SJHCPU(S8)	Make sure to short and ground the LG and FG terminals.[*2]

*1 Filter attachment to the power cable is not required for the A1S63P product with the version (F) and later. However, use the 24VDC panel power equipment that conforms to the CE.

*2 Make sure to attach two ferrite cores to the power line.
Attach them as close to the power supply module as possible.
Use a ferrite core whose damping characteristic is equivalent to that of the RFC-H13 produced by Kitagawa Industries Company, LTD.

20.1.5 Base unit

The following table lists the base units that can be used for compliance with the EMC directives.

Type	Model Name	Applicability
Main Base Unit	A1S38HBEU	Applicable
	A1S3☐B, A1S38HB	N/A
Extension Base Unit	A1S5☐B, A1S6☐B	Applicable

20.1.6 Ferrite core

Use of ferrite cores is effective in reducing conduction noise in the band of about 10MHz and radiated noise of 30 to 100MHz.
It is recommended to attach ferrite cores when the shield of the shielded cable coming out of the control panel does not work effectively, or when emission of the conduction noise from the power line has to be suppressed.
We tested using ferrite cores from TDK Corporation, ZCAT3035-1330 and ZCAT2032-0930, and RFC-H13 from Kitagawa Industries Company, LTD.
Make sure to attach a ferrite core to a cable right before the cable is pulled out of the control panel. If attached to an improper position, the ferrite core does not work effectively.

- Ferrite core
  Type: ZCAT3035-1330, ZCAT2032-0930
  Contact: TDK Corporation
  Type: RFC-H13
  Contact: Kitagawa Industries Company, LTD

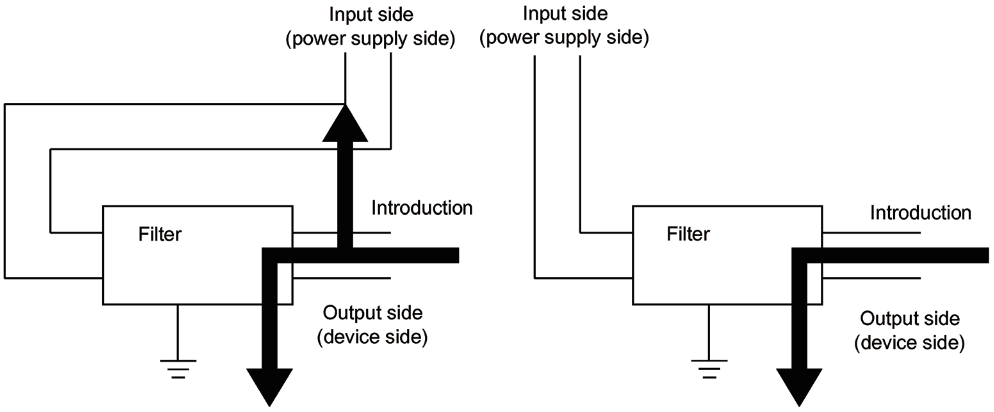
20.1.7 Noise filter (power supply line filter)

A noise filter is effective for suppressing conduction noise. It is not required to attach a noise filter to the power supply line except for some models, however, attaching it can suppress more noise. (The noise filter has the effect on reducing conduction noise of 10MHz or less.). Use any of the following noise filters (double μ type filters) or equivalent.

Model name	FN343-3/01	FN660-6/06	ZHC2203-11
Manufacturer	SCHAFFNER	SCHAFFNER	TDK
Rated current	3A	6A	3A
Rated voltage	250V		

The precautions required when installing a noise filter are described below.

(1) Do not bundle the wires on the input side and output side of the noise filter. When bundled, the output side noise will be induced into the input side wires from which noise has been filtered out.



(a) The noise will be induced when the input and output wires are installed together.

(b) Separate the input wires from the output wires.

(2) Ground the noise filter ground terminal to the control panel with the shortest wire possible (approx. 10cm (3.94in.)).

20.1.8 Power line for external power supply terminal

(1) Use a CE-marked AC/DC power supply for an external power supply of the modules, and the power cable length needs to be less than 30m (98.43 ft.).[*1]

[*1] The power cable length for the A1SJ71QE71N-B5 needs to be less than 3m (9.84 ft.).

(2) Use a CE-marked AC/DC power supply for an external power supply of the A1SJ71QLP21S.

(3) Install noise filters to external supply power terminals of the I/O module and the modules below.
Use noise filters whose damping characteristic is equivalent to that of the MA1206 produced by TDK Lambda Corporation.

- Analog-digital converter module
- Digital-analog converter module
- Analog I/O module
- Temperature input module
- Temperature control module
- Pulse input module
- High-speed counter module
- Positioning module

20.1.9 Installation environment of the CC-Link/LT module and the AS-i module

(1) CC-Link/LT module
Use the module under the environment of Zone A[*1].
For the categories of the following products, refer to the manual came with each product.

- CL1Y4-R1B1
- CL1Y4-R1B2
- CL1XY4-DR1B2
- CL1XY8-DR1B2
- CL1PSU-2A

(2) AS-i module
Use the module under the environment of Zone A[*1].

[*1] Zone defines categories according to industrial environment, specified in the EMC and Low Voltage Directives, EN61131-2.
Zone C: Factory mains (isolated from public mains by dedicated transformer)
Zone B: Dedicated power distribution, secondary surge protection (rated voltage:300V or less)
Zone A: Local power distribution, protected from dedicated power distribution by AC/DC converter and insulation transformer (rated voltage: 120V or less)

20.2 Requirements for Compliance with Low Voltage Directives

The Low Voltage Directives apply to the electrical equipment operating from 50 to 1000VAC or 75 to 1500VDC; the manufacturer must ensure the safety of the equipment. Sections 20.2.1 to Section 20.2.7 provide precautions on installation and wiring of the MELSEC-QnA series programmable controller to conform to The Low Voltage Directives. The descriptions are made based on the requirements and standards of the latest regulation. However, they do not guarantee that any machinery produced according to the contents of this manual is compliant with the above directives. Therefore, manufacturers must finally determine how to make it comply it and how it is compliant with the low voltage directives.

20.2.1 Standard applied for MELSEC-QnA series programmable controller

The standard applied for MELSEC-QnA series programmable controller is EN61010-1 Safety of devices used in measurement, control, or laboratories.

For the modules which operate with the rated voltage of 50 VAC/75 VDC or above, we have developed new models that conform to the above standard.
For the modules which operate with the rated voltage less than 50 VAC or 75 VDC, conventional models can be used, because the low voltage directives do not apply to them.

20.2.2 Precautions when using the QnA series programmable controller

Module selection

(1) POWER SUPPLY MODULE
Since a power supply module with the rated input voltage of 100/200VAC has a potentially hazardous voltage area (42.4V or more at the peak), select a model in which reinforced insulation is provided between the primary and secondary sides.
For those of 24VDC rated input, conventional models can be used.

(2) I/O module
Since an I/O module with the rated input voltage of 100/200VAC has a potentially hazardous voltage area, select a model in which reinforced insulation is provided between the primary and secondary sides.
For those of 24VDC rated input, conventional models can be used.

(3) CPU module, memory card, base unit
Conventional models can be used for these modules, because they only have a 5VDC circuit inside.

(4) Special function module
Conventional models can be used for the special function modules including analog modules, network modules, and positioning modules, because their rated voltage is 24VDC or lower.

(5) Display
Use the CE-marked product.

20.2.3 Power supply

The insulation specification of the power supply module was designed assuming installation category II. Be sure to use the installation category II power supply to the programmable controller.
The installation category indicates the durability level against surge voltage generated by a thunderbolt. Category I has the lowest durability; and category IV has the highest durability.

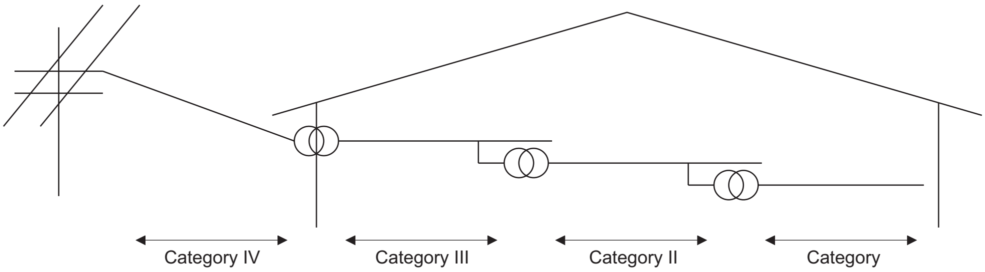


Figure 20.1 Installation Category

Category II indicates a power supply whose voltage has been reduced by two or more levels of isolating transformers from the public power distribution.

20.2.4 Control panel

Because the programmable controller is an open type device (a device designed to be stored within another device), be sure to use it inside the control panel.*

* Also, each network remote station needs to be installed inside the control panel.However, the waterproof type remote station can be installed outside the control panel.

(1) Shock protection

To prevent personnel such as operators who are not familiar with electricity from electric shocks, the control panel must be handled as follows:

(a) Lock the control panel so that only the qualified personnel can open it.

(b) Provide a mechanism so that opening the control panel will automatically stop the power supply.

(c) For electric shock protection, use IP20 or greater control panel.

(2) Dustproof and waterproof features

The control panel also has the dustproof and waterproof functions.Insufficient dustproof and waterproof features lower the insulation withstand voltage, resulting in insulation destruction. As our programmable controllers are designed assuming the pollution level 2, use them in an environment of pollution level 2 or lower.

Pollution level 1:	An environment where the air is dry and conductive dust does not exist.
Pollution level 2:	An environment where conductive dust does not usually exist, however, temporary conductivity may occasionally occur due to accumulated dust. Generally, this is the level for the inside of the IP54-equivalent control panel in a control room or on a shop floor.
Pollution level 3:	An environment where conductive dust exits and conductivity may be generated due to accumulated dust. An environment for a typical factory floor.
Pollution level 4:	Continuous conductivity may occur due to rain, snow, etc.An outdoor environment.

As shown above, the programmable controller can meet pollution level 2 when stored in a control panel equivalent to IP54.

20.2.5 Module installation

(1) Installing modules contiguously
The left side face of each QnA series I/O module is open.When installing I/O modules to the base, do not allow any empty slots between modules.If a slot to the left of a 100/200VAC module is left empty, the circuit board containing the hazardous voltage circuit is exposed.When a slot needs to be left open, be sure to install the blank module (A1SG60).
When using the A1S5☐B(S1) expansion base with no power supply, attach the included cover to the side of the leftmost module.

20.2.6 Grounding

There are two kinds of ground terminals as shown below.Either ground terminal must be used grounded.
Be sure to perform protective grounding to ensure the safety.

Protective grounding⏚	: Ensures the safety of the programmable controller and improves the noise resistance.
Functional grounding⏚	: Improves the noise resistance.

20.2.7 External wiring

(1) Module power supply and external power supply
For the remote module which requires 24VDC as module power supply, the 5/12/24/48VDC I/O module, and the special function module which requires the external power supply, use the 5/12/24/48VDC circuit which is doubly insulated from the hazardous voltage circuit or use the power supply whose insulation is reinforced.

(2) External devices
When a device with a hazardous voltage circuit is externally connected to the programmable controller, use a model whose circuit section of the interface to the programmable controller is intensively insulated from the hazardous voltage circuit.

(3) Reinforced insulation
Reinforced insulation refers to the insulation with the dielectric withstand voltage shown in Table 1.

Reinforced Insulation Withstand Voltage (Installation Category II, source : IEC664)

Rated voltage of hazardous voltage area	Surge withstand voltage (1.2/50μs)
150VAC or less	2500V
300VAC or less	4000V

21 MAINTENANCE AND INSPECTION

In order to use the programmable controller always in good condition, conducting daily and periodical maintenance/inspection on the following items are strongly recommended.

21.1 Daily Inspection

Dairy inspection items recommended are shown in Table 21.1.

Table 21.1 Dairy Inspection

Item	Check item		Content of inspection	Judgement	Action
1	Installation condition of the base unit		Confirm if installation screws are not loose or cover is not detached.	It is installed securely.	Retighten the screw.
2	Installation condition of the I/O modules		Check if the module is not disengaged and if the hook is securely engaged.	The hook should be securely engaged and the module should be positively mounted.	Securely engage the hook.
3	Connection conditions		Loosening of terminal screw	No loosening.	Retighten the terminal screw.
			Proximity of solderless terminals.	There is an appropriate distance.	Correct the distance.
			Connector areas of extension cable	No loosening at connectors.	Retighten the connector fixing screw.
4	LEDs on the main module	POWER SUPPLY MODULE POWER LED	Confirm it is ON.	The LED is ON. (Faulty if it is OFF.)	Refer to Section 22.2.2
		CPU module "RUN" LED	Confirm it is ON in the "RUN" state.	The LED is ON. (Faulty if it is OFF.)	Refer to Section 22.2.3 Section 22.2.4
		CPU module "ERROR" LED	Check that the LED is OFF.	OFF (Faulty if it is ON or flickering.)	Refer to Section 22.2.5
		CPU module "BAT. ARM" LED	Check that the LED is OFF.	OFF (Faulty if it is ON.)	Refer to Section 22.2.7
		Input module LED	Confirm if it correctly turns on and off.	The LED is ON when input is ON, and OFF when input is OFF. (Faulty other than the above.)	Refer to Section 22.2.8
		Output module LED	Confirm if it correctly turns on and off.	The LED is ON when output is ON, and OFF when output is OFF. (Faulty other than the above.)	Refer to Section 22.2.8

21.2 Periodic Inspection

Inspection on items shown below should be conducted once or twice every six months to a year. Conduct the inspection when the equipment is moved or modified, or wiring is changed.

Table 21.2 Periodic inspection

Item	Check item		Content of inspection	Judgement	Corrective action
1	Ambient environment	Ambient temperature	Measure with temperature and humidity gauge. Measure presence of corrosive gases.	0 to 55℃	When used in a panel, temperature inside the panel is the ambient temperature.
		Ambient humidity		10 to 90%RH	
		Atmosphere		There is no corrosive gas present.	
2	Line voltage check		100/200VAC Measure voltage across 100/200VAC terminals.	85 to 264VAC	Change the power supply.
3	Installation condition	Loosening, backlash	Test by moving the module.	Must be installed solidly.	Retighten the screw. For CPU, I/O, or power supply, if loosened, secure it with screws.
		Adhesion of dirt or foreign matters	Visual inspection	No adhesion.	Remove and clean.
4	Connection conditions	Loosening of terminal screw	Retighten with a screwdriver.	No loosening.	Retighten.
		Proximity of solderless terminals	Visual inspection	There is an appropriate distance.	Correct the distance.
		Loosening of connector	Visual inspection	No loosening.	Retighten the connector fixing screw.
5	Battery		Confirm SM51 or SM52 is OFF with a peripheral device in the monitoring mode.	(Preventive maintenance)	Even when there is no low-battery display, replace if specified life is exceeded.

⚠ **WARNING**

- ● Be sure to shut off all phases of the external power supply used by the system before cleaning or retightening the terminal screws or module mounting screws. Failure to do so may result in an electric shock.
  If they are too loose, it may cause a short circuit or malfunctions.
  If too tight, it may cause damage to the screws and/or module, resulting in an accidental drop of the module, short circuit or malfunctions.

21.3 Battery Replacement

Special relay SM51 or SM52 is turned ON when voltage of the battery for backing up programs and power failure compensation function drops. Even though programs and contents of power failure compensation function are not erased immediately when these special relays become ON, the contents could be erased if the ON-status of the special relay fails to be recognized.

Replace the battery before the total latch time after special relay SM51 turns ON reaches the stipulated time.

POINT

SM51 is a battery voltage drop alarm, and it remains ON once turning it ON even if the battery voltage returns to normal.

SM52 is a battery voltage drop alarm, and after turning ON, it goes OFF when the battery voltage returns to normal.

After SM51 and SM52 have turned ON, immediately replace the battery.

SM51 is a battery voltage drop alarm, and it remains ON once turning it ON even if the battery voltage returns to normal.

In order to determine which of these memory's battery has sustained the voltage drop, check the contents of special relay SD51 and SD52.

When the voltage of any memory's battery drops, the bit in SD51 and SD52 that corresponds to each memory turns ON.

SD51, SD52 bit No.	Corresponding memory
Bit 0	Built-in RAM
Bit 1, 2	Memory card

POINT

The relationship of back up between the status of the batteries installed in the CPU module and memory cards is explained below.
The following two points are applied.

1) The battery installed in the CPU module does not back up the RAM memories of the memory cards.
2) The batteries installed in the memory cards do not back up the built-in RAM of the CPU module.

CPU module AC power supply for CPU module	CPU module CPU module battery	Memory card memory Battery	CPU module CPU module memory	Memory card memory Memory
ON	ON	ON	○	○
		OFF	○	○
	OFF	ON	○	○
		OFF	○	○
OFF	ON	ON	○	○
		OFF	○	×
	OFF	ON	×	○
		OFF	×	×

○: Back up is possible.
× : Back up is not possible.

The battery life guideline and the replacement procedures are explained on the following pages.

21.3.1 Battery life

(1) Battery life of CPU module
The CPU module battery life differs depending on the CPU module model.

(a) Q2ASCPU, Q2ASCPU-S1
Table 21.3 shows the battery lives when the Q2ASCPU and Q2ASCPU-S1 are used.

Table 21.3 CPU module battery life

CPU module model	Power-on time ratio[*1]	Battery life[*5]			
		Guaranteed value[*2]	Actual service value (Referencevalue)[*3]		After SM51, SM52 turns ON (Backup power time after an alarm[*4])
			ambient temperature 40 ℃	ambient temperature 25 ℃	
Q2ASCPU	0%	1,800 hours 0.2 years	22,000 hours 2.5 years	43,800 hours 5 years	48 hours 2.0 days
	30%	2,570 hours 0.3 years	31,400 hours 3.6 years	43,800 hours 5 years	48 hours 2.0 days
	50%	3,600 hours 0.4 years	43,800 hours 5 years	43,800 hours 5 years	48 hours 2.0 days
	100%	43,800 hours 5 years	43,800 hours 5 years	43,800 hours 5 years	48 hours 2.0 days
Q2ASCPU-S1	0%	1,150 hours 0.1 years	22,000 hours 2.5 years	43,800 hours 5 years	27 hours 1.0 days
	30%	1,640 hours 0.2 years	31,400 hours 3.6 years	43,800 hours 5 years	27 hours 1.0 days
	50%	2,300 hours 0.3 years	43,800 hours 5 years	43,800 hours 5 years	27 hours 1.0 days
	100%	43,800 hours 5 years	43,800 hours 5 years	43,800 hours 5 years	27 hours 1.0 days

[*1] The power time ratio indicates the percentage of power-on time per day (24 hours). (The power-on time ratio is 50% when the total power-on time is 12 hours and the total power-off time is 12 hours.)

[*2] The guaranteed value represents a battery life at 70 ℃ , which is calculated based on characteristic values of manufacturer-supplied memories (SRAM) and on the assumption of storage within the ambient temperature range of -20 to 75 ℃ (operating ambient temperature of 0 to 55 ℃).

[*3] The actual service value (reference value) represents a battery life that is calculated based on the values measured at storage ambient temperatures of 40 ℃ and 25 ℃ . This value is intended for refe rence only, as it varies with characteristics of the memory.

[*4] The guaranteed time after power-off is 10 minutes when:
- The battery connector is disconnected.
- The battery lead wire is broken

[*5] The battery duration (maximum life) is 5 years (43,800 hours).

Yardsticks for preventive maintenance are as follows:

[1] Replace the battery in four to five years even when it has not been used exceeding the guaranteed value shown in the above table.

[2] Replace the battery when it has been used exceeding the guaranteed value shown in the above table and SM51 is on.

POINT

(1) Use the battery within the time shown by the guaranteed value of the battery life.

(2) If the battery may be used exceeding the guaranteed time, perform ROM operation to protect data in case that the battery will be exhausted during power-off of the programmable controller. Or, after SM51 turns on, back up data within the backup power time.

(3) When the battery (A6BAT) is not connected to the CPU module, its service life is five years.

(4) When the battery-low special relay SM51 turns on, immediately change the battery.
Even if an alarm has not yet occurred, it is recommended to replace the battery periodically according to the operating condition.

(b) Q2ASHCPU, Q2ASHCPU-S1

1) Table 21.4 shows battery lives when the Q2ASHCPU and Q2ASHCPU-S1 of hardware version G or eariler are used. For hardware versions, refer to Section 15.2.

Table 21.4 Battery lives when the Q2ASHCPU and Q2ASHCPU-S1 of hardware version G or earlier are used

CPU module model	Power-on time ratio[*1]	Battery life[*5]			
		Guaranteed value[*2]	Actual service value (Reference value)[*3]		After SM51, SM52 turns ON (Backup power time after an alarm[*4])
			ambient temperature 40 ℃	ambient temperature 25 ℃	
Q2ASHCPU	0%	1,050 hours 0.1 years	8,800 hours 1years	43,800 hours 5 years	24 hours 1.0 days
	30%	1,500 hours 0.2 years	12,570 hours 1.4 years	43,800 hours 5 years	24 hours 1.0 days
	50%	2,100 hours 0.2 years	17,600 hours 2 years	43,800 hours 5 years	24 hours 1.0 days
	100%	43,800 hours 5 years	43,800 hours 5 years	43,800 hours 5 years	24 hours 1.0 days
Q2ASHCPU-S1	0%	860 hours 0.1 years	7,600 hours 0.9 years	40,000 hours 4.6 years	19 hours 1.0 days
	30%	1,220 hours 0.1 years	10,860 hours 1.2 years	43,800 hours 5 years	19 hours 1.0 days
	50%	1,720 hours 0.2 years	15,200 hours 1.7 years	43,800 hours 5 years	19 hours 1.0 days
	100%	43,800 hours 5 years	43,800 hours 5 years	43,800 hours 5 years	19 hours 1.0 days

[*1] The power time ratio indicates the percentage of power-on time per day (24 hours). (The power-on time ratio is 50% when the total power-on time is 12 hours and the total power-off time is 12 hours.)

[*2] The guaranteed value represents a battery life at 70 ℃ , which is calculated based on characteristic values of manufacturer-supplied memories (SRAM) and on the assumption of storage within the ambient temperature range of -20 to 75 ℃ (operating ambient temperature of 0 to 55 ℃).

[*3] The actual service value (reference value) represents a battery life that is calculated based on the values measured at storage ambient temperatures of 40 ℃ and 25 ℃ . This value is intended for refe rence only, as it varies with characteristics of the memory.

[*4] The guaranteed time after power-off is 10 minutes when:.

- The battery connector is disconnected.
- The battery lead wire is broken

[*5] The battery duration (maximum life) is 5 years (43,800 hours).

Yardsticks for preventive maintenance are as follows:

[1] Replace the battery in four to five years even when it has not been used exceeding the guaranteed value shown in the above table.

[2] Replace the battery when it has been used exceeding the guaranteed value shown in the above table and SM51 is on.

POINT

(1) Use the battery within the time shown by the guaranteed value of the battery life.
(2) If the battery may be used exceeding the guaranteed time, perform ROM operation to protect data in case that the battery will be exhausted during power-off of the programmable controller. Or, after SM51 turns on, back up data within the backup power time.
(3) When the battery (A6BAT) is not connected to the CPU module, its service life is five years.
(4) When the battery-low special relay SM51 turns on, immediately change the battery.
Even if an alarm has not yet occurred, it is recommended to replace the battery periodically according to the operating condition.

2) Table 21.5 shows battery lives when the Q2ASHCPU and Q2ASHCPU-S1 of hardware version H or later are used. For hardware versions, refer to Section 15.2.

Table 21.5 Battery lives when the Q2ASHCPU and Q2ASHCPU-S1 of hardware version H or later are used

CPU module model	Power-on time ratio[*1]	Battery life[*5]			
		Guaranteed value[*2]	Actual service value (Reference value)[*3]		After SM51, SM52 turns ON (Backup power time after an alarm[*4])
			ambient temperature 40 ℃	ambient temperature 25 ℃	
Q2ASHCPU	0%	1,050 hours 0.1 years	3,400 hours 0.4 years	4,000 hours 0.5 years	24 hours 1.0 days
	30%	1,500 hours 0.2 years	4,800 hours 0.5 years	5,700 hours 0.7 years	24 hours 1.0 days
	50%	2,100 hours 0.2 years	6,800 hours 0.8 years	8,000 hours 0.9 years	24 hours 1.0 days
	100%	43,800 hours 5 years	43,800 hours 5 years	43,800 hours 5 years	24 hours 1.0 days
Q2ASHCPU-S1	0%	860 hours 0.1 years	3,400 hours 0.4 years	4,000 hours 0.5 years	19 hours 1.0 days
	30%	1,220 hours 0.1 years	4,800 hours 0.5 years	5,700 hours 0.7 years	19 hours 1.0 days
	50%	1,720 hours 0.2 years	6,800 hours 0.8 years	8,000 hours 0.9 years	19 hours 1.0 days
	100%	43,800 hours 5 years	43,800 hours 5 years	43,800 hours 5 years	19 hours 1.0 days

*1 The power time ratio indicates the percentage of power-on time per day (24 hours). (The power-on time ratio is 50% when the total power-on time is 12 hours and the total power-off time is 12 hours.)

*2 The guaranteed value represents a battery life at 70℃, which is calculated based on characteristic values of manufacturer-supplied memories (SRAM) and on the assumption of storage within the ambient temperature range of -20 to 75℃ (operating ambient temperature of 0 to 55℃).

*3 The actual service value (reference value) represents a battery life that is calculated based on the values measured at storage ambient temperatures of 40℃ and 25℃. This value is intended for refe rence only, as it varies with characteristics of the memory.

*4 The guaranteed time after power-off is 10 minutes when:
- The battery connector is disconnected.
- The battery lead wire is broken

*5 The battery duration (maximum life) is 5 years (43,800 hours).

Yardsticks for preventive maintenance are as follows:

[1] Replace the battery in four to five years even when it has not been used exceeding the guaranteed value shown in the above table.

[2] Replace the battery when it has been used exceeding the guaranteed value shown in the above table and SM51 is on.

POINT

(1) Use the battery within the time shown by the guaranteed value of the battery life.

(2) If the battery may be used exceeding the guaranteed time, perform ROM operation to protect data in case that the battery will be exhausted during power-off of the programmable controller. Or, after SM51 turns on, back up data within the backup power time.

(3) When the battery (A6BAT) is not connected to the CPU module, its service life is five years.

(4) When the battery-low special relay SM51 turns on, immediately change the battery.
Even if an alarm has not yet occurred, it is recommended to replace the battery periodically according to the operating condition.

(2) Battery life of memory card

The battery life of memory card differs depending on the memory capacity.The life for each memory is shown in Table 21.6.

Table 21.6 Battery lives of memory cards

Memory card model name	Battery life (Total power failure time) [hr]		
	Guaranteed value (MIN)	Actual value (TYP)	After SM51 is turned ON
Q1MEM-64S	5256	23652	8
Q1MEM-128S	2628	12264	6
Q1MEM-256S	5256	23652	8
Q1MEM-512S	2628	12264	6
Q1MEM-1MS	7008	23652	6
Q1MEM-2MS	2628	12264	6
Q1MEM-64SE	5256	23652	8
Q1MEM-128SE	5256	23652	8
Q1MEM-256SE	5256	23652	8
Q1MEM-512SE	5256	23652	8
Q1MEM-1MSE	2628	12264	6

* Actual value indicates a rough average value and guaranteed value indicates the minimum value.

Yardsticks for preventive maintenance are as follows:

[1] Replace the battery in four to five years even when it has not been used exceeding the guaranteed value shown in the above table.

[2] Replace the battery when it has been used exceeding the guaranteed value shown in the above table and SM51 is on.

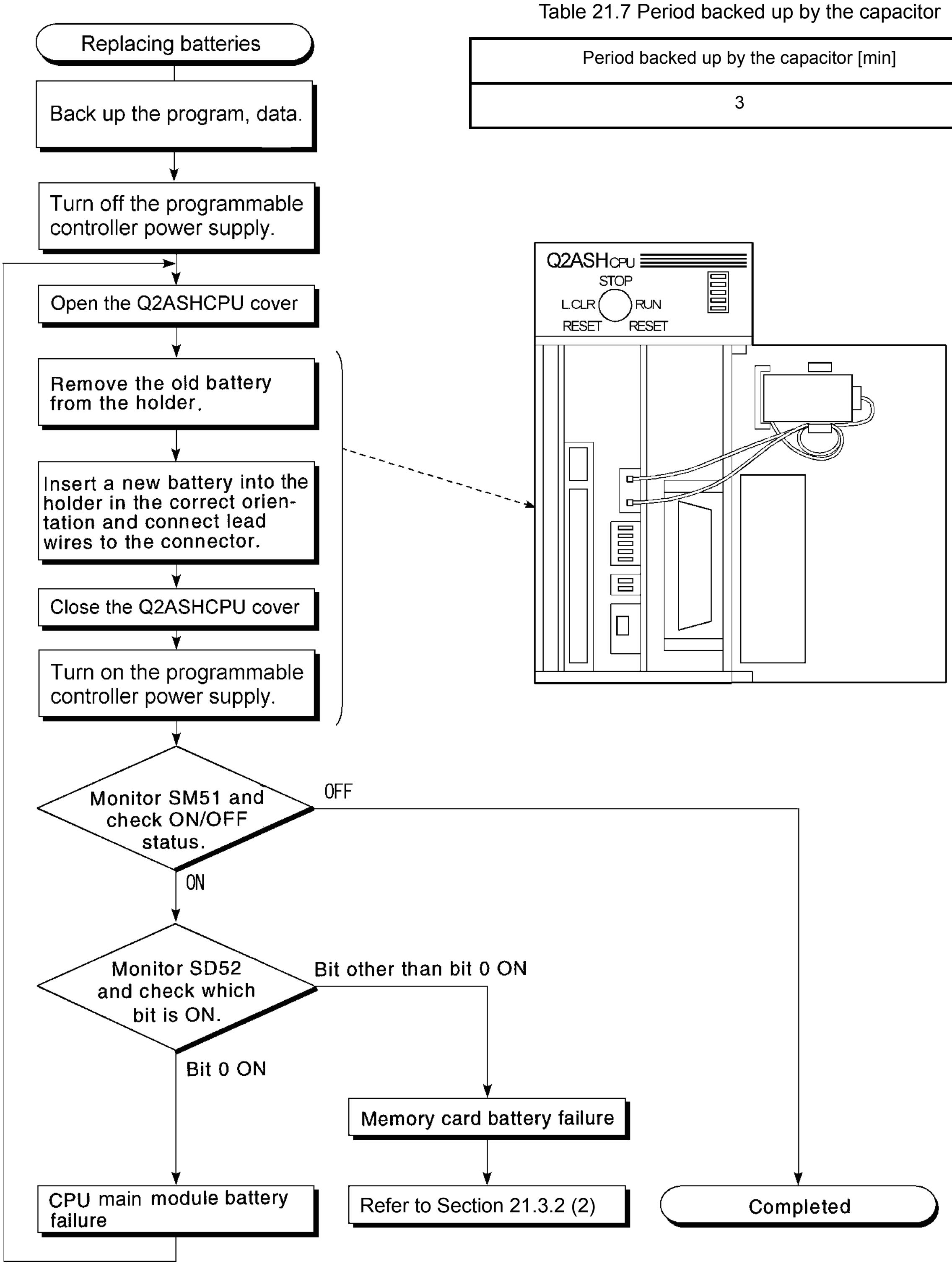
21.3.2 Battery replacement procedure

⚠WARNING ● Correctly connect the battery connector.
Do not charge, disassemble, heat, short-circuit, solder, or throw the battery into the fire.
Incorrect battery handling may cause personal injuries or a fire due to exothermic heat, burst and/or ignition.

(1) CPU module battery replacement procedure
Replace the battery of a CPU module according to the following procedure when life of the battery is over. Even when the battery is removed, memory is backed up by the capacitor for a while. However, if replacement takes longer than the guaranteed value shown in the following table, the content of the memory may be erased, so replace the battery quickly.

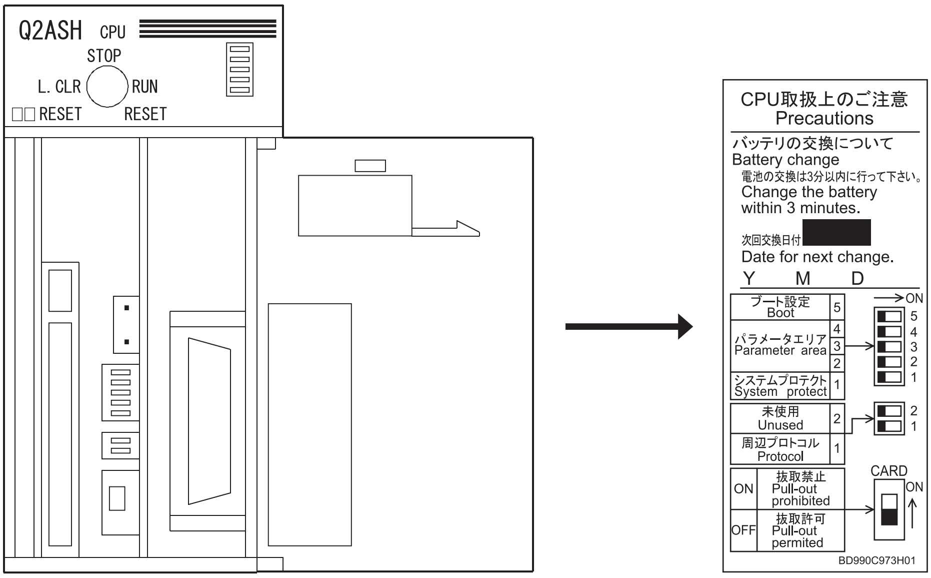
Table 21.7 Period backed up by the capacitor

Period backed up by the capacitor [min]
3



POINT

After replacing a battery, write the date for next battery replacement on the sticker on the back side of the front cover.
Write the proper date by checking the battery life. (Refer to Section 21.3.1)

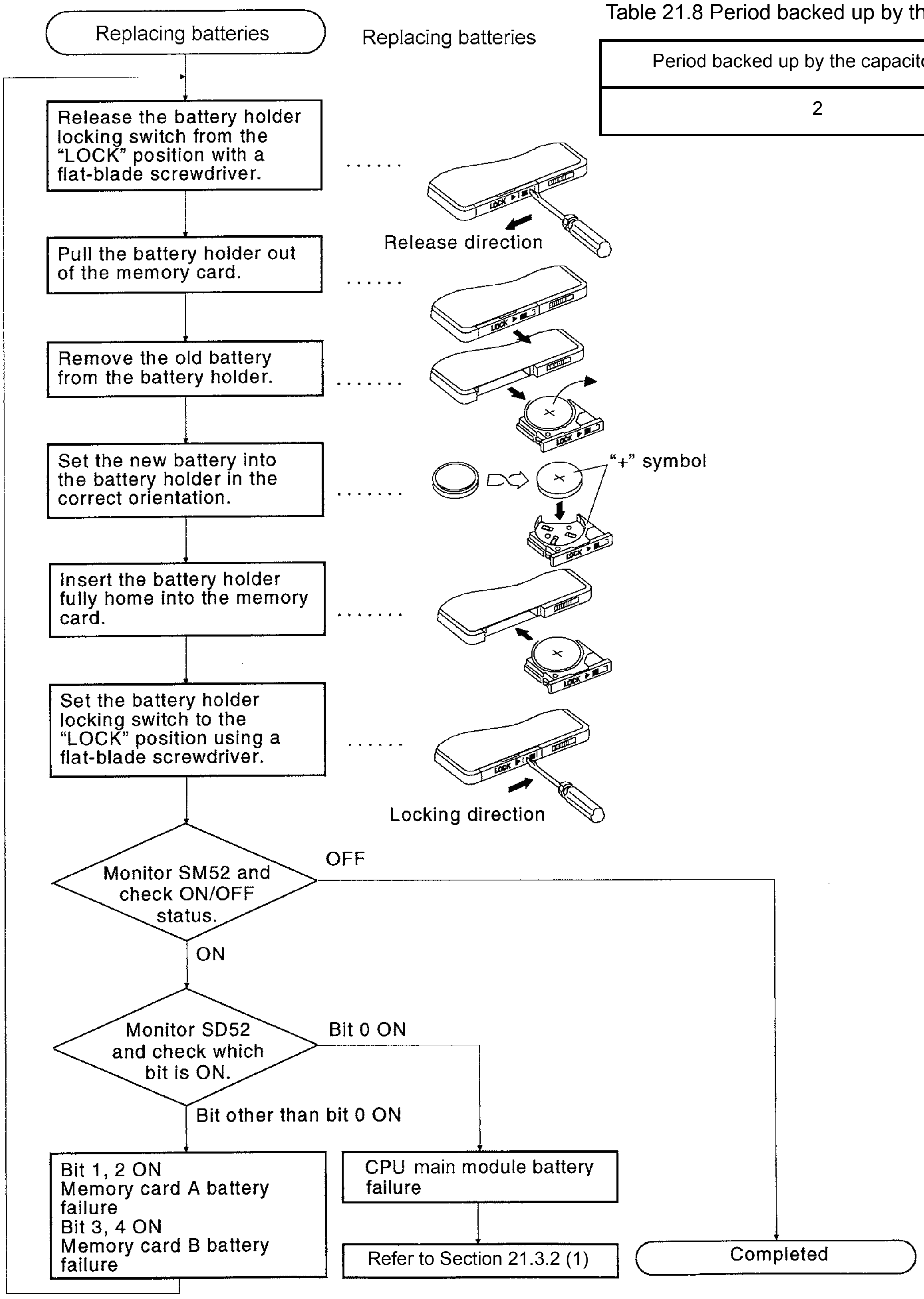


(2) Memory card battery replacement procedure
Replace the memory card battery according to the following procedure when the life is over. Even if the battery is removed, the memory card memory is backed up by a capacitor so that the battery can be replaced while the memory card is out of the CPU module.
However, if the time taken to replace the battery exceeds the guaranteed value indicated in Table 21.6 below, the contents of the memory may be lost. Therefore, change the battery as quickly as possible. While the programmable controller power is ON, the battery can be replaced without removing the memory card in the CPU module. In this case, the memory contents are backed up by the power supply voltage from the power supply module.

Table 21.8 Period backed up by the capacitor

Period backed up by the capacitor [min]
2

Replacing batteries



21.4 When Reoperating a programmable controller After Storing it with a Battery Unconnected

When reoperating after a battery is uncounted and the programmable controller is stored, the memory contents of a CPU module and memory card may be undefined.
Therefore, when resuming the operation, clear the CPU module memory and format the memory in the CPU module by peripheral device.
Afeter doing so, write the memory contents backed up before saving to each memory.
The relationship between the backed-up memory and the batteries is explained below.

Memory			Battery	
			A6BAT installed in a CPU module	Battery incorporates a memory card
CPU module	Built-in RAM		○	×
	Device*		○	×
Memory card	SRAM type		×	○
	SRAM + E^2PROM type	SRAM	×	○
		E^2PROM	– (Battery back up is not required.)	

○:Battery is backed up. ×: Battey is not backed up.

*As for device memory, also clear the latch range.

Before resuming the operation, clear/format the memory for which a battery is backed up in the table above with a peripheral device.
For memory clear/format operations, refer to the following manuals.

- Type SW□IVD-GPPQ GPP Software package Operating Manual (Online)
- GX Developer Operating Manual

POINT

(1) Make sure to back up each memory contents before storing the programmable controller.
When a programmable controller power supply is ON or CPU module reset is cancelled, a CPU module reviews the status of data below, and initializes all the data if detecting an error.
- RAM data in built-in RAM
- Breakdown history
- Latch data (Latch relay (L), latch setting range device set in a parameter), special relay SM900 to SM999, special register SD900 to SD999)
- Sampling trace data

21.5 When a programmable controller is Reoperated After Stored with the Battery Over the Battery Life

If a battery exceeded its guaranteed life is stored and reoperated, the memory contents of CPU module and memory card may be undefined.
Therefore, when resuming the operation, clear the CPU module memory and format the memory in the CPU module by peripheral device.
Afeter doing so, write the memory contents backed up before saving to each memory.
The relationship between the backed-up memory and the batteries is explained below.

Memory			Battery	
			A6BAT installed in a CPU module	Battery incorporates a memory card
CPU module	Built-in RAM		○	×
	Device*		○	×
Memory card	SRAM type		×	○
	SRAM + E2PROM type	SRAM	×	○
		E^2PROM	– (Battery back up is not required.)	

○:Battery is backed up. ×: Battey is not backed up.

*As for device memory, also clear the latch range.

Before resuming the operation, clear/format the memory for which a battery is backed up in the table above with a peripheral device.
For memory clear/format operations, refer to the following manuals.

- Type SW□IVD-GPPQ GPP Software package Operating Manual (Online)
- GX Developer Operating Manual

POINT

(1) Make sure to back up each memory contents before storing a programmable controller.

(2) When a programmable controller power supply is ON or CPU module reset is cancelled, a CPU module reviews the status of data below, and initializes all the data if detecting an error.
- RAM data in built-in RAM
- Breakdown history
- Latch data (Latch relay (L), latch setting range device set in a parameter), special relay SM900 to SM999, special register SD900 to SD999)
- Sampling trace data

22 TROUBLESHOOTING

The description, cause determination, and corrective actions of each error which may occur during system usage are described.

22.1 Fundamentals of Troubleshooting

Besides using obviously highly-reliable devices to increase system reliability, it is an important point to quickly start up the system again when an error occurs.
In order to quickly start up the system, find the cause of the problem and resolve it. There are the following three basic points to be aware of when performing troubleshooting.

(1) Visual confirmation
Confirm the following points:
1) Machine operation (stop status and operation status)
2) Power supply ON/OFF
3) I/O equipment status
4) Wiring status (I/O wires and cable)
5) Display status of each display indicator (POWER LED, RUN LED, ERROR LED, I/O LED, etc.)
6) Status of each setting switch (extension base, power failure compensation, etc.)

After confirming 1) to 6), connect a peripheral device and observe the operation status of the programmable controller and program contents.

(2) Error confirmation
Observe how the error changes by performing the following operations:
1) Set the RUN/STOP key switch to "STOP".
2) Reset using the RUN/STOP key switch.
3) Turn ON/OFF the power supply.

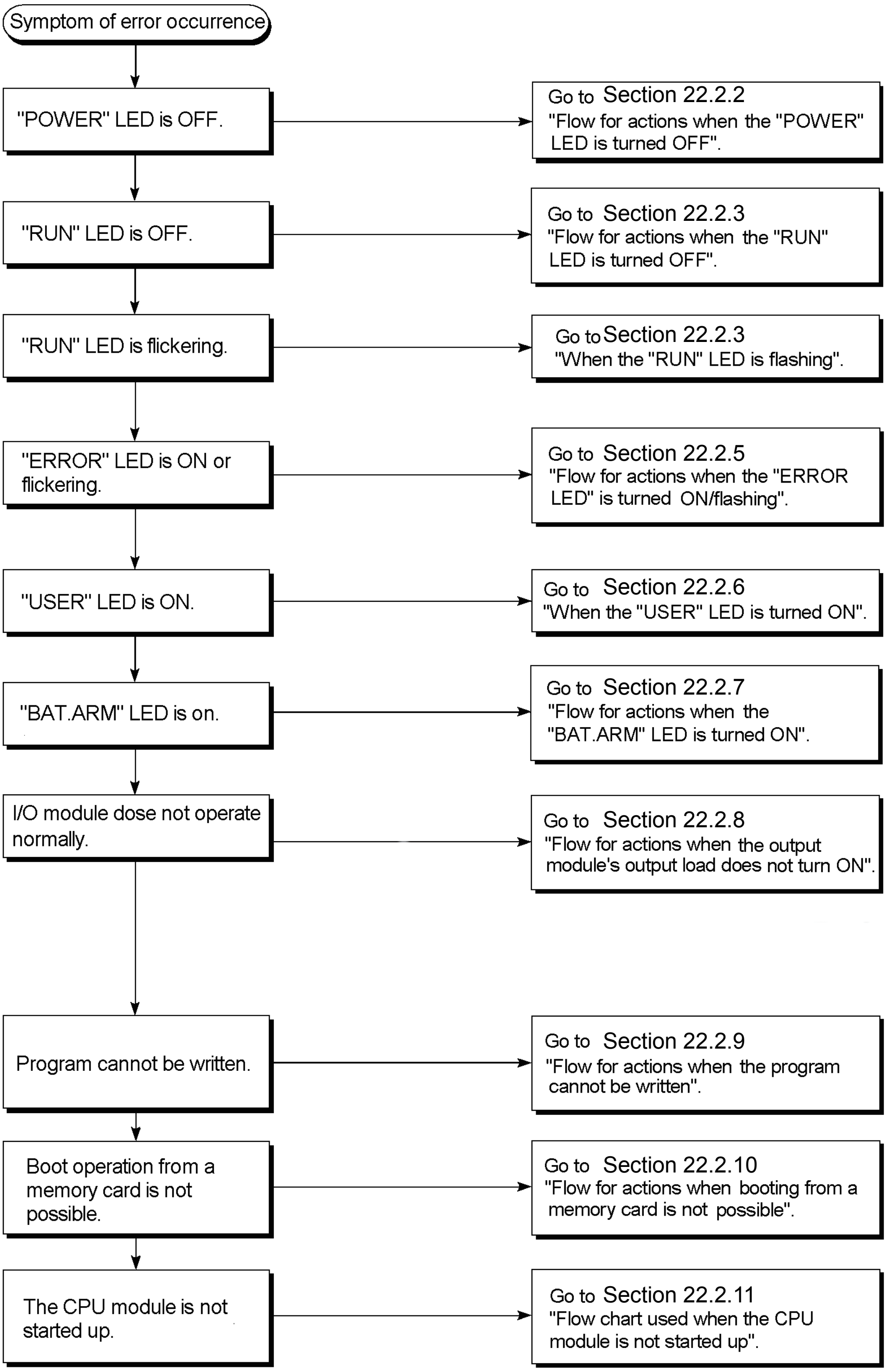
(3) Narrow down the range
By performing the (1) and (2) above, assume the faulty area in the following:
1) Programmable controller or external?
2) I/O module or others?
3) Sequence program?

22.2 Troubleshooting

The error definition determination method, error definition corresponding to the error code, and corrective actions are described.

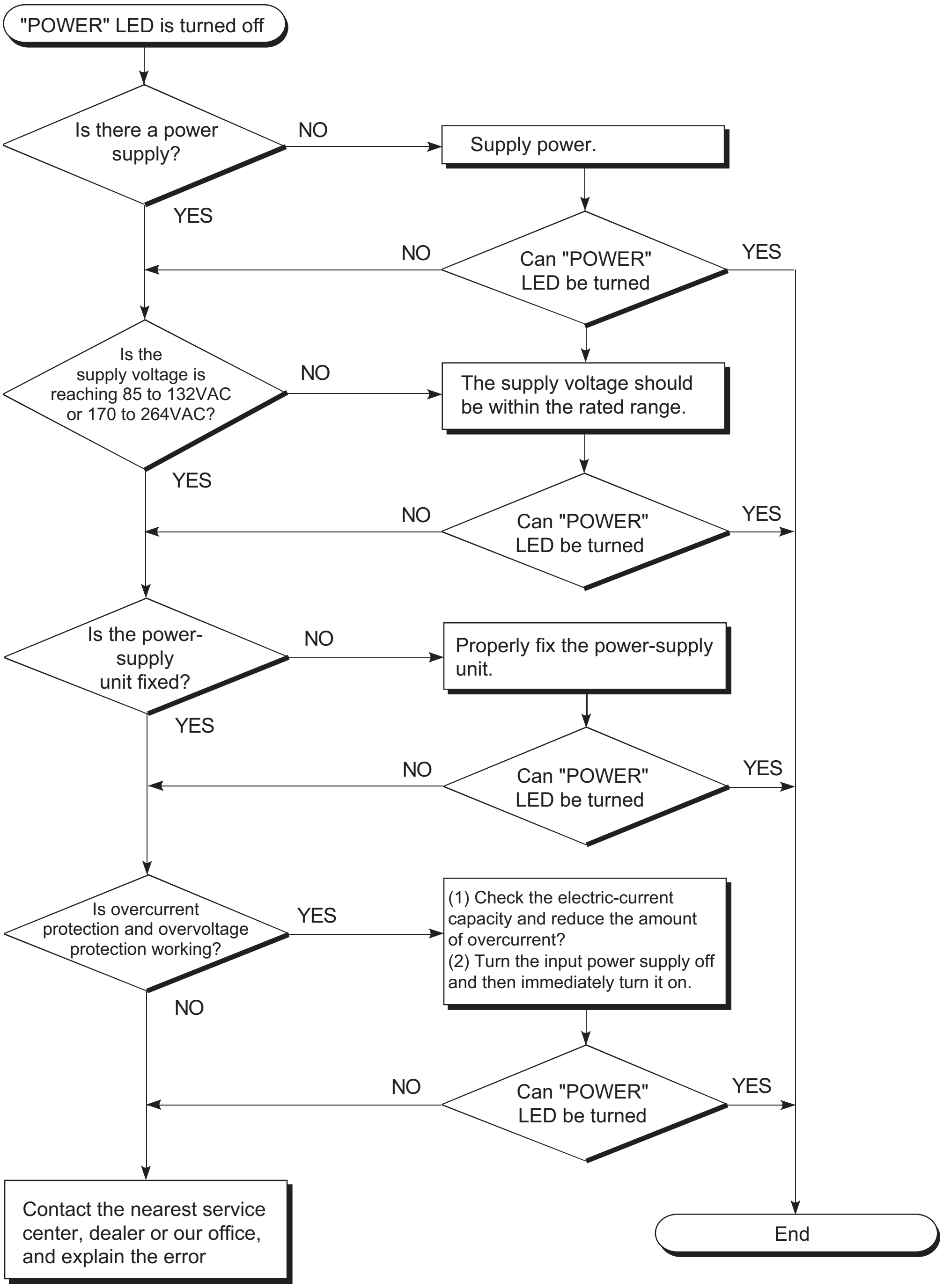
22.2.1 Troubleshooting flowchart

The error definitions are described by events.



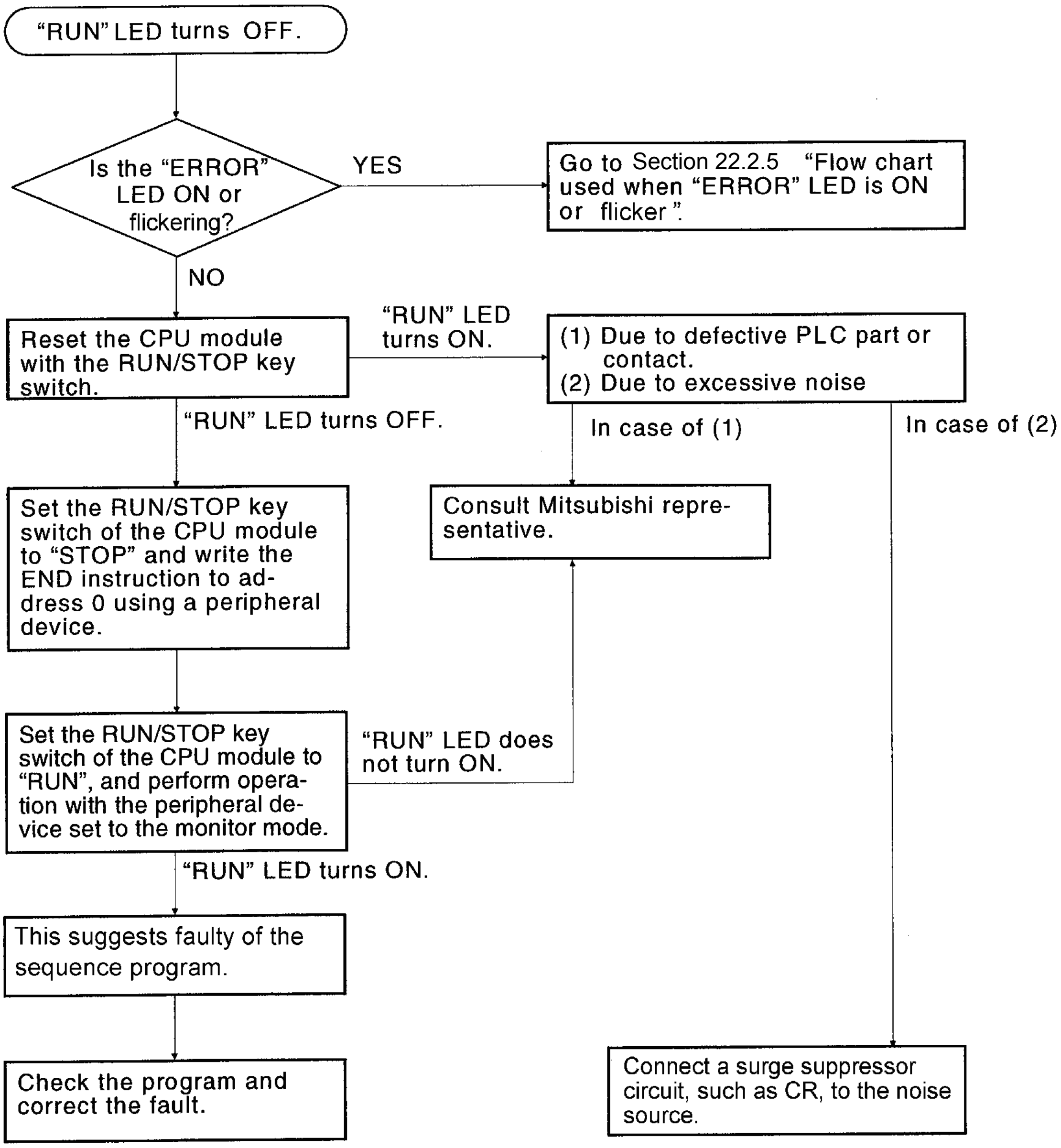
22.2.2 Flow for actions when the "POWER" LED is turned OFF

The flow when the programmable controller power is ON or when the "POWER" LED of the power supply module is ON during operation is described.



22.2.3 Flow for actions when the "RUN" LED is turned OFF

The flow when the "POWER" LED of the CPU module turns OFF during operation is described.

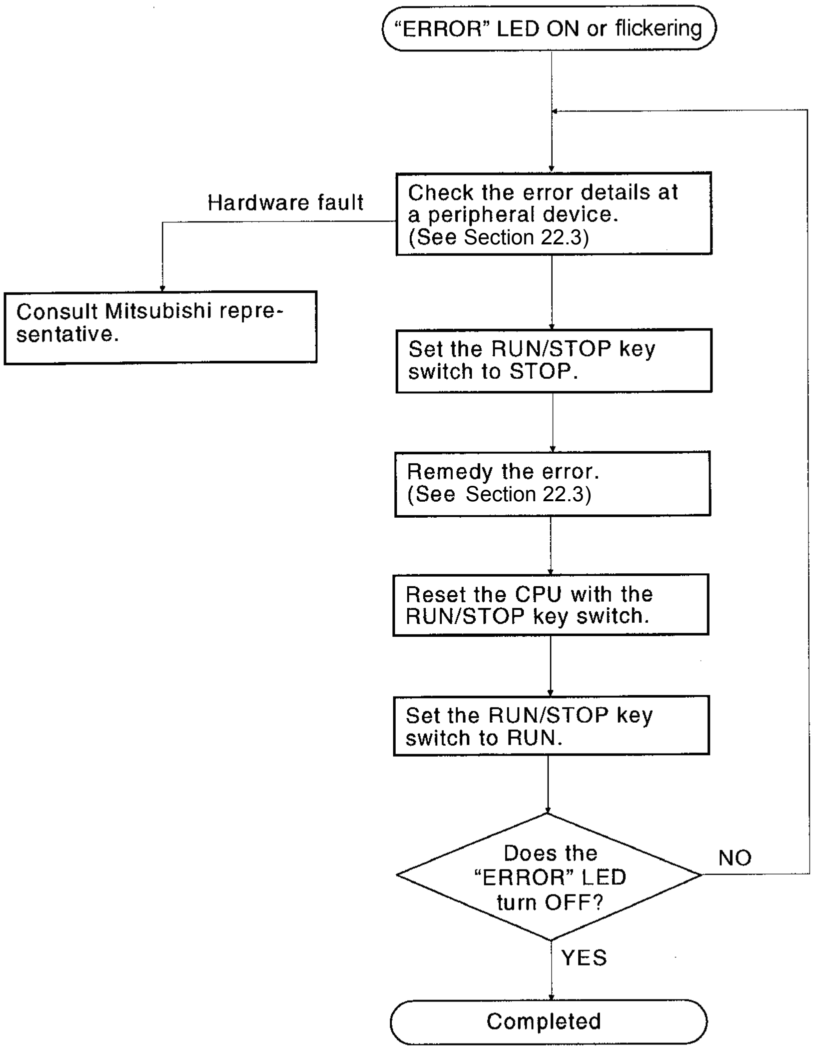


22.2.4 When the "RUN" LED is flashing

Flashing of the "RUN" LED of a CPU module is described below.
With the Q2ASCPU, when the RUN/STOP key switch is turned from STOP to RUN after writing a program in the STOP state, the "RUN" LED flashes. Then, no CPU module error occurs, but the operation stops.
To set the CPU module to RUN, either turn the RUN/STOP key switch to STOP then RUN again, or reset the CPU module using the key switch. The "RUN" LED turns ON.

22.2.5 Flow for actions when the "ERROR LED" is turned ON/flashing

The flow when the programmable controller power is ON, when the operation is started or when the "ERROR" LED of the CPU module is ON/blinking during operation is described.



22.2.6 When the "USER" LED is turned ON

This section describes the case when the "USER" LED of CPU module is turned on.
With the Q2ASCPU, the "USER" LED comes ON when an error is detected by the CHK instruction, or when an annunciator (F), turns ON.
When the "USER" LED is turned ON, monitor SM62 and SM80 of the special relay in the peripheral device monitor mode.
After monitoring and removing the cause, the "USER" LED can be turned OFF by resetting the RUN/STOP key switch or performing the LEDR instruction.

- When SM62 is ON
  With the annunciator (F) ON, the "USER" LED is ON.
  Check the error cause with SD62 to SD79.
- When SM80 is ON,
  With execution of the CHK instruction, the "USER" LED is ON.
  Check the error cause with SD80.

After checking the error cause, remove the cause.
The "USER" LED can be turned OFF by either of the following operations.

- Resetting the system with the RUN/STOP key switch
- Execution of the LEDR instruction with the sequence program

REMARK

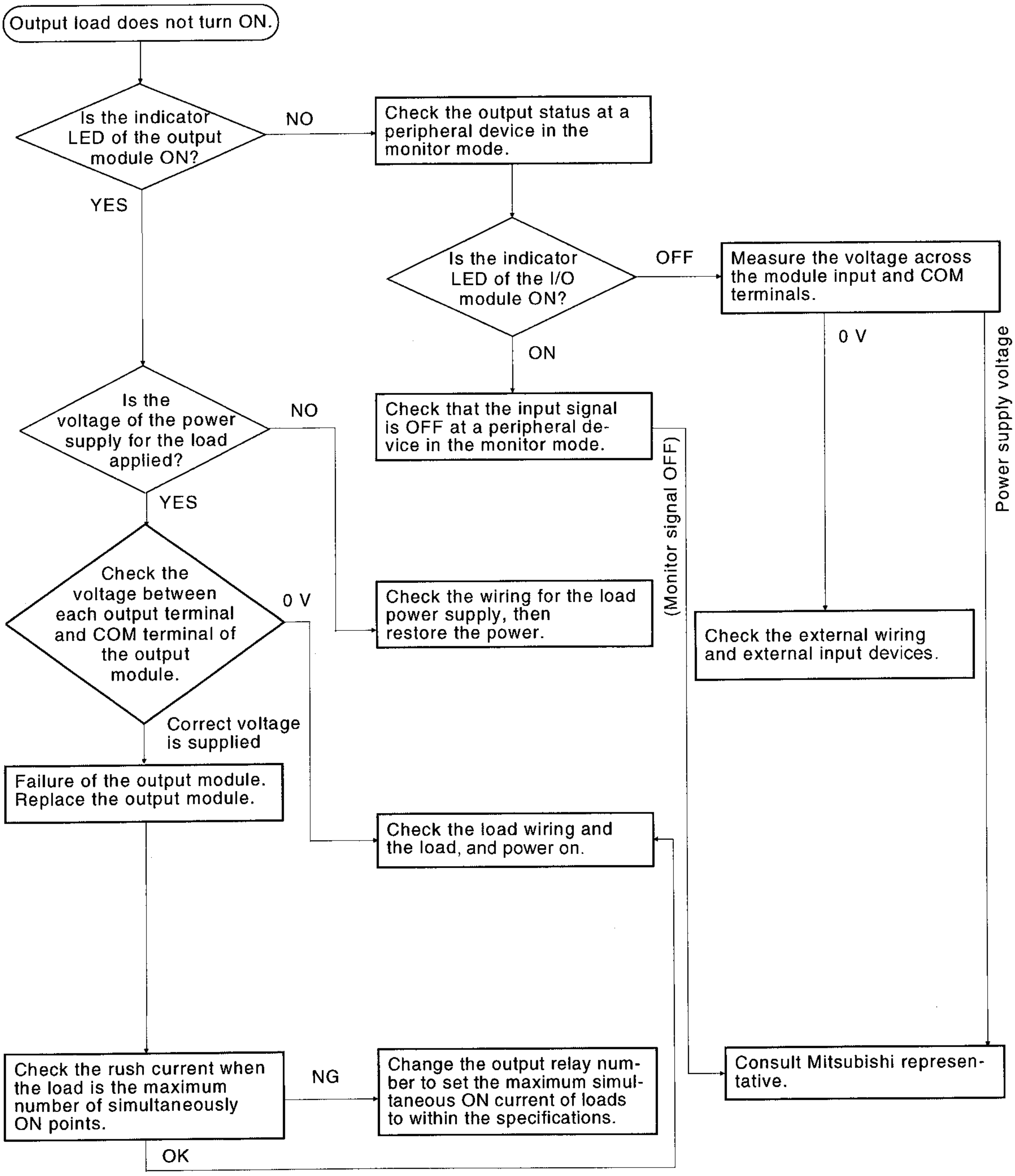
When the RUN/STOP key switch is turned to "L.CLR" several times in a latch clear operation, the "USER" LED flashes to indicate that latch clear processing is in progress.
When the RUN/STOP key switch is turned once more to "L.CLR" while the "USER" LED is flashing, the "USER" LED goes OFF and latch clear processing is ended.

22.2.7 Flow for actions when the "BAT.ARM" LED is turned ON

This section describes the case when the "BAT.ARM" LED of CPU module is turned on.
With the Q2ASCPU, the "BAT.ARM" LED turns ON when the voltage of the battery for a CPU module or a memory card drops.
When the "BAT.ARM" LED turns ON, monitor the special relays (SM51 and SM52) and special registers (SD51 and SD52) in the peripheral device monitor mode, and check if there has been a voltage drop at either of the battery for a CPU module or a memory card.
After monitoring and replacing the battery by a new one, the "BAT.ALM" LED can be turned OFF by resetting the RUN/STOP key switch or performing the LEDR instruction.

22.2.8 Flow for actions when the output module's output load does not turn ON

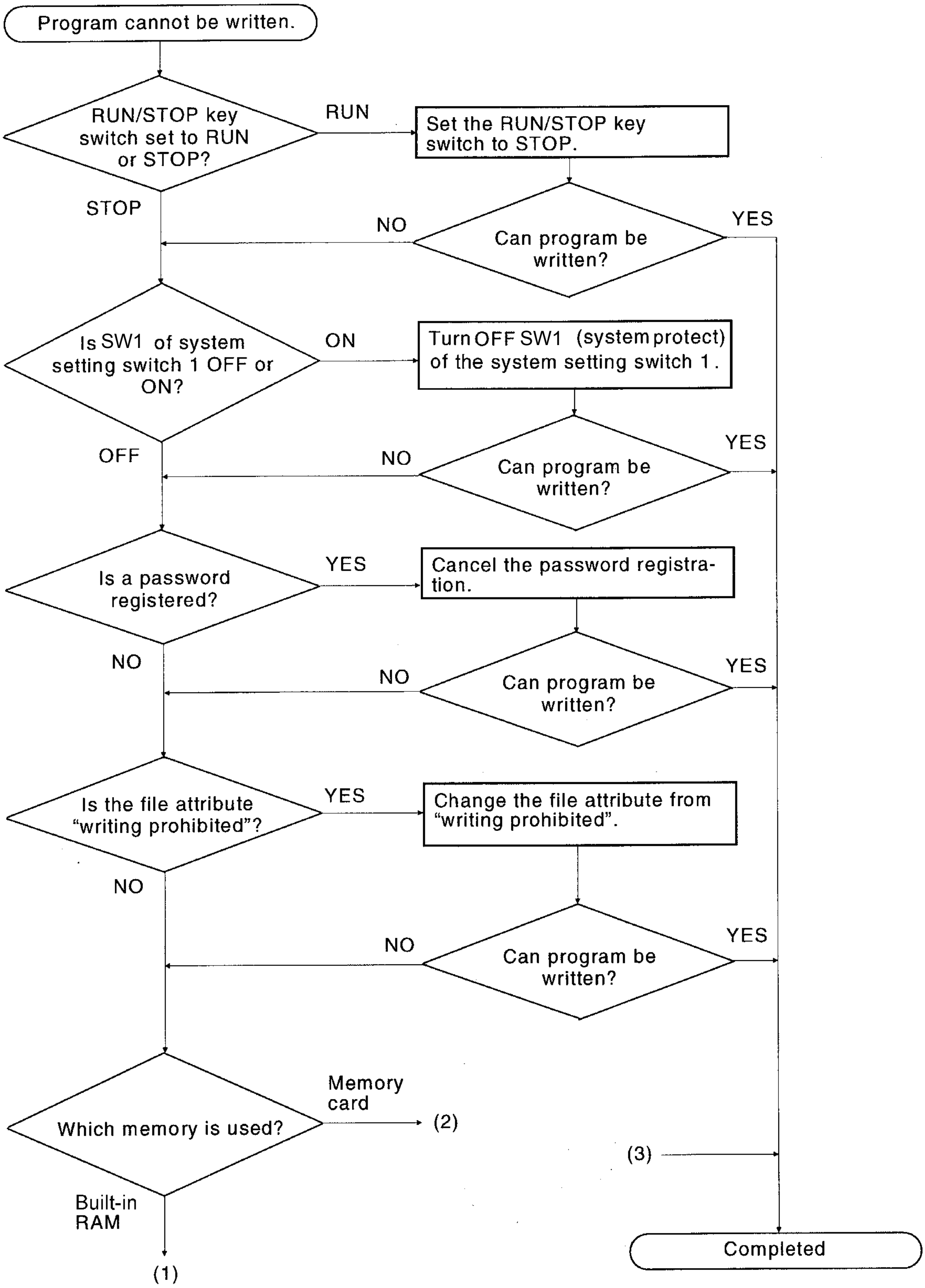
The flow when the output load of the output module is not turned ON during operation is described.

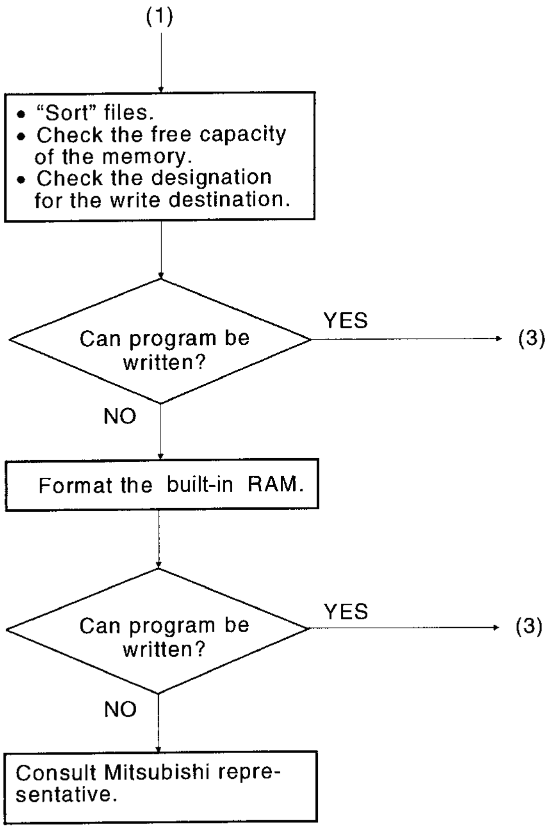


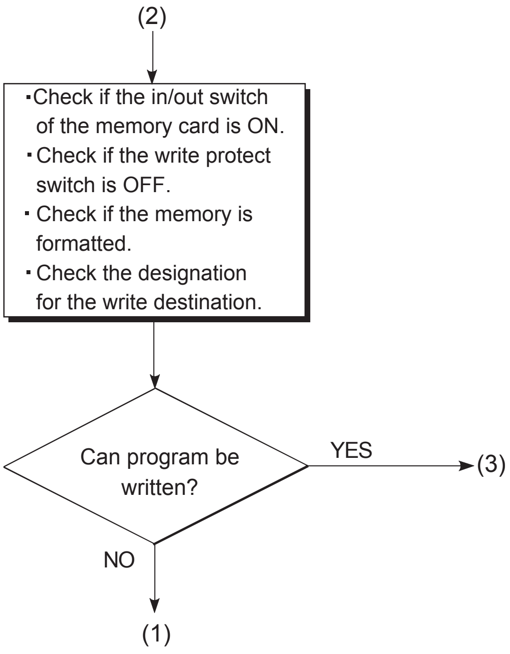
POINT
For problems when the input signal does not turn off or ourput load does not turn off, perform troubleshooting by referring to the fault examples for the I/O modules in Section 22.5.

22.2.9 Flow for actions when the program cannot be written

The flow when a program cannot be written to the CPU module is described.

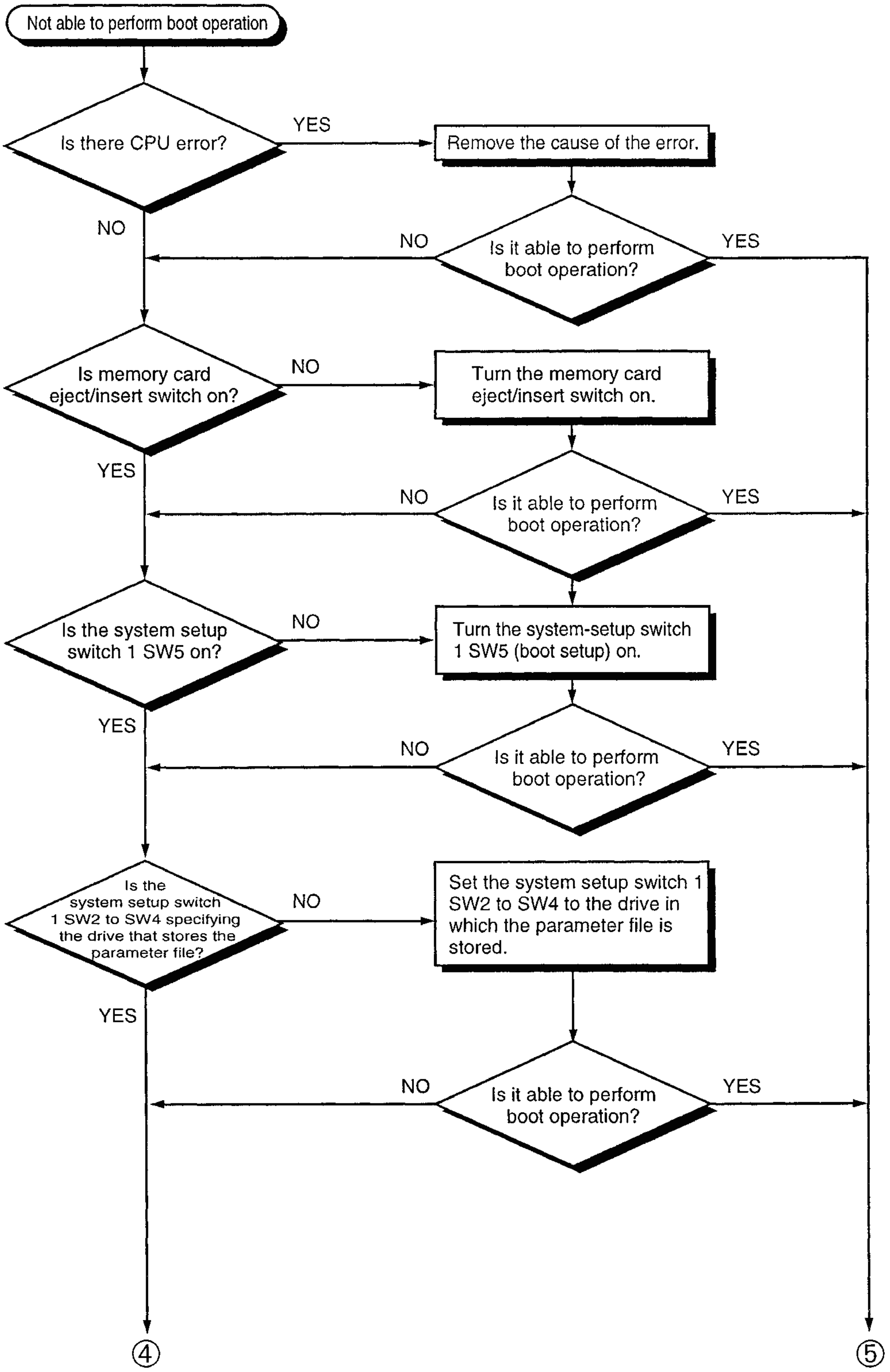


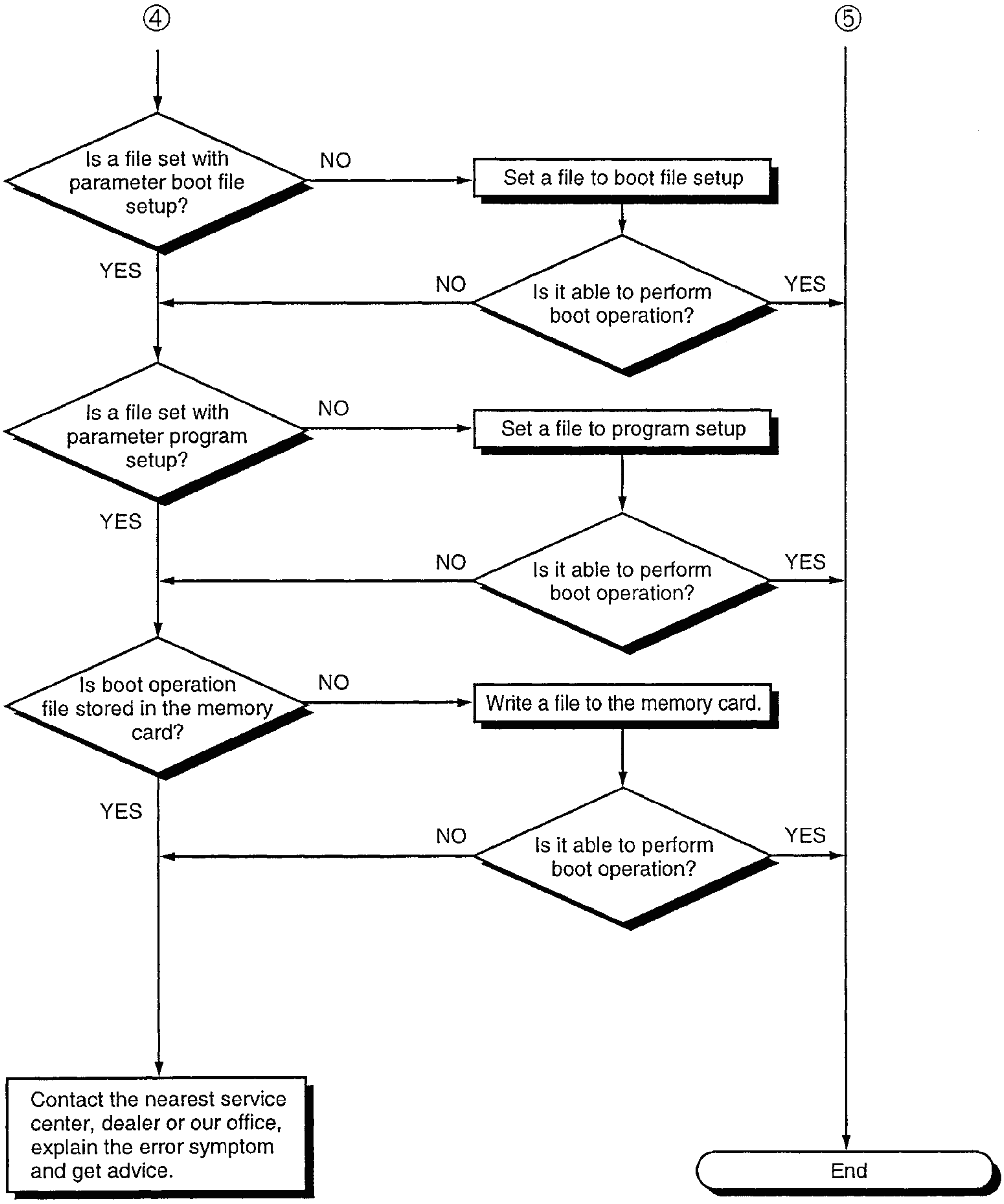

(1)
• "Sort" files.
• Check the free capacity of the memory.
• Check the designation for the write destination.
Can program be written?
YES
(3)
NO
Format the built-in RAM.
Can program be written?
YES
(3)
NO
Consult Mitsubishi representative.


(2)
•Check if the in/out switch of the memory card is ON.
•Check if the write protect switch is OFF.
•Check if the memory is formatted.
•Check the designation for the write destination.
Can program be written?
YES
(3)
NO
(1)

22.2.10 Flow for actions when booting from a memory card is not possible

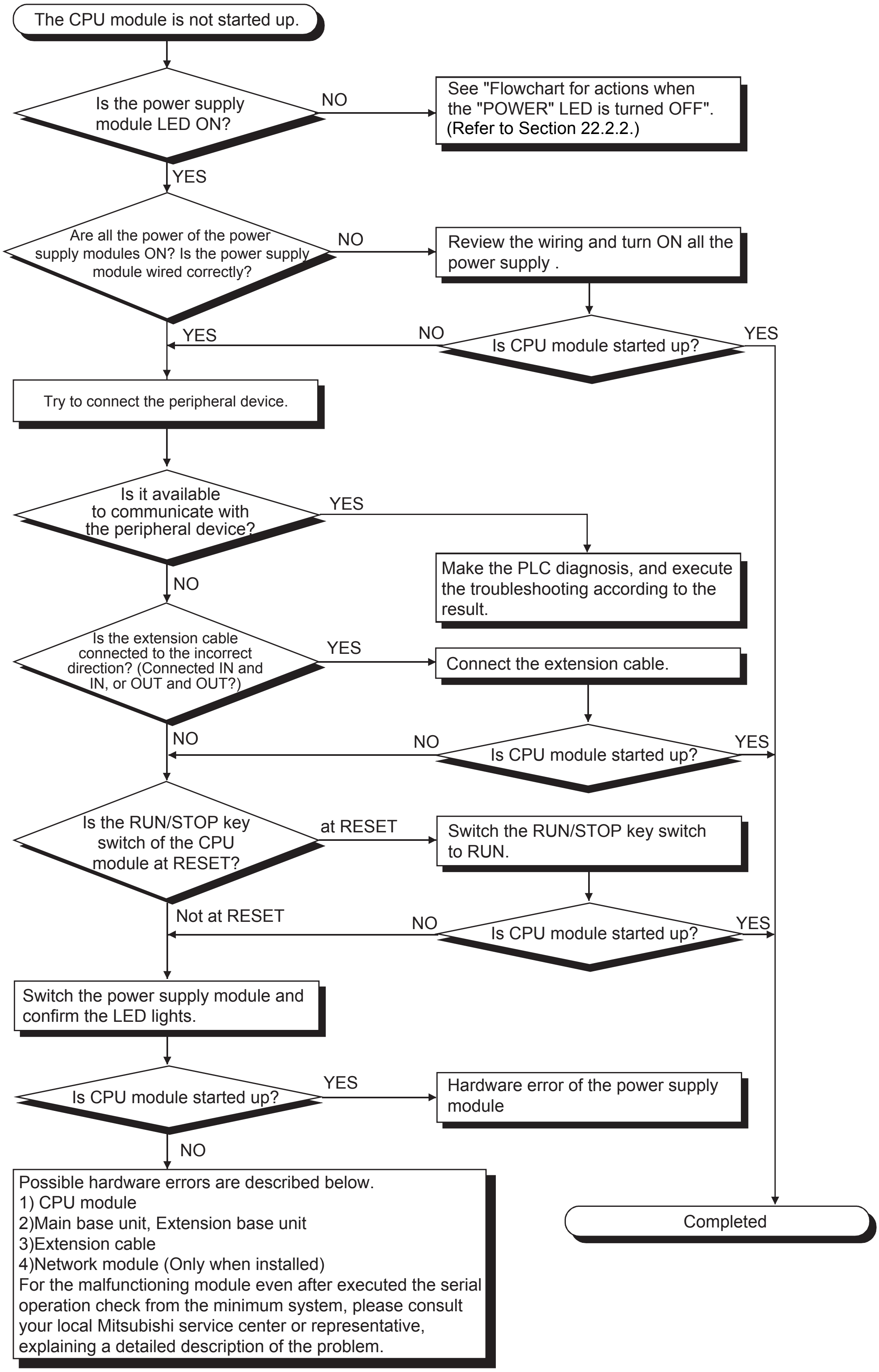
The flow when the CPU module cannot be booted from a memory card is described.




④
⑤
Is a file set with parameter boot file setup?
NO
Set a file to boot file setup
YES
NO
Is it able to perform boot operation?
YES
Is a file set with parameter program setup?
NO
Set a file to program setup
YES
NO
Is it able to perform boot operation?
YES
Is boot operation file stored in the memory card?
NO
Write a file to the memory card.
YES
NO
Is it able to perform boot operation?
YES
Contact the nearest service center, dealer or our office, explain the error symptom and get advice.
End

22.2.11 Flow chart used when the CPU module is not started up

The following shows the flow when the CPU module is not started up.



MEMO

22.3 Error Code List

When an error occurs at PLC power ON, on switching to the RUN status, or during the RUN status, the self-diagnostics function displays the error content (by LED indication, or message display on an LED indicator), and stores the error information at a special relay (SM) and special register (SD).
If an error occurs on a data communicaton request from peripheral devices, a special function module and the network system to the CPU module, error codes (4000H to 4FFFH) are returned to the request source.
QnACPU errors and corrective actions are described in this section.

(1) How to read the error code lists
The following shows the way of reading the error code lists from Section 22.3.3 (1000 to 1999) to Section 22.3.9 (7000 to 10000).

(a) Error code, common information, and individual information
Alphanumeric characters in the parentheses of the titles indicate the special register numbers where the individual information is stored.

(b) Compatible CPUs

○ : Compatible with all the QnACPU and QCPU.
QCPU : Compatible with all the Q series CPU module.
Q00J/Q00/Q01 : Compatible with the Basic model QCPU.
Qn(H) : Compatible with the High Performance model QCPU.
QnPH : Compatible with the process CPU.
QnPRH : Compatible with the redundant CPU.
QnA : Compatible with the QnA series and Q2ASCPU series.
Rem : Compatible with the MELSECNET/H remote I/O module.
Each CPU module : Compatible with the listed CPU module. (Example: Q4AR, Q2AS)

22.3.1 Error Codes

There are errors that is detected by the self-diagnostics function of the CPU module, and that is detected while communicating with the CPU module.

The table below shows the link between the type of error detection, the point of error detection and the error codes.

Error Detection Type	Error Detection Point	Error Code	Reference for Error Contents
Detection by the self-diagnostics function of the CPU module	CPU module	1000 to 10000[*1]	Section 22.3.3 to Section 22.3.9
Detection while communicating with the CPU module	CPU module	4000H to 4FFFH	Appendix 5
	Serial communication	7000H to 7FFFH	Serial Communication Module User's Manual
	CC-Link module	B000H to BFFFH	CC-Link System Master/Local Module User's Manual
	Ethernet module	C000H to CFFFH	Ethernet Interface Module User's Manual
	MELSECNET/10 network module	F000H to FFFFH	QnA/Q4AR MELSECNET/10 Network System Reference Manual

[*1] The error codes of the CPU module are categorizes according to minor errors, moderate errors and major errors.

Minor error:	Errors that CPU module such as a battery error continues the operation (Error code: 1300 to 10000)
Moderate error:	Errors that CPU module such as a WDT error stops the operation (Error code: 1300 to 10000)
Minor error:	Errors that CPU module such as a RAM error stops the operation (Error code: 1000 to 1299)

"The error that the QnACPU continues operation" and "the error that QnACPU stops operation" are determined by "CPU operation status" of the error code list.

22.3.2 Procedure to read an error code

When an error occurs, error codes and error messages can be read with the peripheral devices.

For details on the setting method for each function, refer to the GX Developer Operating Manual or SW☐IVD-GPPQ Operating Manual (Offline).

22.3.3 Error code list (1000 to 1999)

The following shows the error messages from the error code 1000 to 1999, the contents and causes of the errors, and the corrective actions for the errors.

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
1000	**[MAIN CPU DOWN]** Runaway or failure of CPU module or failure of main CPU • Malfunctioning due to noise or other reason • Hardware fault ■**Collateral informationmmon** • Common Information:– • Individual Information:– ■**Diagnostic Timing** • Always	• Take noise reduction measures. • Reset the CPU module and RUN it again.If the same error is displayed again, this suggests a CPU module hardware fault.(Contact your local Mitsubishi representative.)	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
1010	**[END NOT EXECUTE]** Entire program was executed without the execution of an END instruction. • When the END instruction is executed it is read as another instruction code, e.g. due to noise. • The END instruction has been changed to another instruction code somehow. ■**Collateral informationmmon** • Common Information:– • Individual Information:– ■**Diagnostic Timing** • When an END instruction executed	• Take noise reduction measures. • Reset the CPU module and RUN it again. If the same error is displayed again, this suggests a CPU module hardware fault. (Contact your local Mitsubishi representative.)		
1101	**[RAM ERROR]** The sequence program storing built-in RAM/ program memory in the CPU module is faulty. ■**Collateral informationmmon** • Common Information:– • Individual Information:– ■**Diagnostic Timing** • At power ON/ At reset/ When an END instruction executed	• Take noise reduction measures. • Reset the CPU module and RUN it again. If the same error is displayed again,this suggests a CPU module hardware fault.(Contact your local Mitsubishi representative.)		
1102	**[RAM ERROR]** • The work area RAM in the CPU module is faulty. • The standard RAM and extended RAM in the CPU module are faulty. ■**Collateral informationmmon** • Common Information:– • Individual Information:– ■**Diagnostic Timing** • At power ON/ At reset/ When an END instruction executed	• Take noise reduction measures. • Reset the CPU module and RUN it again. If the same error is displayed again,this suggests a CPU module hardware fault.(Contact your local Mitsubishi representative.)		
1103	**[RAM ERROR]** The device memory in the CPU module is faulty. ■**Collateral informationmmon** • Common Information:– • Individual Information:– ■**Diagnostic Timing** • At power ON/At reset	• Take noise reduction measures. • When indexing is performed, check the value of index register to see if it is within the device range. • Reset the CPU module and RUN it again. If the same error is displayed again,this suggests a CPU module hardware fault.(Contact your local Mitsubishi representative.)		
1104	**[RAM ERROR]** The address RAM in the CPU module is faulty. ■**Collateral informationmmon** • Common Information:– • Individual Information:– ■**Diagnostic Timing** • At power ON/At reset	• Take noise reduction measures. • Reset the CPU module and RUN it again.If the same error is displayed again, this suggests a CPU module hardware fault. (Contact your local Mitsubishi representative.)		

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
1105	**[RAM ERROR]** The system RAM in the CPU module is faulty. **■Collateral informationmmon** • Common Information:– • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	• Take noise reduction measures. • Reset the CPU module and RUN it again.If the same error is displayed again, this suggests a CPU module hardware fault. (Contact your local Mitsubishi representative.)	RUN: Off ERR.: Flicker CPU Status: Stop	Q4AR
1200	**[OPE. CIRCUIT ERR.]** The operation circuit for index modification in the CPU module does not operate normally. **■Collateral informationmmon** • Common Information:– • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	This suggests a CPU module hardware fault. (Contact your local Mitsubishi representative.)		QnA
1201	**[OPE. CIRCUIT ERR.]** The hardware (logic) in the CPU module does not operate normally. **■Collateral informationmmon** • Common Information:– • Individual Information:– **■Diagnostic Timing** • At power ON/At reset			
1202	**[OPE. CIRCUIT ERR.]** The operation circuit for sequence processing in the CPU module does not operate normally. **■Collateral informationmmon** • Common Information:– • Individual Information:– **■Diagnostic Timing** • At power ON/At reset			
1203	**[OPE. CIRCUIT ERR.]** The operation circuit for index modification in the CPU module does not operate normally. **■Collateral informationmmon** • Common Information:– • Individual Information:– **■Diagnostic Timing** • When an END instruction executed			Q4AR
1204	**[OPE. CIRCUIT ERR.]** The hardware (logic) in the CPU module does not operate normally. **■Collateral informationmmon** • Common Information:– • Individual Information:– **■Diagnostic Timing** • When an END instruction executed			
1205	**[OPE. CIRCUIT ERR.]** The operation circuit for sequence processing in the CPU module does not operate normally. **■Collateral informationmmon** • Common Information:– • Individual Information:– **■Diagnostic Timing** • When an END instruction executed			QnA

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
1206	**[OPE. CIRCUIT ERR.]** The DSP operation circuit in the CPU module does not operate normally. ■**Collateral informationmmon** • Common Information:– • Individual Information:– ■**Diagnostic Timing** • When instruction executed	This suggests a CPU module hardware fault. (Contact your local Mitsubishi representative.)	RUN: Off ERR.: Flicker CPU Status: Stop	Q4AR
1300	**[FUSE BREAK OFF]** There is an output module with a blown fuse. ■**Collateral informationmmon** • Common Information:Module No.(Slot No.) [For Remote I/O network] Network No./Station No. • Individual Information:– ■**Diagnostic Timing** • Always	• Check ERR. LED of the output modules and replace the fuse of the module whose LED is lit. • Read the common information of the error using the peripheral device and replace the fuse at the output module corresponding to the numerical value (module No.) reading. Alternatively, monitor special registers SD1300 to SD1331 with the peripheral device and change the fuse of the output module whose bit has a value of "1". • When a GOT is bus-connected to the main base unit or extension base unit, check the connection status of the extension cable and the grounding status of the GOT.	RUN: Off/On ERR.: Flicker/On CPU Status: Stop/ Continue[1]	QnA Q4AR
	[FUSE BREAK OFF] • There is an output module with a blown fuse. • External power supply for output load is turned off or disconnected. ■**Collateral informationmmon** • Common Information:Module No.(Slot No.) [For Remote I/O network] Network No./Station No. • Individual Information:– ■**Diagnostic Timing** • Always	• Check ERR. LED of the output modules and replace the module whose LED is lit. • Read the common information of the error using the peripheral device and replace the fuse at the output module corresponding to the numerical value (module No.) reading. Alternatively, monitor special registers SD1300 to SD1331 with the peripheral device and change the fuse of the output module whose bit has a value of "1". • Check whether the external power supply for output load is ON or OFF. • When a GOT is bus-connected to the main base unit or extension base unit, check the connection status of the extension cable and the earth status of the GOT.		Q2AS
1310	**[I/O INT. ERROR]** An interruption has occurred although there is no interrupt module. ■**Collateral informationmmon** • Common Information:– • Individual Information:– ■**Diagnostic Timing** • During interrupt	Any of the mounted modules is experiencing a hardware fault. Therefore, check the mounted modules and change the faulty module. (Contact your local Mitsubishi representative.)	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
1401	**[SP. UNIT DOWN]** When PLC parameter I/O allocation was being made, there was no return signal from the special function module during initial processing stage.(When error is generated, the head I/O number of the special function module that corresponds to the common information is stored.) ■**Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:– ■**Diagnostic Timing** • At power ON/At reset	The CPU module, base unit and/or the special function module that was accessed is experiencing a hardware fault. (Contact your local Mitsubishi representative.)	RUN: Off ERR.: Flicker CPU Status: Stop[2]	

[1] CPU operation can be set in the parameters at error occurrence. (LED indication varies.)

[2] The BAT.ALM LED turns on at BATTERY ERROR.

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
1402	**[SP. UNIT DOWN]** The special function module was accessed during the execution of a FROM/TO instruction set, but there was no response. (When an error is generated, the program error location corresponding to the individual information is stored.) **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:Program error location **■Diagnostic Timing** • During execution of FROM/TO instruction set	The CPU module, base unit and/or the special function module that was accessed is experiencing a hardware fault.(Contact your local Mitsubishi representative.)	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
1411	**[CONTROL-BUS. ERR.]** When performing a parameter I/O allocation the intelligent function module/special function module could not be accessed during initial communications. (On error occurring, the head I/O number of the corresponding intelligent function module/special function module is stored in the common information.) **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:– **■Diagnostic Timing** • At power ON / At reset	Reset the CPU module and RUN it again. If the same error is displayed again, the intelligent function module/special function module, CPU module or base unit is faulty. (Contact your local Mitsubishi representative.)		
1412	**[CONTROL-BUS. ERR.]** The FROM/TO instruction is not executable, due to a control bus error with the intelligent function module/special function module. (On error occurring, the program error location is stored in the individual information.) **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:Program error location **■Diagnostic Timing** • During execution of FROM/TO instruction set			
1421	**[SYS. UNIT DOWN]** Hardware fault at the system management module AS92R. **■Collateral informationmmon** • Common Information:– • Individual Information:– **■Diagnostic Timing** • Always	This suggests a system management module AS92R hardware fault. (Contact your local Mitsubishi representative.)		Q4AR
1500	**[AC/DC DOWN]** • A momentary power supply interruption has occurred. • The power supply went off. **■Collateral informationmmon** • Common Information:– • Individual Information:– **■Diagnostic Timing** • Always	Check the power supply.	RUN: On ERR.: Off CPU Status: Continue	QnA

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
1510	**[DUAL DC DOWN 5V]** The power supply voltage (100 to 240VAC) of either of the two power supply modules on the power supply duplexing extension base unit dropped to or below 85% of the rated voltage. (This can be detected from the control system of the redundant system.) ■**Collateral informationmmon** • Common Information:– • Individual Information:– ■**Diagnostic Timing** • Always	Check the supply voltage of the power supply module. If the voltage is abnormal then replace the power supply module.	RUN: On ERR.: On CPU Status: Continue	Q4AR
1520	**[DC DOWN 5V]** The voltage(100 to 240VAC) of the power supply module on the extension base unit dropped to or below 85% of the rated voltage. (This can be detected from the control system of the stand-alone system or redundant system.) ■**Collateral informationmmon** • Common Information:– • Individual Information:– ■**Diagnostic Timing** • Always	Check the supply voltage of the power supply module. If the voltage is abnormal then replace the power supply module.	RUN: Off ERR.: Flicker CPU Status: Stop	
1530	**[DC DOWN 24V]** The 24 VDC power supplied to the system management module AS92R has dropped below 90% of the rated voltage. (This can be detected from the control system or standby system of the redundant system.) ■**Collateral informationmmon** • Common Information:– • Individual Information:– ■**Diagnostic Timing** • Always	Check the 24VDC power supplied to the system management module AS92R.	RUN: On ERR.: On CPU Status: Continue	
1600	**[BATTERY ERROR[*2]]** • The battery voltage in the CPU module has dropped below stipulated level. • The lead connector of the CPU module battery is not connected. ■**Collateral informationmmon** • Common Information:Drive Name • Individual Information:– ■**Diagnostic Timing** • Always	• Change the battery. • If the battery is for program memory, standard RAM or for the back-up power function, install a lead connector.	RUN: On ERR.: Off CPU Status: Continue	QnA
1601	**[BATTERY ERROR[*2]]** Voltage of the battery on memory card 1 has dropped below stipulated level. ■**Collateral informationmmon** • Common Information:Drive Name • Individual Information:– ■**Diagnostic Timing** • Always	Change the battery.		
1602	**[BATTERY ERROR[*2]]** Voltage of the battery on memory card 2 has dropped below stipulated level. ■**Collateral informationmmon** • Common Information:Drive Name • Individual Information:– ■**Diagnostic Timing** • Always	Change the battery.	RUN: On ERR.: On CPU Status: Continue	

---

*2 The BAT.ALM LED turns on at BATTERY ERROR.

22.3.4 Error code list (2000 to 2999)

The following shows the error messages from the error code 2000 to 2999, the contents and causes of the errors, and the corrective actions for the errors.

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
2000	**[UNIT VERIFY ERR.]** I/O module information power ON is changed. • I/O module (or special function module) not installed properly or installed on the base unit. **■Collateral informationmmon** • Common Information:Module No.(Slot No.) [For Remote I/O network] Network No./Station No. • Individual Information:– **■Diagnostic Timing** • When an END instruction executed	• Read the common information of the error using the peripheral device, and check and/or change the module that corresponds to the numerical value (module number) there. • Alternatively, monitor the special registers SD1400 to SD1431 at a peripheral device, and change the fuse at the output module whose bit has a value of "1". • When a GOT is bus-connected to the main base unit or extension base unit, check the connection status of the extension cable and the grounding status of the GOT.	RUN: Off/On ERR.: Flicker/On CPU Status: Stop/ Continue[*1]	QnA
2100	**[SP. UNIT LAY ERR.]** In PLC parameter I/O allocation settings, a special function module was allocated to a location reserved for an I/O module. Or, the opposite has happened. **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	Reset the PLC parameter I/O allocation setting to conform with the actual status of the special function modules.	RUN: Off ERR.: Flicker CPU Status: Stop	
2101	**[SP. UNIT LAY ERR.]** 13 or more special function modules (not counting the A1SI61) capable of sending an interrupt to the CPU module have been installed. **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	Keep the number of special function modules that can initiate an interrupt (with the exception of the A(1S)I61 module) to 12 or fewer.		
2102	**[SP. UNIT LAY ERR.]** Seven or more serial communication modules (excludes A (1S) J71QC24) have been installed. **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	Keep the number of serial communication modules (excludes A(1S)J71QU24) installed to six or fewer.		
2103	**[SP. UNIT LAY ERR.]** Two or more A (1S) I61 interrupt modules have been mounted. **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	Install only 1 A (1S) I61 module.		

[*1] CPU operation can be set in the parameters at error occurrence. (LED indication varies.)

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
2104	**[SP. UNIT LAY ERR.]** At the MELSECNET/MINI auto refresh network parameter settings, the module allocation that was set is different from the actual module models at the station numbers in the link system. **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	Reset the network parameter MELSECNET/MINI auto refresh unit module allocation setting so that it conforms to the station number of the module that is actually linked.	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
2105	**[SP. UNIT LAY ERR.]** There are too many special function modules that can use dedicated instructions allocated (number of modules installed). (The total of the figures indicated below is above 1344.) (AD59 modules installed × 5) (AD57(S1)/AD58 modules installed × 8) (AJ71C24(S3/S6/S8) modules installed × 10) (AJ71UC24 modules installed × 10) (AJ71C21(S1) modules installed × 29) (AJ71PT32-S3/AJ71T32-S3 modules installed ×125) * (AJ71QC24(R2,R4) modules installed × 29) (AJ71ID1(2)-R4 modules installed × 8) +(AD75 modules installed × 12) total ＞1344 *: When the expansion mode is used. **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	Reduce the number of special function modules installed.		
2106	**[SP.UNIT LAY ERR.]** • Five or more AJ71QLP21 & AJ71QBR11 modules are installed. • Three or more AJ71AP21/R21 & AJ71AT21B modules are installed. • The total number of installed AJ71QLP21, AJ71QBR11, AJ71AP21/R21, and AJ71AT21B modules exceeds five. • The same network numbers or identical station numbers exist in the MELSECNET/10 network system. • Two or more master or load stations exist simultaneously at the MELSECNET(II) or MELSECNET/B data link system. **■Collateral informationmmon** • Common Information:Module No. (Slot No.) • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	• Reduce the AJ71QLP21 and AJ71QBR11 modules to four or less. • Reduce the AJ71AP21/R21 and AJ71AT21B modules to two or less. • Reduce the AJ71QLP21, AJ71QBR11, AJ71AP21/R21 and AJ71AT21B modules to a total of four or less. • Check the network Nos. and station Nos. • Check the station Nos.		
2107	**[SP. UNIT LAY ERR.]** The start X/Y set in the PLC parameter's I/O assignment settings is overlapped with the one for another module. **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	Make the PLC parameter's I/O assignment setting again so it is consistent with the actual status of the special function modules.		

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
2108	**[SP. UNIT LAY ERR.]** A(1S)J71LP21 or A(1S)J71BR11 for use with the AnUCPU network module has been installed. **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	Replace the network module to A(1S)J71QLP21 or A(1S)J71QBR11.	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
2109	**[SP. UNIT LAY ERR.]** The control system and standby system module configurations are different when a redundant system is in the backup mode. **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	Check the module configuration of the standby system.	RUN: Off ERR.: Flicker CPU Status: Stop/ Continue*2	Q4AR
2110	**[SP. UNIT ERROR]** • The location designated by the FROM/TO instruction set is not the special function module. • The module that does not include buffer memory has been specified by the FROM/TO instruction. • The special function module, Network module being accessed is faulty. • Station not loaded was specified using the instruction whose target was the CPU share memory. **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:Program error location **■Diagnostic Timing** • When instruction executed	• Read the individual information of the error using the GX Developer, check the FROM/TO instruction that corresponds to that numerical value (program error location), and correct when necessary. • The special function module that was accessed is experiencing a hardware fault. Therefore, change the faulty module. Alternatively, contact your local Mitsubishi representative.	RUN: Off/On ERR.: Flicker/On CPU Status: Stop/ Continue*1	QnA
2111	**[SP. UNIT ERROR]** • The location designated by a link direct device (J□ \□) is not a network module. • The I/O module (special function module) was nearly removed, completely removed, or mounted during running. **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:Program error location **■Diagnostic Timing** • When instruction executed			
2112	**[SP. UNIT ERROR]** • The module other than special function module is specified by the special function module dedicated instruction. Or, it is not the corresponding special function module. • The module model specified by the special function module dedicated instruction and that specified by the parameter I/O assignment is different. **■Collateral informationmmon** • Common Information:Module No.(Slot No.) • Individual Information:Program error location **■Diagnostic Timing** • When instruction executed/STOP → RUN	• Read the individual information of the error using a peripheral device, and check the special function module dedicated instruction (network instruction) that corresponds to the value (program error part) to make modification. • Set the module model by PLC parameter I/O assignment according to the special function module dedicated instruction setting. Example) Although AJ71QC24N is used actually, AJ71QC24 is set.		

---

*1 CPU operation can be set in the parameters at error occurrence. (LED indication varies.)

*2 The BAT.ALM LED turns on at BATTERY ERROR.

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
2113	**[SP. UNIT ERROR]** Data of special function module to be simulated is not set in the simulation date. **■Collateral informationmmon** • Common Information:FFFFH (fixed) • Individual Information:Program error location **■Diagnostic Timing** • When instruction executed/STOP → RUN	Read the individual information of the error using a peripheral device, and check the special function module /special function module dedicated instruction (network instruction) that corresponds to the value (program error part) to make modification.	RUN: Off/On ERR.: Flicker/On CPU Status: Stop/ Continue[*1]	QnA
2210	**[BOOT ERROR]** There is no boot file in the drive designated by the parameter enabled drive switch even though the Boot DIP switch is ON. **■Collateral informationmmon** • Common Information:Drive name • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	Check and correct the valid parameter drive settings made by the DIP switches. Set the boot file to the drive specified by the parameter drive DIP switches.	RUN: Off ERR.: Flicker CPU Status: Stop	
2300	**[ICM. OPE. ERROR]** • A memory card was removed without switching the memory card in/out switch OFF. • The memory card in/out switch is turned ON although a memory card is not actually installed. **■Collateral informationmmon** • Common Information:Drive name • Individual Information:– **■Diagnostic Timing** • When memory card is inserted or removed/When memory card is inserted	• Remove memory card after placing the memory card in/out switch OFF. • Turn on the card insert switch after inserting a memory card.	RUN: Off/On ERR.: Flicker/On CPU Status: Stop/ Continue[*1]	
2301	**[ICM. OPE. ERROR]** • The memory card has not been formatted. • Memory card format status is incorrect. **■Collateral informationmmon** • Common Information:Drive name • Individual Information:– **■Diagnostic Timing** • When memory card is inserted or removed/When memory card is inserted	• Format memory card. • Reformat memory card.		
2302	**[ICM. OPE. ERROR]** A memory card that cannot be used with the CPU module has been installed. **■Collateral informationmmon** • Common Information:Drive name • Individual Information:– **■Diagnostic Timing** • When memory card is inserted or removed	• Format memory card. • Reformat memory card. • Check memory card.		
2400	**[FILE SET ERROR]** The file designated at the PLC file settings in the parameters cannot be found. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number **■Diagnostic Timing** • At power ON/At reset/ At writing to progurammable controller	• Read the individual information of the error using peripheral device, check to be sure that the parameter drive name and file name correspond to the numerical values there (parameter number), and correct. • Create a file created using parameters, and load it to the CPU module.	RUN: Off ERR.: Flicker CPU Status: Stop	

[*1] CPU operation can be set in the parameters at error occurrence. (LED indication varies.)

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
2401	**[FILE SET ERROR]** The file specified by parameters cannot be made. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number **■Diagnostic Timing** • At power ON/At reset/ At writing to progurammable controller	• Read the individual information of the error using the peripheral device, check to be sure that the parameter drive name and file name correspond to the numerical values there (parameter number), and correct. • Check the space remaining in the memory card.	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
2402	**[FILE SET ERROR]** Though the file register has been set in the pairing setting/tracking setting, the file register does not exist. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number **■Diagnostic Timing** • At power ON/At reset/ At writing to progurammable controller	Confirm the file register and parameter.		Q4AR
2410	**[FILE OPE. ERROR]** • The specified program does not exist in the program memory. This error may occur when the ECALL, EFCALL, PSTOP, PSCAN, POFF or PLOW instruction is executed. • The specified file does not exist. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Program error location **■Diagnostic Timing** • When instruction executed	• Read the individual information of the error using the peripheral device, check to be sure that the program corresponds to the numerical values there (program location), and correct. Create a file created using parameters, and load it to the CPU module. • In case a specified file does not exist, write the file to a target memory and/or check the file specified with the instruction again.	RUN: Off/On ERR.: Flicker/On CPU Status: Stop/ Continue[*1]	QnA
2411	**[FILE OPE. ERROR]** • The file is the one which cannot be specified by the sequence program (such as comment file). • The specified program exists in the program memory, but has not been registered in the program setting of the Parameter dialog box. This error may occur when the ECALL, EFCALL, PSTOP, PSCAN or POFF instruction is executed. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Program error location **■Diagnostic Timing** • When instruction executed	Read the individual information of the error using the peripheral device, check to be sure that the program corresponds to the numerical values there (program location), and correct.		
2412	**[FILE OPE. ERROR]** The SFC program file is one that cannot be designated by the sequence program. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Program error location **■Diagnostic Timing** • When instruction executed	Read the individual information of the error using the peripheral device, check to be sure that the program corresponds to the numerical values there (program location), and correct.		
2413	**[FILE OPE. ERROR]** No data has been written to the file designated by the sequence program. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Program error location **■Diagnostic Timing** • When instruction executed	Read the individual information of the error using the peripheral device, check to be sure that the program corresponds to the numerical values there (program location), and correct. Check to ensure that the designated file has not been write protected.		

[*1] CPU operation can be set in the parameters at error occurrence. (LED indication varies.)

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
2500	**[CAN'T EXE. PRG.]** • There is a program file that uses a device that is out of the range set in the PLC parameter device setting. • After the PLC parameter setting is changed, only the parameter is written into the PLC. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	• Read the common information of the error using the peripheral device, check to be sure that the parameter device allocation setting and the program file device allocation correspond to the numerical values there (file name), and correct if necessary. • If PLC parameter device setting is changed, batch-write the parameter and program file into the PLC.	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
2501	**[CAN'T EXE. PRG.]** There are multiple program files although "none" has been set at the PLC parameter program settings. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	Edit the PLC parameter program setting to "yes". Alternatively, delete unneeded programs.		
2502	**[CAN'T EXE. PRG.]** The program file is incorrect. Alternatively, the file contents are not those of a sequence program. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	Check whether the program version is✻✻✻.QPG, and check the file contents to be sure they are for a sequence program.		
2503	**[CAN'T EXE. PRG.]** There are no program files at all. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:– **■Diagnostic Timing** • At power ON/At reset	• Check program configuration. • Check parameters and program configuration.		
2504	**[CAN'T EXE. PRG.]** Two or more SFC normal programs or control programs have been designated. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:– **■Diagnostic Timing** • At power ON/At reset			

22.3.5 Error code list (3000 to 3999)

The following shows the error messages from the error code 3000 to 3999, the contents and causes of the errors, and the corrective actions for the errors.

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
3000	**[PARAMETER ERROR]** The PLC parameter settings for timer time limit setting, the RUN-PAUSE contact, the common pointer number, general data processing, number of empty slots, system interrupt settings, baud rate setting, and service processing setting are outside the range that can be used by the CPU module. ■**Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number ■**Diagnostic Timing** • At power ON/At reset/STOP → RUN/ At writing to programmable controller	• Read the individual information of the error using the peripheral device, check the parameter item corresponding to the numerical value (parameter No.), and correct it. • Rewrite corrected parameters to the CPU module, reload the CPU power supply and/or reset the module. • If the same error occurs, it is thought to be a hardware error. (Contact your local Mitsubishi representative.)	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
	[PARAMETER ERROR] The parameter settings in the error individual information (special register SD16) are illegal. ■**Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number ■**Diagnostic Timing** • At power ON/At reset/STOP → RUN/ At writing to programmable controller			
3001	**[PARAMETER ERROR]** The parameter settings are corrupted. ■**Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number ■**Diagnostic Timing** • At power ON/At reset/STOP → RUN/ At writing to programmable controller			
3002	**[PARAMETER ERROR]** When "Use the following file" is selected for the file register in the PLC file setting of the PLC parameter dialog box, the specified file does not exist although the file register capacity has been set. ■**Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number ■**Diagnostic Timing** • At power ON/At reset/STOP → RUN/ At writing to programmable controller	• Read the individual information of the error using the peripheral device, check the parameter item corresponding to the numerical value (parameter No.), and correct it. • Rewrite corrected parameters to the CPU module, reload the CPU power supply and/or reset the module. • If the same error occurs, it is thought to be a hardware error. (Contact your local Mitsubishi representative.)		

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
3003	**[PARAMETER ERROR]** **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number **■Diagnostic Timing** • When an END instruction executed	• Read the individual information of the error using the peripheral device, check the parameter item corresponding to the numerical value (parameter No.), and correct it. • If the error is still generated following the correction of the parameter settings, the possible cause is the memory errorm of the CPU module's built-in RAM or program memory or the memory card. (Contact your local Mitsubishi representative.)	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
	[PARAMETER ERROR] The number of devices set at the PLC parameter device settings exceeds the possible CPU module range. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number **■Diagnostic Timing** • At power-On/At reset/STOP → RUN/ At writing to progurammable controller			
3004	**[PARAMETER ERROR]** The parameter file is incorrect. Alternatively, the contents of the file are not parameters. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number **■Diagnostic Timing** • At power-On/At reset/STOP → RUN/ At writing to progurammable controller	Check whether the parameter file version is＊＊＊.QPA, and check the file contents to be sure they are parameters.		
3100	**[LINK PARA. ERROR]** Although the QnACPU is a control station or master station, the network parameters have not been written. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number **■Diagnostic Timing** • At power ON/At reset/STOP → RUN	• Correct and write the network parameters. • If the error occurs after correction, it suggests a hardware fault. (Contact your local Mitsubishi representative.)		
3101	**[LINK PARA. ERROR]** • The network No. specified by a network parameter is different from that of the actually mounted network. • The head I/O No. specified by a network parameter is different from that of the actually mounted I/O unit. • The network class specified by a network parameter is different from that of the actually mounted network. • The network refresh parameter of the MELSECNET/H, MELSECNET/10 is out of the specified area. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number **■Diagnostic Timing** • At power ON/At reset/STOP → RUN	• Check the network parameters and mounting status, and if they differ, match the network parameters and mounting status. If any network parameter has been corrected, write it to the CPU module. • Confirm the setting of the number of extension stages of the extension base units. • Check the connection status of the extension base units and extension cables. When the GOT is bus-connected to the main base unit and extension base units, also check their connection status. If the error occurs after the above checks, the cause is a hardware fault. (Contact your local Mitsubishi representative, explaining a detailed description of the problem.)		
3102	**[LINK PARA. ERROR]** • The network module detected a network parameter error. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number **■Diagnostic Timing** • At power ON/At reset/STOP → RUN[*3]	• Correct and write the network parameters. • If the error occurs after correction, it suggests a hardware fault. (Contact your local Mitsubishi representative.)		

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
3103	**[LINK PARA. ERROR]** • Although the number of modules has been set to one or greater number in the Ethernet network parameter setting, the number of actually mounted module is zero. • The start I/O No. of the Ethernet network parameter differs from the I/O No. of the actually mounted module. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number **■Diagnostic Timing** • At power ON/At reset/STOP → RUN	• Correct and write the network parameters. • If the error occurs after correction, it suggests a hardware fault. (Contact your local Mitsubishi representative.)	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
	[LINK PARA. ERROR] • AJ71QE71 does not exist in the position of I/O number set by the parameter. • I/O number designation is overlapping. • Numbers of the network parameter and loaded AJ71QE71 are different. • Ethernet (parameter + dedicated instruction) is set to more than five. **■Collateral informationmmon** • Common Information:File name/Drive name • Individual Information:Parameter number **■Diagnostic Timing** • At power ON/At reset/STOP → RUN			
3104	**[LINK PARA. ERROR]** • The Ethernet and MELSECNET/10 use the same network number. • The network number, station number or group number set in the network parameter is out of range. • The specified I/O number is outside the range of the used CPU module. • The Ethernet-specific parameter setting is not normal. **■Collateral informationmmon** • Common Information:File name / Drive name • Individual Information:Parameter number **■Diagnostic Timing** • At power ON/At reset/STOP → RUN			
3105	**[LINK PARA. ERROR]** The contents of the Ethernet parameter are incorrect. **■Collateral informationmmon** • Common Information:File name / Drive name • Individual Information:Parameter number **■Diagnostic Timing** • At power ON/At reset/STOP → RUN	Write after correcting parameters.		
3107	**[LINK PARA. ERROR]** • The CC-Link parameter setting is incorrect. • The set mode is not allowed for the version of the mounted CC-Link module. **■Collateral informationmmon** • Common Information:File name • Individual Information:Parameter number **■Diagnostic Timing** • At power ON/At reset/STOP → RUN	Check the parameter setting.		

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
3200	**[SFC PARA. ERROR]** The parameter setting is illegal. • Though Block 0 was set to "Automatic start" in the SFC setting of the PLC parameter dialog box, Block 0 does not exist. **■Collateral informationmmon** • Common Information:File name • Individual Information:Parameter number **■Diagnostic Timing** • STOP → RUN	Read the common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
3201	**[SFC PARA. ERROR]** The block parameter setting is illegal. **■Collateral informationmmon** • Common Information:File name • Individual Information:Parameter number **■Diagnostic Timing** • STOP → RUN			
3202	**[SFC PARA. ERROR]** The number of step relays specified in the device setting of the PLC parameter dialog box is less than that used in the program. **■Collateral informationmmon** • Common Information:File name • Individual Information:Parameter number **■Diagnostic Timing** • STOP → RUN			
3203	**[SFC PARA. ERROR]** The execution type of the SFC program specified in the program setting of the PLC parameter dialog box is other than scan execution. **■Collateral informationmmon** • Common Information:File name • Individual Information:Parameter number **■Diagnostic Timing** • At power ON/At reset/STOP → RUN			

---

*3 The diagnostic timing of CPU modules except for Universal QCPU can be performed only when switching the CPU modules to run.

22.3.6 Error code list (4000 to 4999)

The following shows the error messages from the error code 4000 to 4999, the contents and causes of the errors, and the corrective actions for the errors.

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
4000	**[INSTRCT. CODE ERR]** • The program contains an instruction code that cannot be decoded. • An unusable instruction is included in the program. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • At power ON/At reset/STOP → RUN When instruction executed	Read the common information of the error using a peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
4001	**[INSTRCT. CODE ERR]** The program contains a dedicated instruction for SFC although it is not an SFC program. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • At power ON/At reset/STOP → RUN When instruction executed			
4002	**[INSTRCT. CODE ERR]** • The name of dedicated instruction specified by the program is incorrect. • The dedicated instruction specified by the program cannot be executed by the specified module. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • At power ON/At reset/STOP → RUN When instruction executed			
4003	**[INSTRCT. CODE ERR]** The number of devices for the dedicated instruction specified by the program is incorrect. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • At power ON/At reset/STOP → RUN When instruction executed			
4004	**[INSTRCT. CODE ERR]** The device which cannot be used by the dedicated instruction specified by the program is specified. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • At power ON/At reset/STOP → RUN When instruction executed			

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
4010	**[MISSING END INS.]** There is no END (FEND) instruction in the program. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • At power ON/At reset/STOP → RUN	Read the common information of the error using a peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
4020	**[CAN'T SET(P)]** The total number of internal file pointers used by the program exceeds the number of internal file pointers set in the parameters. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • At power ON/At reset/STOP → RUN			
4021	**[CAN'T SET(P)]** • The common pointer Nos. assigned to files overlap. • The local pointer Nos. assigned to files overlap. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • At power ON/At reset/STOP → RUN			
4030	**[CAN'T SET(I)]** The allocation pointer Nos. assigned by files overlap. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • At power ON/At reset/STOP → RUN			
4100	**[OPERATION ERROR]** The instruction cannot process the contained data. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Read the common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off/On ERR.: Flicker/On CPU Status: Stop/ Continue[*1]	
4101	**[OPERATION ERROR]** • The number of setting data dealt with the instruction exceeds the applicable range. • The storage data and constant of the device specified by the instruction exceeds the applicable range. • When writing to the host CPU shared memory, the write prohibited area is specified for the write destination address. • The range of storage data of the device specified by the instruction is duplicated. • The device specified by the instruction exceeds the range of the number of device points. • The interrupt pointer No. specified by the instruction exceeds the applicable range. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed			

[*1] CPU operation can be set in the parameters at error occurrence. (LED indication varies.)

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
4102	**[OPERATION ERROR]** • The network No. or station No. specified for the dedicated instruction is wrong. • The link direct device (J□\□) setting is incorrect. • The module No./ network No./number of character strings exceeds the range that can be specified. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Read the common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off/On ERR.: Flicker/On CPU Status: Stop/ Continue[*1]	QnA
4103	**[OPERATION ERROR]** The configuration of the PID dedicated instruction is incorrect. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed			
4104	**[OPERATION ERROR]** The number of settings is beyond the range. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Read the common information of the error using peripheral device, and check and correct the program corresponding to that value (program error location).		Q4AR
4107	**[OPERATION ERROR]** Numbers of execution to the CC-Link instruction are beyond 32. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Set the numbers of execution to the CC-Link instruction to 32 or less.		QnA
4108	**[OPERATION ERROR]** The CC-Link parameter is not set when the CC-Link instruction is executed. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Execute the CC-Link instruction after setting the CC-Link parameter.		
4200	**[FOR NEXT ERROR]** No NEXT instruction was executed following the execution of a FOR instruction. Alternatively, there are fewer NEXT instructions than FOR instructions. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Read the common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off ERR.: Flicker CPU Status: Stop	

[*1] CPU operation can be set in the parameters at error occurrence. (LED indication varies.)

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
4201	**[FOR NEXT ERROR]** A NEXT instruction was executed although no FOR instruction has been executed. Alternatively, there are more NEXT instructions than FOR instructions. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Read the common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
4202	**[FOR NEXT ERROR]** More than 16 nesting levels are programmed. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Keep nesting levels at 16 or under.		
4203	**[FOR NEXT ERROR]** A BREAK instruction was executed although no FOR instruction has been executed prior to that. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Read the common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.		
4210	**[CAN'T EXECUTE(P)]** The CALL instruction is executed, but there is no subroutine at the specified pointer. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed			
4211	**[CAN'T EXECUTE(P)]** There was no RET instruction in the executed subroutine program. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed			
4212	**[CAN'T EXECUTE(P)]** The RET instruction exists before the FEND instruction of the main routine program. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed			
4213	**[CAN'T EXECUTE(P)]** More than 16 nesting levels are programmed. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Keep nesting levels at 16 or under.		

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
4220	**[CAN'T EXECUTE(I)]** Though an interrupt input occurred, the corresponding interrupt pointer does not exist. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Read the common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
4221	**[CAN'T EXECUTE(I)]** An IRET instruction does not exist in the executed interrupt program. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed			
4223	**[CAN'T EXECUTE(I)]** The IRET instruction exists before the FEND instruction of the main routine program. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed			
4230	**[INST. FORMAT ERR.]** The number of CHK and CHKEND instructions is not equal. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed			
4231	**[INST. FORMAT ERR.]** The number of IX and IXEND instructions is not equal. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed			
4235	**[INST. FORMAT ERR.]** The configuration of the check conditions for the CHK instruction is incorrect. Alternatively, a CHK instruction has been used in a low speed execution type program. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed			
4300	**[EXTEND INST. ERR.]** The designation of a MELSECNET/MINI-S3 master module control instruction was wrong. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Read the common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off/On ERR.: Flicker/On CPU Status: Stop/ Continue[*1]	

---

*1 CPU operation can be set in the parameters at error occurrence. (LED indication varies.)

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
4301	**[EXTEND INST. ERR.]** The designation of an AD57/AD58 control instruction was wrong. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • When instruction executed	Read the common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off/On ERR.: Flicker/On CPU Status: Stop/ Continue[*1]	QnA
4400	**[SFCP. CODE ERROR]** No SFCP or SFCPEND instruction in SFC program. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • STOP → RUN	Write the program to the CPU module again using GX Developer.		
4410	**[CAN'T SET(BL)]** The block number designated by the SFC program exceeds the range. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • STOP → RUN			
4411	**[CAN'T SET(BL)]** Block number designations overlap in SFC program. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • STOP → RUN			
4420	**[CAN'T SET(S)]** A step number designated in an SFC program exceeds the range. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • STOP → RUN			
4421	**[CAN'T SET(S)]** Total number of steps in all SFC programs exceed the maximum. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • STOP → RUN			
4422	**[CAN'T SET(S)]** Step number designations overlap in SFC program. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • STOP → RUN			

[*1] CPU operation can be set in the parameters at error occurrence. (LED indication varies.)

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
4500	**[SFCP. FORMAT ERR.]** The numbers of BLOCK and BEND instructions in an SFC program are not equal. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • STOP → RUN	Write the program to the CPU module again using the peripheral device.	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
4501	**[SFCP. FORMAT ERR.]** The configuration of the STEP* to TRAN* to TSET to SEND instructions in the SFC program is incorrect. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • STOP → RUN			
4502	**[SFCP. FORMAT ERR.]** The structure of the SFC program is illegal. • STEPI* instruction does not exist in the block of the SFC program. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • STOP → RUN			
4503	**[SFCP. FORMAT ERR.]** The structure of the SFC program is illegal. • The step specified in the TSET instruction does not exist. • In jump transition, the host step number was specified as the destination step number. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • STOP → RUN	• Write the program to the CPU module again using GX Developer. • Read the common information of the error using GX Developer, and check and correct the error step corresponding to that value (program error location).		
4504	**[SFCP. FORMAT ERR.]** The structure of the SFC program is illegal. • The step specified in the TAND instruction does not exist. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • STOP → RUN	Write the program to the CPU module again using GX Developer.		
4600	**[SFCP. OPE. ERROR]** The SFC program contains data that cannot be processed. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • When instruction executed	Read common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off/On ERR.: Flicker/On CPU Status: Stop/ Continue[*1]	

[*1] CPU operation can be set in the parameters at error occurrence. (LED indication varies.)

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
4601	**[SFCP. OPE. ERROR]** Exceeds device range that can be designated by the SFC program. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • When instruction executed	Read common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off/On ERR.: Flicker/On CPU Status: Stop/ Continue[*1]	QnA
4602	**[SFCP. OPE. ERROR]** The START instruction in an SFC program is preceded by an END instruction. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • When instruction executed			
4610	**[SFCP. EXE. ERROR]** The active step information at presumptive start of the SFC program is incorrect. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • STOP → RUN	Read common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem. The program is automatically subjected to an initial start.	RUN: On ERR.: On CPU Status: Continue	
4611	**[SFCP. EXE. ERROR]** Key-switch was reset during RUN when presumptive start was designated for SFC program. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • STOP → RUN			
4620	**[BLOCK EXE. ERROR]** Startup was executed at a block in the SFC program that was already started up. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • When instruction executed	Read common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off ERR.: Flicker CPU Status: Stop	
4621	**[BLOCK EXE. ERROR]** Startup was attempted at a block that does not exist in the SFC program. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:– ■**Diagnostic Timing** • When instruction executed	• Read the common information of the error using GX Developer, and check and correct the error step corresponding to that value (program error location). • Turn ON if the special relay SM321 is OFF.		

[*1] CPU operation can be set in the parameters at error occurrence. (LED indication varies.)

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
4630	**[STEP EXE. ERROR]** Startup was executed at a block in the SFC program that was already started up. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Read common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
4631	**[STEP EXE. ERROR]** • Startup was attempted at the step that does not exist in the SFC program. Or, the step that does not exist in the SFC program was specified for end. • Forced transition was executed based on the transition condition that does not exit in the SFC program. Or, the transition condition for forced transition that does not exit in the SFC program was canceled. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	• Read the common information of the error using the peripheral device, and check and correct the error step corresponding to that value (program error location). • Turn ON if the special relay SM321 is OFF.		
4632	**[STEP EXE. ERROR]** There were too many simultaneous active steps in blocks that can be designated by the SFC program. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed	Read common information of the error using the peripheral device, check error step corresponding to its numerical value (program error location), and correct the problem.		
4633	**[STEP EXE. ERROR]** There were too many simultaneous active steps in all blocks that can be designated. **■Collateral informationmmon** • Common Information:Program error location • Individual Information:– **■Diagnostic Timing** • When instruction executed			

22.3.7 Error code list (5000 to 5999)

The following shows the error messages from the error code 5000 to 5999, the contents and causes of the errors, and the corrective actions for the errors.

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
5000	**[WDT ERROR]** • The scan time of the initial execution type program exceeded the initial execution monitoring time specified in the PLC RAS setting of the PLC parameter. **■Collateral informationmmon** • Common Information:Time (value set) • Individual Information:Time (value actually measured) **■Diagnostic Timing** • Always	• Read the individual information of the error from the peripheral device, check its value (time), and shorten the scan time. • Change the initial execution monitoring time or the WDT value in the PLC RAS setting of the PLC parameter. • Resolve the endless loop caused by jump transition.	RUN: Off ERR.: Flicker CPU Status: Stop	QnA
5001	**[WDT ERROR]** • The scan time of the program exceeded the WDT value specified in the PLC RAS setting of the PLC parameter. **■Collateral informationmmon** • Common Information:Time (value set) • Individual Information:Time (value actually measured) **■Diagnostic Timing** • Always	• Read the individual information of the error from the peripheral device, check its value (time), and shorten the scan time. • Change the initial execution monitoring time or the WDT value in the PLC RAS setting of the PLC parameter. • Resolve the endless loop caused by jump transition. • Check the number of interrupt program executions with the peripheral device and reduce the number of interrupts.		
5010	**[PRG. TIME OVER]** The program scan time exceeded the constant scan setting time specified in the PLC RAS setting of the PLC parameter. **■Collateral informationmmon** • Common Information:Time (value set) • Individual Information:Time (value actually measured) **■Diagnostic Timing** • Always	• Review the constant scan setting time. • Review the constant scan setting time and low speed program execution time in the PLC parameter so that the excess time of constant scan can be fully secured.	RUN: On ERR.: On CPU Status: Continue	
	[PRG. TIME OVER] The low speed program execution time specified in the PLC RAS setting of the PLC parameter exceeded the excess time of the constant scan. **■Collateral informationmmon** • Common Information:Time (value set) • Individual Information:Time (value actually measured) **■Diagnostic Timing** • Always			
5011	**[PRG. TIME OVER]** The scan time of the low speed execution type program exceeded the low speed execution watch time specified in the PLC RAS setting of the PLC parameter dialog box. **■Collateral informationmmon** • Common Information:Time (value set) • Individual Information:Time (value actually measured) **■Diagnostic Timing** • Always	Read the individual information of the error using the peripheral device, check the numerical value (time) there, and shorten scan time if necessary. Change the low speed execution watch time in the PLC RAS setting of the PLC parameter dialog box.		

22.3.8 Error code list (6000 to 6999)

The following shows the error messages from the error code 6000 to 6999, the contents and causes of the errors, and the corrective actions for the errors

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
6000	**[PRG. VERIFY ERR.]** The control system and standby system in the redundant system do not have the same programs and parameters. (This can be detected from the standby system of the redundant system.) **■Collateral informationmmon** • Common Information:File name • Individual Information:– **■Diagnostic Timing** • Always	Synchronise the programs and parameters of the control system and standby system.	RUN: Off ERR.: Flicker CPU Status: Stop	Q4AR
6010	**[MODE. VERIFY ERR.]** The operational status of the control system and standby system in the redundant system is not the same. (This can be detected from the standby system of the redundant system.) **■Collateral informationmmon** • Common Information:– • Individual Information:– **■Diagnostic Timing** • Always	Synchronise the operation statuses of the control system and standby system.	RUN: On ERR.: On CPU Status: Continue	
6100	**[TRUCKINERR.]** A CPU module tracking memory error was detected during initial. (This can be detected from the control system or standby system of the redundant system.) **■Collateral informationmmon** • Common Information:– • Individual Information:– **■Diagnostic Timing** • At power ON/At reset/STOP → RUN	Hardware fault of the CPU module. (Please contact your local nearest Mitsubishi or sales representative, explaining a detailed description of the problem. Change the CPU modules in order of the standby system CPU module and control system CPU module.)		
6101	**[TRUCKIN ERR.]** The CPU module detected an error during the handshake for tracking. (This can be detected from the control system or standby system of the redundant system.) **■Collateral informationmmon** • Common Information:– • Individual Information:– **■Diagnostic Timing** • When an END instruction executed	Check the condition of the other stations.		
6200	**[CONTROL EXE.]** The standby system in a redundant system is switched to the control system. (This can be detected from the standby system of the redundant system.) **■Collateral informationmmon** • Common Information:Reason(s) for system switching • Individual Information:– **■Diagnostic Timing** • Always	Check the control system condition.	RUN: On ERR.: Off CPU Status: Continue	

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
6210	**[CONTROL WAIT]** The control system in a redundant system is switched to the standby system. (This can be detected from the standby system of the redundant system.) ■**Collateral informationmmon** • Common Information:Reason(s) for system switching • Individual Information:– ■**Diagnostic Timing** • Always	Check the control system condition.	RUN: On ERR.: Off CPU Status: Continue	Q4AR
6220	**[CAN'T EXE. CHANGE]** • Since the standby system is in an error or similar status in the redundant system, the control system cannot be switched to the standby system. • When an attempt was made to execute system switching, the control system could not be switched to the standby system due to a network error of the control system. (This can be detected from the control system of the redundant system.) ■**Collateral informationmmon** • Common Information:Reason(s) for system switching • Individual Information:– ■**Diagnostic Timing** • At switching request	Check the standby system condition.	RUN: On ERR.: On CPU Status: Continue	
6221	**[CAN'T EXE. CHANGE]** Switching is disabled because of a bus switching module error. (This can be detected from the control system of the redundant system.) ■**Collateral informationmmon** • Common Information:Reason(s) for system switching • Individual Information:– ■**Diagnostic Timing** • At switching request	This is a bus switching module hardware fault. (Contact your local Mitsubishi representative.)		
6230	**[DUAL SYS. ERROR]** The link module mounted on the standby system CPU module is the remote master station. ■**Collateral informationmmon** • Common Information:– • Individual Information:– ■**Diagnostic Timing** • Always	Check the system configuration status.		

22.3.9 Error code list (7000 to 10000)

The following shows the error messages from the error code 7000 to 10000, the contents and causes of the errors, and the corrective actions for the errors.

Error Code	Error Contents and Cause	Corrective Action	LED Status CPU Status	Corresponding CPU
9000	**[F****]** Annunciator (F) was set ON ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:Annunciator number ■**Diagnostic Timing** • When instruction executed	Read the individual information of the error using the peripheral device, and check the program corresponding to the numerical value (annunciator number).	RUN: On ERR.: On/Off [*2] CPU Status: Continue	QnA
			RUN: USER LED On ERR.: USER LED On CPU Status: Continue	
9010	**[<CHK>ERR ***-***]** Error detected by the CHK instruction. ■**Collateral informationmmon** • Common Information:Program error location • Individual Information:Failure No. ■**Diagnostic Timing** • When instruction executed	Read the individual information of the error using the peripheral device, and check the program corresponding to the numerical value (error number) there.	RUN: On ERR.: Off CPU Status: Continue	
			RUN: USER LED On ERR.: USER LED On CPU Status: Continue	

---

*2 For the Basic model QCPU, the special register (SD207 to DS209) for LED indication priority can turn off the indication. (The LED indication is always OFF for the High Performance model QCPU, Process CPU, Redundant CPU, and Universal model QCPU.)

22.3.10 Canceling of Errors

Q series CPU module can perform the cancel operation for errors only when the errors allow the CPU module to continue its operation.
To cancel the errors, follow the steps shown below.

1) Eliminate the cause of the error.
2) Store the error code to be canceled in the special register SD50.
3) Energize the special relay SM50 (OFF → ON).
4) The error to be canceled is canceled.

After the CPU module is reset by the canceling of the error, the special relays, special registers, and LEDs associated with the error are returned to the status under which the error occurred.
If the same error occurs again after the cancellation of the error, it will be registered again in the error history.

When multiple enunciators(F) detected are canceled, the first one with No. F only is canceled.

Refer to the following manual for details of error canceling.
→ QCPU User's Manual (Function Explanation, Program Fundamentals)

POINT

(1) When the error is canceled with the error code to be canceled stored in the SD50, the lower one digit of the code is neglected.
(Example)
If error codes 2100 and 2101 occur, and error code 2100 to cancel error code 2101.
If error codes 2100 and 2111 occur, error code 2111 is not canceled even if error code 2100 is canceled.

(2) Errors developed due to trouble in other than the CPU module are not canceled even if the special relay (SM50) and special register (SD50) are used to cancel the error.
(Example)
Since "SP. UNIT DOWN" is the error that occurred in the base unit (including the extension cable), intelligent function module, etc. the error cause cannot be removed even if the error is canceled by the special relay (SM50) and special register (SD50).
Refer to the error code list and remove the error cause.

22.4 Resetting Errors

The CPU module allows error resetting only for the errors that does not block the CPU module operation.

The procedure for resetting an error is as follows.

1) Eliminate the cause of the error.
2) Store the error code to be reset to special register SD50.
3) Switch special relay SM50 from OFF to ON.
4) The error is reset.

If the CPU module is returned with the error reset, the special relay and special register relating to the error, and the LED/LED indicator return to their state before the error occurred.

If the same error occurs again after the error has been reset, it is recorded in breakdown history again.

To reset multiple detected annunciators, only the first detected F number is reset.

POINT

When storing the error code to be reset in SD50 at error reset, the lower one digit of the code number is ignored.

Example:

When error codes 2100 and 2101 occurred, resetting of error code 2100 results in also resetting of error code 2101.

When error codes 2100 and 2111 occurred, resetting of error code 2100 does not result in resetting of error code 2111.

22.5 Fault Examples with I/O Modules

Examples of faults concerning I/O circuits and the corrective actions are explained.

22.5.1 Faults with the input circuit and the corrective actions

The following is an input-circuit fault example and its corrective action.

Table 22.2 Faults with the input circuit and the corrective actions

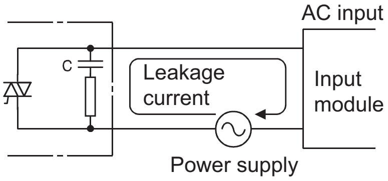
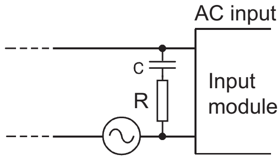
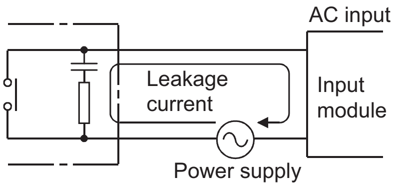
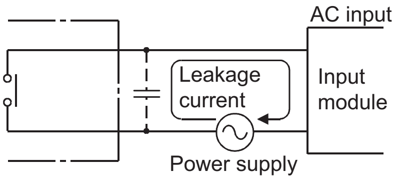
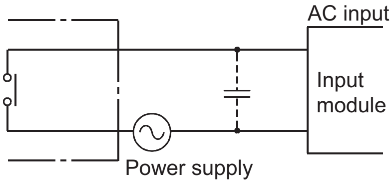
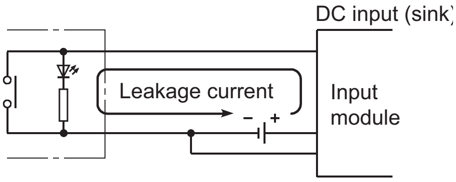
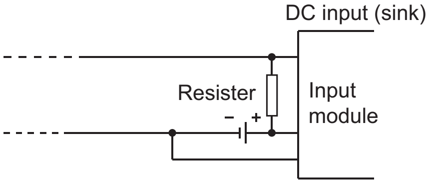
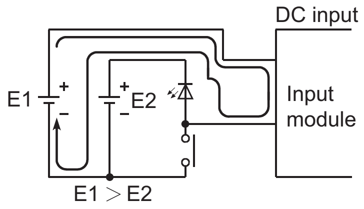
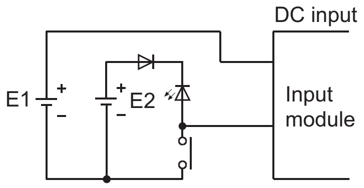
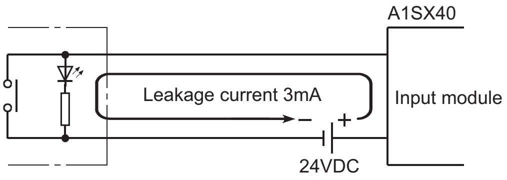
	Situation	Cause	Countermeasure
Example 1	The input signal does not turn off.	• Leakage current of input switch (driven by contactless switch, etc.)  	• Connect the proper resistor so that the input modules terminal to terminal voltage is under the OFF voltage value.  Recommend μ0.1 to 0.47μF + 47 to 120Ω (1/2W) for CRs constant.
Example 2	The input signal does not turn off.	• Drive by limit switch with neon lamp  	• Same as the example 1. • Or, provide a totally independent display circuit separately.
Example 3	The input signal does not turn off.	• Line capacity C of the leak current twisted pair cable due to line capacity of the wiring cable is about 100PF/m.  	• Same as the example 1. • However, leakage current does not occur when a power supply is on the side of input device as shown below. 
Example 4	The input signal does not turn off.	• Driven by a switch with LED indication  	• Connect an appropriate resistance so that voltage between the terminal of the input module and the common is lower than the OFF voltage as shown below.  * An example of calculation of resistor to be connected is provided on the following page.

Table 22.2 Faults with the input circuit and the corrective actions

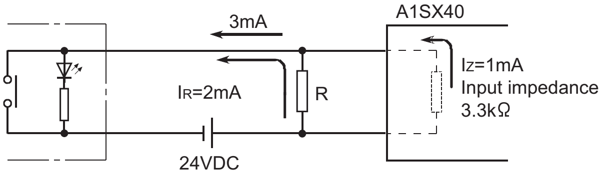
	Situation	Cause	Countermeasure
Example 5	The input signal does not turn off.	• Sneak path due to the use of two power supplies.  	• Use only one power supply. • Connect revolving path preventive diode (figure below). 

<Sample calculation for Example 4>

When a switch with LED indicator, giving leaking current of 3mA at maximum when 24VDC power is supplied to the A1SX40.



(1) 1mA or less OFF current of the A1SX40 is not satisfied. Hence, connect a resistor as shown below.



(2) Calculate the resistance value R as shown below.
To satisfy 1mA or less OFF current of the A1SX40, connect a resistor which flows 2mA or more.
$I_R : I_Z = Z(\text{Input impedance}) : R$

$$R < \frac{I_Z}{I_R} \times Z(\text{Input impedance}) = \frac{1.0}{2.0} \times 3.3 = 1.65[\text{k}\Omega]$$

Supposing that the resistance R is 1.5kΩ , the power capacity W of resistor R is:
$W = (\text{Input voltage})^2/R = 26.4^2/1500 = 0.465$ [W]

(3) Connect a resistor of 1.5(kΩ) and 2 to 3(W) to a terminal which may cause an error, since the power capacity of a resistor is selected so that it will be 3 to 5 times greater than the actual power consumption.

(4) Also,OFF voltage when resistor R is connected will be as follows.

$$\frac{1}{\frac{1}{1.5[\text{k}\Omega]} + \frac{1}{3.3[\text{k}\Omega]}} \times 3[\text{mA}] = 3.09[\text{V}]$$

- This satisfies 4V or less OFF voltage of A1SX40.

22.5.2 Faults in the output circuit

Faults concerning output circuits and the corrective actions are explained.

Table22.3 Faults in the output circuit

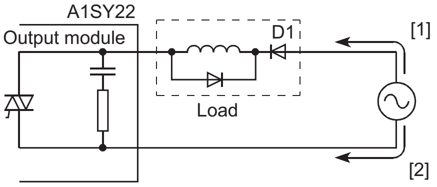
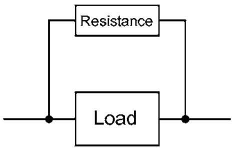
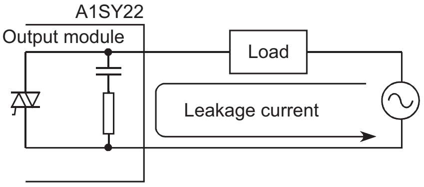
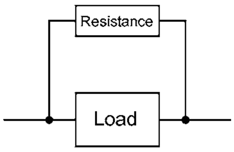
	Situation	Cause	Countermeasure
Example 1	An excessive voltage is applied to the load when output is off. (Triac output)	When the load is subjected to half wave rectification inside (Solenoids have these types.)  • When the polarity of the power supply is (1), C is charged, and when the polarity is (2), the voltage charged in C + voltage of the power supply are applied to the both ends of D1. The maximum value of the voltage is about 2.2E.	• Connect a resistor at several tensΩ to several hundred of kΩ to the both ends of the load. (With this kind of usage, there is no problem with the output element, but the diode built-in to the load may deteriorate and burn-out.) 
Example 2	Load does not turn OFF. (Triac output)	• Leak current caused by built-in noise supressor 	• Connect a resistor to the both ends of the load. (If the wire distance from the output module to the load is great, then it may have leakage current by line capacity. It is necessary to take precautions.) 

Table22.3 Faults in the output circuit

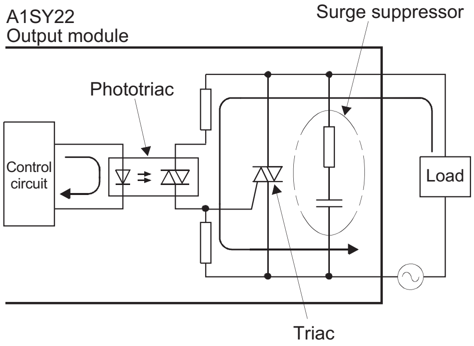
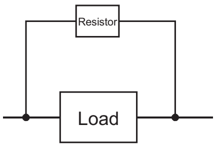
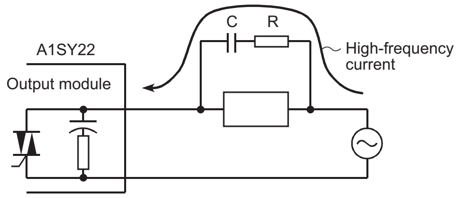
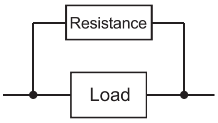
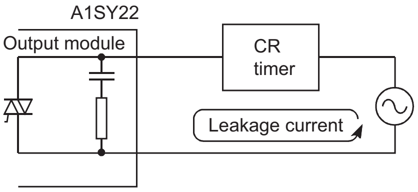
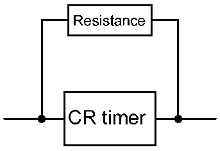
	Situation	Cause	Countermeasure
Example 3	The load is not turned OFF. (Triac output)	• The load current is lower than the minimum load current.  • When the load current is lower than the minimum load current of the output module, the triac does not operate since the load current flows into a phototriac as shown below. When an inductive load is connected, the load may not be turned OFF since surge at the time of OFF is applied to the phototriac.	• Connect a resistor to both ends of a load so that the load current is higher than the minimum load current. 
Example 4	Load turns OFF with a delay. (Triac output)	• Leakage current due to surge suppressor for the load. 	• Disconnect the surge suppressor from across the loads, leaving only the resistance. (If the wire distance from the output module to the load is great, then it may have leakage current by line capacity. It is necessary to take precautions.)  Guideline of resistance. For 100VAC 5 to 10KΩ, 5 to 3W For 200VAC 10 to 20KΩ, 15 to 10W
Example 5	When load is CR type timer, the time limit fluctuates. (Triac output)		• Connect a resistance between the CR timer terminals. (In some timers, internal circuit may be half wave rectification type, so the caution as to the example 1 is necessary here.) (If the wire distance from the output module to the load is great, then it may have leakage current by line capacity. It is necessary to take precautions.)  The constant value of resistor should be calculated according to the load.

Table22.3 Faults in the output circuit

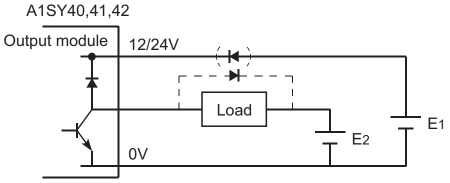
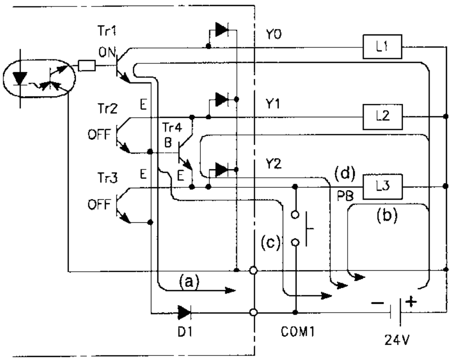
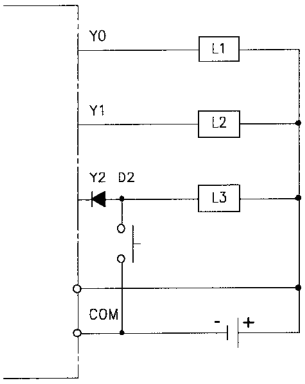
	Situation	Cause	Countermeasure
Example 6	Load does not turn OFF. (Transistor output with clamp diode)	• Sneak path due to the use of two power supplies.  • Sneak path occurs when E1 < E2.	• Use only one power supply. • Connect a diode for a sneak path. (When a relay or similar load is used, a free-wheel diode must be connected across the load.) (Shown by dotted line in the figure at left)
Example 7	When an external switch is connected in parallel between the output and common, the voltage between Y1 and COM1 drops to between 0 and 24V even though the output Y1 which is not connected to the external switch is OFF. Especially when the load L2 is relatively small, (Load current of several mA only) such as LED lamps and photocouplers, the outputs drop. A1SY40 A1SY41 A1SY42	Incorrect output by parasitic transistor (Tr4)  Y2 can turn the load L3 on either from a PC or PB. When PB is ON, Y0 is ON with a PC, and Y1 is OFF: (1) L1(current (a)) and L3 (current (b)) turn ON. (2) A potential difference to COM1occurs in the emitter E of Tr1 to Tr3 since diode D1 is connected between COM1 and the emitter. (3) The transistors A1SY40 to 42, etc., are accompanied by a parasitic transistor (Tr4). (4) The potential difference described in (2) above is supplied between the base (B) of Tr4 and emitter (E), which causes the base current (c) to flow.(Tr4 turns ON.) (5) The current in (4) causes the collector current (d) to flow, and voltage Y1 drops to between 0 and 24V.	 As shown above, connect diode D2 of I_F = 1A class to output Y2 which is connected to an external switch. (This revents currents (3) and (4) from flowing.) However, check the operation voltage of L3 as the amount of voltage drop of Y2 at power ON increases for 0.6 to 1V.

Table22.3 Faults in the output circuit

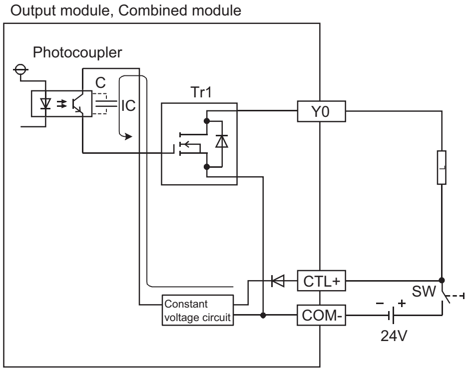
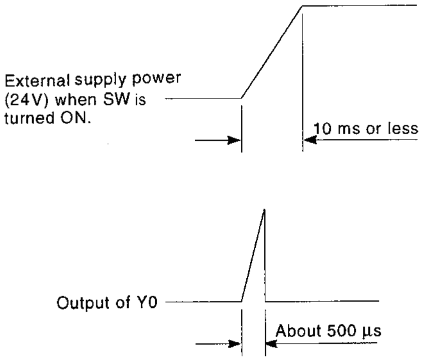
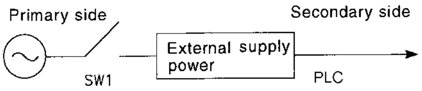
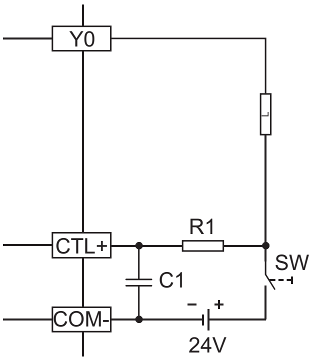
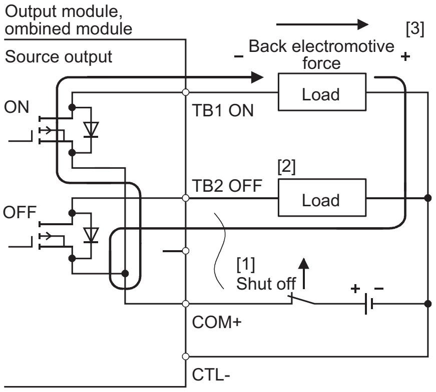
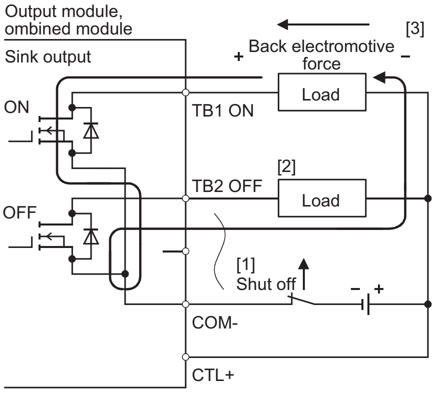
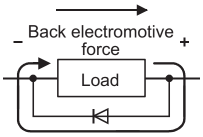
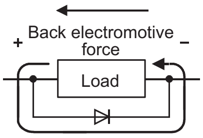
	Situation	Cause	Countermeasure
Example 8	The load is momentarily turned ON when the external supply power is started up. (Transistor output)	Incorrect output due to the floating capacitance (C) between the collector and emitter of the photocoupler (This does not affect normal loads, but in case of highly sensitive loads (such as solid state relays), incorrect outputs may occur.) Photocoupler  (1) If the external supply power is suddenly started up, current Ic flows due to the floating capacitance (C) between the collector and emitter of the photocoupler. (2) Current Ic flows to the base of transistor Tr1in the next stage, and output Y0 turns ON for about 500μs.  	(1) After checking the external supply power takes at least 10ms to start up when turned it ON/OFF, set the switch SW1 at the primary side of the external supply power.  (2) If setting the switch at the secondary side of the external supply power is required, connect a capacitor and resistor so that the start-up of the power is slowly performed (Longer than 10ms).  R1: Number + Ω *1 *2 Power capacity ≧ (External supply power current)^2 × Resistive value × (3 to 5) C1: Several hundreds of μF, 50 mV *1 For the current consumption of the external supply power, refer to the manual attached to the module to be used. *2 Select the resistance for power capacity in the range of between 3 and 5 times higher than the actual power consumption. Example: R1 = 40 Ω , C1 = 300 μF Calculate the time constant as follows: $C1 \times R1 = 300 \times 10^{-6} \times 40$ $= 12 \times 10^{-3}$S =12ms

Table22.3 Faults in the output circuit

	Situation	Cause	Countermeasure
Example 9	The load which was turned OFF is turned ON for a moment at power-off. (Transistor output)	The load [2] which was turned OFF may be turned ON due to back electromotive force at the time of power-off [1] if an inductive load is used.  	To prevent the generation of the back electromotive force, connect diode in parallel with load where the back electromotive force has been generated. Source output [3]  Sink output [3] 

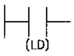
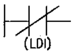
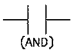
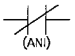
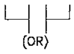
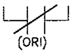
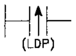
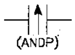
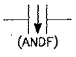
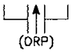
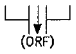
APPENDICES

APPENDIX 1 INSTRUCTION LIST

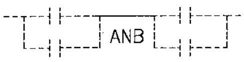
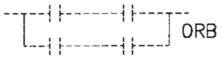
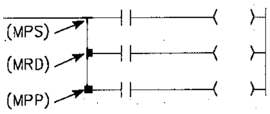
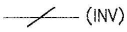
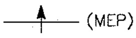
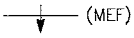
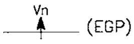
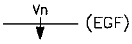
For details on SFC-related instructions, refer to the QnACPU Programming Manual (SFC).

Appendix 1.1 Sequence Instructions

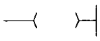
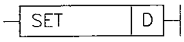
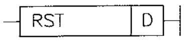
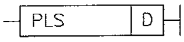
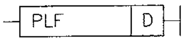
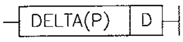
(1) Contact instructions

Classification	Symbol	Description
Contact		• Logical operation start (N/O contact logical operation start)
		• Logical NOT operation start (N/C contact logical operation start)
		• Logical product (N/O contact series connection)
		• Logical product NOT (N/C contact series connection)
		• Logical sum (N/O contact parallel connection)
		• Logical sum NOT (N/C contact parallel connection)
		• Rising edge pulse operation start
		• Falling edge pulse operation start
		• Rising edge pulse series connection
		• Falling edge pulse series connection
		• Rising edge pulse parallel connection
		• Falling edge pulse parallel connection

(2) Association commands

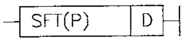
Classification	Symbol	Description
Association		• ANDs logical blocks (series connection of blocks).
		• ORs logical blocks (parallel connection of blocks).
		• Stores the operation result.
		• Reads the operation result from MPS.
		• Reads the operation result from MPS and clears the result.
		• Inverts the operation result.
		• Converts the operation result to a rising edge pulse.
		• Converts the operation result to falling edge pulse.
		• Converts the operation result to rising edge pulse (stored at Vn).
		• Converts the operation result to falling edge pulse (stored at Vn).

(3) Output instructions

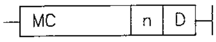
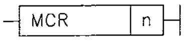
Classification	Symbol	Description
Output		• Device output
		• Sets a device.*
		• Resets a device.
		• Generates one-program cycle pulse at the rising edge of an input signal.
		• Generates one-program cycle pulse at the falling edge of an input signal.
		• Inverts device output.
		• Converts a direct output to pulse.

* When specifying input (X) for the target device, specify the device number out of the actual input (X) range.

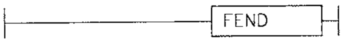
(4) Shift instructions

Classification	Symbol	Description
Shift		• Shifts a device 1 bit.

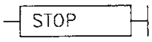
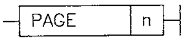
(5) Master control instructions

Classification	Symbol	Description
Master control		• Master control start
		• Master control reset

(6) End instructions

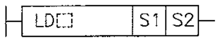
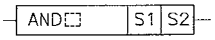
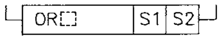
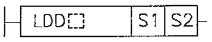
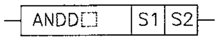
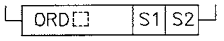
Classification	Symbol	Description
Program end		• Ends the main program.
		• Ends the sequence program.

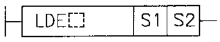
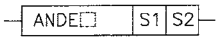
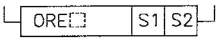
(7) Other instructions

Classification	Symbol	Description
Stop		• Stops sequence operation when the input condition is met. • Sequence program execution can be resumed by turning the RUN/ STOP key switch to RUN.
No processing	$\overline{\text{(NOP)}}$	• No processing (for program erasure or space)
		• No processing (for starting a new page during printout)
		• No processing (for managing the rest of the program as starting from step 0 of page "n")

Appendix 1.2 Basic Instructions

(1) Comparison operation instructions

Classification	Symbol	Description
16-bit data comparison		[=] • Conductive status when (S1) = (S2) • Non-conductive status when (S1)≠(S2) [< >] • Conductive status when (S1)≠(S2) • Non-conductive status when (S1) = (S2) [>] • Conductive status when (S1)>(S2) • Non-conductive status when (S1)≦(S2) [< =] • Conductive status when (S1)≦(S2) • Non-conductive status when (S1)>(S2) [<] • Conductive status when (S1)<(S2) • Non-conductive status when (S1)≧(S2) [> =] • Conductive status when (S1)≧(S2) • Non-conductive status when (S1)<(S2)
		
		
32-bit data comparison		[=] • Conductive status when (S1 + 1, S1) = (S2 + 1, S2) • Non-conductive status when (S1 + 1, S1)≠(S2 + 1, S2) [< >] • Conductive status when (S1 + 1, S1)≠(S2 + 1, S2) • Non-conductive status when (S1 + 1, S1) = (S2 + 1, S2) [>] • Conductive status when (S1 + 1, S1)>(S2 + 1, S2) • Non-conductive status when (S1 + 1, S1)≦(S2 + 1, S2) [< =] • Conductive status when (S1 + 1, S1)≦(S2 + 1, S2) • Non-conductive status when (S1 + 1, S1)>(S2 + 1, S2) [<] • Conductive status when (S1 + 1, S1)<(S2 + 1, S2) • Non-conductive status when (S1 + 1, S1)≧(S2 + 1, S2) [> =] • Conductive status when (S1 + 1, S1)≧(S2 + 1, S2) • Non-conductive status when (S1 + 1, S1)<(S2 + 1, S2)
		
		

Classification	Symbol	Description
Real number data comparison		= • Conductive status when (S1 + 1, S1) = (S2 + 1, S2) • Non-conductive status when (S1 + 1, S1)≠(S2 + 1, S2) <> • Conductive status when (S1 + 1, S1)≠(S2 + 1, S2) • Non-conductive status when (S1 + 1, S1) = (S2 + 1, S2) > • Conductive status when (S1 + 1, S1)>(S2 + 1, S2) • Non-conductive status when (S1 + 1, S1)≦(S2 + 1, S2) <= • Conductive status when (S1 + 1, S1)≦(S2 + 1, S2) • Non-conductive status when (S1 + 1, S1)>(S2 + 1, S2) < • Conductive status when (S1 + 1, S1)<(S2 + 1, S2) • Non-conductive status when (S1 + 1, S1)≧(S2 + 1, S2) >= • Conductive status when (S1 + 1, S1)≧(S2 + 1, S2) • Non-conductive status when (S1 + 1, S1)<(S2 + 1, S2)
		
		

Classification	Symbol	Description
Character string data comparison	LD$□ S1 S2	• Compares character strings (S1) and (S2) character by character. Condition for "match": Character string in which all characters match Condition for "larger character string": Character string that includes characters with larger character codes, or the longer character string Condition for "smaller character string": Character string that includes characters with smaller character codes, or the shorter character string = • Conductive status when (character string S1) = (character string S2) • Non-conductive status when (character string S1)≠(character string S2) < > • Conductive status when (S1 + 1, S1)≠(S2 + 1, S2) • Non-conductive status when (S1 + 1, S1) = (S2 + 1, S2) > • Conductive status when (character string S1)>(character string S2) • Non-conductive status when (character string S1)≦(character string S2) < • Conductive status when (character string S1)<(character string S2) • Non-conductive status when (character string S1)≠(character string S2) >= • Conductive status when (character string S1)≧(character string S2) • Non-conductive status when (character string S1)<(character string S2) <= • Conductive status when (character string S1)≦(character string S2) • Non-conductive status when (character string S1)>(character string S2)
	AND$□ S1 S2	
	OR$□ S1 S2	

Classification	Symbol	Description
Block data comparison	BKCMP=(P) | S1 | S2 | D | n	• Compares n points of data from (S1) with n points of data from (S2) in 1 word units, and stores the comparison result in the n points starting from the bit device specified by (D).
	BKCMP<>(P) | S1 | S2 | D | n	
	BKCMP>(P) | S1 | S2 | D | n	
	BKCMP<=(P) | S1 | S2 | D | n	
	BKCMP<(P) | S1 | S2 | D | n	
	BKCMP>=(P) | S1 | S2 | D | n	

(2) Arithmetic operation instructions

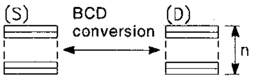
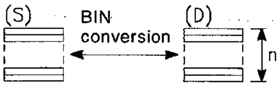
Classification	Symbol	Description
BIN 16-bit addition/ subtraction	+(P) S D	• (D) + (S) → (D)
	+(P) S1 S2 D	• (S1) + (S2) → (D)
	−(P) S D	• (D) - (S) → (D)
	−(P) S1 S2 D	• (S1) - (S2) → (D)
BIN 32-bit addition/ subtraction	D+(P) S D	• (D+1, D) + (S+1, S) → (D+1, D)
	D+(P) S1 S2 D	• (S1+1, S1) + (S2+1, S2) → (D+1, D)
	D−(P) S D	• (D+1, D) – (S+1, S) → (D+1, D)
	D−(P) S1 S2 D	• (S1+1, S1) – (S2+1, S2) → (D+1, D)
BIN 16-bit multiplication/division	*(P) S1 S2 D	• (S1) × (S2) → (D+1, D)
	/(P) S1 S2 D	• (S1)/(S2) → quotient (D), remainder (D+1)
BIN 32-bit multiplication/division	D*(P) S1 S2 D	• (S1+1, S1) × (S2+1, S2) → (D+3, D+2, D+1, D)
	D/(P) S1 S2 D	• (S1+1, S1)/(S2+1, S2) → quotient (D+1, D), remainder (D+3, D+2)

Classification	Symbol	Description
BCD 4-digit addition/ subtraction	B+(P) S D	• (D) + (S)→ (D)
	B+(P) S1 S2 D	• (S1) + (S2)→ (D)
	B-(P) S D	• (D) - (S)→ (D)
	B-(P) S1 S2 D	• (S1) - (S2)→ (D)
BCD 8-digit addition/ subtraction	DB+(P) S D	• (D+1, D) + (S+1, S)→ (D+1, D)
	DB+(P) S1 S2 D	• (S1+1, S1) + (S2+1, S2)→ (D+1, D)
	DB-(P) S D	• (D+1, D) – (S+1, S)→ (D+1, D)
	DB-(P) S1 S2 D	• (S1+1, S1) – (S2+1, S2)→ (D+1, D)
BCD 4-digit multiplication/division	B*(P) S1 S2 D	• (S1) × (S2)→ (D+1, D)
	B/(P) S1 S2 D	• (S1)/(S2)→ quotient (D), remainder (D+1)
BCD 8-digit multiplication/division	DB*(P) S1 S2 D	• (S1+1, S1) × (S2+1, S2)→ (D+3, D+2, D+1, D)
	DB/(P) S1 S2 D	• (S1+1, S1)/(S2+1, S2)→ quotient (D+1, D), remainder (D+3, D+2)
Floating point data addition/subtraction	E+(P) S D	• (D+1, D) + (S+1, S)→ (D+1, D)
	E+(P) S1 S2 D	• (S1+1, S1) + (S2+1, S2)→ (D+1, D)
	E-(P) S D	• (D+1, D) – (S+1, S)→ (D+1, D)
	E-(P) S1 S2 D	• (S1+1, S1) – (S2+1, S2)→ (D+1, D)

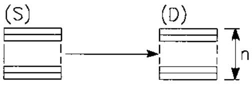
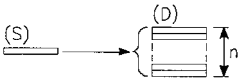
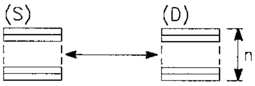
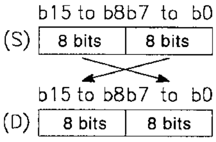
Classification	Symbol	Description
Floating point data multiplication/division	E*(P) S1 S2 D	• (S1+1, S1) × (S2+1, S2) → (D+1, D)
	E/(P) S1 S2 D	• (S1+ 1, S1)/(S2+1, S2) → quotient (D+1, D)
Character string data addition	$+(P) S D	• Associates the character string specified at (S) to the character string specified at (D) and stores the result to devices starting from (D).
	$+(P) S1 S2 D	• Associates the character string specified at (S2) to the character string specified at (S1) and stores the result to devices starting from (D).
BIN block addition/ subtraction	BK+ S1 S2 D n	• Adds n points of data from (S1) and n points of data from (S2) in a batch and stores the result to devices starting from (D).
	BK− S1 S2 D n	
BIN data increment	INC(P) D	• (D) + 1 → (D)
	DINC(P) D	• (D+1, D) + 1 → (D)
BIN data decrement	DEC(P) D	• (D) – 1 → (D)
	DDEC(P) D	• (D+1, D) – 1 → (D)

(3) Data conversion instructions

Classification	Symbol	Description
BCD conversion	BCD(P) S D	• (S) $\xrightarrow{\text{BCD conversion}}$ (D) BIN (0 to 9999)
	DBCD(P) S D	• (S + 1, S) $\xrightarrow{\text{BCD conversion}}$ (D + 1, D) BIN (0 to 99999999)
BIN conversion	BIN(P) S D	• (S) $\xrightarrow{\text{BIN conversion}}$ (D) BCD (0 to 9999)
	DBIN(P) S D	• (S + 1, S) $\xrightarrow{\text{BIN conversion}}$ (D + 1, D) BCD (0 to 99999999)
Floating point → BIN conversion	INT(P) S D	• (S + 1, S) $\xrightarrow{\text{BIN conversion}}$ (D) Real number (–32768 to 32767)
	DINT(P) S D	• (S + 1, S) $\xrightarrow{\text{BIN conversion}}$ (D) Real number (–2147483648 to 2147483647)
BIN → floating point conversion	FLT(P) S D	• (S + 1, S) $\xrightarrow{\text{Floating decimal point conversion}}$ (D) Real number (–32768 to 32767)
	DFLT(P) S D	• (S + 1, S) $\xrightarrow{\text{Floating decimal point conversion}}$ (D + 1, D) Real number (–2147483648 to 2147483647)
BIN 16-bit ←→ 32-bit conversion	DBL(P) S D	• (S) $\xrightarrow{\text{Conversion to 32-bit data}}$ (D + 1, D) BIN (–32768 to 327767)
	WORD(P) S D	• (S + 1, S) $\xrightarrow{\text{16-bit data conversion}}$ (D) BIN (–32768 to 32767)
BIN → gray code conversion	GRY(P) S D	• (S) $\xrightarrow{\text{Gray code conversion}}$ (D) BIN (–32768 to 327767)
	DGRY(P) S D	• (S + 1, S) $\xrightarrow{\text{Gray code conversion}}$ (+1, DD) BIN (–32768 to 327767)

Classification	Symbol	Description
Gray code → BIN conversion	CBIN(P) S D	• $\underline{(S)} \xrightarrow{\text{Gray code conversion}} (D)$ Gray code (–32768 to 327767)
	DGBIN(P) S D	• $\underline{(S+1, S)} \xrightarrow{\text{Gray code conversion}} (D+1, D)$ Gray code (–2147483648 to 2147483647)
2's complement	NEG(P) D	• $\overline{(D)} \longrightarrow (D)$ BIN data
	DNEG(P) D	• $\overline{\underline{(D+1, D)}} \longrightarrow (D+1, D)$ BIN data
	ENEG(P) D	• $\overline{\underline{(D+1, D)}} \longrightarrow (D+1, D)$ Real number data
Block conversion	BKBCD(P) S D n	• Converts n points of BIN data from (S) to BCD data in a batch and stores the result to devices starting from (D). 
	BKBIN(P) S D n	• Converts n points of BCD data from (S) to BIN data in a batch and stores the result to devices starting from (D). 

(4) Data transfer instructions

Classification	Symbol	Description
16-bit data transfer	MOV(P) S D	• (S) ⟶ (D)
32-bit data transfer	DMOV(P) S D	• (S+1, S) ⟶ (D+1, D)
Floating point data transfer	EMOV(P) S D	• (S+1, S) ⟶ (D+1, D)
Character string data transfer	$MOV(P) S D	• Transfers the character string specified at (S) to devices starting with the device specified at (D).
16-bit data negation transfer	CML(P) S D	• ($\overline{S}$) ⟶ ($\overline{D}$)
32-bit data negation transfer	DCML(P) S D	• ($\overline{S+1, S}$) ⟶ (D+1, D)
Data block transfer	BMOV(P) S D n	
Same data block transfer	FMOV(P) S D n	
16-bit data exchange	XCH(P) S D	• (S) ⟷ (D)
32-bit data exchange	DXCH(P) S D	• (S+1, S) ⟷ (D+1, D)
Block data exchange	BXCH(P) S D n	
Upper/lower byte swap	SWAP(P) S D	

(5) Program branch instructions

Classification	Symbol	Description
Jump	CJ Pn	• Causes a jump to Pn when the input condition is met.
	SCJ Pn	• Causes a jump to Pn beginning with the scan after the one in which the input condition is met.
	JMP Pn	• Causes a jump to Pn unconditionally.
	GOEND	• Causes a jump to the END instruction when the input condition is met.

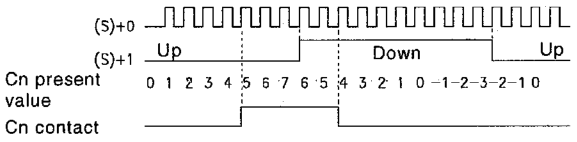
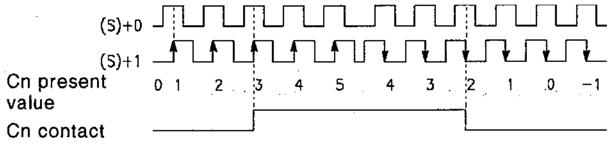
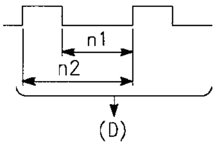
(6) Program execution control instructions

Classification	Symbol	Description
Interrupt disable	DI	• Disables execution of interrupt programs.
Interrupt enable	EI	• Cancels the execution disabled status for interrupt programs.
Interrupt disable/ enable setting	IMASK S	• Disables or enables execution of individual interrupt programs.
Return	IRET	• Returns from the interrupt program to the sequence program.

(7) I/O refresh instruction

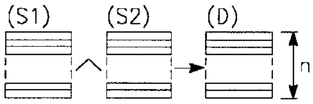
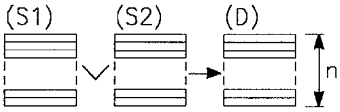
Classification	Symbol	Description
I/O refresh	RFS D n	• Executes partial refresh for the specified I/O part in a scan.

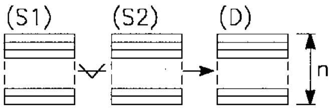
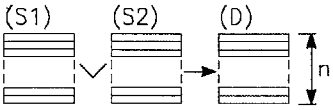
(8) Other convenient instructions

Classification	Symbol	Description
Up/down counter	UDCNT1 | S | D | n	
	UDCNT2 | S | D | n	
Teaching timer	TTMR | D | n	• (TTMR ON time) x n → (D) ↑ n=0:1, n=0:10, n=2:100
Special timer	STMR | S | n | D	• Four bit devices starting with the bit device specified at (D) perform the following operations in accordance with the ON/OFF status of the STMR instruction. (D) + 0: Off delay timer output (D) + 1: One shot timer output after OFF (D) + 2: One shot timer output after ON (D) + 3: On delay timer
Shortest path control	ROTC | S | n1 | n2 | D	• Rotates a rotary table that is partitioned into n1 from the position at which it is stopped to the position specified by (S+1) in the direction that gives the shortest path.
Ramp signal	RAMP | n1 | n2 | D1 | n3 | D2	• Changes the device data specified at (D1) in the range of n1 to n2 in n3 scans.
Pulse density	SPD | S | n | D	• Counts the pulse input of the device specified at (S) for the time specified at n and stores the result in the device specifid at (D).
Pulse output	PLSY | n1 | n2 | D	• (n1)Hz ——→ (D) Outputs "n2" times.
Pulse width modulation	PWM | n1 | n2 | D	
Matrix input	MTR | S | D1 | D2 | n	• Consecutively reads the data of n rows of 16 devices starting from the device specified at (S1) and stores it in devices starting from the device specified at (D2).

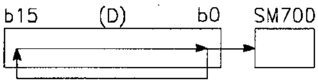
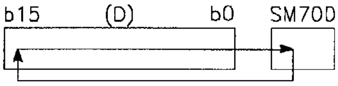
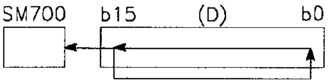
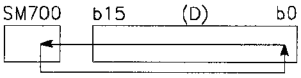
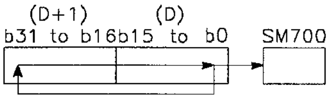
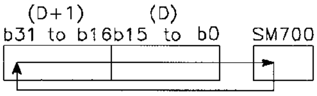
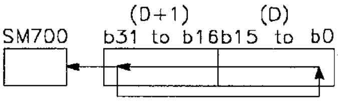
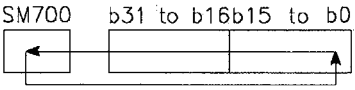
Appendix 1.3 Application Instructions

(1) Logical operation instructions

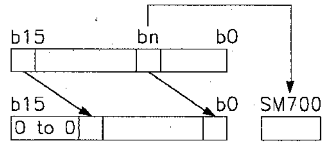
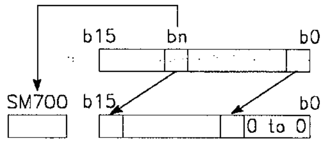
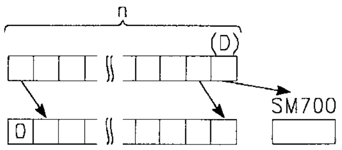
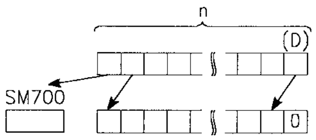
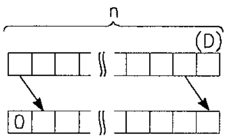
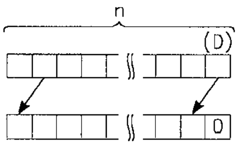
Classification	Symbol	Description
Logical product	WAND(P) S D	• (D)∧(S)→(D)
	WAND(P) S1 S2 D	• (S1)∧(S2)→(D)
	DAND(P) S D	• (D+1, D)∧(S+1, S)→(D+1, D)
	DAND(P) S1 S2 D	• (S1+1, S1)∧(S2+1, S2)→(D+1, D)
	BKAND(P) S1 S2 D n	
Logical sum	WOR(P) S D	• (D)∨(S)→(D)
	WOR(P) S1 S2 D	• (S1)∨(S2)→(D)
	DOR(P) S D	• (D+1, D)∨(S+1, S)→(D+1, D)
	DOR(P) S1 S2 D	• (S1+1, S1)∨(S2+1, S2)→(D+1, D)
	BKOR(P) S1 S2 D n	

Classification	Symbol	Description
Exclusive logical sum	WXOR(P) S D	• $(D)\veebar(S)\rightarrow(D)$
	WXOR(P) S1 S2 D	• $(S1)\veebar(S2)\rightarrow(D)$
	DXOR(P) S D	• $(D+1, D)\veebar(S+1, S)\rightarrow(D+1, D)$
	DXOR(P) S1 S2 D	• $(S1+1, S1)\veebar(S2+1, S2)\rightarrow(D+1, D)$
	BKXOR(P) S1 S2 D n	
Not exclusive logical sum	WNXR(P) S D	• $\overline{(D)\veebar(S)}\rightarrow(D)$
	WNXR(P) S1 S2 D	• $\overline{(S1)\veebar(S2)}\rightarrow(D)$
	DNXR(P) S D	• $\overline{(D+1, D)\veebar(S+1, S)}\rightarrow(D+1, D)$
	DNXR(P) S1 S2 D	• $\overline{(S1+1, S1)\veebar(S2+1, S2)}\rightarrow(D+1, D)$
	BKNXR(P) S1 S2 D n	

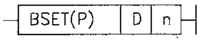
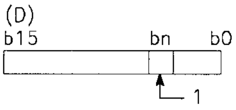
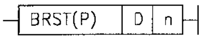
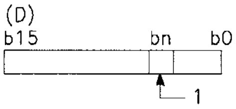
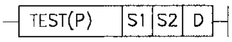
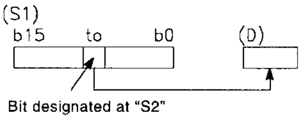
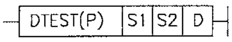
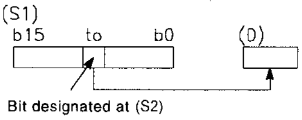
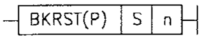
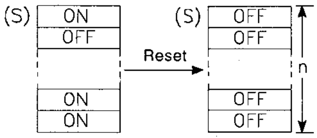
(2) Rotation instructions

Classification	Symbol	Description
Right rotation	ROR(P) D n	 Rotates “n” bits to the right.
	RCR(P) D n	 Rotates “n” bits to the right.
Left rotation	ROL(P) D n	 Rotates “n” bits to the left.
	RCL(P) D n	 Rotates “n” bits to the left.
Right rotation	DROR(P) D n	 Rotates “n” bits to the right.
	DRCR(P) D n	 Rotates “n” bits to the right.
Left rotation	DROL(P) D n	 Rotates “n” bits to the left.
	DRCL(P) D n	 Rotates “n” bits to the left.

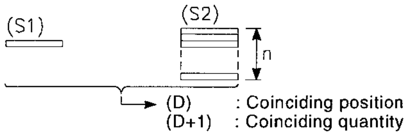
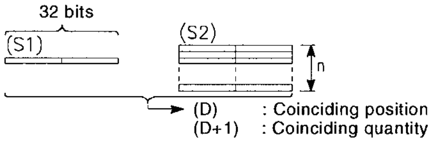
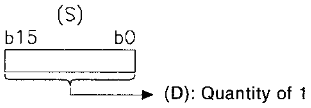
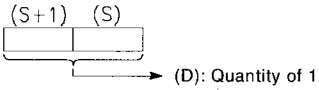
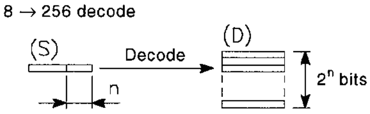
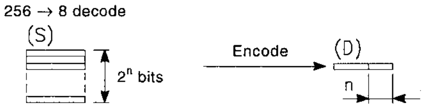
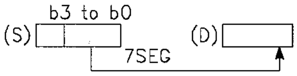
(3) Shift instructions

Classification	Symbol	Description
n bit shift	SFR(P) D n	
	SFL(P) D n	
1 bit shift	BSFR(P) D n	
	BSFL(P) D n	
1 word shift	DSFR(P) D n	
	DSFL(P) D n	

(4) Bit processing instructions

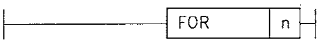
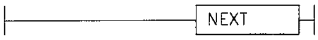
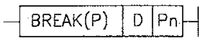
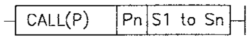
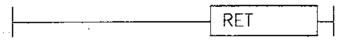
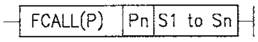
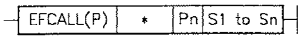
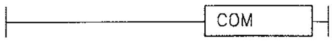
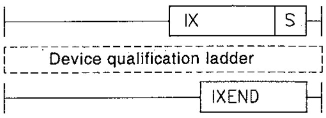
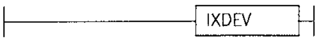
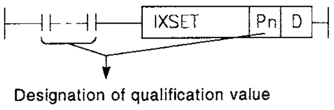
Classification	Symbol	Description
Bit set/reset		
		
Bit test		
		
Bit device batch reset		

(5) Data processing instructions

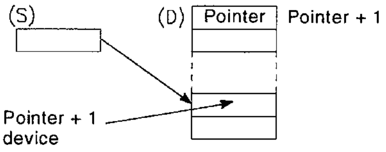
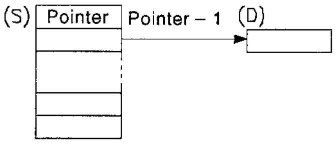
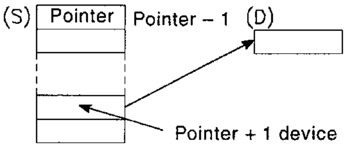
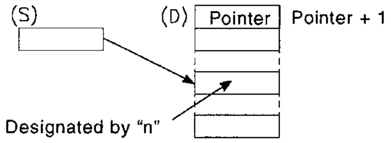
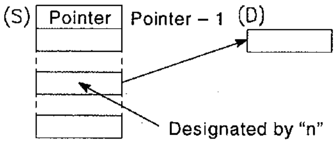
Classification	Symbol	Description
Data search	SER(P) S1 S2 D n	
	DSER(P) S1 S2 D n	
Bit check	SUM(P) S D	
	DSUM(P) S D	
Decode	DECO(P) S D n	
Encode	ENCO(P) S D n	
7-segment decode	SEG(P) S D	

Classification	Symbol	Description
Dissociation/ Association	DIS(P) S D n	• Dissociates the 16-bit data specified at (S) into 4-bit units, and stores these data in the least significant four bits of n devices starting with the one specified at (D). (n≦4)
	UNI(P) S D n	• Associates the least significant 4-bit data of n devices starting from the one specified at (S) and stores this data in the device specified at (D). (n≦4)
	NDIS(P) S1 D S2	• Dissociates data of the devices starting with the one specified at (S1) into the specified bits starting with the one specified by (S2), and stores this data in sequence starting at the device specified at (D).
	NUNI(P) S1 D S2	• Associates each of the data starting from the one specified at (S1) to the data of the devices starting from the one specified by (S2) and stores the data to the devices in equence starting at the device specified at (D).
	WTOB(P) S D n	• Dissociates the 16-bit data that starts from the device specified at (S) into 8-bit units, and stores the n points of data to the devices in sequence starting from the one specified at (D).
	BTOW(P) S D n	• Associates the lower 8 bits of 16-bit data for n points starting from the one specified at (S) to give 16-bit data, and stores the data to the devices in sequence starting from the one specified at (D).
Search	MAX(P) S D n	• Searches the n points of data starting from the device specified at (S) in 16-bit units, and stores the maximum value to the device specified at (D).
	MIN(P) S D n	• Searches the n points of data starting from the device specified at (S) in 16-bit units, and stores the minimum value to the device specified at (D).
	DMAX(P) S D n	• Searches the 2 × n points of data starting from the device specified at (S) in 32-bit units, and stores the maximum value to the device specified at (D).
	DMIN(P) S D n	• Searches the 2 × n points of data starting from the device specified at (S) in 32-bit units, and stores the minimum value in the device specified at (D).
Sort	SORT S1 n S2 D1 D2 • S2: Number of comparisons executed at one time • D1: Device turned ON on completion of sorting • D2: For system use	• Sorts n points of data starting from the device specified at (S1) in 16-bit units. [Max. number of scans required: {n × (n – 1)}/2 scans]
	DSORT S1 n S2 D1 D2 • S2: Number of comparisons executed at one time • D1: Device turned ON on completion of sorting • D2: For system use	• Sorts 2 × n points of data starting from the device specified at (S1) in 32-bit units. [Max. number of scans required: {n × (n – 1)}/2 scans]

(6) Structured program instruction

Classification	Symbol	Description
Repeat		• Executes the program section between FOR and NEXT n times.
		
		• Forcibly ends execution of the program section between FOR and NEXT and causes a jump to Pn.
Subroutine program call		• Executes the subroutine program Pn when the input condition is met. (S1 to Sn are arguments for the subroutine program.0≦n≦5)
		• Causes a return from the subroutine program.
		• Executes no-execution processing for the subroutine program Pn when the input condition is not met.
	 *: Program name	• Executes the subroutine program Pn of the specified program when the input condition is met. (S1 to Sn are arguments for the subroutine program.0≦n≦5)
	 *: Program name	• Executes no-execution processing for the subroutine program Pn of the specified program when the input condition is not met.
		• Executes link refresh and general data processing.
Ladder indexing		• Indexes each of the devices used in the device qualification ladder.
		• Stores the qualification value for indexing at IX to IXEND to the devices starting from the one specified at (D).
		

(7) Table operation instructions

Classification	Symbol	Description
Table processing	FIFW(P) S D	
	FIFR(P) S D	
	FPOR(P) S D	
	FINS(P) S D n	
	FDEL(P) S D n	

(8) Buffer memory access instructions

Classification	Symbol	Description
Data read	FROM n1 n2 D n3	• Reads data in 16-bit units from special function modules.
	DFRO n1 n2 D n3	• Reads data in 32-bit units from special function modules.
Data write	TO(P) n1 n2 S n3	• Writes data in 16-bit units to special function modules.
	DTO(P) n1 n2 S n3	• Writes data in 32-bit units to special function modules.

(9) Display instructions

Classification	Symbol	Description
ASCII print	* When SM701 is OFF PR S D	• Outputs ASCII codes in the 8 points of devices (16 characters) starting from the one specified at (S) to an output module.
	* When SM701 is ON PR S D	• Outputs ASCII codes in the devices starting from the one specified at (S) and ending at 00H, to an output module.
	PRC S D	• Converts the device comment specified at (S) to ASCII codes and outputs the result to an output module.
Display	LED S	• Displays ASCII codes in the 8 points of devices (corresponding to 16 characters) starting from the one specified at (S) on the LED indicator.
	LEDC S	• Displays the comment of the device specified at (S) on the LED indicator.
Reset	LEDR	• Resets annunciators and LED indication.

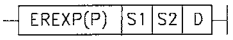
(10) Debugging and fault diagnostics instructions

Classification	Symbol	Description
Error check	CHKST	• Executes the CHK instruction when it is executed. • Causes a jump to the step following the step of the CHK instruction when it is not executed.
		• When normal → SM80: OFF, SD80: 0 • When abnormal → SM80: ON, SD80: fault No.
	CHKCIR	• Indicates the start of ladder pattern change for the ladders to be checked with the CHK instruction.
	CHKEND	• Indicates the end of ladder pattern change for the ladders to be checked with the CHK instruction.
Status latch	SLT	• Executes status latch.
	SLTR	• Resets the status latch to enable re-execution of status latch.
Sampling trace	STRA	• Triggers sampling trace.
	STRAR	• Resets the sampling trace to enable re-execution of sampling trace.
Program trace	PTRA	• Triggers program trace.
	PTRAR	• Resets the program trace to enable re-execution of program trace.
	PTRAEXE	• Executes program trace.

(11) Text string processing instructions

Classification	Symbol	Description
BIN ↓ Decimal ASCII	BINDA(P) S D	• Converts the 1-word BIN data specified at (S) into 5-digit decimal ASCII values, and stores them to the word devices starting from the one specified at (D).
	DBINDA(P) S D	• Converts the 2-word BIN data specified at (S) into 10-digit decimal ASCII values, and stores them to the word devices starting from the one specified at (D).
BIN ↓ Hexadecimal ASCII	BINHA(P) S D	• Converts the 1-word BIN data specified at (S) into 4-digit hexadecimal ASCII values, and stores them to the word devices starting from the one specified at (D).
	DBINHA(P) S D	• Converts the 2-word BIN data specified at (S) into 8-digit hexadecimal ASCII values, and stores them to the word devices starting from the one specified at (D).
BCD ↓ ASCII	BCDDA(P) S D	• Converts the 1-word BCD value specified at (S) into 4-digit decimal ASCII values, and stores them to the word devices starting from the one specified at (D).
	DBCDDA(P) S D	• Converts the 2-word BCD value specified at (S) into 8-digit decimal ASCII values, and stores them to the word devices starting from the one specified at (D).
Decimal ASCII ↓ BIN	DABIN(P) S D	• Converts the 5-digit decimal ASCII value specified at (S) to a 1-word BIN value, and stores this to the word device specified at (D).
	DDABIN(P) S D	• Converts the 10-digit decimal ASCII values specified at (S) to a 2-word BIN value, and stores this to the word device specified at (D).
Hexadecimal ASCII ↓ BIN	HABIN(P) S D	• Converts the 4-digit hexadecimal ASCII values specified at (S) to a 1-word BIN value, and stores this to the word device specified at (D).
	DHABIN(P) S D	• Converts the 8-digit decimal ASCII values designated at (S) to a 2-word BIN value, and stores this at the word device number designated at (D).
ASCII ↓ BCD	DABCD(P) S D	• Converts the 4-digit decimal ASCII values specified at (S) to a 1-word BCD value, and stores this to the word device specified at (D).
	DDABCD(P) S D	• Converts the 8-digit decimal ASCII values specified at (S) to a 2-word BCD value, and stores this to the word devices specified at (D).
Device comment read	COMRD(P) S D	• Stores the comment data of the device specified at (S) to the device specified at (D).
Text string length detection	LEN(P) S D	• Stores the length of the character string data (number of characters) that is stored in the device specified at (S) to the device specified at (D).

Classification	Symbol	Description
BIN ↓ Decimal text string	STR(P) | S1 | S2 | D	• Converts the 1-word BIN value specified at (S2) into a decimal character string with the total number of digits and number of fraction part digits specified at (S1), and stores it in the device specified at (D).
	DSTR(P) | S1 | S2 | D	• Converts the 2-word BIN value specified at (S2) into a decimal character string with the total number of digits and number of fraction part digits specified at (S1), and stores it in the device specified at (D).
Decimal text string ↓ BIN	VAL(P) | S | D1 | D2	• Converts the character string that includes a decimal point specified at (S) to a 1-word BIN value and the number of fraction part digits, and stores them to the devices specified at (D1) and (D2).
	DVAL(P) | S | D1 | D2	• Converts the character string that includes a decimal point specified at (S) to a 2-word BIN value and the number of fraction part digits, and stores them to the devices specified at (D1) and (D2).
Floating point ↓ Character string	ESTR(P) | S1 | S2 | D	• Converts the floating point data specified at (S) to a character string and stores it in the devices specified at (D).
Character string ↓ Floating decimal point	EVAL (P) | S | D	• Converts the character string specified at (S) to a floating point data and stores it in the devices specified at (D).
Hexadecimal BIN ↓ ASCII	ASC(P) | S | D | n	• Converts the 1-word BIN value in the devices starting from the one specified at (S) to hexadecimal ASCII data, and stores them to the word devices starting from the one specified at (D) for n characters.
ASCII ↓ Hexadecimal BIN	HEX(P) | S | D | n	• Converts the hexadecimal ASCII data in the devices starting from the one specified at (S) to BIN values for n characters, and stores them to the devices starting from the one specified at (D).
Character string processing	RIGHT(P) | S | D | n	• Stores n characters from the final character of the character string specified at (S) to the devices specified at (D).
	LEFT(P) | S | D | n	• Stores n characters from the initial character of the character string specified at (S) to the devices specified at (D).
	MIDR(P) | S1 | D | S2	• Stores the specified number of characters from the position specified at (S2) of the character string specified at (S1) to the devices specified at (D).
	MIDW(P) | S1 | D | S2	• Stores the character string specified at (S1) for the specified number of characters to the position specified at (S2) of the devices specified at (D).
	INSTR(P) | S1 | S2 | D | n	• Searches for the character string specified at (S1) from the nth character of the character string specified at (S2) and stores the position where a match is found to (D).
Floating point data ↓ BCD resolution	EMOD(P) | S1 | S2 | D	• Converts the floating point data specified at (S1) to a BCD data with the number of fraction part digits specified at (S2), and stores this data to the devices specified at (D).

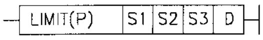
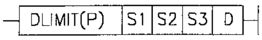
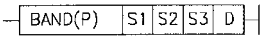
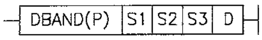
Classification	Symbol	Description
BCD ↓ Floating point data		• Converts the BCD data specified at (S1) to a floating point data with the number of fraction part digits specified at (S2) and stores this data to the devices specified at (D).

(12) Special function instructions

Classification	Symbol	Description
Trigonometric function (floating point data)	SIN(P) | S | D	• Sin(S+1, S) → (D+1, D)
	COS(P) | S | D	• Cos(S+1, S) → (D+1, D)
	TAN(P) | S | D	• Tan(S+1, S) → (D+1, D)
	ASIN(P) | S | D	• Sin^{-1}(S+1, S) → (D+1, D)
	ACOS(P) | S | D	• Cos^{-1}(S+1, S) → (D+1, D)
	ATAN(P) | S | D	• Tan^{-1}(S+1, S) → (D+1, D)
Degree ⟷ radian conversion	RAD(P) | S | D	• (S + 1, S) → (D + 1, D) Degree → radian conversion
	DEC(P) | S | D	• (S + 1, S) → (D + 1, D) Radian → degree conversion
	SQR(P) | S | D	• $\sqrt{(S + 1, S)}$ → (D + 1, D)
Exponent operation	EXP(P) | S | D	• $e^{(S + 1, S)}$ → (D + 1, D)
Natural logarithm	LOG(P) | S | D	• Log e(S + 1, S) → (D + 1, D)
Square root	BSQR(P) | S | D	• $\sqrt{(S)}$ → (D) + 0 Integer part + 1 Fraction part
	BDSQR(P) | S | D	• $\sqrt{(S + 1, S)}$ → (D) + 0 Integer part + 1 Fraction part

Classification	Symbol	Description
Trigonometric function	BSIN(P) S D	• Sin (S) → (D) + 0 Sign + 1 Integer part + 2 Fraction part
	BCOS(P) S D	• Cos (S) → (D) + 0 Sign + 1 Integer part + 2 Fraction part
	BTAN(P) S D	• Tan (S) → (D) + 0 Sign + 1 Integer part + 2 Fraction part
	BASIN(P) S D	• Sin^{-1} (S) → (D) + 0 Sign + 1 Integer part + 2 Fraction part
	BACOS(P) S D	• Cos^{-1} (S) →(D) + 0 Sign + 1 Integer part + 2 Fraction part
	BATAN(P) S D	• Tan^{-1} (S) →(D) + 0 Sign + 1 Integer part + 2 Fraction part

(13) Data control instructions

Classification	Symbol	Description
Upper/lower limit control		• Processes the value specified at (S3) to a data in the range defined by the upper and lower limits set at (S1) and (S2), and stores it to the word device specified at (D). · When S3<S1 · · · · · The value at (S1) is stored to (D). · When S1≦S3≦S2 · · · The value at (S3) is stored to (D). · When S2<S3 · · · · · The value at (S2) is stored to (D).
		• Processes the value specified at (S3+1, S3) to a data in the range defined by the upper and lower limits set at (S1+1, S1) and (S2+1, S2), and stores it to the word device specified at (D+1, D). · When (S3+1, S3)<(S1+1, S1) · · · · · The value at (S1+1, S1) is stored to (D+1, D). · When (S1+1, S1)≦(S3+1, S3)≦(S2+1, S2) · · · · · The value at (S3+1, S3) is stored to (D+1, D). · When (S2, S2+1)<(S3, S3+1) · · · · · The value at (S2+1, S2) is stored to (D+1, D).
Dead zone control		• Taking the area set by (S1) and (S2) as the dead band, if the input value specified at (S3) is within the dead band, "0" is stored to the word device specified at (D) and if it is outside the dead band, the value obtained by subtracting the dead band upper/lower limit value from the input value is stored to the word device specified at (D). · When S1≦S3≦S2 · · · · 0→D · When S3<S1 · · · · · · S3 – S1→D · When S3>S2 · · · · · · S3 – S2→D
		• Taking the area set by (S1+1, S1) and (S2+1, S2) as the dead band, if the input value specified at (S3+1, S3) is within the dead band, "0" is stored to the word device specified at (D) and if it is outside the dead band, the value obtained by subtracting the dead band upper/lower limit value from the input value is stored to the word device specified at (D). · When (S1+1, S1)≦(S3+1, S3)≦(S2+1, S2) · · · · · · · · · · · · · · · · 0→(D+1, D) · When (S3+1, S3)<(S1+1, S1) · · · · · (S3+1, S3) – (S1+1, S1)→(D+1, D) · When (S3+1, S3)>(S2+1, S2) · · · · · (S3+1, S3) – (S2+1, S2)→(D+1, D)

Classification	Symbol	Description
Zone control	ZONE(P) S1 S2 S3 D	• By setting positive and negative bias values for the input value specified at (S3) with (S1) and (S2), calculates the value for S1 + bias, and stores it to the word device specified at (D). ▪ When S3 = 0 ・ ・ ・ 0 → D ▪ When S3 > 0 ・ ・ ・ S3 + S2 → D ▪ When S3 < 0 ・ ・ ・ S3 – S1 → D
	DZONE(P) S1 S2 S3 D	• By setting positive and negative bias values for the input value specified at (S3+1, S3) with (S1+1, S1) and (S2+1, S2), calculates the value for S1 + bias, and stores it to the word device specified at (D+1, D). ▪ When (S3+1, S3) = 0 ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ 0 → (D+1, D) ▪ When (S3+1, S3) > 0 ・ ・ ・ (S3+1, S3) – (S2+1, S2) → (D+1, D) ▪ When (S3+1, S3) < 0 ・ ・ ・ (S3+1, S3) + (S1+1, S1) → (D+1, D)

(14) Switching instructions

Classification	Symbol	Description
Block No. setting	RSET(P) S	• Changes the block No. of an extension file register to the number specified at (S).
	QDRSET(P) File name	• Sets the name of a file to be used as a file register.
	QCDSET(P) File name	• Sets the name of a file to be used as a comment register.

(15) Clock instructions

Classification	Symbol	Description
Clock data read/write	DATERD(P) | D	• (Clock element)→(D) + 0 Year + 1 Month + 2 Day + 3 Hour + 4 Minute + 5 Second + 6 Day of the week
	DATEWR(P) | S	• (S) + 0 Year → (Clock element) + 1 Month + 2 Day + 3 Hour + 4 Minute + 5 Second + 6 Day of the week
	DATE+(P) | S1 | S2 | D	(S1) Hour / Minute / Second + (S2) Hour / Minute / Second → (D) Hour / Minute / Second
	DATE−(P) | S1 | S2 | D	(S1) Hour / Minute / Second − (S2) Hour / Minute / Second → (D) Hour / Minute / Second
	SECOND(P) | S | D	(S) Hour / Minute / Second → (D) Second (Lower level) / Second (Upper level)
	HOUR(P) | S | D	(S) Second (Lower level) / Second (Upper level) → (D) Hour / Minute / Second

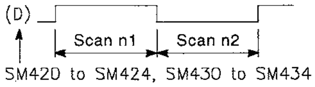
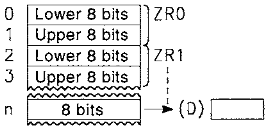
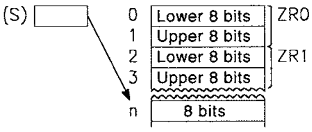
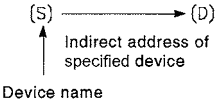
(16) Instructions for peripheral devices

Classification	Symbol	Description
Input/output to peripheral device	MSG S	• Stores the message specified at (S) to the QnACPU. This message is displayed at the peripheral device.
	PKEY D	• Stores the data input from a peripheral device to the device specified at (D).

(17) Program instructions

Classification	Symbol	Description
Program execution status switch	PSTOP(P) Program name	• Sets the specified program in the standby status.
	POFF(P) Program name	• Turns OFF the coil of the specified program's OUT instruction and sets the program to the standby status.
	PSCAN(P) Program name	• Registers the specified program as a scan execution type program.
	PLOW(P) Program name	• Registers the specified program as a low-speed execution type program.

(18) Other instructions

Classification	Symbol	Description
WDT reset	WDT(P)	• Resets the WDT in a sequence program.
Timing clock	DUTY n1 n2 D	
Direct read/write in 1 byte unit	ZRRDB(P) n D	
	ZRWRB(P) n S	
Indirect address set	ADRSET(P) S D	
Numeral key input from keyboard	KEY S n D1 D2	• Fetches ASCII data to the input module specified at (S) for 8 points, converts the data to hexadecimal values, and stores them in the devices starting with the one specified at (D1).

Appendix 1.4 Data Link Instructions

(1) Link refresh instructions

Classification	Symbol	Description
Specified network refresh	J(P).ZCOM | Jn	• Performs link refresh for the network module corresponding to the specified network No. in network n.
	G(P).ZCOM | Un	• Refreshes the network module corresponding to the specified I/O number in network n.

(2) QnA link dedicated instructions

Classification	Symbol	Description
Data read/write from/to other stations	J(P).READ Jn S1 S2 D1 D2	• Reads data from word devices of another station.
	G(P).READ Un S1 S2 D1 D2	
	J(P).WRITE Jn S1 S2 D1 D2	• Writes data to word devices of another station.
	G(P).WRITE Un S1 S2 D1 D2	
Data send/receive to/from other stations	J(P).SEND Jn S1 S2 D	• Sends data (message) to another station.
	G(P).SEND Un S1 S2 D	
	J(P).RECV Jn S D1 D2	• Receives data (message) from another station.
	G(P).RECV Un S D1 D2	
Processing request to other stations	J(P).REQ Jn S1 S2 D1 D2	Executes remote RUN/STOP for another station.
	G(P).REQ Un S1 S2 D1 D2	
Data read/write from/to a special function module at a remote I/O station	J(P).ZNFR Jn S1 S2 D	Reads data from a special function module installed at a remote station in the MELSECNET/10 network.
	G(P).ZNFR Un S1 S2 D	
	J(P).ZNTO Jn S1 S2 D	Writes data to a special function module at a remote I/O station in the MELSECNET/10 network.
	G(P).ZNTO Un S1 S2 D	

* (The GP. *** instructions can also be used for the AJ71QC24N)

(3) A series link instructions

Classification	Symbol	Description
Word device read from specified station	J(P).ZNRD Jn n1 S D1 n2 D2	• Reads the data of T, C, D, and W devices of other stations in the MELSECNET(II) or MELSECNET/10 system.
Word device write to specified station	J(P).ZNWR Jn n1 D1 S n2 D2	• Reads the data of T, C, D, and W devices of other stations on the MELSECNET(II) or MELSECNET/10 network.
Data read/write from a special function module at a remote I/O station	G(P).RFRP Un n1 D1 n2 D2	• Reads data from the special function module installed at a remote I/O station in the MELSECNET(II) system.
	G(P).RTOP Un n1 S n2 D	• Writes data to the special function module installed at a remote I/O station in the MELSECNET(II) system.

(4) Routing parameter instructions

Classification	Symbol	Description
Routing information read	Z(P).RTREAD n D	• Reads the data of the transfer destination network with the number specified by n in the routing parameters and stores the data to the devices starting from (D).
Routing information registration	Z(P).RTWRITE n S	• Registers the routing data in the devices starting from (S) to the area for the transfer destination network with the number specified by n in the parameters.

Appendix 1.5 PID Control Instructions

Classification	Symbol	Description
PID control data set	PIDINIT S	Registers the PID control data in the devices starting from the one specified at (S) to the PLC CPU.
PID control execution	PIDCONT S	Performs PID operation on the basis of the set value (SV) and process value (PV) set in the devices starting from the one specified at (S), and stores the operation result in the manipulated value (MV) area.
PID control status monitor	PID57 n S1 S2	Displays, in the form of a bar graph, the PID control status of the loop with the number specified at (S1) on the display for the AD57 specified at n. At the start of execution of PID control monitor, static image elements of other than the bar graph and numerical data are displayed by issuing the initial screen display request specified at (S2).
Specified loop operation stop	PIDSTOP n	Stops operation for the loop whose number is specified at n.
Specified loop operation start	PIDRUN n	Starts operation for the loop whose number is specified at n.
Specified loop parameter change	PIDPRMW n S	Changes the operation parameters of the loop whose number is specified at n to the data set in the devices starting from the one whose number is specified at (S).

Appendix 1.6 Special Function Module Instructions

(1) Instructions compatible with all versions
The following instructions can be used for modules with all versions.

Classification	Function	Instruction Symbol
AD61(S1) control instruction	Preset data setting	RVWR1, PVWR2
	Set value data setting for larger/smaller/ matched judgments	SVWR1, SVWR2
	Present value read	PVRD1, PVRD2
AD59(S1) control instruction	Character outputting for the intended number to a printer	PRN
	Character outputting up to the 00H code to a printer	PR
	Data read from memory card	GET
	Data write to memory card	PUT
AJ71C24 (-S3/S6/S8) control instruction	Data send for the specified number of bytes in no-protocol mode	PRN
	Data send up to the 00H code in no-protocol mode	PR
	Data receive in no-protocol mode	INPUT
	Communications status read	SPBUSY
	Send/receive processing forced interruption	SPCLR
AJ71C21(S1) control instruction	Data send for the specified number of bytes	PRN2, PRN4
	Data send up to the 00H code	PR2, PR4
	Data receive	INPUT2, INPUT4
	Data read from RAM memory	GET
	Data write to RAM memory	PUT
	Communication processing forced interruption	SPBUSY

Classification	Function	Instruction Symbol
AJ71PT32-S3 control instruction	Key input from operation box	INPUT
	Data send for the specified number of bytes in no-protocol mode	PRN
	Data send up to the 00_H code in no-protocol mode	PR
	Data receive in no-protocol mode	INPUT
	Communications with remote terminal modules	MINI, MINIEND
	Error reset for remote terminal module	MINIERR
	Communications status read	SPBUSY
	Communication processing forced interruption	SPCLR

Classification	Function	Instruction Symbol
AD75 control instruction	Display mode setting	CMODE
	Canvas screen display	CPS1
	VRAM display address change	CPS2
	Canvas data transfer to the VRAM area	CMOV
	Display area clear	CLS
	VRAM area clear	CLV
	Screen scrolling	CSCRU, CSCRD
	Cursor display	CON1, CON2
	Cursor erase	COFF
	Cursor position setting	LOCATE
	Forward/reverse rotation specification for characters	CNOR, CREV
	Forward/reverse rotation switching for characters	CRDSP, SRDSPV
	Character display color specification	COLOR
	Character color change	CCDSP, CCDSPV
	ASCII character display	PR, PRN
	ASCII character write to VRAM	PRV, PRNV
	Character display	EPR, EPRN
	Character write to VRAM	EPRV, EPRNV
	Concatenated display of same character	CR1, CR2, CC1, CC2
	- (minus) display	CINMP
	- (hyphen) display	CINHP
	. (period, decimal point) display	CINPT
	Numeric character display	CIN0 to CIN9
	Alphabet character display	CINA to CINZ
	Space display	CINSP
	Specified column clear display	CINCLR
	ASCII code conversion of specified character strings	INPUT
	VRAM data read	GET
	VRAM data write	PUT
	Display status read	STAT

Classification	Function	Instruction Symbol
AJ71ID□-R4 control instruction	ID controller initial setting	IDINIT1, IDINIT2
	Data read from ID data carrier	IDRD1, IDRD2
	Data write to ID data carrier	IDWD1, IDWD2
	Continuous read from ID data carrier	IDARD1, IDARD2
	Continuous write to ID data carrier	IDAWD1, IDAWD2
	Data compare with ID data carrier	IDCMP1, IDCMP2
	Same data batch write to ID data carrier	IDFILL1, IDFILL2
	Copy between ID data carriers	IDCOPY1, IDCOPY2
	ID data carrier clear	IDCLR1, IDCLR2
	ID data carrier use end	IDOFF1, IDOFF2
	ID data carrier use start	IDON1, IDON2
AJ71QC24 control instruction*	Writes the user registration frame to the E^2PROM for the AJ71QC24N.	PUTE
	Reads the user registration frame from the E^2PROM for the AJ71QC24N.	GETE
	Data send with the dedicated protocol using the "on demand" function	ONDEMAND
	Data send for the specified number of bytes in no-protocol mode	OUTPUT
	Data send in accordance with the send schedule table in no-protocol mode	PRR
	Data receive in no-protocol mode	INPUT
	Data send with the bi-directional protocol	BIDOUT
	Data receive with the bi-directional protocol	BIDIN
	Communication status read	SPBUSY
	Device read from other stations	READ
	Device write to other stations	SWRITE
	Data send to other stations	SEND
	Data receive from other stations	RECV
	Transient transmission request to other stations	REQ

* The AJ71QC24N can be used with QnA link instructions designated for use with special function modules (G(P). ***).

(2) Instructions added after function version B

With function version B, the following instructions can be used in addition to the instructions in (1).

Refer to Section 2.2 for the function version.

Classification	Function	Instruction Symbol
AJ71ID□-R4 control instruction	Comparison read from ID data carrier	IDCRD1, IDCRD2
	Comparison write to ID data carrier	IDCWD1, IDCWD2
	Continuous comparison read from ID data carrier	IDSRD1, IDSRD2
	Continuous comparison write to ID data carrier	IDSWD1, IDSWD2
	Continuous high-speed read from ID data carrier	IDFRD1, IDFRD2
	Continuous high-speed write to ID data carrier	IDFWD1, IDFWD2
CC-Link control instruction	Read from the buffer memory of the intelligent device station	RIRD
	Write to the buffer memory of the intelligent device station	RIWT
	Write to the buffer memory of the intelligent device station(with handshake)	RISEND
	Read from the buffer memory of the intelligent device station (with handshake)	RIRCV
	Read from master station buffer memory for automatic update	RIFR
	Write to master station buffer memory for automatic update	RITO
	Intelligent device station communication	CCL, CCLEND
	Intelligent device station communication status read	SPCBUSY
	Intelligent device station communication processing interrupt	SPCCLR
	Remote register (RWr) read	RDGET
	Remote register (RWw) write	RDPUT
	Remote register (RWr) monitor	RDMON

Classification	Function		Instruction Symbol
AD75 contorol instruction	1 axis positioning start		PSTART
	Interpolation positioning start		PHOSTA
	OPR start		PZPR
	Current value change request		PADCH
	Forward JOG start/stop		PJOG+
	Reverse JOG start/stop		PJOG-
	Manual pulse generator operation enable/ disable		PMPG
	Speed change request		PSPCH
	Axis error reset		PERRST
	Basic parameter setting		PBPSET
	Detail parameter setting		PEPSET
	OPR data setting		POPSET
	Positioning data setting		PPOSET
	Positioning start data setting		PSDSET
	Positioning special start data setting		PSPSET
	Condition data setting		PCTSET
	Error/warning number read		PEWR
	Monitor data read		PMDRD
	Positioning data I/F setting		PIFSET
AJ71QE71 control instruction	Parameter setting		EPRSET
	QnA compatible transmission/ receiving instruction	Other station device read	READ SREAD
		Other station device write	WRITE SWRITE
		Data send	SEND
		Data receive	RECV
		Other station transient request	REQ
	A compatible send/ receive instruction	Other station device read	ZNRD
		Other station device write	ZNWR

APPENDIX 2 Special Relay List

Special relays, SM, are internal relays whose applications are fixed in the PLC.
For this reason, they cannot be used by sequence programs in the same way as the normal internal relays.
However, they can be turned ON or OFF as needed in order to control the CPU module and remote I/O modules.

The heading descriptions in the following special relay lists are shown in Table App. 2.1.

Table App. 2.1. Explanation of special relay list

Item	Function of Item
Number	• Indicates special register number
Name	• Indicates name of special register
Meaning	• Indicates contents of special register
Explanation	• Discusses contents of special register in more detail
Set by (When set)	• Indicates whether the relay is set by the system or user, and, if it is set by the system, when setting is performed. <Set by> S : Set by system U : Set by user (sequence programs or test operations from GX Developer) S/U : Set by both system and user <When set> Indicated only for registers set by system Each END : Set during each END processing Initial : Set only during initial processing (when power supply is turned ON, or when going from STOP to RUN) Status change : Set only when there is a change in status Error : Set when error occurs Instruction execution : Set when instruction is executed Request : Set only when there is a user request (through SM, etc.) System switching : Set when system switching is executed.
Corresponding ACPU M9□ □ □	• Indicates the corresponding special relay (M9□□□) of the ACPU. (When the contents are changed, the special relay is represented M9□□□ format change.) • New indicates the special relay newly added to the QnACPU.
Corresponding CPU	Indicates the corresponding CPU module type name. QnA : Indicates the QnA series and Q2ASCPU series. Each CPU module model name: Indicates the relevant specific CPU module. (Example: Q4AR, Q2AS)

For details on the following items, refer to the following manuals:

- Networks → Manual of the corresponding network module
- SFC → QCPU(Q mode)/QnACPU Programming Manual (SFC)

(1) Diagnostic Information

Table App. 2.2. Special relay

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM0	Diagnostic errors	OFF : No error ON : Error	• Turns ON if an error occurs as a result of diagnosis. (Includes when an annunciator is ON, and when an error is detected with CHK instruction) • Remains ON even if the condition is restored to normal thereafter.	S (Error)	New	QnA
SM1	Self-diagnostic error	OFF : No self-diagnosis errors ON : Self-diagnosis	• Turns ON if an error occurs as a result of diagnosis. (Does not include when an annunciator is ON or when an error is detected by the CHK instruction) • Remains ON even if the condition is restored to normal thereafter.	S (Error)	M9008	QnA
SM5	Error common information	OFF : No error common information ON : Error common information	• When SM0 is ON, turns ON if there is error common information	S (Error)	New	QnA
SM16	Error individual information	OFF : No error individual information ON : Error individual information	• When SM0 is ON, turns ON if there is error individual information	S (Error)	New	QnA
SM50	Error reset	OFF → ON: Error reset	• Conducts error reset operation	U	New	QnA
SM51	Battery low latch	OFF : Normal ON : Battery low	• Turns ON if battery voltage at CPU module or memory card drops below rated value. • Remains ON even if the battery voltage returns to normal thereafter. • Synchronizes with the BAT. ALARM/BAT. LED.	S (Error)	M9007	QnA
SM52	Battery low	OFF : Normal ON : Battery low	• Same as SM51, but turns OFF subsequently when battery voltage returns to normal.	S (Error)	M9006	QnA
SM53	AC/DC DOWN detection	OFF : AC/DC DOWN not detected ON : AC/DC DOWN detected	• Turns ON if an instantaneous power failure of within 20ms occurs during use of the AC power supply module. Reset when the power supply is switched OFF, then ON.	S (Error)	M9005	QnA
			• Turns ON if an instantaneous power failure of within 1ms occurs during use of the DC power supply module. Reset when the power supply is switched OFF, then ON.			QnA
SM54	MINI link error	OFF : Normal ON : Error	• Turns ON if MINI (S3) link error is detected at even one of the installed AJ71PT32 (S3) modules. • Remains ON even if the condition is restored to normal thereafter.	S (Error)	M9004	QnA
SM56	Operation error	OFF : Normal ON : Operation error	• ON when operation error is generated • Remains ON if the condition is restored to normal thereafter.	S (Error)	M9011	QnA
SM60	Blown fuse detection	OFF : Normal ON : Module with blown fuse	• Turns ON if there is at least one output module whose fuse has blown. • Remains ON if the condition is restored to normal thereafter. • Blown fuse status is checked even for remote I/O station output modules.	S (Error)	M9000	QnA
SM61	I/O module verify error	OFF : Normal ON : Error	• Turns ON if the I/O module differs from the status registered at power on. • Remains ON if the condition is restored to normal thereafter. • I/O module verification is also conducted for remote I/O station modules.	S (Error)	M9002	QnA
SM62	Annunciator detection	OFF : Not detected ON : Detected	• Goes ON if even one annunciator F goes ON.	S (Instruction execution)	M9009	QnA
SM80	CHK detection	OFF : Not detected ON : Detected	• Goes ON if error is detected by CHK instruction. • Remains ON if the condition is restored to normal thereafter.	S (Instruction execution)	New	QnA

Table App. 2.2. Special relay

Number	Name	Meaning	Explanation		Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM90	Startup of monitoring timer for step transition (Enabled only when SFC program exists)	OFF : Not started(monitoring timer reset) ON : Started(monitoring timer started)	Corresponds to SD90	• Goes ON when measurement of step transition monitoring timer is commenced. • Resets step transition monitoring timer when it goes OFF.	U	M9108	QnA
SM91			Corresponds to SD91			M9109	
SM92			Corresponds to SD92			M9110	
SM93			Corresponds to SD93			M9111	
SM94			Corresponds to SD94			M9112	
SM95			Corresponds to SD95			M9113	
SM96			Corresponds to SD96			M9114	
SM97			Corresponds to SD97			New	
SM98			Corresponds to SD98			New	
SM99			Corresponds to SD99			New	

(2) System information

Table App. 2.3. Special relay

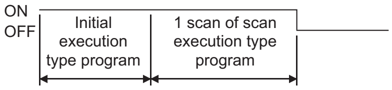
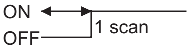
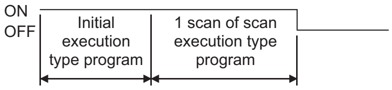
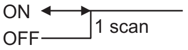
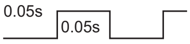
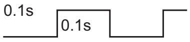
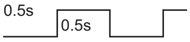
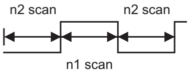
Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM202	LED OFF command	OFF → ON : LED OFF	• When this relay goes from OFF to ON, the LEDs corresponding to the individual bits at SD202 go off	U	New	QnA
SM203	STOP contact	STOP status	• Goes ON at STOP status	S (Status change)	M9042	QnA
SM204	PAUSE contact	PAUSE status	• Goes ON at PAUSE status	S (Status change)	M9041	QnA
SM205	STEP-RUN contact	STEP-RUN status	• Goes ON at STEP-RUN status	S (Status change)	M9054	QnA
SM206	PAUSE enable coil	OFF : PAUSE disabled ON : PAUSE enabled	• PAUSE status is entered if this relay is ON when the PAUSE contact goes ON	U	M9040	QnA
SM210	Clock data set request	OFF : Ignored ON : Set request	• When this relay goes from OFF to ON and after END instruction execution of subsequent scan, clock data stored in SD210 to SD213 are written to the CPU module.	U	M9025	QnA
SM211	Clock data error	OFF : No error ON : Error	• ON when error is generated in clock data (SD210 to SD213) value, and OFF if no error is detected.	S (Request)	M9026	QnA
SM212	Clock data display	OFF : Ignored ON : Display	• Displays clock data as month, day, hour, minute, and second at the LED display at front of CPU module.(Enabled only for Q3ACPU and Q4ACPU)	U	M9027	Q3A Q4A Q4AR
SM213	Clock data read request	OFF : Ignored ON : Read request	• When this relay is ON, clock data is read to SD210 to SD213 as BCD values.	U	M9028	QnA
SM250	Max. loaded I/O read	OFF : Ignored ON : Read	• When this relay goes from OFF to ON, maximum loaded I/O number is read to SD250.	U	New	QnA
SM251	I/O change flag	OFF : No replacement ON : Replacement	• By turning this relay ON after setting the head I/O number of the replaced I/O module to SD251, the I/O module can be replaced online (with power on). (Only one module can be replaced for each setting.) • Turn this relay ON in the test mode of the program or peripheral device for an I/O module change during RUN, or in the test mode of the peripheral device for an I/O change during STOP. • Do not execute a RUN/STOP mode change until I/O module change is finished.	U	M9094	Q2A(S1) Q3A Q4A Q4AR
SM252	I/O change OK	OFF : Replacement prohibited ON : Replacement enabled	• Goes ON when I/O replacement is OK.	S (END)	New	
SM255	MELSECNET/10 module 1 information	OFF : Operative network ON : Standby network	• Goes ON for standby network(If no designation has been made concerning active or standby, active is assumed.)	S (Initial)	New	QnA
SM256		OFF : Reads ON : Does not read	• For refresh from link to CPU module (B, W, etc.) indicate whether to read from the link module.	U	New	
SM257		OFF : Writes ON : Does not write	• For refresh from CPU module to link (B, W, etc.), designate whether to write to the link module.	U	New	
SM260	MELSECNET/10 module 2 information	OFF : Operative network ON : Standby network	• Goes ON for standby network (If no designation has been made concerning active or standby, active is assumed.)	S (Initial)	New	QnA
SM261		OFF : Reads ON : Does not read	• For refresh from link to CPU module (B, W, etc.) indicate whether to read from the link module.	U	New	
SM262		OFF : Writes ON : Does not write	• For refresh from CPU module to link (B, W, etc.), designate whether to write to the link module.	U	New	
SM265	MELSECNET/10 module 3 information	OFF : Operative network ON : Standby network	• Goes ON for standby network (If no designation has been made concerning active or standby, active is assumed.)	S (Initial)	New	QnA
SM266		OFF : Reads ON : Does not read	• For refresh from link to CPU module (B, W, etc.) indicate whether to read from the link module.	U	New	
SM267		OFF : Writes ON : Does not write	• For refresh from CPU module to link (B, W, etc.), designate whether to write to the link module.	U	New	
SM270	MELSECNET/10 module 4 information	OFF : Operative network ON : Standby network	• Goes ON for standby network (If no designation has been made concerning active or standby, active is assumed.)	S (Initial)	New	QnA
SM271		OFF : Reads ON : Does not read	• For refresh from link to CPU module (B, W, etc.) indicate whether to read from the link module.	U	New	
SM272		OFF : Writes ON : Does not write	• For refresh from CPU module to link (B, W, etc.), designate whether to write to the link module.	U	New	
SM280	CC-Link error	OFF : Normal ON : Error	• Goes ON when a CC-Link error is detected in any of the installed CC-Link module. Remains ON if the condition is restored to normal thereafter.	S (Error)	New	QnA

Table App. 2.3. Special relay

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM320	Presence/absence of SFC program	OFF : SFC program absent ON : SFC program present	• Turns ON when an SFC program is registered. • OFF when an SFC program is not registered.	S (Initial)	M9100	QnA
SM321	Start/stop SFC program	OFF : SFC program not executed (stop) ON : SFC program executed (start)	• Initial value is set at the same value as SM320. (Goes ON automatically if SFC program is present.) • Turn this relay from ON to OFF to stop program execution. • Turn this relay from OFF to ON to resume program execution.	S (Initial)/U	M9101form at change	QnA
SM322	SFC program start status	OFF : Initial start ON : Resume start	• The SFC program starting mode in the SFC setting of the PLC parameter dialog box is set as the initial value. At initial start: OFF At continued start: ON	S (Initial)/U	M9102form at change	QnA
SM323	Presence/absence of continuous transition for entire block	OFF : Continuous transition not effective ON : Continuous transition effective	Set the presence/absence of continuous transition for the block where "Continuous transition bit" of the SFC data device has not been set.	U	M9103	QnA
SM324	Continuous transition prevention flag	OFF : When transition is executed ON : When no transition	• OFF during operation in the continuous transition mode or during continuous transition, and ON when continuous transition is not executed. • Always ON during operation in the no continuous transition mode.	S (Instruction execution)	M9104	QnA
				S (Status change)	New	QnA
SM325	Output mode at block stop	OFF : OFF ON : Preserves	Select whether the coil outputs of the active steps are held or not at the time of a block stop. • As the initial value, the output mode at a block stop in the parameter is OFF when the coil outputs are OFF, and ON when the coil outputs are held. • All coil outputs go OFF when this relay is OFF. • Coil outputs are preserved when this relay is ON.	S (Initial)/U	M9196	QnA
SM326	SFC device clear mode	OFF : Clear device ON : Preserves device	Selects the device status when the stopped CPU is run after the sequence program or SFC program has been modified when the SFC program exists.	U	New	QnA
SM327	Output during end step execution	OFF : Hold step output turned OFF (cleared) ON : Hold step output held	• Select the device status at the time of switching from STOP to program write to RUN.(All devices except the step relay)	S (Initial)/U	New	QnA
SM330	Operation mode for low speed execution type program	OFF : Asynchronous mode ON : Synchronous mode	Select whether the low speed execution type program will be executed in the asynchronous mode or in the synchronous mode. • Asynchronous mode (this relay is turned OFF.) Mode in which the operation of the low speed execution type program is performed continuously within the excess time. • Synchronous mode (this relay is turned ON.) Mode in which the operation of the low speed execution type program is not performed continuously and operation is performed from the next scan if there is excess time.	U	New	QnA

(3) System clocks/counters

Table App. 2.4. Special relay

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM400	Always ON		• Normally is ON	S (Every END processing)	M9036	QnA
SM401	Always OFF		• Normally is OFF	S (Every END processing)	M9037	QnA
SM402	After RUN, ON for 1 scan only		• After RUN, ON for 1 scan only. • This connection can be used for scan execution type programs only. • When an initial execution type program is used, this relay turns OFF at the END processing of the scan execution type program in the first scan after RUN. 	S (Every END processing)	M9038	QnA
SM403	After RUN, OFF for 1 scan only		• After RUN, OFF for 1 scan only. • This connection can be used for scan execution type programs only. • When an initial execution type program is used, this relay turns OFF at the END processing of the scan execution type program in the first scan after RUN. 	S (Every END processing)	M9039	QnA
SM404	Low speed execution type programON for 1 scan only after RUN		• After RUN, ON for 1 scan only. • This connection can be used for low speed execution type programs only.	S (Every END processing)	New	QnA
SM405	Low speed execution type programAfter RUN, OFF for 1 scan only		• After RUN, OFF for 1 scan only. • This connection can be used for low speed execution type programs only.	S (Every END processing)	New	QnA
SM410	0.1 second clock		• Repeatedly changes between ON and OFF at each designated time interval. • When PLC power supply is turned OFF or a CPU module reset is performed, goes from OFF to start. (Note that the ON-OFF status changes when the designated time has elapsed during the execution of the program.)	S (Status change)	M9030	QnA
SM411	0.2 second clock				M9031	QnA
SM412	1 second clock				M9032	QnA
SM413	2 second clock				M9033	QnA
SM414	2n second clock		• This relay alternates between ON and OFF at intervals of the time (unit: s) specified in SD414. • When PLC power supply is turned OFF or a CPU module reset is performed, goes from OFF to start. (Note that the ON-OFF status changes when the designated time has elapsed during the execution of the program.)	S (Status change)	M9034form at change	QnA
SM420	User timing clock No.0		• Relay repeats ON/OFF switching at fixed scan intervals. • When PLC power supply is turned ON or a CPU module reset is performed, goes from OFF to start. • The ON/OFF intervals are set with the DUTY instruction DUTY n1 n2 SM420 n1: ON scan interval n2: OFF scan interval	S (Every END processing)	M9020	QnA
SM421	User timing clock No.1				M9021	
SM422	User timing clock No.2				M9022	
SM423	User timing clock No.3				M9023	
SM424	User timing clock No.4				M9024	
SM430	User timing clock No.5		• For use with SM420 to SM424 low speed programs	S (Every END processing)	New	QnA
SM431	User timing clock No.6					
SM432	User timing clock No.7					
SM433	User timing clock No.8					
SM434	User timing clock No.9					

(4) Scan information

Table App. 2.5. Special relay

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM510	Low speed program execution flag	OFF : Completed or not executed ON : Execution under way.	• Goes ON when low speed execution type program is executed.	S (Every END processing)	New	QnA
SM551	Reads module service interval	OFF : Ignored ON : Read	• When this relay goes from OFF to ON, the module service interval designated by SD550 is read to SD551 to SD552.	U	New	QnA

(5) Drive information

Table App. 2.6. Special relay

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM600	Memory card (A) usable flags	OFF : Unusable ON : Use enabled	• ON when memory card (A) is ready for use by user	S (Status change)	New	QnA
SM601	Memory card (A) protect flag	OFF : No protect ON : Protect	• Goes ON when memory card (A) protect switch is ON	S (Status change)	New	QnA
SM602	Drive 1 flag	OFF : No drive 1 ON : Drive 1 present	• Turns ON when the mounted memory card (A) is RAM	S (Status change)	New	QnA
SM603	Drive 2 flag	OFF : No drive 2 ON : Drive 2 present	• Turns ON when the mounted memory card (A) is ROM	S (Status change)	New	QnA
SM604	Memory card (A) in-use flag	OFF : Not used ON : In use	• Goes ON when memory card (A) is in use	S (Status change)	New	QnA
SM605	Memory card (A) remove/insert prohibit flag	OFF : Remove/insert enabled ON : Remove/insert prohibited	• Goes ON when memory card (A) cannot be inserted or removed	U	New	QnA
SM620	Memory card B usable flags	OFF : Unusable ON : Use enabled	• ON when memory card B is ready for use by user	S (Initial)	New	Q2A(S1) Q3A Q4A Q4AR
SM621	Memory card B protect flag	OFF : No protect ON : Protect	• Goes ON when memory card B protect switch is ON	S (Initial)	New	Q2A(S1) Q3A Q4A Q4AR
SM622	Drive 3 flag	OFF : No drive 3 ON : Drive 3 present	• Goes ON when drive 3 (card 2 RAM area) is present	S (Initial)	New	Q2A(S1) Q3A Q4A Q4AR
SM623	Drive 4 flag	OFF : No drive 4 ON : Drive 4 present	• Goes ON when drive 4 (card 2 ROM area) is present	S (Initial)	New	Q2A(S1) Q3A Q4A Q4AR
SM624	Memory card B in-use flag	OFF : Not used ON : In use	• Goes ON when memory card B is in use	S (Status change)	New	Q2A(S1) Q3A Q4A Q4AR
SM625	Memory card B remove/insert prohibit flag	OFF : Remove/insert enabled ON : Remove/insert prohibited	• Goes ON when memory card B cannot be inserted or removed	U	New	Q2A(S1) Q3A Q4A Q4AR
SM640	File register use	OFF : File register not used ON : File register in use	• Goes ON when file register is in use	S (Status change)	New	QnA
SM650	Comment use	OFF : File register not used ON : File register in use	• Goes ON when comment file is in use	S (Status change)	New	QnA
SM660	Boot operation	OFF : Internal memory execution ON : Boot operation in progress	• Goes ON while boot operation is in process • Goes OFF if boot designation switch is OFF	S (Status change)	New	QnA
SM672	Memory card A file register access range flag	OFF : Within access range ON : Outside access range	• Goes ON when access is made to area outside the range of file register of memory card A(Set within END processing.) • Reset at user program	S/U	New	QnA
SM673	Memory card B file register access range flag	OFF : Within access range ON : Outside access range	• Goes ON when access is made outside the range of file register of memory card B.(Set within END processing.) • Reset at user program	S/U	New	Q2A(S1) Q3A Q4A Q4AR

(6) Instruction-Related Special Relays

Table App. 2.7. Special relay

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM700	Carry flag	OFF : Carry OFF ON : Carry ON	• Carry flag used in application instruction	S (Instruction execution)	M9012	QnA
SM701	Number of output characters selection	Switching the number of output characters and the output pattern	• Used for the PR, PRC, BINDA, DBINDA, BINHA, DBINHA, BCDDA, DBCDDA, or COMRD instruction • For details, refer to the QCPU (Q Mode)/QnACPU Programming Manual (Common Instructions).	U	M9049	QnA Qn(H) QnPH QnPRH QnU
SM702	Search method	OFF : Search next ON : 2-part search	• Designates method to be used by search instruction. • Data must be arranged for 2-part search.	U	New	QnA
SM703	Sort order	OFF : Ascending order ON : Descending order	• The sort instruction is used to designate whether data should be sorted in ascending order or in descending order.	U	New	QnA
SM704	Block comparison	OFF : Non-match found ON : All match	• Goes ON when all data conditions have been met for the BKCMP instruction.	S (Instruction execution)	New	QnA
SM707	Selection of real number instruction processing type	OFF : Speed oriented ON : Accuracy oriented	• When SM707 is OFF, real number instructions are processed at high speed. • When it is ON, real number instructions are processed with high accuracy.	U	New	Q4AR
SM710	CHK instruction priority ranking flag	OFF : Conditions priority ON : Pattern priority	• Remains as originally set when OFF. • CHK priorities updated when ON.	S (Instruction execution)	New	QnA
SM711	Divided transmission status	OFF : Other than during divided processing ON : During divided processing	• In processing of AD57(S1), goes ON when screen is split for transfer, and goes OFF when split processing is completed	S (Instruction execution)	M9065	QnA
SM712	Transmission processing selection	OFF : Batch processing ON : Divided processing	• In processing of AD57(S1), goes ON when canvas screen is divided for transfer.	S (Instruction execution)	M9066	QnA
SM714	Communication request registration area BUSY signal	OFF : Communication request to remote terminal module enabled ON : Communication request to remote terminal module disabled	• Used to determine whether communications requests to remote terminal modules connected to the AJ71PT32-S3 can be executed or not.	S (Instruction execution)	M9081	QnA
SM715	EI flag	OFF : During DI ON : During EI	• ON when EI instruction is being executed.	S (Instruction execution)	New	QnA
SM736	PKEY instruction execution in progress flag	OFF : Instruction not executed ON : Instruction execution	• ON when PKEY instruction is being executed. Goes OFF when CR is input, or when input character string reaches 32 characters.	S (Instruction execution)	New	QnA
SM737	Keyboard input reception flag for PKEY instruction	OFF : Keyboard input reception enabled ON : Keyboard input reception disabled	• Goes ON when keyboard input is being conducted. Goes when keyboard input has been stored at the CPU.	S (Instruction execution)	New	QnA
SM738	MSG instruction reception flag	OFF : Instruction not executed ON : Instruction execution	• Goes ON when MSG instruction is executed	S (Instruction execution)	New	QnA
SM774	PID bumpless processing (for complete derivative)	OFF : Matched ON : Not matched	• Specifies whether to match the set value (SV) with the process value (PV) or not in the manual mode.	U	New	QnA
SM775	Selection of refresh processing during COM instruction execution	OFF : Performs link refresh ON : Performs no link refresh	• Select whether link refresh processing will be performed or not when only communication with the CPU module is made at the execution of the COM instruction.	U	New	QnA
SM776	Enable/disable local device at CALL	OFF : Local device disabled ON : Local device enabled	• Set whether the local device of the subroutine program called at execution of the CALL instruction is valid or invalid.	U	New	QnA
SM777	Enable/disable local device in interrupt program	OFF : Local device disabled ON : Local device enabled	• Set whether the local device at execution of the interrupt program is valid or invalid.	U	New	QnA
SM780	CC-Link dedicated instruction executable	OFF : CC-Link dedicated instruction executable ON : CC-Link dedicated instruction not executable	• Switches ON when the number of the CC-Link dedicated instructions that can be executed simultaneously reaches 32. Switches OFF when the number goes below 32.	U	New	QnA

(7) Debug

Table App. 2.8. Special relay

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM800	Sampling trace preparation	OFF : Not ready ON : Ready	• Goes ON when sampling trace is ready	S (Status change)	New	QnA
SM801	Sampling trace start	OFF : Suspend ON : Start	• Sampling trace started when this goes ON • Suspended when OFF (Related special M all OFF)	U	M9047	QnA
SM802	Sampling trace execution in progress	OFF : Suspend ON : Start	• Goes ON during execution of sampling trace	S (Status change)	M9046	QnA
SM803	Sampling trace trigger	OFF → ON: Start	• Sampling trace trigger goes ON when this goes from OFF to ON (Identical to STRA instruction execution status)	U	New	QnA
SM804	After sampling trace trigger	OFF : Not after trigger ON : After trigger	• Goes ON after sampling trace trigger	S (Status change)	New	QnA
SM805	Sampling trace completed	OFF : Not completed ON : End	• Goes ON at completion of sampling trace	S (Status change)	M9043	QnA
SM806	Status latch preparation	OFF : Not ready ON : Ready	• Goes ON when status latch is ready	S (Status change)	New	QnA
SM807	Status latch command	OFF → ON: Latch	• Runs status latch command	U	New	QnA
SM808	Status latch completion	OFF : Latch not completed ON : Latch completed	• Comes ON when status latch is completed.	S (Status change)	M9055	QnA
SM809	Status latch clear	OFF → ON: Clear	• Enable next status latch	U	New	QnA
SM810	Program trace preparation	OFF : Not ready ON : Ready	• Goes ON when program trace is ready	S (Status change)	New	QnA
SM811	Start program trace	OFF : Suspend ON : Start	• Program trace started when this goes ON • Suspended when OFF (Related special M all OFF)	S (Status change)	New	QnA
SM812	Program trace execution under way	OFF : Suspend ON : Start	• ON when program trace execution is underway	U	New	QnA
SM813	Program trace trigger	OFF → ON: Start	• Program trace trigger goes ON when this goes from OFF to ON (Identical to PTRA instruction execution status)	S (Status change)	New	QnA
SM814	After program trace trigger	OFF : Not after trigger ON : After trigger	• Goes ON after program trace trigger	S (Status change)	New	QnA
SM815	Program trace completion	OFF : Not completed ON : End	• Goes ON at completion of program trace	S (Status change)	New	QnA
SM820	Step trace preparation	OFF : Not ready ON : Ready	• Goes ON after program trace registration, at ready	S (Status change)	New	QnA
SM821	Step trace starts	OFF : Suspend ON : Start	• Select whether execution of step trace is started or suspended. • When this goes ON, step trace is started • Suspended when OFF (Related special M all OFF)	U	M9182form at change	QnA
SM822	Step trace execution underway	OFF : Suspend ON : Start	• Goes ON when step trace execution is underway • Goes OFF at completion or suspension	S (Status change)	M9181	QnA
SM823	After step trace trigger	OFF : Not after trigger ON : Is after first trigger	• Goes ON if even 1 block within the step trace being executed is triggered. • Goes OFF when step trace is commenced.	S (Status change)	New	QnA
SM824	After Step trace trigger	OFF : Is not after all triggers ON : Is after all triggers	• Goes ON if all blocks within the step trace being executed are triggered. • Goes OFF when step trace is commenced.	S (Status change)	New	QnA
SM825	Step trace completed	OFF : Not completed ON : End	• Goes ON at step trace completion. • Goes OFF when step trace is commenced.	S (Status change)	M9180	QnA
SM826	Sampling trace error	OFF : Normal ON : Errors	• Goes ON if error occurs during execution of sampling trace.	S (Status change)	New	QnA
SM827	Status latch error	OFF : Normal ON : Errors	• Goes ON if error occurs during execution of status latch.	S (Status change)	New	QnA
SM828	Program trace error	OFF : Normal ON : Errors	• Goes ON if error occurs during execution of program trace.	S (Status change)	New	QnA

(8) Latch area

Table App. 2.9. Special relay

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM900	Power off file	OFF : No power off file ON : Power off file present	• Goes ON if a file is present during access when power is interrupted.	S (Status change)/ U	New	QnA
SM910	RKEY registration flag	OFF : Keyboard input notregistered ON : Keyboard input registered	• Goes ON at registration of keyboard input. • OFF if keyboard input is not registered.	S (Instruction execution)	New	QnA

(9) A to QnA conversion correspondences

Special relays SM1000 to SM1255 are the relays which correspond to ACPU special relays M9000 to M9255 after A to QnA conversion.

These special relays are all set by the system, and cannot be set by the user program.

To turn them ON/OFF by the user program, change the special relays in the program into those of QnACPU.

However, some of SM1084 and SM1200 to SM1255 (corresponding to M9084 and M9200 to M9255 before conversion) can be turned ON/OFF by the user program, if they could be turned ON/OFF by the user program before conversion.For details on the ACPU special relays, see the user's manuals for the individual CPUs, and MELSECNET or MELSECNET/B Data Link System Reference Manuals

The following are additional explanations about the Special Relay for Modification column.

① When a special relay for modification is provided, the device number should be changed to the provided QnACPU special relay.

② When ⊟ is provided, the converted special relay can be used for the device number.

③ When ☒ is provided, the device number does not work with QnACPU.

Table App. 2.10. Special relay

ACPU Special Relay	Special Relay after Conversion	Special Relay for Modification	Name	Meaning	Details	Corresponding CPU
M9000	SM1000	–	Fuse blown	OFF : Normal ON : Module with blown fuse	• Turned on when there is one or more output modules of which fuse has been blown. • Remains ON if the condition is restored to normal thereafter. • Output modules of remote I/O stations are also checked fore fuse condition.	QnA
M9002	SM1002	–	I/O module verify error	OFF : Normal ON : Error	• Turned on if the status of I/O module is different form entered status when power is turned on. • Remains ON if the condition is restored to normal thereafter. • I/O module verification is done also to remote I/O station modules. • Reset is enabled only when special registers SD1116 to SD1123 are reset.	QnA
M9004	SM1004	–	NIMI link master module error	OFF : Normal ON : Error	• Goes ON if MINI (S3) link error is detected at even one of the installed AJ71PT32 (S3) modules. • Remains ON if the condition is restored to normal thereafter.	QnA
M9005	SM1005	—	AC DOWN detection	OFF : AC DOWN not detected ON : AC DOWN detected	• Turns ON if an instantaneous power failure of within 20ms occurs during use of the AC power supply module. • Reset when the power supply is switched OFF, then ON.	QnA
					• Turns ON if an instantaneous power failure of within 1ms occurs during use of the DC power supply module. • Reset when the power supply is switched OFF, then ON.	QnA
M9006	SM1006	–	Battery low	OFF : Normal ON : Battery low	• Turns ON when the battery voltage drops to or below the specified. • Turns OFF when the battery voltage returns to normal thereafter.	QnA
M9007	SM1007	–	Battery low latch	OFF : Normal ON : Battery low	• Turns ON when the battery voltage drops to or below the specified. • Remains ON if the battery voltage returns to normal thereafter.	QnA
M9008	SM1008	SM1	Self-diagnosis error	OFF : No error ON : Error	• Turned on when error is found as a result of self-diagnosis.	QnA
M9009	SM1009	SM62	Annunciator detection	OFF : No F number detected ON : F number detected	• Turned on when OUT F of SET F instruction is executed. • Switched off when SD1124 data is cleared to zero.	QnA

Table App. 2.10. Special relay

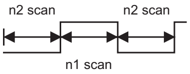
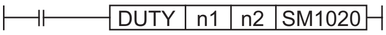
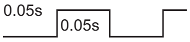
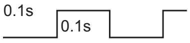
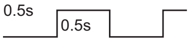
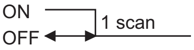
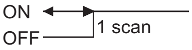
ACPU Special Relay	Special Relay after Conversion	Special Relay for Modification	Name	Meaning	Details	Corresponding CPU
M9011	SM1011	SM56	Operation error flag	OFF : No error ON : Error	• Turned on when operation error occurs during execution of application instruction. • Remains ON if the condition is restored to normal thereafter.	QnA
M9012	SM1012	SM700	Carry flag	OFF : Carry OFF ON : Carry ON	• Carry flag used in application instruction.	QnA
M9016	SM1016	×	Data memory clear flag	OFF : Ignored ON : Output claered	• Clears the data memory including the latch range (other than special relays and special registers) in remote run mode from computer, etc. when SM1016 is on.	–
M9017	SM1017	×	Data memory clear flag	OFF : Ignored ON : Output claered	• Clears the unlatched data memory (other than special relays and special egisters) in remote run mode from computer, etc. when SM1017 is on.	–
M9020	SM1020	–	User timing clock No.0		• Relay which repeats on/off at intervals of predetermined scan. • When power is turned on or reset is per-formed, the clock starts with off. Set the intervals of on/off by DUTY instruction.  n1: ON scan interval n2: OFF scan interval	QnA
M9021	SM1021	–	User timing clock No.1			QnA
M9022	SM1022	–	User timing clock No.2			QnA
M9023	SM1023	–	User timing clock No.3			QnA
M9024	SM1024	–	User timing clock No.4			QnA
M9025	SM1025	–	Clock data set request	OFF : Ignored ON : Set request present used	• Writes the clock data stored in SD1025 to SD1028 to the CPU module after the END instruction is executed in the scan in which SM1025 turned from OFF to ON.	QnA
M9026	SM1026	–	Clock data error	OFF : No error ON : Error	• Switched on by clock data (SD1025 to SD1028) error	QnA
M9027	SM1027	–	Clock data display	OFF : Ignored ON : Display	• Clock data is read from SD1025 to SD1028 and month, day, hour, minute and minute are indicated on the CPU module front LED display.	Q3A Q4A Q4AR
M9028	SM1028	–	Clock data read request	OFF : Ignored ON : Read request	• Reads clock data to SD1025 to SD1028 in BCD when SD1028 is on.	QnA
M9029	SM1029	×	Batch processing of data communications requests	OFF : Batch processing not conducted ON : Batch processing conducted	• The SM1029 relay is turned on using a sequence program to process all data communication requests accepted during one scan in the END processing of that scan. • The batch processing of the data communication requests can be turned on and off during running. • The default is OFF (processed one at a time for each END processing in the order in which data communication requests are accepted).	–
M9030	SM1030	–	0.1 second clock		• 0.1 second, 0.2 second, 1 second and 2 second, clocks are generated. • Not turned on or off per scan but turned on and off even during scan if corresponding time has elapsed. • Starts with off when PLC power supply is turned on or CPU module reset is performed.	QnA
M9031	SM1031	–	0.2 second clock			
M9032	SM1032	–	1 second clock			
M9033	SM1033	–	2 second clock			
M9034	SM1034	–	2n minute clock(1 minute clock)*		• Alternates between ON and OFF according to the seconds specified at SD414. (Default: n = 30) • Not turned on or off per scan but turned on and off even during scan if corresponding time has elapsed. • Starts with off when PLC power supply is turned on or CPU module reset is performed..	QnA
M9036	SM1036	–	Always ON		• Used as dummy contacts of initialization and application instruction in sequence program. • SM1038 and SM1037 are turned on and off without regard to position of key switch on CPU module front. SM1038 and SM1039 are under the same condition as RUN status except when the key switch is at STOP position, and turned off and on. Switched off if the key switch is in STOP position. SM1038 is on for one scan only and SM1039 is off for one scan only if the key switch is not in STOP position.	QnA
M9037	SM1037	–	Always OFF			
M9038	SM1038	–	ON for 1 scan only after RUN			
M9039	SM1039	–	RUN flag(After RUN, OFF for 1 scan only)			

Table App. 2.10. Special relay

ACPU Special Relay	Special Relay after Conversion	Special Relay for Modification	Name	Meaning	Details	Corresponding CPU
M9040	SM1040	SM206	PAUSE enable coil	OFF : PAUSE disabled ON : PAUSE enabled	• When RUN key switch is at PAUSE position or pause contact has turned on and if SM204 is on, PAUSE mode is set and SM206 is turned on.	QnA
M9041	SM1041	SM204	PAUSE status contact	OFF : PAUSE not in effect ON : PAUSE in effect		
M9042	SM1042	SM203	STOP status contact	OFF : STOP not in effect ON : STOP in effect	• Switched on when the RUN key switch or RUN/STOP switch is in STOP position.	QnA
M9043	SM1043	SM805	Sampling trace completed	OFF : Sampling trace in progress ON : Sampling trace completed	• Turns on when sampling trace is performed by the number of times set by the peripheral device after the STRA instruction is executed. This relay is reset by executing the STRAR instruction.	QnA
M9045	SM1045	×	Watchdog timer (WDT) reset	OFF : Does not reset WDT ON : Resets WDT	• The SM1015 relay is turned on to reset the WDT when the ZCOM instruction and data communication request batch processing are executed (used when the scan time exceeds 200 ms).	–
M9046	SM1046	SM802	Sampling trace	OFF : Trace not in progress ON : Trace in progress	• Switched on during sampling trace.	QnA
M9047	SM1047	SM801	Sampling trace preparations	OFF : Sampling trace suspended ON : Sampling trace started	• Sampling trace is not executed unless SM801 is turned ON. Sampling trace is suspended when SM801 goes OFF.	QnA
M9049	SM1049	SM701	Switching the number of output characters	OFF : Output until NULL code encountered ON : 16 characters output	• When SM701 is OFF, characters up to NULL (00H) code are output. • When SM701 is ON, ASCII codes of 16 characters are output.	QnA
M9051	SM1051	×	CHG instruction execution disable	OFF : Enabled ON : Disable	• Switched ON to disable the CHG instruction. • Switched ON when program transfer is requested. Automatically switched OFF when transfer is complete.	–
M9052	SM1052	×	SEG instruction switch	OFF : 7SEG segment display ON : I/O partial refresh	• When SM1052 is ON, the SEG instruction is executed as an I/O partial refresh instruction. When SM1052 is OFF, the SEG instruction is executed as a 7-SEG display instruction.	–
M9054	SM1054	SM205	STEP RUN flag	OFF : STEP RUN not in effect ON : STEP RUN in effect	• Switched on when the RUN key switch is in STEP RUN position.	QnA
M9055	SM1055	SM808	Status latch completion flag	OFF : Not completed ON : Completed	• Turned on when status latch is completed. • Turned off by reset instruction.	QnA
M9056	SM1056	×	Main side P, I set request	OFF : Other than when P, I set being requested ON : P, I set being requested	• Provides P, I set request after transfer of the other program (for example subprogram when main program is being run) is complete during run. Automatically switched off when P, I setting is complete.	–
M9057	SM1057	×	Sub side P, I set request	OFF : Other than when P, I set being requested ON : P, I set being requested		–
M9058	SM1058	×	Main side P, I set completion	Momentarily ON at P, I set completion	• Turned ON once when the P, I set has been completed, and then turned OFF again.	–
M9059	SM1059	×	Sub program P, I set completion	Momentarily ON at P, I set completion		–
M9060	SM1060	×	Sub program 2 P, I set request	OFF : Other than when P, I set being requested ON : P, I set being requested	• Provides P, I set request after transfer of the other program (for example subprogram when main program is being run) is complete during run. Automatically switched off when P, I setting is complete.	–
M9061	SM1061	×	Sub program 3 P, I set request	OFF : Other than when P, I set being requested ON : P, I set being requested		–

*: 1 minute clock indicates the name of the special relay (M9034) of the ACPU.

Table App. 2.10. Special relay

ACPU Special Relay	Special Relay after Conversion	Special Relay for Modification	Name	Meaning	Details	Corresponding CPU
M9065	SM1065	SM711	Divided transfer status	OFF : Divided processing not underway ON : During divided processing	• Turned on when canvas screen transfer to AD57(S1)/ AD58 is done by divided processing, and turned off at completion of divided processing	QnA
M9066	SM1066	SM712	Transfer processing switch	OFF : Batch transfer ON : Divided transfer	• Turned on when canvas screen transfer to AD57(S1)/ AD58 is done by divided processing.	QnA
M9070	SM1070	×	A8UPU/ A8PUJrequired search time*2	OFF : Read time not shortened ON : Read time shortened	• Turned ON to shorten the search time in the A8UPU/ A8PUJ. (In this case, the scan time is extended by 10 %.)	–
M9081	SM1081	SM714	Communication request registration area BUSY signal	OFF : Empty spaces in communication request registration area ON : No empty spaces in communication request registration area	• Indication of communication enable/disable to remote terminal modules connected to the AJ71PT32-S3, A2C or A52G.	QnA
M9084	SM1084	×	Error check	OFF : Error check executed ON : No error check	It is set whether the error checks below are performed or not when the END instruction is processed (to set the END instruction processing time). • Check for fuse blown. • Check of battery • Collation check of I/O module	–
M9091	SM1091	×	Operation error details flag	OFF : No error ON : Error	• Turns ON when the detail factor of the operation error is stored into SD1091. • Remains ON if the condition is restored to normal thereafter.	–
M9094	SM1094	SM251	I/O exchange flag	OFF : Exchanged ON : Not exchanged	• The I/O module can be changed online (with power on) when SM251 is turned ON after the head I/O number of the I/O module is set to SD251. (One module only is allowed to be changed by one setting.) • To be switched on in the program or peripheral device test mode to change the module during CPU RUN. To be switched on in peripheral device test mode to change the module during CPU STOP. • RUN/STOP mode must not be changed until I/O module change is complete.	QnA
M9100	SM1100	SM320	Presence/absence of SFC program	OFF : SFC programs not used ON : SFC programs used	• Turned on if the SFC program is registered. • Turned off if the SFC program is not registered.	QnA
M9101	SM1101	SM321	Start/stop SFC program	OFF : SFC programs stop ON : SFC programs start	• The value in SM320 is set as the initial value. (The relay automatically turns ON when the SFC program is present.) • When this relay turns from ON to OFF, execution of the SFC program stops. • When this relay turns from OFF to ON, execution of the SFC program resumes.	QnA
M9102	SM1102	SM322	SFC program start status	OFF : Initial start ON : Resume start	• The SFC program start mode in the SFC setting of the PLC parameter dialog box is set as the initial value. At initial start: OFF At continue start: ON	QnA

*2: The A8UPU/A8PUJ is not available for the QnACPU.

Table App. 2.10. Special relay

ACPU Special Relay	Special Relay after Conversion	Special Relay for Modification	Name	Meaning	Details	Corresponding CPU
M9103	SM1103	SM323	Presence/absence of continuous transition	OFF : Continuous transition not effective ON : Continuous transition effective	• Set whether continuous transition will be performed for the block where the "continuous transition bit" of the SFC information device is not set.	QnA
M9104	SM1104	SM324	Continuous transition suspension flag	OFF : When transition is completed ON : When no transition	• OFF during operation in the continuous transition mode or during continuous transition, and ON when continuous transition is not executed. • Always ON during operation in the no continuous transition mode.	QnA
M9108	SM1108	SM90	Step transition monitoring timer start (equivalent of SD90)	OFF : Monitoring timer reset ON : Monitoring timer reset start	• Turns ON when the measurement of the step transition monitoring timer is started. Turning this relay OFF resets the step transition monitoring timer.	QnA
M9109	SM1109	SM91	Step transition monitoring timer start (equivalent of SD91)			
M9110	SM1110	SM92	Step transition monitoring timer start (equivalent of SD92)			
M9111	SM1111	SM93	Step transition monitoring timer start (equivalent of SD93)			
M9112	SM1112	SM94	Step transition monitoring timer start (equivalent of SD94)			
M9113	SM1113	SM95	Step transition monitoring timer start (equivalent of SD95)			
M9114	SM1114	SM96	Step transition monitoring timer start (equivalent of SD96)			
M9180	SM1180	SM825	Active step sampling trace completion flag	OFF : Trace started ON : Trace completed	• Set when sampling trace of all specified blocks is completed. Reset when sampling trace is started.	QnA
M9181	SM1181	SM822	Active step sampling trace execution flag	OFF : Trace not being executed ON : Trace execution under way	• Set when sampling trace is being executed.Reset when sampling trace is completed or suspended	QnA
M9182	SM1182	SM821	Active step sampling trace permission	OFF : Trace disable/ suspend ON : Trace enable	• Selects sampling trace execution enable/disable. ON: Sampling trace execution is enabled. OFF: Sampling trace execution is disabled. If turned off during sampling trace execution, trace is suspended.	QnA
M9196	SM1196	SM325	Operation output at block stop	OFF : Coil output OFF ON : Coil output ON	• Selects the operation output when block stop is executed. ON: Retains the ON/OFF status of the coil being used by using operation output of the step being executed at block stop. OFF: All coil outputs are turned off. (Operation output by the SET instruction is retained regardless of the ON/OFF status of M9196.)	QnA
M9197	SM1197	×	Switch between blown fuse and I/O verify error display	SM 1197 / SM 1198 / I/O numbers to be displayed OFF / OFF / X/Y0 to 7F0 ON / OFF / X/Y800 to FF0 OFF / ON / X/Y1000 to 17F0 ON / ON / X/Y1800 to 1FF0	Switches I/O numbers in the fuse blow module storage registers (SD1100 to SD1107) and I/O module verify error storage registers (SD1116 to SD1123) according to the combination of ON/OFF of the SM1197 and SM1198.	–
M9198	SM1198	×				
M9199	SM1199	×	Data recovery of online sampling trace/status latch	OFF : Data recovery disabled ON : Data recovery enabled	• Recovers the setting data stored in the CPU module at restart when sampling trace/status latch is executed. • SM1199 should be ON to execute again. (Unnecessary when writing the data again from peripheral devices.)	–

Table App. 2.10. Special relay

ACPU Special Relay	Special Relay after Conversion	Special Relay for Modification	Name	Meaning	Details	Corresponding CPU
M9200	SM1200	–	ZNRD instruction (LRDP instruction for ACPU) reception (for master station)	OFF : Not accepted ON : Accepted	• Depends on whether or not the ZNRD (word device read) instruction has been received. • Used in the program as an interlock for the ZNRD instruction. • Use the RST instruction to reset.	QnA
M9201	SM1201	–	ZNRD instruction (LRDP instruction for ACPU) completion (for master station)	OFF : Not completed ON : End	• Depends on whether or not the ZNRD (word device read) instruction execution is complete. • Used as a condition contact for resetting M9200 and M9201 after the ZNRD instruction is complete. • Use the RST instruction to reset.	QnA
M9202	SM1202	–	ZNWR instruction (LWTP instruction for ACPU) reception (for master station)	OFF : Not accepted ON : Accepted	• Depends on whether or not the ZNWR (word device write) instruction has been received. • Used in the program as an interlock for the ZNWR instruction. • Use the RST instruction to reset.	QnA
M9203	SM1203	–	ZNWR instruction (LWTP instruction for ACPU) completion (for master station)	OFF : Not completed ON : End	• Depends on whether or not the ZNWR (word device write) instruction execution is complete. • Used as a condition contact to reset M9202 and M9203 after the ZNWR instruction is complete. • Use the RST instruction to reset.	QnA
M9204	SM1204	–	ZNRD instruction (LRDP instruction for ACPU) reception (for local station)	OFF : Not completed ON : End	• On indicates that the ZNRD instruction is complete at the local station.	QnA
M9205	SM1205	–	ZNWR instruction (LWTP instruction for ACPU) reception (for local station)	OFF : Not completed ON : End	• On indicates that the ZNWR instruction is complete at the local station.	QnA
M9206	SM1206	–	Host station link parameter error	OFF : Normal ON : Abnormal	• Depends on whether or not the link parameter setting of the host is valid.	QnA
M9207	SM1207	–	Link parameter check results	OFF : Match ON : Mismatch	• Depends on whether or not the link parameter setting of the master station in tier two matches that of the master station in tier three in a three-tier system. (Valid for only the master station in a three-tier system.)	QnA
M9208	SM1208	–	Sets master station B and W transmission range (for lower link master stations only)	OFF : Transmits to tier2 and tier 3 ON : Transmits to tier2 only	• Depends on whether or not the B and W data controlled by higher-link master station (host station) is sent to lower-link local stations (tertiary stations). When SM1208 is OFFB and W of host station is sent to tertiary stations. When SM1208 is ONB and W of host station is not sent to tertiary stations.	QnA
M9209	SM1209	–	Link parameter check command (for lower link master stations only)	OFF : Executing the check function ON : Check non-execution	• Set to ON not to match B and W of the higher and lower links. • When SM1209 is ON, the link parameters of the higher and lower link are not checked. • When SM1209 is OFF, the link parameters of the higher and lower link are checked.	QnA
M9210	SM1210	–	Link card error (for master station)	OFF : Normal ON : Abnormal	Control is performed depending on whether the link card hardware is faulty or not.	QnA
M9211	SM1211	–	Link module error (for local station use)	OFF : Normal ON : Abnormal	Control is performed depending on whether the link card hardware is faulty or not.	QnA
M9224	SM1224	–	Link status	OFF : Online ON : Offline,station-to-station test, or self-loopback test	Depends on whether the master station is online or offline or is in station-to-station test or self-loopback test mode.	QnA
M9225	SM1225	–	Forward loop error	OFF : Normal ON : Abnormal	Depends on the error condition of the forward loop line.	QnA
M9226	SM1226	–	Reverse loop error	OFF : Normal ON : Abnormal	Depends on the error condition of the reverse loop line.	QnA
M9227	SM1227	–	Loop test status	OFF : Not being executed ON : Forward or reverse loop test execution underway	Depends on whether or not the master station is executing a forward or a reverse loop test.	QnA
M9232	SM1232	–	Local station operation status	OFF : RUN or STEP RUN status ON : STOP or PAUSE status	Control is performed depending on whether a local station is in the STOP or PAUSE mode.	QnA
M9233	SM1233	–	Local station error detect status	OFF : No errors ON : Error detection	Depends on whether or not a local station has detected an error in another station.	QnA

Table App. 2.10. Special relay

ACPU Special Relay	Special Relay after Conversion	Special Relay for Modification	Name	Meaning	Details	Corresponding CPU
M9235	SM1235	–	Local station, remote I/O station parameter error detect status	OFF : No errors ON : Error detection	Depends on whether or not a local or a remote I/O station has detected any link parameter error in the master station	QnA
M9236	SM1236	–	Local station, remote I/O station initial communications status	OFF : No communications ON : Communications underway	Depends on the results of initial communication between a local or remote I/O station and the master station. (Parameter communication, etc.)	QnA
M9237	SM1237	–	Local station, remote I/O station error	OFF : Normal ON : Abnormal	Depends on the error condition of a local or remote I/O station.	QnA
M9238	SM1238	–	Local station, remote I/O station forward or reverse loop error	OFF : Normal ON : Abnormal	Depends on the error condition of the forward and reverse loop lines of a local or a remote I/O station.	QnA
M9240	SM1240	–	Link status	OFF : Online ON : Offline, station-to-stationtest, or self-loopback test	Depends on whether the local station is online or offline, or is in station-to-station test or self-loopback test mode.	QnA
M9241	SM1241	–	Forward loop line error	OFF : Normal ON : Abnormal	Depends on the error condition of the forward loop line.	QnA
M9242	SM1242	–	Reverse loop line error	OFF : Normal ON : Abnormal	Depends on the error condition of the reverse loop line.	QnA
M9243	SM1243	–	Loopback implementation	OFF : Loopback not being conducted ON : Loopback implementation	Depends on whether or not loopback is occurring at the local station.	QnA
M9246	SM1246	–	Data not received	OFF : Reception ON : No reception	Depends on whether or not data has been received from the master station.	QnA
M9247	SM1247	–	Data not received	OFF : Reception ON : No reception	Depends on whether or not a tier three station has received data from its master station in a three-tier system.	QnA
M9250	SM1250	–	Parameters not received	OFF : Reception ON : No reception	Depends on whether or not link parameters have been received from the master station.	QnA
M9251	SM1251	–	Link relay	OFF : Normal ON : Abnormal	Depands on the data link condition at the local station.	QnA
M9252	SM1252	–	Loop test status	OFF : Not being executed ON : Forward or reverse loop test execution underway	Depends on whether or not the local station is executing a forward or a reverse loop test.	QnA
M9253	SM1253	–	Master station operation status	OFF : RUN or STEP RUN status ON : STOP or PAUSE status	Control is performed depending on whether the master station is in the STOP or PAUSE mode.	QnA
M9254	SM1254	–	Local station other than host station operation status	OFF : RUN or STEP RUN status ON : STOP or PAUSE status	Control is performed depending on whether a local station other than the host is in the STOP or PAUSE mode.	QnA
M9255	SM1255	–	Local station other than host station error	OFF : Normal ON : Abnormal	Depends on whether or not a local station other than the host is in error.	QnA

(10) Process control instructions

Table App. 2.11. Special relay

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM1500	Hold mode	OFF : No-hold ON : Hold	• Specifies whether or not to hold the output value when a range over occurs for the S.IN instruction range check.	U	New	Q4AR
SM1501	Hold mode	OFF : No-hold ON : Hold	• Specifies whether or not the output value is held when a range over occurs for the S.OUT instruction range check.	U	New	Q4AR

(11) For redundant systems (Host system CPU information *1)

SM1510 to SM1599 are only valid for redundant systems.

All off for standalone systems.

Table App. 2.12. Special relay

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM1510	Operation mode	OFF : Redundant system backup mode, stand-alone system ON : Redundant system separate mode	• Turns on when the operating mode is redundant system separate.	S (Each END)	New	Q4AR
SM1511	Start mode when power supply is on	OFF : System A fixed mode ON : Previous control system latch mode	• Turns on when the start mode for a redundant system when the power is turned on is the previous control system latch mode.	S (Initial)	New	Q4AR
SM1512	Start mode when CPU is started	OFF : Initial start ON : Hot start	• Turns on when the CPU module operation mode is hot start when the redundant system is started up.	S (Initial)	New	Q4AR
SM1513	Operation status when CPU is started	OFF : Initial start ON : Hot start	• Turns on when the CPU module operation mode is hot start when the redundant system is actually start up.	S (Initial)	New	Q4AR
SM1514	Operation mode at CPU module change	OFF : Initial start ON : Hot start	• Turns on when the operation is hot start when the CPU module operation is switched for a redundant system.	S (Initial)	New	Q4AR
SM1515	Output hold mode	OFF : Output reset ON : Output hold	• Turns on when the output mode during a stop error is output hold.	S (Each END)	New	Q4AR
SM1516	Operation system status	OFF : Control system ON : Standby system	• Turns on when the CPU module operation system status is the standby system.	S (Status change)	New	Q4AR

*1: The information of the host CPU module is stored.

Table App. 2.12. Special relay

Number	Name	Meaning	Explanation			Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM1517	CPU module startup status	OFF : Power supply on startup ON : Operation system switch start up	• Turns on when the CPU module is started up by the operation system switch. • Reset using the user program.			S (Status change)/ U	New	Q4AR
SM1518	Tracking execution mode	OFF : Batch transfer mode ON : Carryover mode	• When this relay is turned OFF, the start of tracking is delayed until it is executable if the tracking memory is being used at END. • When this relay is turned ON, the start of tracking is carried over to next END if the tracking memory is being used at END.			U	New	Q4AR
SM1520	Data tracking transfer trigger specification	OFF : No trigger ON : Trigger	SM1520	Block 1	When data is transferred by the data tracking instruction S. TRUCK, the target block is specified as trigger.	U	New	Q4AR
SM1521			SM1521	Block 2				
SM1522			SM1522	Block 3				
SM1523			SM1523	Block 4				
SM1524			SM1524	Block 5				
SM1525			SM1525	Block 6				
SM1526			SM1526	Block 7				
SM1527			SM1527	Block 8				
SM1528			SM1528	Block 9				
SM1529			SM1529	Block 10				
SM1530			SM1530	Block 11				
SM1531			SM1531	Block 12				
SM1532			SM1532	Block 13				
SM1533			SM1533	Block 14				
SM1534			SM1534	Block 15				
SM1535			SM1535	Block 16				
SM1536			SM1536	Block 17				
SM1537			SM1537	Block 18				
SM1538			SM1538	Block 19				
SM1539			SM1539	Block 20				
SM1540			SM1540	Block 21				
SM1541			SM1541	Block 22				
SM1542			SM1542	Block 23				
SM1543			SM1543	Block 24				
SM1544			SM1544	Block 25				
SM1545			SM1545	Block 26				
SM1546			SM1546	Block 27				
SM1547			SM1547	Block 28				
SM1548			SM1548	Block 29				

Table App. 2.12. Special relay

Number	Name	Meaning	Explanation			Set by (When Set)	Corresponding ACPU M9□□□	Corresponding CPU
SM1549	Data tracking transmission link specification	OFF : No trigger ON : Trigger	SM1549	Block 30	When data is transferred by the data tracking instruction S. TRUCK, the target block is specified as trigger.	U	New	Q4AR
SM1550			SM1550	Block 31				
SM1551			SM1551	Block 32				
SM1552			SM1552	Block 33				
SM1553			SM1553	Block 34				
SM1554			SM1554	Block 35				
SM1555			SM1555	Block 36				
SM1556			SM1556	Block 37				
SM1557			SM1557	Block 38				
SM1558			SM1558	Block 39				
SM1559			SM1559	Block 40				
SM1560			SM1560	Block 41				
SM1561			SM1561	Block 42				
SM1562			SM1562	Block 43				
SM1563			SM1563	Block 44				
SM1564			SM1564	Block 45				
SM1565			SM1565	Block 46				
SM1566			SM1566	Block 47				
SM1567			SM1567	Block 48				
SM1568			SM1568	Block 49				
SM1569			SM1569	Block 50				
SM1570			SM1570	Block 51				
SM1571			SM1571	Block 52				
SM1572			SM1572	Block 53				
SM1573			SM1573	Block 54				
SM1574			SM1574	Block 55				
SM1575			SM1575	Block 56				
SM1576			SM1576	Block 57				
SM1577			SM1577	Block 58				
SM1578			SM1578	Block 59				
SM1579			SM1579	Block 60				
SM1580			SM1580	Block 61				
SM1581			SM1581	Block 62				
SM1582			SM1582	Block 63				
SM1583			SM1583	Block 64				
SM1590	Switching status from the network module	OFF : Normal ON : Switching unsuccessful	• Turns ON when switching could not be executed normally if the network module had detected a network fault and issued a switching request to the host CPU module.			S (Error ocurrs)	New	Q4AR

(12)For redundant system (Other system CPU information *1)

SM1600 to SM1650 only valid for the CPU redundant system backup mode, so they cannot be refreshed during the separate mode.
Either the backup mode or the separate mode is valid for the SM4651 to SM1699.

SM1600 to SM1699 are all turned off for stand-alone system.

Table App. 2.13. Special relay

Number	Name	Meaning	Explanation	Set by (When Set)	Corresponding Host SM□□ *2	Corresponding CPU
SM1600	Diagnosis error	OFF : No error ON : Error	• Turns on if a error occurs in the diagnosis results. (Including external diagnosis) • Remains on even if returns to normal thereafter.	S (Each END)	SM0	Q4AR
SM1601	Self diagnosis error	OFF : No self diagnosis error ON : Self diagnosis error	• Turns on when an error occurs in the self-diagnosis results. • Remains on even if returns to normal thereafter.	S (Each END)	SM1	Q4AR
SM1605	Error common information	OFF : No error common information ON : Error common information	• Turns on when there is error common information and the SM1600 is on.	S (Each END)	SM5	Q4AR
SM1616	Error individual information	OFF : No error individual information ON : Error individual information	• Turns on when there is error individual information and the SM1600 is on.	S (Each END)	SM16	Q4AR
SM1653	STOP contact	STOP status	• Turns on when in the STOP status.	S (Each END)	SM203	Q4AR
SM1654	PAUSE contact	PAUSE status	• Turns on when in the PAUSE status.	S (Each END)	SM204	Q4AR
SM1655	STEP-RUN contact	STEP-RUN status	• Turns on when in the STEP-RUN status.	S (Each END)	SM205	Q4AR

*1 Stores other system CPU diagnostic information and system information.

*2 This shows the special relay(SM☐ ☐) for the host system CPU.

(13)For redundant system (tracking)

Either the backup mode or the separate mode is valid for SM1700 to SM1799.

All is turned off for stand-alone system.

Table App. 2.14. Special relay

Number	Name	Meaning	Explanation			Set by (When Set)	Corresponding ACPU M9□□□	Corresponding CPU
SM1700	Tracking execution flag	OFF : Execution not possible ON : Execution possible	• Turns on when tracking can be normally executed.			S (status change)	New	Q4AR
SM1712	Transfer trigger completion flag	OFF : Transfer uncompleted ON : Transfer completed	SM1712	Block 1	Turns ON only during one scan when the transmission of the corresponding data is completed.	S (status change)	New	Q4AR
SM1713			SM1713	Block 2				
SM1714			SM1714	Block 3				
SM1715			SM1715	Block 4				
SM1716			SM1716	Block 5				
SM1717			SM1717	Block 6				
SM1718			SM1718	Block 7				
SM1719			SM1719	Block 8				
SM1720			SM1720	Block 9				
SM1721			SM1721	Block 10				
SM1722			SM1722	Block 11				
SM1723			SM1723	Block 12				
SM1724			SM1724	Block 13				
SM1725			SM1725	Block 14				
SM1726			SM1726	Block 15				
SM1727			SM1727	Block 16				
SM1728			SM1728	Block 17				
SM1729			SM1729	Block 18				
SM1730			SM1730	Block 19				
SM1731			SM1731	Block 20				
SM1732			SM1732	Block 21				
SM1733			SM1733	Block 22				
SM1734			SM1734	Block 23				
SM1735			SM1735	Block 24				
SM1736			SM1736	Block 25				
SM1737			SM1737	Block 26				
SM1738			SM1738	Block 27				
SM1739			SM1739	Block 28				
SM1740			SM1740	Block 29				
SM1741			SM1741	Block 30				
SM1742			SM1742	Block 31				
SM1743			SM1743	Block 32				
SM1744			SM1744	Block 33				
SM1745			SM1745	Block 34				
SM1746			SM1746	Block 35				
SM1747			SM1747	Block 36				
SM1748			SM1748	Block 37				
SM1749			SM1749	Block 38				
SM1750			SM1750	Block 39				
SM1751			SM1751	Block 40				
SM1752			SM1752	Block 41				
SM1753			SM1753	Block 42				
SM1754			SM1754	Block 43				
SM1755			SM1755	Block 44				
SM1756			SM1756	Block 45				
SM1757			SM1757	Block 46				
SM1758			SM1758	Block 47				
SM1759			SM1759	Block 48				

Table App. 2.14. Special relay

Number	Name	Meaning	Explanation			Set by (When Set)	Corres-ponding ACPU M9□□□	Corresponding CPU
SM1760	Transmission trigger end flag	OFF : Transmission uncompleted ON : Transmission end	SM1760	Block 49	Turns ON only during one scan when the transmission of the corresponding data is completed.	S (status change)	New	Q4AR
SM1761			SM1761	Block 50				
SM1762			SM1762	Block 51				
SM1763			SM1763	Block 52				
SM1764			SM1764	Block 53				
SM1765			SM1765	Block 54				
SM1766			SM1766	Block 55				
SM1767			SM1767	Block 56				
SM1768			SM1768	Block 57				
SM1769			SM1769	Block 58				
SM1770			SM1770	Block 59				
SM1771			SM1771	Block 60				
SM1772			SM1772	Block 61				
SM1773			SM1773	Block 62				
SM1774			SM1774	Block 63				
SM1775			SM1775	Block 64				

APPENDIX 3 Special Register List

The special registers, SD, are internal registers with fixed applications in the PLC.
For this reason, it is not possible to use these registers in sequence programs in the same way that normal registers are used.
However, data can be written as needed in order to control the CPU modules and remote I/O modules.
Data stored in the special registers are stored as BIN values if no special designation has been made to the contrary.

The heading descriptions in the following special register lists are shown in Table App. 3.1.

Table App. 3.1. Special register

Item	Function of Item
Number	• Indicates special register number
Name	• Indicates name of special register
Meaning	• Indicates contents of special register
Explanation	• Discusses contents of special register in more detail
Set by (When set)	• Indicates whether the relay is set by the system or user, and, if it is set by the system, when setting is performed. <Set by> S : Set by system U : Set by user (sequence programs or test operations from GX Developer) S/U : Set by both system and user <When set> Indicated only for registers set by system Each END : Set during each END processing Initial : Set only during initial processing (when power supply is turned ON, or when going from STOP to RUN) Status change : Set only when there is a change in status Error : Set when error occurs Instruction execution : Set when instruction is executed Request : Set only when there is a user request (through SM, etc.) System switching : Set when system switching is executed.
Corresponding ACPU M9□□□	• Indicates corresponding special register in ACPU (When the contents are changed, the special register is represented D9□□□ format change.) • New indicates the special register newly added to the QnACPU.
Corresponding CPU	Indicates the relevant CPU module. QnA : Can be applied to QnA series and Q2ASCPU series Each CPU type name : Can be applied only to the specific CPU. (e.g. Q4AR, Q2AS)

For details on the following items, refer to the following manuals:

- Networks → Manual of the corresponding network module
- SFC → QCPU(Q mode)/QnACPU Programming Manual (SFC)

(1) Diagnostic Information

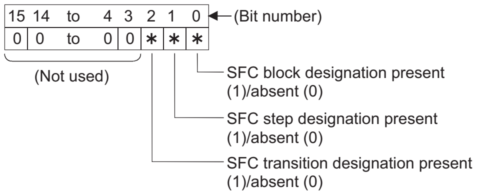
Table App. 3.2. Special register

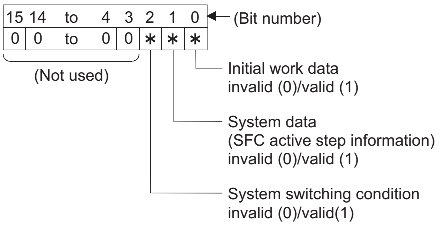
Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD0	Diagnostic errors	Diagnosis error code	• Error codes for errors found by diagnosis are stored as BIN data. • Contents identical to latest fault history information.	S (Error)	D9008 format change	QnA
SD1	Clock time for diagnosis error occurrence	Clock time for diagnosis error occurrence	• Year (last two digits) and month that SD0 data was updated is stored as BCD 2-digit code. b15 to b8 b7 to b0 Year (0 to 99) Month (1 to 12) (Example) October, 1995 H9510	S (Error)	New	QnA
SD2			• The day and hour that SD0 was updated is stored as BCD 2-digit code. b15 to b8 b7 to b0 Day (1 to 31) Hour (0 to 23) (Example) 10 a.m. on 25th H2510			QnA
SD3			• The minute and second that SD0 data was updated is stored as BCD 2-digit code. b15 to b8 b7 to b0 Minutes (0 to 59) Seconds (0 to 59) (Example) 35 min. 48 sec. H3548			QnA
SD4	Error information categories	Error information category code	Category codes which help indicate what type of information is being stored in the common information areas (SD5 through SD15) and the individual information areas (SD16 through SD26) are stored here. The category code for judging the the error information type is stored. b15 to b8 b7 to b0 Individual information category codes Common information category codes • The common information category codes store the following codes: 0 : No error 1: Unit/module No. 2: File name/Drive name 3: Time (value set) 4: Program error location 5: System switching cause (for Q4ARCPU only) 6: Power supply No. • The individual information category codes store the following codes: 0: No error 1: (Empty) 2: File name/Drive name 3: Time (value actually measured) 4: Program error location 5: Parameter number 6: Annunciator number 7: CHK instruction failure No.	S (Error)	New	QnA

Table App. 3.2. Special register

<table>
<tr><th>Number</th><th>Name</th><th>Meaning</th><th>Explanation</th><th>Set by (When Set)</th><th>Corres-ponding ACPU D9□□□</th><th>Corresponding CPU</th></tr>
<tr><td>SD5</td><td rowspan="11">Error common information</td><td rowspan="11">Error common information</td><td rowspan="11">
• Common information corresponding to the error codes (SD0) is stored here.
• The following five types of information are stored here:
• The error common information type can be judged by the "common information category code" in SD4. (The values of the "common information category code" stored in SD4 correspond to following 1) to 5).)
1) Slot No.
<table>
<tr><th>Number</th><th>Meaning</th></tr>
<tr><td>SD5</td><td>Slot No. *1</td></tr>
<tr><td>SD6</td><td>I/O No. *2</td></tr>
<tr><td>SD7</td><td rowspan="9">(Empty)</td></tr>
<tr><td>SD8</td></tr>
<tr><td>SD9</td></tr>
<tr><td>SD10</td></tr>
<tr><td>SD11</td></tr>
<tr><td>SD12</td></tr>
<tr><td>SD13</td></tr>
<tr><td>SD14</td></tr>
<tr><td>SD15</td></tr>
</table>
*1: Definitions of slot No.
<Slot No.>
Value used to identify the slot of each base unit and the module mounted on that slot.
• The I/O slot 0 (slot on the right side of the CPU slot) of the main base unit is defined as the slot of "Slot No. = 0".
• The slot Nos. are consecutively assigned to the slots of the base units in order of the main base unit and extension base units 1 to 7.
• When the number of base unit slots has been set in the I/O assignment setting of the PLC parameter dialog box, the slot Nos. are assigned for only the number of set slots.
*2: When 0FFFFH is stored into SD6 (I/O No.), the I/O No. cannot be identified due to overlapping I/O No., etc. in the I/O assignment setting of the PLC parameter dialog box. Therefore, identify the error location using SD5.
2) File name/Drive name
<table>
<tr><th>Number</th><th colspan="2">Meaning</th></tr>
<tr><td>SD5</td><td colspan="2">Drive</td></tr>
<tr><td>SD6</td><td colspan="2" rowspan="4">File name
(ASCII code: 8 characters)</td></tr>
<tr><td>SD7</td></tr>
<tr><td>SD8</td></tr>
<tr><td>SD9</td></tr>
<tr><td>SD10</td><td>Extension *3</td><td>2EH(.)</td></tr>
<tr><td>SD11</td><td colspan="2">(ASCII code: 3 characters)</td></tr>
<tr><td>SD12</td><td colspan="2" rowspan="4">(Empty)</td></tr>
<tr><td>SD13</td></tr>
<tr><td>SD14</td></tr>
<tr><td>SD15</td></tr>
</table>
(Example) File name = ABCDEFGH. IJK
<table>
<tr><th>b15 to b8</th><th>b7 to b0</th></tr>
<tr><td>42H(B)</td><td>41H(A)</td></tr>
<tr><td>44H(D)</td><td>43H(C)</td></tr>
<tr><td>46H(F)</td><td>45H(E)</td></tr>
<tr><td>48H(H)</td><td>47H(G)</td></tr>
<tr><td>49H(I)</td><td>2EH(.)</td></tr>
<tr><td>4BH(K)</td><td>4AH(J)</td></tr>
</table>
</td><td rowspan="11">S (Error)</td><td rowspan="11">New</td><td rowspan="11">QnA</td></tr>
<tr><td>SD6</td></tr>
<tr><td>SD7</td></tr>
<tr><td>SD8</td></tr>
<tr><td>SD9</td></tr>
<tr><td>SD10</td></tr>
<tr><td>SD11</td></tr>
<tr><td>SD12</td></tr>
<tr><td>SD13</td></tr>
<tr><td>SD14</td></tr>
<tr><td>SD15</td></tr>
</table>

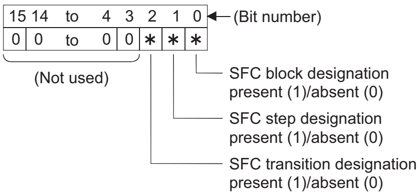
Table App. 3.2. Special register

<table>
<tr><th>Number</th><th>Name</th><th>Meaning</th><th>Explanation</th><th>Set by
(When Set)</th><th>Corres-ponding ACPU D9□□□</th><th>Corresponding CPU</th></tr>
<tr><td>SD5</td><td rowspan="11">Error common information</td><td rowspan="11">Error common information</td><td rowspan="6">
3) Time (value set)
<table>
<tr><th>Number</th><th>Meaning</th></tr>
<tr><td>SD5</td><td>Time : 1μs units (0 to 999μs)</td></tr>
<tr><td>SD6</td><td>Time : 1ms units (0 to 65535ms)</td></tr>
<tr><td>SD7</td><td rowspan="9">(Empty)</td></tr>
<tr><td>SD8</td></tr>
<tr><td>SD9</td></tr>
<tr><td>SD10</td></tr>
<tr><td>SD11</td></tr>
<tr><td>SD12</td></tr>
<tr><td>SD13</td></tr>
<tr><td>SD14</td></tr>
<tr><td>SD15</td></tr>
</table>
4) Program error location
<table>
<tr><th>Number</th><th colspan="2">Meaning</th></tr>
<tr><td>SD5</td><td colspan="2" rowspan="4">File name
(ASCII code: 8 characters)</td></tr>
<tr><td>SD6</td></tr>
<tr><td>SD7</td></tr>
<tr><td>SD8</td></tr>
<tr><td>SD9</td><td>Extension *3</td><td>2EH(.)</td></tr>
<tr><td>SD10</td><td colspan="2">(ASCII code: 3 characters)</td></tr>
<tr><td>SD11</td><td colspan="2">Pattern *4</td></tr>
<tr><td>SD12</td><td colspan="2">Block No.</td></tr>
<tr><td>SD13</td><td colspan="2">Step No./transition condition</td></tr>
<tr><td>SD14</td><td colspan="2">Sequence step No. (L)</td></tr>
<tr><td>SD15</td><td colspan="2">Sequence step No. (H)</td></tr>
</table>
*4 : Contents of pattern data


</td><td rowspan="6">S (Error)</td><td rowspan="6">New</td><td rowspan="6">QnA</td></tr>
<tr><td>SD6</td></tr>
<tr><td>SD7</td></tr>
<tr><td>SD8</td></tr>
<tr><td>SD9</td></tr>
<tr><td>SD10</td></tr>
<tr><td>SD11</td><td rowspan="5">
5) Reason(s) for system switching
<table>
<tr><th>Number</th><th>Meaning</th></tr>
<tr><td>SD5</td><td>System switching condition
(0: automatic system switching/
1: manual system switching)</td></tr>
<tr><td>SD6</td><td>System switching direction
(0:standby system to control system/
1: control system to standby system)</td></tr>
<tr><td>SD7</td><td>Tracking flag *5</td></tr>
<tr><td>SD8</td><td rowspan="8">(Empty)</td></tr>
<tr><td>SD9</td></tr>
<tr><td>SD10</td></tr>
<tr><td>SD11</td></tr>
<tr><td>SD12</td></tr>
<tr><td>SD13</td></tr>
<tr><td>SD14</td></tr>
<tr><td>SD15</td></tr>
</table>
*5 : Tracking flag contents
Shows whether or not the tracking data is valid.


</td><td rowspan="5">S (Error)</td><td rowspan="5">New</td><td rowspan="5">Q4AR</td></tr>
<tr><td>SD12</td></tr>
<tr><td>SD13</td></tr>
<tr><td>SD14</td></tr>
<tr><td>SD15</td></tr>
</table>

Table App. 3.2. Special register

<table>
<tr><th>Number</th><th>Name</th><th>Meaning</th><th>Explanation</th><th>Set by (When Set)</th><th>Corres-ponding ACPU D9□□□</th><th>Corresponding CPU</th></tr>
<tr><td>SD16</td><td rowspan="11">Error common information</td><td rowspan="11">Error common information</td><td rowspan="11">
- Individual information corresponding to error codes (SD0) is stored here.
- There are the following seven different types of information are stored.
- The error individual information type can be judged by the "individual information category code" in SD4. (The values of the "individual information category code" stored in SD4 correspond to following 1) to 7).)
1) (Empty)
2) File name/Drive name
<table>
<tr><th>Number</th><th colspan="2">Meaning</th></tr>
<tr><td>SD16</td><td colspan="2">Drive</td></tr>
<tr><td>SD17</td><td colspan="2" rowspan="4">File name
(ASCII code: 8 characters)</td></tr>
<tr><td>SD18</td></tr>
<tr><td>SD19</td></tr>
<tr><td>SD20</td></tr>
<tr><td>SD21</td><td rowspan="2">Extension ∗3
(ASCII code: 3 characters)</td><td>2EH(.)</td></tr>
<tr><td>SD22</td><td></td></tr>
<tr><td>SD23</td><td colspan="2" rowspan="4">(Empty)</td></tr>
<tr><td>SD24</td></tr>
<tr><td>SD25</td></tr>
<tr><td>SD26</td></tr>
</table>
(Example) File name = ABCDEFGH. IJK
<table>
<tr><th>b15 to b8</th><th>b7 to b0</th></tr>
<tr><td>42H(B)</td><td>41H(A)</td></tr>
<tr><td>44H(D)</td><td>43H(C)</td></tr>
<tr><td>46H(F)</td><td>45H(E)</td></tr>
<tr><td>48H(H)</td><td>47H(G)</td></tr>
<tr><td>49H(I)</td><td>2EH(.)</td></tr>
<tr><td>4BH(K)</td><td>4AH(J)</td></tr>
</table>
3) Time (value actually measured)
<table>
<tr><th>Number</th><th>Meaning</th></tr>
<tr><td>SD16</td><td>Time : 1µs units (0 to 999µs)</td></tr>
<tr><td>SD17</td><td>Time : 1ms units (0 to 65535ms)</td></tr>
<tr><td>SD18</td><td rowspan="9">(Empty)</td></tr>
<tr><td>SD19</td></tr>
<tr><td>SD20</td></tr>
<tr><td>SD21</td></tr>
<tr><td>SD22</td></tr>
<tr><td>SD23</td></tr>
<tr><td>SD24</td></tr>
<tr><td>SD25</td></tr>
<tr><td>SD26</td></tr>
</table>
4) Program error location
<table>
<tr><th>Number</th><th colspan="2">Meaning</th></tr>
<tr><td>SD16</td><td colspan="2" rowspan="4">File name
(ASCII code: 8 characters)</td></tr>
<tr><td>SD17</td></tr>
<tr><td>SD18</td></tr>
<tr><td>SD19</td></tr>
<tr><td>SD20</td><td>Extension ∗3</td><td>2EH(.)</td></tr>
<tr><td>SD21</td><td colspan="2">(ASCII code: 3 characters)</td></tr>
<tr><td>SD22</td><td colspan="2">Pattern ∗6</td></tr>
<tr><td>SD23</td><td colspan="2">Block No.</td></tr>
<tr><td>SD24</td><td colspan="2">Step No./transition No.</td></tr>
<tr><td>SD25</td><td colspan="2" rowspan="2">Sequence step No. (L)
Sequence step No. (H)</td></tr>
<tr><td>SD26</td></tr>
</table>
*6 : Contents of pattern data


5) Parameter No.
<table>
<tr><th>Number</th><th>Meaning</th></tr>
<tr><td>SD16</td><td>Parameter No. ∗7</td></tr>
<tr><td>SD17</td><td rowspan="10">(Empty)</td></tr>
<tr><td>SD18</td></tr>
<tr><td>SD19</td></tr>
<tr><td>SD20</td></tr>
<tr><td>SD21</td></tr>
<tr><td>SD22</td></tr>
<tr><td>SD23</td></tr>
<tr><td>SD24</td></tr>
<tr><td>SD25</td></tr>
<tr><td>SD26</td></tr>
</table>
6) Annunciator number /
7) CHK instruction malfunction number
<table>
<tr><th>Number</th><th>Meaning</th></tr>
<tr><td>SD16</td><td>No.</td></tr>
<tr><td>SD17</td><td rowspan="10">(Empty)</td></tr>
<tr><td>SD18</td></tr>
<tr><td>SD19</td></tr>
<tr><td>SD20</td></tr>
<tr><td>SD21</td></tr>
<tr><td>SD22</td></tr>
<tr><td>SD23</td></tr>
<tr><td>SD24</td></tr>
<tr><td>SD25</td></tr>
<tr><td>SD26</td></tr>
</table>
*7: For details of the parameter No., refer to the User's Manual of the CPU module used.
</td><td rowspan="11">S (Error)</td><td rowspan="11">New</td><td rowspan="11">QnA</td></tr>
<tr><td>SD17</td></tr>
<tr><td>SD18</td></tr>
<tr><td>SD19</td></tr>
<tr><td>SD20</td></tr>
<tr><td>SD21</td></tr>
<tr><td>SD22</td></tr>
<tr><td>SD23</td></tr>
<tr><td>SD24</td></tr>
<tr><td>SD25</td></tr>
<tr><td>SD26</td></tr>
</table>

*3 : Extensions are shown below.

Table App. 3.3. Extension name

SDn	**SDn+1**		**Extension Name**	**File Type**
Higher 8 bits	**Lower 8 bits**	**Higher 8 bits**		
51H	50H	41H	QPA	Parameters
51H	50H	47H	QPG	• Sequence program • SFC program
51H	43H	44H	QCD	Device comment
51H	44H	49H	QDI	Initial device value
51H	44H	52H	QDR	File register
51H	44H	53H	QDS	Simulation data
51H	44H	4CH	QDL	Local device
51H	54H	44H	QTD	Sampling trace data
51H	54H	4CH	QTL	Status latch data
51H	54H	50H	QTP	Program trace data
51H	54H	52H	QTR	SFC trace file
51H	46H	44H	QFD	Breakdown history data

Table App. 3.2. Special register

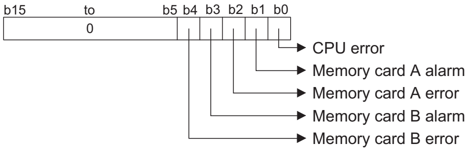
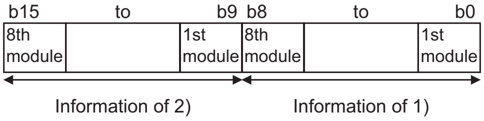
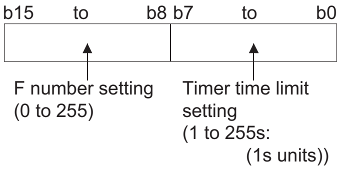
Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD50	Error reset	Error number that performs error rese	• Stores error number that performs error reset	U	New	QnA
SD51	Battery low latch	Bit pattern indicating where battery voltage drop occurred	• All corresponding bits go 1(ON) when battery voltage drops. • Subsequently, these remain 1(ON) even after battery voltage has been returned to normal.  • In the alarm, data can be held within the time specified for battery low. • The error indicates the complete discharge of the battery.	S (Error)	New	QnA
SD52	Battery low	Bit pattern indicating where battery voltage drop occurred	• Same configuration as SD51 above • Turns to 0 (OFF) when the battery voltage returns to normal thereafter.	S (Error)	New	QnA
SD53	AC/DC DOWN detection	Number of times for AC/DC DOWN detection	• Every time the input voltage falls to or below 85% (AC power)/65% (DC power) of the rating during operation of the CPU module, the value is incremented by 1 and stored in BIN code.	S (Error)	D9005	QnA
SD54	MINI link errors	Error detection state	1) When any of X(n+0)/X(n+20), X(n+6)/X(n+26), X(n+7)/X(n+27) and X(n+8)/X(n+28) of the mounted MINI(-S3) turns ON, the bit of the corresponding station turns to 1 (ON). 2) Turns to 1 (ON) when communication between the mounted MINI(-S3) and CPU module cannot be made. 		S (Error)	QnA
SD60	Number of module with blown fuse	Number of module with blown fuse	• Value stored here is the lowest station I/O number of the module with the blown fuse.		S (Error)	QnA
SD61	I/O module verify error number	I/O module verify error module number	• The lowest I/O number of the module where the I/O module verification number took place.		S (Error)	QnA

Table App. 3.2. Special register

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD62	Annunciator number	Annunciator number	• The first annunciator number (F number) to be detected is stored here.	S (Instruction execution)	D9009	QnA
SD63	Number of annunciators	Number of annunciators	• Stores the number of annunciators searched.	S (Instruction execution)	D9124	QnA
SD64	Table of detected annunciator numbers	Annunciator detection number	When F goes ON due to OUT F or SET F, the F numbers which go progressively ON from SD64 through SD79 are registered. The F numbers turned OFF by RST F are deleted from SD64 - SD79, and the F numbers stored after the deleted F numbers are shifted to the preceding registers. Execution of the LEDR instruction shifts the contents of SD64 to SD79 up by one. (This can also be done by using the INDICATOR RESET switch on the of the Q3A/Q4ACPU.) After 16 annunciators have been detected, detection of the 17th will not be stored from SD64 through SD79.	S (Instruction execution)	D9125	QnA
SD65					D9126	
SD66					D9127	
SD67					D9128	
SD68					D9129	
SD69					D9130	
SD70					D9131	
SD71					D9132	
SD72					New	
SD73					New	
SD74					New	
SD75					New	
SD76					New	
SD77					New	
SD78					New	
SD79					New	
SD80	CHK number	CHK number	• Error codes detected by the CHK instruction are stored as BCD code.	S (Instruction execution)	New	QnA
SD90	Step transition monitoring timer setting value (Enabled only when SFC program exists)	F number for timer set value and time over error	Corresponds to SM90 • Set the annunciator number (F number) that will be turned ON when the step transition monitoring timer setting or monitoring timeout occurs.  • Turning ON any of SM90 to SM99 during an active step starts the timer, and if the transition condition next to the corresponding step is not met within the timer time limit, the set annunciator (F) turns ON.	U	D9108	QnA
SD91			Corresponds to SM91		D9109	
SD92			Corresponds to SM92		D9110	
SD93			Corresponds to SM93		D9111	
SD94			Corresponds to SM94		D9112	
SD95			Corresponds to SM95		D9113	
SD96			Corresponds to SM96		D9114	
SD97			Corresponds to SM97		New	
SD98			Corresponds to SM98		New	
SD99			Corresponds to SM99		New	

		SET F50	SET F25	SET F99	RST F25	SET F15	SET F70	SET F65	SET F38	SET F110	SET F151	SET F210	LEDR	
SD62	0	50	50	50	50	50	50	50	50	50	50	50	99	(Number detected)
SD63	0	1	2	3	2	3	4	5	6	7	8	9	8	(Number of annunciators detected)
SD64	0	50	50	50	50	50	50	50	50	50	50	50	99	(Number detected)
SD65	0	0	25	25	99	99	99	99	99	99	99	99	15	
SD66	0	0	0	99	0	15	15	15	15	15	15	15	70	
SD67	0	0	0	0	0	0	70	70	70	70	70	70	65	
SD68	0	0	0	0	0	0	0	65	65	65	65	65	38	
SD69	0	0	0	0	0	0	0	0	38	38	38	38	110	
SD70	0	0	0	0	0	0	0	0	0	110	110	110	151	
SD71	0	0	0	0	0	0	0	0	0	0	151	151	210	
SD72	0	0	0	0	0	0	0	0	0	0	0	210	0	
SD73	0	0	0	0	0	0	0	0	0	0	0	0	0	
SD74	0	0	0	0	0	0	0	0	0	0	0	0	0	
SD75	0	0	0	0	0	0	0	0	0	0	0	0	0	
SD76	0	0	0	0	0	0	0	0	0	0	0	0	0	
SD77	0	0	0	0	0	0	0	0	0	0	0	0	0	
SD78	0	0	0	0	0	0	0	0	0	0	0	0	0	
SD79	0	0	0	0	0	0	0	0	0	0	0	0	0	

(2) System information

Table App. 3.4. Special register

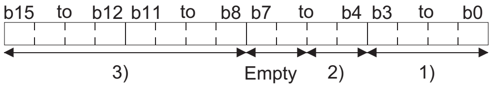
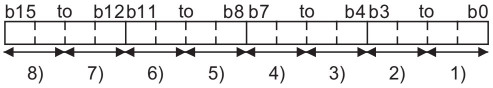
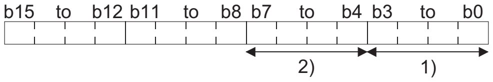
Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD200	Status of switch	Status of CPU switch	• The CPU switch status is stored in the following format:  1): CPU switch status — 0: RUN 1: STOP 2: L.CLR 2): Memory card switch — b4 corresponds to memory card A, and b5 corresponds to memory card B. 0: OFF, 1: ON 3): DIP switch — b8 through b12 correspond to SW1 through SW5 of system setting switch 1. b14 and b15 correspond to SW1 and SW2 of system setting switch 2, respectively. OFF at 0; ON at 1	S (Every END processing)	New	QnA
SD201	LED status	Status of CPU-LED	• The following bit patterns store the status of the LEDs on the CPU module: • 0 is off, 1 is on, and 2 is flicker  1): RUN 5): BOOT 2): ERROR 6): CARD A (memory card A) 3): USER 7): CARD B (memory card B) 4): BAT.ALARM 8): Empty	S (Status change)	New	QnA
SD203	Operating status of CPU	Operating status of CPU	• The CPU operating status is stored as indicated in the following figure:  1): Operating status of CPU — 0: RUN 1: STEP-RUN 2: STOP 3: PAUSE 2): STOP/PAUSE cause — 0: Instruction in remote operation program from RUN/STOP switch 1: Remote contact 2: Remote operation from GX Developer/ serial communication, etc. 3: Internal program instruction Note: Priority is earliest first — 4: Error	S (Every END processing)	D9015 format change	QnA

Table App. 3.4. Special register

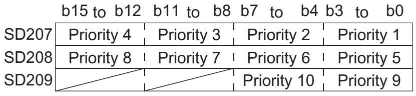
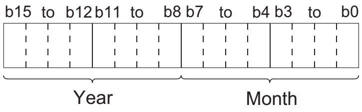
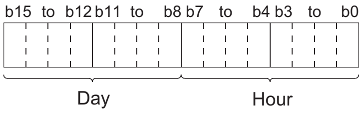
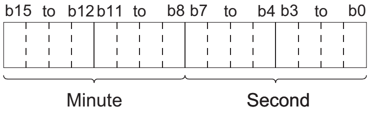
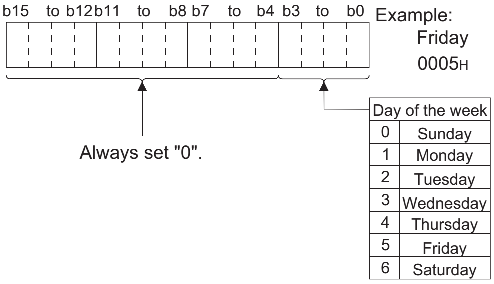
Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD207	LED display priority ranking	Priorities 1 to 4	• When error is generated, the LED display (flicker) is made according to the error number setting priorities. (The Basic model QCPU supports only the annunciator (error item No. 7). • The Universal model QCPU sets execution/non-execution of LED display of the error corresponding to the each priority ranking when the error occurs. • The setting areas for priorities are as follows:  Default Value SD207 = 4321H SD208 = 8765H (0765H for Redundant CPU) SD209 = 00A9H • No display is made if "0" is set.	U	D9038	QnA
SD208		Priorities 5 to 8			D9039 format change	
SD209		Priorities 9 to 10			New	
SD210	Clock data	Clock data (year, month)	• The year (last two digits) and month are stored as BCD code as shown below:  Example: July, 1993 9307H	S (Request)/U	D9025	QnA
SD211	Clock data	Clock data (day, hour)	• The day and hour are stored as BCD code as shown below:  Example: 31st, 10 a.m. 3110H		D9026	
SD212	Clock data	Clock data (minute, second)	• The minutes and seconds (after the hour) are stored as BCD code as shown below:  Example: 35 min, 48 s 3548H		D9027	
SD213	Clock data	Clock data (day of week)	• The day of the week is stored as BCD code as shown below. 	S (Request)/U	D9028	QnA

Table App. 3.4. Special register

<table>
<tr><th>Number</th><th>Name</th><th>Meaning</th><th>Explanation</th><th>Set by (When Set)</th><th>Corres-ponding ACPU D9□□□</th><th>Corresponding CPU</th></tr>
<tr><td>SD220</td><td rowspan="8">LED display data</td><td rowspan="8">LED display data</td><td rowspan="8">• LED display ASCII data (16 characters) stored here.
<table>
<tr><th></th><th>b15 to b8</th><th>b7 to b0</th></tr>
<tr><td>SD220</td><td>15th character from the right</td><td>16th character from the right</td></tr>
<tr><td>SD221</td><td>13th character from the right</td><td>14th character from the right</td></tr>
<tr><td>SD222</td><td>11th character from the right</td><td>12th character from the right</td></tr>
<tr><td>SD223</td><td>9th character from the right</td><td>10th character from the right</td></tr>
<tr><td>SD224</td><td>7th character from the right</td><td>8th character from the right</td></tr>
<tr><td>SD225</td><td>5th character from the right</td><td>6th character from the right</td></tr>
<tr><td>SD226</td><td>3rd character from the right</td><td>4th character from the right</td></tr>
<tr><td>SD227</td><td>1st character from the right</td><td>2nd character from the right</td></tr>
</table></td><td rowspan="8">S (When changed)</td><td rowspan="8">New</td><td rowspan="8">QnA</td></tr>
<tr><td>SD221</td></tr>
<tr><td>SD222</td></tr>
<tr><td>SD223</td></tr>
<tr><td>SD224</td></tr>
<tr><td>SD225</td></tr>
<tr><td>SD226</td></tr>
<tr><td>SD227</td></tr>
<tr><td>SD251</td><td>Head I/O number for replacement</td><td>Head I/O No. for module replacement</td><td>• Stores the upper two digits of the head I/O number of an I/O module that is removed/replaced in the online status (with power on). (Default value: 100H).</td><td>U</td><td>D9094</td><td>Q2A(S1)
Q3A
Q4A
Q4AR</td></tr>
<tr><td>SD253</td><td>RS422 transmission speed</td><td>RS422 transmission speed</td><td>• Stores transmission speed of RS422.
0 : 9600bps 1 : 19.2kbps 2 : 38.4kbps</td><td>S (When changed)</td><td>New</td><td>QnA</td></tr>
</table>

Table App. 3.4. Special register

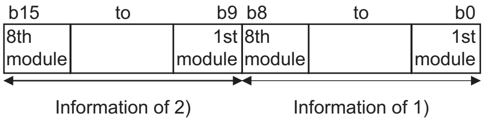
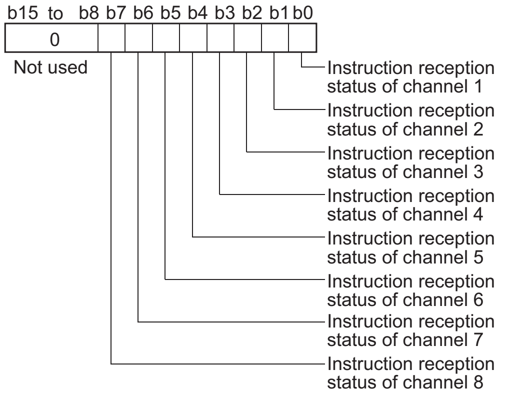
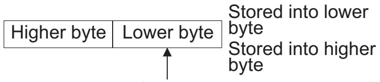
Number	Name	Meaning		Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD254	MELSECNET/ 10 information	Number of modules installed		• Indicates the number of mounted MELSECNET/10 modules.	S (Initial)	New	QnA
SD255		Information from 1st module	I/O No.	• Indicates I/O number of mounted MELSECNET/10 module			
SD256			Network No.	• Indicates network No. of mounted MELSECNET/10 module			
SD257			Group number	• Indicates group No. of mounted MELSECNET/10 module			
SD258			Station No.	• Indicates station No. of mounted MELSECNET/10 module			
SD259			Standby informa-tion	• In the case of standby stations, the module number of the standby station is stored. (1 to 4)			
SD260 to SD264		Information from 2nd module		• Configuration is identical to that for the first module.			
SD265 to SD269		Information from 3rd module		• Configuration is identical to that for the first module.			
SD270 to SD274		Information from 4th module		• Configuration is identical to that for the first module.			
SD280	CC-Link error	Error detection status		• When Xn0 of the mounted CC-Link module turns ON, the bit of the corresponding station turns to 1 (ON). • When either Xn1 or XnF of the mounted CC-Link module turns OFF, the bit of the corresponding station turns to 1 (ON). • Turns to 1 (ON) when communication between the mounted CC-Link module and CPU module cannot be made. 	S (Error)	New	QnA
SD290	Device assignment (Same as parameter contents)	Number of points assigned for X		• Stores the number of points currently set for X devices	S (Initial)	New	QnA
SD291		Number of points assigned for Y		• Stores the number of points currently set for Y devices			
SD292		Number of points assigned for M		• Stores the number of points currently set for M devices			
SD293		Number of points assigned for L		• Stores the number of points currently set for L devices			
SD294		Number of points assigned for B		• Stores the number of points currently set for B devices			
SD295		Number of points assigned for F		• Stores the number of points currently set for F devices			
SD296		Number of points assigned for SB		• Stores the number of points currently set for SB devices			
SD297		Number of points assigned for V		• Stores the number of points currently set for V devices			
SD298		Number of points assigned for S		• Stores the number of points currently set for S devices			
SD299		Number of points assigned for T		• Stores the number of points currently set for T device			
SD300		Number of points assigned for ST		• Stores the number of points currently set for ST devices			
SD301		Number of points assigned for C		• Stores the number of points currently set for C devices			
SD302		Number of points assigned for D		• Stores the number of points currently set for D devices			
SD303		Number of points assigned for W		• Stores the number of points currently set for W devices			
SD304		Number of points assigned for SW		• Stores the number of points currently set for SW devices			

Table App. 3.4. Special register

Number	Name	Meaning		Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD340	Ethernet information	No. of modules installed		• Indicates the number of mounted Ethernet module.	S (Initial)	New	QnA
SD341		Information of 1st module	I/O No.	• Indicates I/O No. of mounted Ethernet module			
SD342			Network No.	• Indicates network No. of mounted Ethernet module			
SD343			Group No.	• Indicates group No. of mounted Ethernet module			
SD344			Station No.	• Indicates station No. of mounted Ethernet module			
SD345 to SD346			IP address	• Indicates IP address of mounted Ethernet module			
SD347			Error code	• Indicates error code of mounted Ethernet module			
SD348 to SD354		Information from 2nd module		• Configuration is identical to that for the first module.			
SD355 to SD361		Information from 3rd module		• Configuration is identical to that for the first module.			
SD362 to SD368		Information from 4th module		• Configuration is identical to that for the first module.			
SD380	Ethernet instruction reception status	Instruction reception status of 1st module		 ON: Received (Channel is being used.) OFF: Not received (Channel is not used.)	S (Initial)	New	QnA
SD392	Software version	Internal system software version		• Stores the internal system software version in ASCII code.  For version "A", for example, "41H" is stored. Note: The internal system software version may differ from the version indicated by the version symbol printed on the case.	S (Initial)	D9060	QnA

(3) System clocks/counters

Table App. 3.5. Special register

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD412	1 second counter	Number of counts in 1-second units	• Following programmable controller CPU module RUN, 1 is added each second • Count repeats from 0 to 32767 to -32768 to 0	S (Status change)	D9022	QnA
SD414	2n second clock setting	2n second clock units	• Stores value n of 2n second clock (Default is 30) • Setting can be made between 1 and 32767	U	New	QnA
SD420	Scan counter	Number of counts in each scan	• Incremented by 1 for each scan execution after the CPU module is set to RUN. (Not counted by the scan in an initial execution type program.) • Count repeats from 0 to 32767 to -32768 to 0	S (Every END processing)	New	QnA
SD430	Low speed scan counter	Number of counts in each scan	• Incremented by 1 for each scan execution after the CPU module is set to RUN. • Count repeats from 0 to 32767 to -32768 to 0 • Used only for low speed execution type programs	S (Every END processing)	New	QnA

(4) Scan information

Table App. 3.6. Special register

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD500	Execution program No.	Program No. in execution	• Program number of program currently being executed is stored as BIN value.	S (Status change)	New	QnA
SD510	Low speed excution type program No.	Low speed execution type program No. in execution	• Program number of low speed excution type program No. currently being executed is stored as BIN value. • Enabled only when SM510 is ON.	S (Every END processing)	New	QnA
SD520	Current scan time	Current scan time (in 1 ms units)	• The current scan time is stored into SD520 and SD521. (Measurement is made in 100 µs units.) SD520: Stores the ms place. (Storage range: 0 to 65535) SD521: Stores the µs place. (Storage range: 0 to 900) (Example) When the current scan time is 23.6ms, the following values are stored. SD520 = 23 SD521 = 600	S (Every END processing)	D9017 format change	QnA
SD521		Current scan time (in 100 µs units)		S (Every END processing)	New	
SD522	Initial scan time	Initial scan time (in 1 ms units)	• Stores the scan time of an initial execution type program into SD522 and SD523. (Measurement is made in 100 µs units.) SD522: Stores the ms place. (Storage range: 0 to 65535) SD523: Stores the µs place. (Storage range: 0 to 900)	S (First END processing)	New	QnA
SD523		Initial scan time (in 100 µs units)				
SD524	Minimum scan time	Minimum scan time (in 1 ms units)	• Stores the minimum value of the scan time except that of an initial execution type program into SD524 and SD525. (Measurement is made in 100 µs units.) SD524: Stores the ms place. (Storage range: 0 to 65535) SD525: Stores the µs place. (Storage range: 0 to 900)	S (Every END processing)	D9018 format change	QnA
SD525		Minimum scan time (in 100 µs units)		S (Every END processing)	New	
SD526	Maximum scan time	Maximum scan time (in 1 ms units)	• Stores the maximum value of the scan time except that of an initial execution type program into SD526 and SD527. (Measurement is made in 100 µs units.) SD526: Stores the ms place. (Storage range: 0 to 65535) SD527: Stores the µs place.	S (Every END processing)	D9019 format change	QnA
SD527		Maximum scan time (in 100 µs units)			New	
SD528	Current scan time for low speed execution type programs	Current scan time (in 1 ms units)	• Stores the current scan time of a low speed execution type program into SD528 and SD529. (Measurement is made in 100 µs units.) SD528: Stores the ms place. (Storage range: 0 to 65535) SD529: Stores the µs place. (Storage range: 0 to 900)	S (Every END processing)	New	QnA
SD529		Current scan time (in 100 µs units)				
SD532	Minimum scan time for low speed execution type programs	Minimum scan time (in 1 ms units)	• Stores the minimum value of the scan time of a low speed execution type program into SD532 and SD533. (Measurement is made in 100 µs units.) SD532: Stores the ms place. (Storage range: 0 to 65535) SD533: Stores the µs place. (Storage range: 0 to 900)	S (Every END processing)	New	QnA
SD533		Minimum scan time (in 100 µs units)				
SD534	Maximum scan time for low speed execution type programs	Maximum scan time (in 1 ms units)	• Stores the maximum value of the scan time except that of the first scan of a low speed execution type program into SD534 and SD535. (Measurement is made in 100 µs units.) SD534: Stores the ms place. (Storage range: 0 to 65535) SD535: Stores the µs place. (Storage range: 0 to 900)	S (Every END processing)	New	QnA
SD535		Maximum scan time (in 100 µs units)				
SD540	END processing time	END processing time (in 1 ms units)	• Stores the time from the end of a scan execution type program to the start of the next scan into SD540 and SD541. (Measurement is made in 100 µs units.) SD540: Stores the ms place. (Storage range: 0 to 65535) SD541: Stores the µs place. (Storage range: 0 to 900) (Storage range: 0 to 900)	S (Every END processing)	New	QnA
SD541		END processing time (in 100 µs units)				
SD542	Constant scan wait time	Constant scan wait time (in 1 ms units)	• Stores the wait time for constant scan setting into SD542 and SD543. (Measurement is made in 100 µs units. (For the Universal model QCPU, in 1µs units.)) SD542: Stores the ms place. (Storage range: 0 to 65535) SD543: Stores the µs place. (Storage range: 0 to 900 (For the Universal model QCPU, storage range is 0 to 999))	S (Every END processing)	New	QnA
SD543		Constant scan wait time (in 100 µs units)				

Table App. 3.6. Special register

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD544	Cumulative execution time for low speed execution type programs	Cumulative execution time for low speed execution type programs (in 1 ms units)	• Stores the cumulative execution time of a low speed execution type program into SD544 and SD545. (Measurement is made in 100 µs units.) SD544: Stores the ms place. (Storage range: 0 to 65535) SD545: Stores the µs place. (Storage range: 0 to 900) • Cleared to 0 after the end of one low speed scan.	S (Every END processing)	New	QnA
SD545		Cumulative execution time for low speed execution type programs (in 100 µs units)				
SD546	Execution time for low speed execution type programs	Execution time for low speed execution type programs (in 1 ms units)	• Stores the execution time of a low speed execution type program during one scan into SD546 and SD547. (Measurement is made in 100 µs units.) SD546: Stores the ms place. (Storage range: 0 to 65535) SD547: Stores the µs place. (Storage range: 0 to 900) • Stored every scan.	S (Every END processing)	New	QnA
SD547		Execution time for low speed execution type programs (in 100 µs units)				
SD548	Scan execution type program execution time	Scan execution type program execution time (in 1 ms units)	• Stores the execution time of a scan execution type program during one scan into SD548 and SD549. (Measurement is made in 100 µs units.) SD548: Stores the ms place. (Storage range: 0 to 65535) SD549: Stores the µs place. (Storage range: 0 to 900) • Stored every scan.	S (Every END processing)	New	QnA
SD549		Scan execution type program execution time (in 100 µs units)				
SD550	Service interval measurement module	Unit/module No.	• Sets I/O number for module that measures service interval.	U	New	QnA
SD551	Service interval time	Module service interval (in 1 ms units)	• Stores the service interval for the module specified in SD550 into SD551 and SD552 when SM551 is turned ON. (Measurement is made in 100 µs units.) SD551: Stores the ms place. (Storage range: 0 to 65535) SD552: Stores the µs place. (Storage range: 0 to 900)	S (Request)	New	QnA
SD552		Module service interval (in 100 µs units)				

(5) Drive information

Table App. 3.7. Special register

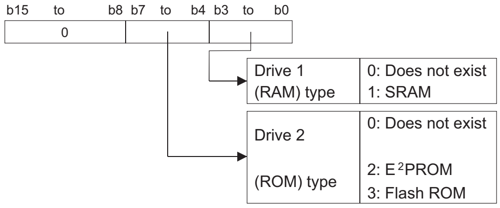
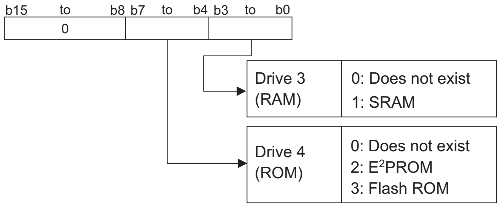
Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD600	Memory card A typs	Memory card A typs	• Indicates the type of memory card A installed.  b15 to b8 b7 to b4 b3 to b0 0 Drive 1 (RAM) type 0: Does not exist 1: SRAM Drive 2 (ROM) type 0: Does not exist 2: E^{2}PROM 3: Flash ROM	S (Initial and card removal)	New	QnA
SD602	Drive 1 (Memory card RAM) capacity	Drive 1 capacity	• Drive 1 capacity is stored in 1 k byte units. • (Empty capacity after format is stored.)	S (Initial and card removal)	New	QnA
SD603	Drive 2 (Memory card ROM) capacity	Drive 2 capacity	• Drive 2 capacity is stored in 1 k byte units.	S (Initial and card removal)	New	QnA
SD604	Memory card A use conditions	Memory card A use conditions	• The use conditions for memory card A are stored as bit patterns. (In use when ON) • The significance of these bit patterns is indicated below: b0 : Boot operation (QBT) b1 : Parameters (QPA) b2 : Device comments (QCD) b3 : Device initial value (QDI) b4 : File register (QDR) b5 : Sampling trace (QTD) b6 : Status latch (QTL) b7 : Program trace (QTP) b8 : Simulation data (QDS) b9 : CPU fault history (QFD) b10 : SFC trace (QTR) b11 : Local device (QDL) b12 : Not used b13 : Not used b14 : Not used b15 : Not used	S (Status change)	New	QnA
SD620	Memory card B typs	Memory card B typs	• Indicates memory card B type installed  b15 to b8 b7 to b4 b3 to b0 0 Drive 3 (RAM) 0: Does not exist 1: SRAM Drive 4 (ROM) 0: Does not exist 2: E^{2}PROM 3: Flash ROM	S (Initial/Card installation and removal)	New	Q2A(S1) Q3A Q4A Q4AR
SD622	Drive 3 (Standard RAM) capacity	Drive 3 capacity	• Drive 3 capacity is stored in 1 k byte units. (Empty capacity after format is stored.)	S (Initial/Card installation and removal)	New	Q2A(S1) Q3A Q4A Q4AR
SD623	Drive 4 (Standard ROM) capacity	Drive 4 capacity	• Drive 4 capacity is stored in 1 k byte units. (Empty capacity after format is stored.)	S (Initial/Card installation and removal)	New	Q2A(S1) Q3A Q4A Q4AR
SD624	Memory card B use conditions	Memory card B use conditions	• The use conditions for memory card B are stored as bit patterns. (In use when ON) • The significance of these bit patterns is indicated below: b0 : Boot operation (QBT) b1 : Parameters (QPA) b2 : Device comments (QCD) b3 : Device initial value (QDI) b4 : File register (QDR) b5 : Sampling trace (QTD) b6 : Status latch (QTL) b7 : Program trace (QTP) b8 : Simulation data (QDS) b9 : CPU fault history (QFD) b10 : SFC trace (QTS) b11 : Local device (QDL) b12 : Local device (QDL) b13 : Not used b14 : Not used b15 : Not used	S (Status change)	New	Q2A(S1) Q3A Q4A Q4AR
SD640	File register drive	Drive number:	• Stores drive number being used by file register	S (Status change) *10	New	QnA

Table App. 3.7. Special register

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD641	File register file name	File register file name	• Stores file register file name (with extension) selected at parameters or by use of QDRSET instruction as ASCII code. b15 to b8 / b7 to b0 SD641: 2nd character / 1st character SD642: 4th character / 3rd character SD643: 6th character / 5th character SD644: 8th character / 7th character SD645: 1st character of extension / 2EH(.) SD646: 3rd character of the extension / 2nd character of the extension	S (Status change)	New	QnA
SD642						
SD643						
SD644						
SD645						
SD647	File register capacity	File register capacity	• Stores the data capacity of the currently selected file register in 1 k word units.	S (Status change)	New	QnA
SD648	File register block number	File register block number	• Stores the currently selected file register block number.	S (Status change)	D9035	QnA
SD650	Comment drive	Comment drive number	• Stores the comment drive number selected at the parameters or by the QCDSET instruction.	S (Status change)	New	QnA
SD651	Comment file name	Comment file name	• Stores the comment file name (with extension) selected at the parameters or by the QCDSET instruction in ASCII code. b15 to b8 / b7 to b0 SD651: 2nd character / 1st character SD652: 4th character / 3rd character SD653: 6th character / 5th character SD654: 8th character / 7th character SD655: 1st character of the extension / 2EH(.) SD656: 3rd character of the extension / 2nd character of the extension	S (Status change)	New	QnA
SD652						
SD653						
SD654						
SD655						
SD656						
SD660	Boot operation designation file	Boot designation file drive number	• Stores the drive number where the boot designation file (*.QBT) is being stored.	S (Initial)	New	QnA
SD661		File name of boot designation file	• Stores the file name of the boot designation file (*.QBT). b15 to b8 / b7 to b0 SD661: 2nd character / 1st character SD662: 4th character / 3rd character SD663: 6th character / 5th character SD664: 8th character / 7th character SD665: 1st character of the extension / 2EH(.) SD666: 3rd character of the extension / 2nd character of the extension	S (Initial)	New	QnA
SD662						
SD663						
SD664						
SD665						
SD666						

(6) Instruction-Related Registers

Table App. 3.8. Special register

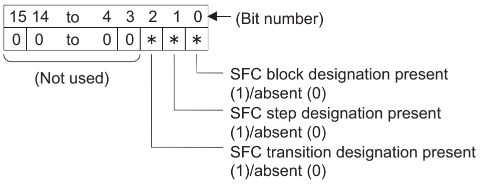
Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD705	Mask pattern	Mask pattern	• During block operations, turning SM705 ON makes it possible to use the mask pattern being stored at SD705 (or at SD705 and SD706 if double words are being used) to operate on all data in the block with the masked values.	U	New	QnA
SD706						
SD714	Number of empty communication request registration areas	0 to 32	• Stores the number of empty blocks in the communications request area for remote terminal modules connected to the MELSECNET/MINI-S3.	S (During execution)	D9081	QnA
SD715	IMASK instruction mask pattern	Mask pattern	• Patterns masked by use of the IMASK instruction are stored in the following manner: b15 b1 b0 SD715 I15 to I1 I0 SD716 I31 to I17 I16 SD717 I47 to I33 I32	S (During execution)	New	QnA
SD716						
SD717						
SD718	Accumulator	Accumulator	• For use as replacement for accumulators used in A series programs.	S/U	New	QnA
SD719						
SD730	No. of empty areas for CC-Link communication reguest register area	0 to 32	• Stores the number of empty registration area for the request for communication with the intelligent device station connected to A(1S)J61QBT61.	S (During execution)	New	QnA
SD736	PKEY input	PKEY input	• Special register that temporarily stores keyboard data input by means of the PKEY instruction.	S (During execution)	New	QnA

Table App. 3.8. Special register

<table>
<tr><th>Number</th><th>Name</th><th>Meaning</th><th>Explanation</th><th>Set by (When Set)</th><th>Corres-ponding ACPU D9□□□</th><th>Corresponding CPU</th></tr>
<tr><td>SD738</td><td rowspan="32">Message storage</td><td rowspan="32">Message storage</td><td rowspan="32">• Stores the message designated by the MSG instruction.
<table>
<tr><th></th><th>b15 to b8</th><th>b7 to b0</th></tr>
<tr><td>SD738</td><td>2nd character</td><td>1st character</td></tr>
<tr><td>SD739</td><td>4th character</td><td>3rd character</td></tr>
<tr><td>SD740</td><td>6th character</td><td>5th character</td></tr>
<tr><td>SD741</td><td>8th character</td><td>7th character</td></tr>
<tr><td>SD742</td><td>10th character</td><td>9th character</td></tr>
<tr><td>SD743</td><td>12th character</td><td>11th character</td></tr>
<tr><td>SD744</td><td>14th character</td><td>13th character</td></tr>
<tr><td>SD745</td><td>16th character</td><td>15th character</td></tr>
<tr><td>SD746</td><td>18th character</td><td>17th character</td></tr>
<tr><td>SD747</td><td>20th character</td><td>19th character</td></tr>
<tr><td>SD748</td><td>22nd character</td><td>21st character</td></tr>
<tr><td>SD749</td><td>24th character</td><td>23rd character</td></tr>
<tr><td>SD750</td><td>26th character</td><td>25th character</td></tr>
<tr><td>SD751</td><td>28th character</td><td>27th character</td></tr>
<tr><td>SD752</td><td>30th character</td><td>29th character</td></tr>
<tr><td>SD753</td><td>32nd character</td><td>31st character</td></tr>
<tr><td>SD754</td><td>34th character</td><td>33rd character</td></tr>
<tr><td>SD755</td><td>36th character</td><td>35th character</td></tr>
<tr><td>SD756</td><td>38th character</td><td>37th character</td></tr>
<tr><td>SD757</td><td>40th character</td><td>39th character</td></tr>
<tr><td>SD758</td><td>42nd character</td><td>41st character</td></tr>
<tr><td>SD759</td><td>44th character</td><td>43rd character</td></tr>
<tr><td>SD760</td><td>46th character</td><td>45th character</td></tr>
<tr><td>SD761</td><td>48th character</td><td>47th character</td></tr>
<tr><td>SD762</td><td>50th character</td><td>49th character</td></tr>
<tr><td>SD763</td><td>52nd character</td><td>51st character</td></tr>
<tr><td>SD764</td><td>54th character</td><td>53rd character</td></tr>
<tr><td>SD765</td><td>56th character</td><td>55th character</td></tr>
<tr><td>SD766</td><td>58th character</td><td>57th character</td></tr>
<tr><td>SD767</td><td>60th character</td><td>59th character</td></tr>
<tr><td>SD768</td><td>62nd character</td><td>61st character</td></tr>
<tr><td>SD769</td><td>64th character</td><td>63rd character</td></tr>
</table></td><td rowspan="32">S (During execution)</td><td rowspan="32">New</td><td rowspan="32">QnA</td></tr>
<tr><td>SD739</td></tr>
<tr><td>SD740</td></tr>
<tr><td>SD741</td></tr>
<tr><td>SD742</td></tr>
<tr><td>SD743</td></tr>
<tr><td>SD744</td></tr>
<tr><td>SD745</td></tr>
<tr><td>SD746</td></tr>
<tr><td>SD747</td></tr>
<tr><td>SD748</td></tr>
<tr><td>SD749</td></tr>
<tr><td>SD750</td></tr>
<tr><td>SD751</td></tr>
<tr><td>SD752</td></tr>
<tr><td>SD753</td></tr>
<tr><td>SD754</td></tr>
<tr><td>SD755</td></tr>
<tr><td>SD756</td></tr>
<tr><td>SD757</td></tr>
<tr><td>SD758</td></tr>
<tr><td>SD759</td></tr>
<tr><td>SD760</td></tr>
<tr><td>SD761</td></tr>
<tr><td>SD762</td></tr>
<tr><td>SD763</td></tr>
<tr><td>SD764</td></tr>
<tr><td>SD765</td></tr>
<tr><td>SD766</td></tr>
<tr><td>SD767</td></tr>
<tr><td>SD768</td></tr>
<tr><td>SD769</td></tr>
<tr><td>SD780</td><td>Remaining No. of simultaneous execution of CC-Link dedicated instruction</td><td>0 to 32</td><td>• Stores the remaining number of simultaneous execution of the CC-Link dedicated instructions.</td><td>U</td><td>New</td><td>QnA</td></tr>
</table>

(7) Debug

Table App. 3.9. Special register

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD806	Status latch file name	Status latch file name	• Stores file name (with extension) from point in time when status latch was conducted as ASCII code. b15 to b8 / b7 to b0 SD806: 2nd character / 1st character SD807: 4th character / 3rd character SD808: 6th character / 5th character SD809: 8th character / 7th character SD810: 1st character of the extension / 2EH(.) SD811: 3rd character of the extension / 2nd character of the extension	S (During execution)	New	QnA
SD807						
SD808						
SD809						
SD810						
SD811						
SD812	Status latch step	Status latch step	• Stores step number from point in time when status latch was conducted. SD812: Pattern *1 SD813: Block No. SD814: Step No./transition condition No. SD815: Sequence step No. (L) SD816: Sequence step No. (H) *1: Contents of pattern data 	S (During execution)	D9055 format change	QnA
SD813						
SD814						
SD815						
SD816						

(8) Latch area

Table App. 3.10. Special register

<table>
<tr><th>Number</th><th>Name</th><th>Meaning</th><th>Explanation</th><th>Set by (When Set)</th><th>Corres-ponding ACPU D9□□□</th><th>Corresponding CPU</th></tr>
<tr><td>SD900</td><td>Drive where power was interrupted</td><td>Access file drive number during power loss</td><td>• Stores drive number if file was being accessed during power loss.</td><td>S (Status change)</td><td>New</td><td>QnA</td></tr>
<tr><td>SD901</td><td rowspan="6">File name active during power loss</td><td rowspan="6">Access file name during power loss</td><td rowspan="6">• Stores file name (with extension) in ASCII code if file was being accessed during power loss.
<table>
<tr><th></th><th>b15 to b8</th><th>b7 to b0</th></tr>
<tr><td>SD901</td><td>2nd character</td><td>1st character</td></tr>
<tr><td>SD902</td><td>4th character</td><td>3rd character</td></tr>
<tr><td>SD903</td><td>6th character</td><td>5th character</td></tr>
<tr><td>SD904</td><td>8th character</td><td>7th character</td></tr>
<tr><td>SD905</td><td>1st character of the extension</td><td>2EH(.)</td></tr>
<tr><td>SD906</td><td>3rd character of the extension</td><td>2nd character of the extension</td></tr>
</table></td><td rowspan="6">S (Status change)</td><td rowspan="6">New</td><td rowspan="6">QnA</td></tr>
<tr><td>SD902</td></tr>
<tr><td>SD903</td></tr>
<tr><td>SD904</td></tr>
<tr><td>SD905</td></tr>
<tr><td>SD906</td></tr>
<tr><td>SD910</td><td rowspan="16">RKEY input</td><td rowspan="16">RKEY input</td><td rowspan="16">• Stored in sequence that PU key code was entered.
<table>
<tr><th></th><th>b15 to b8</th><th>b7 to b0</th></tr>
<tr><td>SD910</td><td>2nd character</td><td>1st character</td></tr>
<tr><td>SD911</td><td>4th character</td><td>3rd character</td></tr>
<tr><td>SD912</td><td>6th character</td><td>5th character</td></tr>
<tr><td>SD913</td><td>8th character</td><td>7th character</td></tr>
<tr><td>SD914</td><td>10th character</td><td>9th character</td></tr>
<tr><td>SD915</td><td>12th character</td><td>11th character</td></tr>
<tr><td>SD916</td><td>14th character</td><td>13th character</td></tr>
<tr><td>SD917</td><td>16th character</td><td>15th character</td></tr>
<tr><td>SD918</td><td>18th character</td><td>17th character</td></tr>
<tr><td>SD919</td><td>20th character</td><td>19th character</td></tr>
<tr><td>SD920</td><td>22nd character</td><td>21st character</td></tr>
<tr><td>SD921</td><td>24th character</td><td>23rd character</td></tr>
<tr><td>SD922</td><td>26th character</td><td>25th character</td></tr>
<tr><td>SD923</td><td>28th character</td><td>27th character</td></tr>
<tr><td>SD924</td><td>30th character</td><td>29th character</td></tr>
<tr><td>SD925</td><td>32nd character</td><td>31st character</td></tr>
</table></td><td rowspan="16">S (During execution)</td><td rowspan="16">New</td><td rowspan="16">QnA</td></tr>
<tr><td>SD911</td></tr>
<tr><td>SD912</td></tr>
<tr><td>SD913</td></tr>
<tr><td>SD914</td></tr>
<tr><td>SD915</td></tr>
<tr><td>SD916</td></tr>
<tr><td>SD917</td></tr>
<tr><td>SD918</td></tr>
<tr><td>SD919</td></tr>
<tr><td>SD920</td></tr>
<tr><td>SD921</td></tr>
<tr><td>SD922</td></tr>
<tr><td>SD923</td></tr>
<tr><td>SD924</td></tr>
<tr><td>SD925</td></tr>
</table>

(9) A to QnA conversion

ACPU special registers D9000 to D9255 correspond to QnA special registers SD1000 to SD1255 after A to Q/QnA conversion.
These special registers are all set by the system, and cannot be set by the user program.
To set data by the user program, correct the program for use of the QnACPU special registers.
However, some of SD1200 to SD1255 (corresponding to D9200 to 9255 before conversion) can be set by the user program if they could be set by the user program before conversion.
For details on the ACPU special registers, refer to the user's manual for the corresponding CPU, and MELSECNET or MELSECNET/B Data Link System Reference Manuals.

REMARK

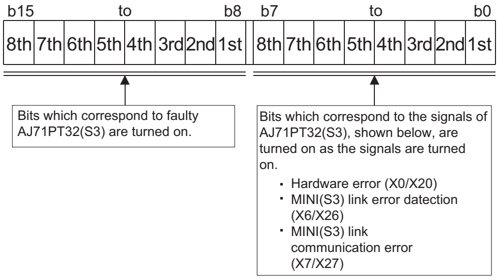
Supplemental explanation on "Special Register for Modification" column

① For the device numbers for which a special register for modification is specified, modify it to the special register for QnACPU.

② For the device numbers for which ⊟ is specified, special register after conversion can be used.

③ Device numbers for which ⊠ is specified do not function for QnACPU.

Table App. 3.11. Special register

<table>
<tr><th>ACPU Special Register</th><th>Special Register after Conversion</th><th>Special Register for Modification</th><th>Name</th><th>Meaning</th><th>Details</th><th>Corresponding CPU</th></tr>
<tr><td>D9000</td><td>SD1000</td><td>–</td><td>Fuse blown</td><td>Number of module with blown fuse</td><td>• When fuse blown modules are detected, the first I/O number of the lowest number of the detected modules is stored in hexadecimal. (Example: When fuses of Y50 to 6F output modules have blown, "50" is stored in hexadecimal) To monitor the number by peripheral devices, perform monitor operation given in hexadecimal. (Cleared when all contents of SD1100 to SD1107 are reset to 0.)
• Fuse blow check is executed also to the output modules of remote I/O stations.</td><td>QnA</td></tr>
<tr><td>D9001</td><td>SD1001</td><td>–</td><td>Fuse blown</td><td>Number of module with blown fuse</td><td>• Stores the module numbers corresponding to setting switch numbers or base slot numbers when fuse blow occurred.
<table>
<tr><th colspan="2">AJ02 I/O module</th><th colspan="2">Extension base unit</th></tr>
<tr><th>Setting switch</th><th>Stored data</th><th>Base unit slot No.</th><th>Stored data</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>4</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>5</td></tr>
<tr><td>2</td><td>2</td><td>2</td><td>6</td></tr>
<tr><td>3</td><td>3</td><td>3</td><td>7</td></tr>
<tr><td>4</td><td>4</td><td></td><td></td></tr>
<tr><td>5</td><td>5</td><td></td><td></td></tr>
<tr><td>6</td><td>6</td><td></td><td></td></tr>
<tr><td>7</td><td>7</td><td></td><td></td></tr>
</table>
• For the remote I/O station, the value of (module I/O No./10H) + 1 is stored.</td><td>QnA</td></tr>
<tr><td>D9002</td><td>SD1002</td><td>–</td><td>I/O module verify error</td><td>I/O module verify error module number</td><td>• If I/O modules, of which data are different from data entered, are detected when the power is turned on, the first I/O number of the lowest number unit among the detected units is stored in hexadecimal. (Storing method is the same as that of SD1000.) To monitor the number by peripheral devices, perform monitor operation given in hexadecimal. (Cleared when all contents of SD1116 to SD1123 are reset to 0.)
• I/O module verify check is executed also to the modules of remote I/O terminals.</td><td>QnA</td></tr>
<tr><td>D9004</td><td>SD1004</td><td>–</td><td>MINI link master module errors</td><td>Stores setting status made at parameters</td><td>• Error status of the MINI(S3) link detected on loaded AJ71PT32(S3) is stored.


Bits which correspond to faulty AJ71PT32(S3) are turned on.
Bits which correspond to the signals of AJ71PT32(S3), shown below, are turned on as the signals are turned on.
・Hardware error (X0/X20)
・MINI(S3) link error datection (X6/X26)
・MINI(S3) link communication error (X7/X27)</td><td>QnA</td></tr>
<tr><td rowspan="2">D9005</td><td rowspan="2">SD1005</td><td rowspan="2">–</td><td rowspan="2">AC DOWN counter</td><td rowspan="2">Number of times for AC DOWN</td><td>• When the AC power supply module is used, 1 is added at occurrence of an instantaneous power failure of within 20ms. (The value is stored in BIN code.) It is reset when the power supply is switched from OFF to ON.</td><td>QnA</td></tr>
<tr><td>• When the DC power supply module is used, 1 is added at occurrence of an instantaneous power failure of within 1ms. (The value is stored in BIN code.) It is reset when the power supply is switched from OFF to ON.</td><td>QnA</td></tr>
<tr><td>D9008</td><td>SD1008</td><td>SD0</td><td>Self-diaghostic error</td><td>Self-diaghostic error number</td><td>• When error is found as a result of self-diagnosis, error number is stored in BIN code.</td><td>QnA</td></tr>
</table>

Table App. 3.11. Special register

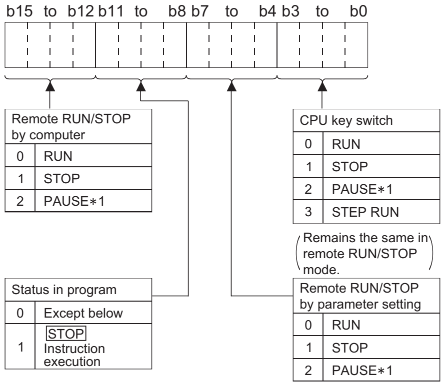
ACPU Special Register	Special Register after Conversion	Special Register for Modification	Name	Meaning	Details	Corresponding CPU
D9009	SD1009	SD62	Annunciator detection	F number at which external failure has occurred	• When one of F0 to 2047 is turned on by OUT F or SET F , the F number, which has been detected earliest among the F numbers which have turned on, is stored in BIN code. • SD62 can be cleared by RST F or LEDR instruction. If another F number has been detected, the clearing of SD62 causes the next number to be stored in SD62.	Q2AS Q2A
					• When one of F0 to 2047 is turned on by OUT F or SET F , the F number, which has been detected earliest among the F numbers which have turned on, is stored in BIN code. • SD62 can be cleared by executing RST F or LEDR instruction or moving INDICATOR RESET switch on CPU module front to ON position. If another F number has been detected, clearing of SD62 stores the next F number into SD62.	Q3A Q4A Q4AR
D9015	SD1015	SD203	Operating status of CPU	Operating status of CPU	• The operation status of CPU as shown below are stored in SD203.  *1: When the CPU mdoule is in RUN mode and SM1040 is off, the CPU module remains in RUN mode if changed to PAUSE mode.	QnA
D9016	SD1016	×	Program number	0: Main program (ROM) 1: Main program (RAM) 2: Subprogram 1 (RAM) 3: Subprogram 2 (RAM) 4: Subprogram 3 (RAM) 5: Subprogram 1 (ROM) 6: Subprogram 2 (ROM) 7: Subprogram 3 (ROM) 8: Main program (E^2PROM) 9: Subprogram 1 (E^2PROM) A: Subprogram 2 (E^2PROM) B: Subprogram 3 (E^2PROM)	• Indicates which sequence program is run presently. One value of 0 to B is stored in BIN code.	–
D9017	SD1017	SD520	Scan time	Minimum scan time (10 ms units)	• If scan time is smaller than the content of SD520, the value is newly stored at each END. Namely, the minimum value of scan time is stored into SD520 in BIN code.	QnA
D9018	SD1018	SD524	Scan time	Scan time (10 ms units)	• At every END, the scan time is stored in BIN code and always rewritten.	QnA
D9019	SD1019	SD526	Scan time	Maximum scan time (10 ms units)	• If scan time is larger than the content of SD526, the value is newly stored at each END. Namely, the maximum value of scan time is stored into SD526 in BIN code.	QnA

Table App. 3.11. Special register

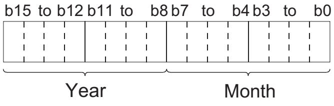
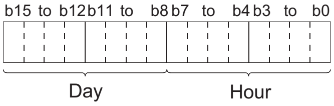
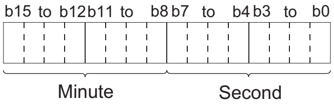
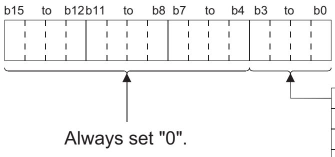
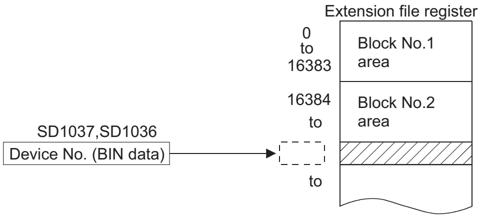
ACPU Special Register	Special Register after Conversion	Special Register for Modification	Name	Meaning	Details	Corresponding CPU
D9020	SD1020	×	Constant scan	Constant scan time (User sets in 10 ms units)	• Sets the interval between consecutive program starts in multiples of 10 ms. 0 : No setting 1 to 200 : Set. Program is executed at intervals of (set value)× 10 ms.	–
D9021	SD1021	–	Scan time	Scan time (1 ms units)	• At every END, the scan time is stored in BIN code and always rewritten.	QnA
D9022	SD1022	SD412	1 second counter	Count in units of 1s.	• When the PC CPU starts running, it starts counting 1 every second. • It starts counting up from 0 to 32767, then down to -32768 and then again up to 0. Counting repeats this routine.	QnA
D9025	SD1025	–	Clock data	Clock data (year, month)	• The year (last two digits) and month are stored as BCD code as shown below.  Example: 1987, July H8707	QnA
D9026	SD1026	–	Clock data	Clock data (day, hour)	• The day and hour are stored as BCD code as shown below.  Example: 31st, 10 a.m. H3110	QnA
D9027	SD1027	–	Clock data	Clock data (minute, second)	• The minute and second are stored as BCD code as shown below.  Example: 35 min, 48 sec. H3548	QnA
D9028	SD1028	–	Clock data	Clock data (day of week)	• The day of the week is stored as BCD code as shown below.  Example: Friday H0005 Day of the week 0 Sunday 1 Monday 2 Tuesday 3 Wednesday 4 Thursday 5 Friday 6 Saturday	QnA
D9035	SD1035	SD648	Extension file register	Use block No.	• Stores the block No. of the extension file register being used in BCD code.	QnA
D9036	SD1036	×	Extension file registerfor designation of device number	Device number when individual devices from extension file register are directly accessed	• Designate the device number for the extension file register for direct read and write in 2 words at SD1036 and SD1037 in BIN data. Use consecutive numbers beginning with R0 of block No. 1 to designate device numbers. 	–
D9037	SD1037	×				–

Table App. 3.11. Special register

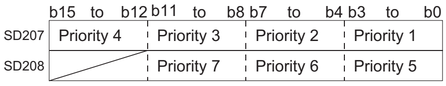
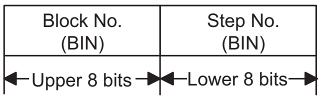
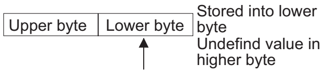
ACPU Special Register	Special Register after Conversion	Special Register for Modification	Name	Meaning	Details	Corresponding CPU
D9038	SD1038	SD207	LED display priority ranking	Priorities 1 to 4	• Sets priority of ERROR LEDs which illuminate (or flicker) to indicate errors with error code numbers. • Configuration of the priority setting areas is as shown below.  • For details, refer to the applicable CPUs User's Manual and the ACPU Programming manual (Fundamentals).	QnA
D9039	SD1039	SD208		Priorities 5 to 7		QnA
D9044	SD1044	×	For sampling trace	Step or time during sampling trace	• Turned on/off with a peripheral device. When STRA or STRAR is executed, the value stored in SD1044 is used as the sampling trace condition. At scanning--------0 At time--------------Time (10 msec unit) The value is stored into SD1044 in BIN code.	–
D9049	SD1049	×	Work area for SFC	Block number of extension file register	• Stores the block number of the expansion file register which is used as the work area for the execution of a SFC program in a binary value. • Stores "0" if an empty area of 16K bytes or smaller, which cannot be expansion file register No. 1, is used or if SM320 is OFF.	–
D9050	SD1050	×	SFC program error number	Error code generated by SFC program	• Stores error code of errors occurred in the SFC program in BIN code. 0 : No error 80: SFC program parameter error 81: SFC code error 82: Number of steps of simultaneous execution exceeded 83: Block start error 84: SFC program operation error	–
D9051	SD1051	×	Error block	Block number where error occurred	• Stores the block number in which an error occurred in the SFC program in BIN code. In the case of error 83 the starting block number is stored.	–
D9052	SD1052	×	Error step	Step number where error occurred	• Stores the step number, where error code 84 occurred in an SFC program, in BIN value. • Stores "0" when error code 80, 81 or 82 occurred. • Stores the block stating step number when error code 83 occurs.	–
D9053	SD1053	×	Error transition	Transition condition number where error occurred	• Stores the transition condition number, where error code 84 occurred in an SFC program, in BIN value. Stores "0" when error code 80, 81, 82 or 83 occurred.	–
D9054	SD1054	×	Error sequence step	Sequence step number where error occurred	• Stores the sequence step number of transfer condition and operation output in which error 84 occurred in the SFC program in BIN code.	–
D9055	SD1055	SD812	Status latch execution step number	Status latch step	• Stores the step number when status latch is executed. • Stores the step number in a binary value if status latch is executed in a main sequence program. • Stores the block number and the step number if status latch is executed in a SFC program. 	QnA
D9060	SD1060	SD392	Software version	Software version of internal software	• Stores the software version of the internal system in ASCII code.  For version "A", for example, "41_H" is stored. Note: The software version of the initial system may differ from the version indicated by the version information printed on the rear of the case.	QnA
D9072	SD1072	×	PLC communication check	Data check of serial communication module	• In the self-loopback test of the serial communication module, the serial communication module writes/reads data automatically to make communication checks.	–
D9081	SD1081	SD714	Number of empty blocks in communications request registrtion area	Number of empty blocks in communications request registration area	• Stores the number of empty blocks in the communication request registration area to the remote terminal module connected to the MELSECNET/MINI-S3 master unit, A2CCPU or A52GCPU.	QnA

Table App. 3.11. Special register

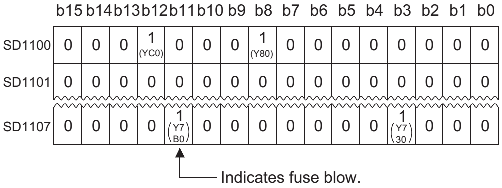
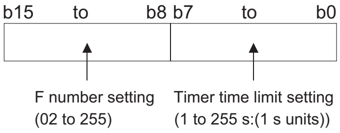
ACPU Special Register	Special Register after Conversion	Special Register for Modification	Name	Meaning	Details	Corresponding CPU
D9085	SD1085	×	Register for setting time check value	1 s to 65535 s	• Sets the time check time of the data link instructions (ZNRD, ZNWR) for the MELSECNET/10. • Setting range : 1 s to 65535 s (1 to 65535) • Setting unit: 1 s • Default value : 10 s (If 0 has been set, default 10 s is applied)	–
D9090	SD1090	×	Number of special functions modules over	Number of special functions modules over	• For details, refer to the manual of each microcomputer program package.	–
D9091	SD1091	×	Detailed error code	Self-diagnosis detailed error code	• Stores the detail code of cause of an instruction error.	–
D9094	SD1094	SD251	Head I/O number of I/O module to be replaced	Head I/O number of I/O module to be replaced	• Stores the first two digits of the head I/O number of the I/O module, which will be dismounted/mounted online (with power on), in BIN value. Example) Input module X2F0 → H2F	QnA
D9100	SD1100	–	Fuse blown module	Bit pattern in units of 16 points, indicating the modules whose fuses have blown	• Output module numbers (in units of 16 points), of which fuses have blown, are entered in bit pattern. (Preset output module numbers when parameter setting has been performed.)  b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0 SD1100 0 0 0 1 (YC0) 0 0 0 1 (Y80) 0 0 0 0 0 0 0 0 SD1101 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 SD1107 0 0 0 0 1 (Y7B0) 0 0 0 0 0 0 0 1 (Y730) 0 0 0 For a module whose number of output points exceeds 16 points, all bits corresponding to output module numbers within the number of output points occupied by the module (in increments of 16 points) turn on. (Example) When a 64-point module is mounted on the slot 0, b0 to b3 turn on when the fuse has blown. • Fuse blow check is executed also to the output module of remote I/O station. (If normal status is restored, clear is not performed. Therefore, it is required to perform clear by user program.)	QnA
D9101	SD1101					
D9102	SD1102					
D9103	SD1103					
D9104	SD1104					
D9105	SD1105					
D9106	SD1106					
D9107	SD1107					
D9108	SD1108	–	Step transfer monitoring timer setting	Timer setting valve and the F number at time out	• Set the value of the step transition monitoring timer and the annunciator number (F number) that will be turned ON when the monitoring timer times out.  • By turning ON any of SM1108 to SM1114, the monitoring timer starts. If the transition condition following a step which corresponds to the timer is not established within set time, set annunciator (F) is turned on.)	QnA
D9109	SD1109					
D9110	SD1110					
D9111	SD1111					
D9112	SD1112					
D9113	SD1113					
D9114	SD1114					

Table App. 3.11. Special register

<table>
<tr><th>ACPU Special Register</th><th>Special Register after Conversion</th><th>Special Register for Modification</th><th>Name</th><th>Meaning</th><th>Details</th><th>Corresponding CPU</th></tr>
<tr><td>D9116</td><td>SD1116</td><td rowspan="8">–</td><td rowspan="8">I/O module verification error</td><td rowspan="8">Bit pattern, in units of 16 points, indicating the modules with verification errors.</td><td rowspan="8">

- When I/O modules, of which data are different from those entered at power-ON, have been detected, the I/O module numbers (in units of 16 points) are entered in bit pattern. (Preset I/O module numbers set in parmeters when parameter setting has been performed.)

<table>
<tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>SD1116</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (XY 0)</td></tr>
<tr><td>SD1117</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (XY 190)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>SD1123</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (XY 7B0)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
</table>

Indicates an I/O module verify error.

For a module whose number of I/O points exceeds 16 points, all bits corresponding to I/O module numbers within the number of I/O points occupied by the module (in increments of 16 points) turn on.
(Example) When a 64-point module is mounted on the slot 0, b0 to b3 turn on when an error is detected.

- I/O module verify check is executed also to remote I/O station modules.
(If normal status is restored, clear is not performed. Therefore, it is required to perform clear by user program.)

</td><td rowspan="8">QnA</td></tr>
<tr><td>D9117</td><td>SD1117</td></tr>
<tr><td>D9118</td><td>SD1118</td></tr>
<tr><td>D9119</td><td>SD1119</td></tr>
<tr><td>D9120</td><td>SD1120</td></tr>
<tr><td>D9121</td><td>SD1121</td></tr>
<tr><td>D9122</td><td>SD1122</td></tr>
<tr><td>D9123</td><td>SD1123</td></tr>
<tr><td>D9124</td><td>SD1124</td><td>SD63</td><td>Number of annuciator detections</td><td>Number of annuciator detections</td><td>

- When one of F0 to 255 (F0 to 2047 for AuA and AnU) is turned on by $\boxed{\text{SET F}}$ 1 is added to the contents of SD63. When $\boxed{\text{RST F}}$ or $\boxed{\text{LEDR}}$ instruction is executed, 1 is subtracted from the contents of SD63.
(If the INDICATOR RESET switch is provided to the CPU module, pressing the switch can execute the same processing.)
- Quantity, which has been turned on by $\boxed{\text{SET F}}$ is stored into SD63 in BIN code. The value of SD63 is maximum 8.

</td><td>QnA</td></tr>
<tr><td>D9125</td><td>SD1125</td><td>SD64</td><td rowspan="8">Annunciator detection number</td><td rowspan="8">Annunciator detection number</td><td rowspan="8">

- When any of F0 to 2047 is turned on by $\boxed{\text{SET F}}$, the annunciator numbers (F numbers) that are turned on in order are registered into D9125 to D9132.
- The F number turned off by $\boxed{\text{RST F}}$ is erased from any of D9125 to D9132, and the F numbers stored after the erased F number are shifted to the preceding registerers.

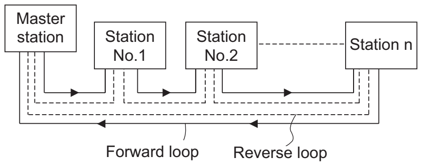
By executing $\boxed{\text{LEDR}}$ instruction, the contents of SD64 to SD71 are shifted upward by one. (For A3N, A3HCPU, it can be performed by use of INDICATOR RESET switch on front of CPU module.)
When there are 8 annunciator detections, the 9th one is not stored into SD64 to SD71 even if detected.

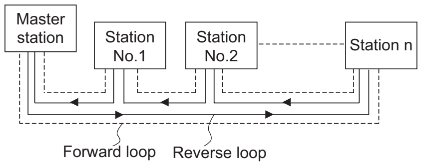
<table>
<tr><th></th><th></th><th>SET F50</th><th>SET F25</th><th>SET F99</th><th>RST F25</th><th>SET F15</th><th>SET F70</th><th>SET F65</th><th>SET F38</th><th>SET F110</th><th>SET F151</th><th>SET F210</th><th>LEDR</th></tr>
<tr><td>SD62</td><td>0</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>99</td></tr>
<tr><td>SD63</td><td>0</td><td>1</td><td>2</td><td>3</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>8</td><td>8</td></tr>
<tr><td>SD64</td><td>0</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>50</td><td>99</td></tr>
<tr><td>SD65</td><td>0</td><td>0</td><td>25</td><td>25</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>99</td><td>15</td></tr>
<tr><td>SD66</td><td>0</td><td>0</td><td>0</td><td>99</td><td>0</td><td>15</td><td>15</td><td>15</td><td>15</td><td>15</td><td>15</td><td>15</td><td>70</td></tr>
<tr><td>SD67</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>70</td><td>70</td><td>70</td><td>70</td><td>70</td><td>70</td><td>65</td></tr>
<tr><td>SD68</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>65</td><td>65</td><td>65</td><td>65</td><td>65</td><td>38</td></tr>
<tr><td>SD69</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>38</td><td>38</td><td>38</td><td>38</td><td>110</td></tr>
<tr><td>SD70</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>110</td><td>110</td><td>110</td><td>151</td></tr>
<tr><td>SD71</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>151</td><td>151</td><td>210</td></tr>
</table>

</td><td rowspan="8">QnA</td></tr>
<tr><td>D9126</td><td>SD1126</td><td>SD65</td></tr>
<tr><td>D9127</td><td>SD1127</td><td>SD66</td></tr>
<tr><td>D9128</td><td>SD1128</td><td>SD67</td></tr>
<tr><td>D9129</td><td>SD1129</td><td>SD68</td></tr>
<tr><td>D9130</td><td>SD1130</td><td>SD69</td></tr>
<tr><td>D9131</td><td>SD1131</td><td>SD70</td></tr>
<tr><td>D9132</td><td>SD1132</td><td>SD71</td></tr>
</table>

(10)Special register list dedicated for QnA

Table App. 3.12. Special register

<table>
<tr><th>ACPU Special Register</th><th>Special Register after Conversion</th><th>Special Register for Modification</th><th>Name</th><th>Meaning</th><th>Details</th><th>Corresponding CPU</th></tr>
<tr><td>D9200</td><td>SD1200</td><td>–</td><td>ZNRD instruction processing result (LRDP for ACPU)</td><td>0 : Normal end
2 : ZNRD instruction setting fault
3 : Error at relevant station
4 : Relevant station ZNRD execution disabled</td><td>Stores the execution result of the ZNRD (word device read) instruction
• ZNRD instruction setting faultFaulty setting of the ZNRD instruction constant, source, and/or destination.
• Corresponding station errorOne of the stations is not communicating.
• ZNRD cannot be executed in the corresponding station The specified station is a remote I/O station.</td><td>QnA</td></tr>
<tr><td>D9201</td><td>SD1201</td><td>–</td><td>ZNWR instruction processing result (LWTP for ACPU)</td><td>0 : Normal end
2 : ZNWR instruction setting fault
3 : Error at relevant station
4 : Relevant station ZNWR execution disabled</td><td>Stores the execution result of the ZNWR (word device write) instruction.
• ZNWR instruction setting faultFaulty setting of the ZNWR instruction constant, source, and/or destination.
• Corresponding station errorOne of the stations is not communicating.
• ZNWR cannot be executed in the corresponding station The specified station is a remote I/O station.</td><td>QnA</td></tr>
<tr><td>D9202</td><td>SD1202</td><td>–</td><td rowspan="2">Local station link type</td><td>Stores conditions for up to numbers 1 to 16</td><td rowspan="2">Stores whether the slave station corresponds to MELSECNET or MELSECNET II.
• Bits corresponding to the MELSECNET II stations become "1."
• Bits corresponding to the MELSECNET stations or unconnected become "0."
<table>
<tr><th rowspan="2">Device number</th><th colspan="16">Bit</th></tr>
<tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>SD1202</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr>
<tr><td>SD1203</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr>
<tr><td>SD1241</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr>
<tr><td>SD1242</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr>
</table>
• If a local station goes down during the operation, the contents before going down are retained.
Contents of SD1224 to SD1227 and SD1228 to SD1231 are ORed.
If the corresponding bit is "0", the corresponding bit of the special register above becomes valid.
• If the host (master) station goes down, the contents before going down are also retained.</td><td>QnA</td></tr>
<tr><td>D9203</td><td>SD1203</td><td>–</td><td>Stores conditions for up to numbers 17 to 32</td><td>QnA</td></tr>
<tr><td>D9204</td><td>SD1204</td><td>–</td><td>Link status</td><td>0 : Forward loop, during data link
1 : Reverse loop, during data link
2 : Loopback implemented in forward/ reverse directions
3 : Loopback implemented only in forward direction
4 : Loopback implemented only in reverse direction
5 : Data link disabled</td><td>Stores the present path status of the data link.
• Data link in forward loop


• Data link in reverse loop


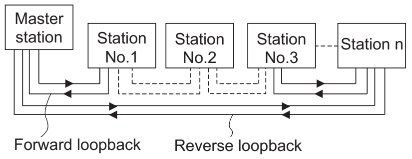
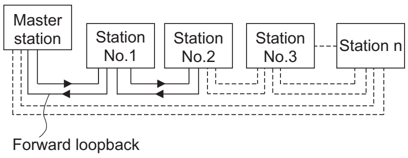
• Loopback implemented in forward/reverse directions

 </td><td>QnA</td></tr>
</table>

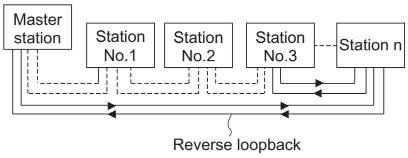
Table App. 3.12.Special register

<table>
<tr><th>ACPU Special Register</th><th>Special Register after Conversion</th><th>Special Register for Modification</th><th>Name</th><th>Meaning</th><th>Details</th><th>Corresponding CPU</th></tr>
<tr><td>D9204</td><td>SD1204</td><td>–</td><td>Link status</td><td>0 : Forward loop, during data link
1 : Reverse loop, during data link
2 : Loopback implemented in forward/ reverse directions
3 : Loopback implemented only in forward direction
4 : Loopback implemented only in reverse direction
5 : Data link disabled</td><td>

- Loopback in forward loop only

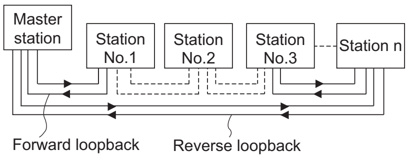


- Loopback in reverse loop only



</td><td>QnA
QnA</td></tr>
<tr><td>D9205</td><td>SD1205</td><td>–</td><td>Station implementing loopback</td><td>Station that implemented forward loopback</td><td rowspan="2">

Stores the local or remote I/O station number at which loopback is being executed.



In the above example, 1 is stored into D9205 and 3 into D9206.
If data link returns to normal status (data link in forward loop), values in D9205 and D9206 remain 1 and 3. To return them to "0", therefore, use a sequence program or perform reset operation.

</td><td>QnA</td></tr>
<tr><td>D9206</td><td>SD1206</td><td>–</td><td>Station implementing loopback</td><td>Station that implemented reverse loopback</td><td>QnA</td></tr>
<tr><td>D9210</td><td>SD1210</td><td>–</td><td>Number of retries</td><td>Stored as cumulative value</td><td>Stores the number of retry times due to transmission error.
Count stops at maximum of "FFFFH" .
To return the value to "0", perform reset operation.</td><td>QnA</td></tr>
<tr><td>D9211</td><td>SD1211</td><td>–</td><td>Number of times loop selected</td><td>Stored as cumulative value</td><td>Stores the number of times the loop line has been switched to reverse loop or loopback.
Count stops at maximum of "FFFFH".
To return the value to "0", perform reset operation.</td><td>QnA</td></tr>
<tr><td>D9212</td><td>SD1212</td><td>–</td><td>Local station operation status</td><td>Stores conditions for up to numbers 1 to 16</td><td rowspan="4">

Stores the local station numbers which are in STOP or PAUSE mode.

<table>
<tr><th rowspan="2">Device number</th><th colspan="16">Bit</th></tr>
<tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>SD1212</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr>
<tr><td>SD1213</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr>
<tr><td>SD1214</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr>
<tr><td>SD1215</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr>
</table>

When a local station is switched to STOP or PAUSE mode, the bit corresponding to the station number in the register becomes "1".
Example: When station 7 switches to STOP mode, b6 in SD1212 becomes "1" , and when SD1212 is monitored, its value is "64 (40H)".

</td><td rowspan="4">QnA</td></tr>
<tr><td>D9213</td><td>SD1213</td><td>–</td><td>Local station operation status</td><td>Stores conditions for up to numbers 17 to 32</td></tr>
<tr><td>D9214</td><td>SD1214</td><td>–</td><td>Local station operation status</td><td>Stores conditions for up to numbers 33 to 48</td></tr>
<tr><td>D9215</td><td>SD1215</td><td>–</td><td>Local station operation status</td><td>Stores conditions for up to numbers 49 to 64</td></tr>
<tr><td>D9216</td><td>SD1216</td><td>–</td><td>Local station error detect status</td><td>Stores conditions for up to numbers 1 to 16</td><td rowspan="4">

Stores the local station numbers which are in error.

<table>
<tr><th rowspan="2">Device number</th><th colspan="16">Bit</th></tr>
<tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>SD1216</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr>
<tr><td>SD1217</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr>
<tr><td>SD1218</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr>
<tr><td>SD1219</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr>
</table>

If a local station detects an error, the bit corresponding to the station number becomes "1".
Example: When station 6 and 12 detect an error, b5 and 11 in SD1216 become "1", and when SD1216 is monitored, its value is "2080 (820H)".

</td><td rowspan="4">QnA</td></tr>
<tr><td>D9217</td><td>SD1217</td><td>–</td><td>Local station error detect status</td><td>Stores conditions for up to numbers 17 to 32</td></tr>
<tr><td>D9218</td><td>SD1218</td><td>–</td><td>Local station error detect status</td><td>Stores conditions for up to numbers 33 to 48</td></tr>
<tr><td>D9219</td><td>SD1219</td><td>–</td><td>Local station error detect status</td><td>Stores conditions for up to numbers 49 to 64</td></tr>
</table>

Table App. 3.12.Special register

<table>
<tr><th>ACPU Special Register</th><th>Special Register after Conversion</th><th>Special Register for Modification</th><th>Name</th><th>Meaning</th><th>Details</th><th>Corresponding CPU</th></tr>
<tr><td>D9220</td><td>SD1220</td><td>–</td><td>Local station parameters non-conforming; remote I/O station I/O assignment error</td><td>Stores conditions for up to numbers 1 to 16</td><td rowspan="4">Stores the local station numbers which contain mismatched parameters or of remote station numbers for which incorrect I/O assignment has been made.
<table>
<tr><th rowspan="2">Device number</th><th colspan="16">Bit</th></tr>
<tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>SD1220</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr>
<tr><td>SD1221</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr>
<tr><td>SD1222</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr>
<tr><td>SD1223</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr>
</table>
If a local station acting as the master station of tier three detects a parameter error or a remote I/O station whose I/O assignment is abnormal, the bit of the device number corresponding to the station number of that local station or remote I/O station turns to "1".
Example: When local station 5 and remote I/O station 14 detect an error, b4 and b13 in SD1220 become "1" , and when SD1220 is monitored, its value is "8208 (2010H) " .</td><td rowspan="4">QnA</td></tr>
<tr><td>D9221</td><td>SD1221</td><td>–</td><td>Local station parameters non-conforming; remote I/O station I/O assignment error</td><td>Stores conditions for up to numbers 17 to 32</td></tr>
<tr><td>D9222</td><td>SD1222</td><td>–</td><td>Local station parameters non-conforming; remote I/O station I/O assignment error</td><td>Stores conditions for up to numbers 33 to 48</td></tr>
<tr><td>D9223</td><td>SD1223</td><td>–</td><td>Local station parameters non-conforming; remote I/O station I/O assignment error</td><td>Stores conditions for up to numbers 49 to 64</td></tr>
<tr><td>D9224</td><td>SD1224</td><td>–</td><td>Local station and remote I/O station initial communications underway</td><td>Stores conditions for up to numbers 1 to 16</td><td rowspan="4">Stores the local or remote station numbers while they are communicating the initial data with their relevant master station.
<table>
<tr><th rowspan="2">Device number</th><th colspan="16">Bit</th></tr>
<tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>SD1224</td><td>L/R 16</td><td>L/R 15</td><td>L/R 14</td><td>L/R 13</td><td>L/R 12</td><td>L/R 11</td><td>L/R 10</td><td>L/R 9</td><td>L/R 8</td><td>L/R 7</td><td>L/R 6</td><td>L/R 5</td><td>L/R 4</td><td>L/R 3</td><td>L/R 2</td><td>L/R 1</td></tr>
<tr><td>SD1225</td><td>L/R 32</td><td>L/R 31</td><td>L/R 30</td><td>L/R 29</td><td>L/R 28</td><td>L/R 27</td><td>L/R 26</td><td>L/R 25</td><td>L/R 24</td><td>L/R 23</td><td>L/R 22</td><td>L/R 21</td><td>L/R 20</td><td>L/R 19</td><td>L/R 18</td><td>L/R 17</td></tr>
<tr><td>SD1226</td><td>L/R 48</td><td>L/R 47</td><td>L/R 46</td><td>L/R 45</td><td>L/R 44</td><td>L/R 43</td><td>L/R 42</td><td>L/R 41</td><td>L/R 40</td><td>L/R 39</td><td>L/R 38</td><td>L/R 37</td><td>L/R 36</td><td>L/R 35</td><td>L/R 34</td><td>L/R 33</td></tr>
<tr><td>SD1227</td><td>L/R 64</td><td>L/R 63</td><td>L/R 62</td><td>L/R 61</td><td>L/R 60</td><td>L/R 59</td><td>L/R 58</td><td>L/R 57</td><td>L/R 56</td><td>L/R 55</td><td>L/R 54</td><td>L/R 53</td><td>L/R 52</td><td>L/R 51</td><td>L/R 50</td><td>L/R 49</td></tr>
</table>
The bit corresponding to the station number which is currently communicating the initial settings becomes "1" .
Example: When stations 23 and 45 are communicating, b6 of SD1225 and b12 of SD1226 become "1", and when SD1225 is monitored, its value is "64 (40H)", and when SD1226 is monitored, its value is "4096 (1000H) " .</td><td rowspan="4">QnA</td></tr>
<tr><td>D9225</td><td>SD1225</td><td>–</td><td>Local station and remote I/O station initial communications underway</td><td>Stores conditions for up to numbers 17 to 32</td></tr>
<tr><td>D9226</td><td>SD1226</td><td>–</td><td>Local station and remote I/O station initial communications underway</td><td>Stores conditions for up to numbers 33 to 48</td></tr>
<tr><td>D9227</td><td>SD1227</td><td>–</td><td>Local station and remote I/O station initial communications underway</td><td>Stores conditions for up to numbers 49 to 64</td></tr>
<tr><td>D9228</td><td>SD1228</td><td>–</td><td>Local station and remote I/O station error</td><td>Stores conditions for up to numbers 1 to 16</td><td rowspan="4">Stores the local or remote station numbers which are in error.
<table>
<tr><th rowspan="2">Device number</th><th colspan="16">Bit</th></tr>
<tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>SD1228</td><td>L/R 16</td><td>L/R 15</td><td>L/R 14</td><td>L/R 13</td><td>L/R 12</td><td>L/R 11</td><td>L/R 10</td><td>L/R 9</td><td>L/R 8</td><td>L/R 7</td><td>L/R 6</td><td>L/R 5</td><td>L/R 4</td><td>L/R 3</td><td>L/R 2</td><td>L/R 1</td></tr>
<tr><td>SD1229</td><td>L/R 32</td><td>L/R 31</td><td>L/R 30</td><td>L/R 29</td><td>L/R 28</td><td>L/R 27</td><td>L/R 26</td><td>L/R 25</td><td>L/R 24</td><td>L/R 23</td><td>L/R 22</td><td>L/R 21</td><td>L/R 20</td><td>L/R 19</td><td>L/R 18</td><td>L/R 17</td></tr>
<tr><td>SD1230</td><td>L/R 48</td><td>L/R 47</td><td>L/R 46</td><td>L/R 45</td><td>L/R 44</td><td>L/R 43</td><td>L/R 42</td><td>L/R 41</td><td>L/R 40</td><td>L/R 39</td><td>L/R 38</td><td>L/R 37</td><td>L/R 36</td><td>L/R 35</td><td>L/R 34</td><td>L/R 33</td></tr>
<tr><td>SD1231</td><td>L/R 64</td><td>L/R 63</td><td>L/R 62</td><td>L/R 61</td><td>L/R 60</td><td>L/R 59</td><td>L/R 58</td><td>L/R 57</td><td>L/R 56</td><td>L/R 55</td><td>L/R 54</td><td>L/R 53</td><td>L/R 52</td><td>L/R 51</td><td>L/R 50</td><td>L/R 49</td></tr>
</table>
The bit corresponding to the station number with the error becomes "1" .
Example: When local station 3 and remote I/O station 14 have an error, b2 and b13 of SD1228 become "1", and when SD1228 is monitored, its value is "8196 (2004H)".</td><td rowspan="4">QnA</td></tr>
<tr><td>D9229</td><td>SD1229</td><td>–</td><td>Local station and remote I/O station error</td><td>Stores conditions for up to numbers 17 to 32</td></tr>
<tr><td>D9230</td><td>SD1230</td><td>–</td><td>Local station and remote I/O station error</td><td>Stores conditions for up to numbers 33 to 48</td></tr>
<tr><td>D9231</td><td>SD1231</td><td>–</td><td>Local station and remote I/O station error</td><td>Stores conditions for up to numbers 49 to 64</td></tr>
</table>

Table App. 3.12.Special register

<table>
<tr><th>ACPU Special Register</th><th>Special Register after Conversion</th><th>Special Register for Modification</th><th>Name</th><th>Meaning</th><th>Details</th><th>Corresponding CPU</th></tr>
<tr><td>D9232</td><td>SD1232</td><td>–</td><td>Local station and remote I/O station loop error</td><td>Stores conditions for up to numbers 1 to 8</td><td rowspan="8">Stores the local or remote station number at which a forward or reverse loop error has occurred
<table>
<tr><th rowspan="2">Device number</th><th colspan="16">Bit</th></tr>
<tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td rowspan="2">SD1232</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td></tr>
<tr><td colspan="2">L/R8</td><td colspan="2">L/R7</td><td colspan="2">L/R6</td><td colspan="2">L/R5</td><td colspan="2">L/R4</td><td colspan="2">L/R3</td><td colspan="2">L/R2</td><td colspan="2">L/R1</td></tr>
<tr><td rowspan="2">SD1233</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td></tr>
<tr><td colspan="2">L/R16</td><td colspan="2">L/R15</td><td colspan="2">L/R14</td><td colspan="2">L/R13</td><td colspan="2">L/R12</td><td colspan="2">L/R11</td><td colspan="2">L/R10</td><td colspan="2">L/R9</td></tr>
<tr><td rowspan="2">SD1234</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td></tr>
<tr><td colspan="2">L/R24</td><td colspan="2">L/R23</td><td colspan="2">L/R22</td><td colspan="2">L/R21</td><td colspan="2">L/R20</td><td colspan="2">L/R19</td><td colspan="2">L/R18</td><td colspan="2">L/R17</td></tr>
<tr><td rowspan="2">SD1235</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td></tr>
<tr><td colspan="2">L/R32</td><td colspan="2">L/R31</td><td colspan="2">L/R30</td><td colspan="2">L/R29</td><td colspan="2">L/R28</td><td colspan="2">L/R27</td><td colspan="2">L/R26</td><td colspan="2">L/R25</td></tr>
<tr><td rowspan="2">SD1236</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td></tr>
<tr><td colspan="2">L/R40</td><td colspan="2">L/R39</td><td colspan="2">L/R38</td><td colspan="2">L/R37</td><td colspan="2">L/R36</td><td colspan="2">L/R35</td><td colspan="2">L/R34</td><td colspan="2">L/R33</td></tr>
<tr><td rowspan="2">SD1237</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td></tr>
<tr><td colspan="2">L/R48</td><td colspan="2">L/R47</td><td colspan="2">L/R46</td><td colspan="2">L/R45</td><td colspan="2">L/R44</td><td colspan="2">L/R43</td><td colspan="2">L/R42</td><td colspan="2">L/R41</td></tr>
<tr><td rowspan="2">SD1238</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td></tr>
<tr><td colspan="2">L/R56</td><td colspan="2">L/R55</td><td colspan="2">L/R54</td><td colspan="2">L/R53</td><td colspan="2">L/R52</td><td colspan="2">L/R51</td><td colspan="2">L/R50</td><td colspan="2">L/R49</td></tr>
<tr><td rowspan="2">SD1239</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td><td>R</td><td>F</td></tr>
<tr><td colspan="2">L/R64</td><td colspan="2">L/R63</td><td colspan="2">L/R62</td><td colspan="2">L/R61</td><td colspan="2">L/R60</td><td colspan="2">L/R59</td><td colspan="2">L/R58</td><td colspan="2">L/R57</td></tr>
</table>
"F" in the above table indicates a forward loop line, and "R" a reverse loop line. The bit of the device number corresponding to the station number of the local station or remote I/O station that has a forward loop line or reverse loop line error.
Example: When the forward loop line of station 5 has an error, b8 of SD1232 become "1", and when SD1232 is monitored, its value is "256 (100H)".</td><td rowspan="8">QnA</td></tr>
<tr><td>D9233</td><td>SD1233</td><td>–</td><td>Local station and remote I/O station loop error</td><td>Stores conditions for up to numbers 9 to 16</td></tr>
<tr><td>D9234</td><td>SD1234</td><td>–</td><td>Local station and remote I/O station loop error</td><td>Stores conditions for up to numbers 17 to 24</td></tr>
<tr><td>D9235</td><td>SD1235</td><td>–</td><td>Local station and remote I/O station loop error</td><td>Stores conditions for up to numbers 25 to 32</td></tr>
<tr><td>D9236</td><td>SD1236</td><td>–</td><td>Local station and remote I/O station loop error</td><td>Stores conditions for up to numbers 33 to 40</td></tr>
<tr><td>D9237</td><td>SD1237</td><td>–</td><td>Local station and remote I/O station loop error</td><td>Stores conditions for up to numbers 41 to 48</td></tr>
<tr><td>D9238</td><td>SD1238</td><td>–</td><td>Local station and remote I/O station loop error</td><td>Stores conditions for up to numbers 49 to 56</td></tr>
<tr><td>D9239</td><td>SD1239</td><td>–</td><td>Local station and remote I/O station loop error</td><td>Stores conditions for up to numbers 57 to 64</td></tr>
<tr><td>D9240</td><td>SD1240</td><td>–</td><td>Number of times communications errors detected</td><td>Stores cumulative total of receive errors</td><td>Stores the number of times the following transmission errors have been detected:
CRC, OVER, AB. IF
Count is made to a maximum of FFFFH.
To return the value to "0", perform reset operation.</td><td>QnA</td></tr>
<tr><td>D9241</td><td>SD1241</td><td rowspan="2">–</td><td rowspan="2">Local station link type</td><td>Stores conditions for up to numbers 33 to 48</td><td rowspan="2">Stores whether the slave station corresponds to MELSECNET or MELSECNET II.
• Bits corresponding to the MELSECNET II stations become "1."
• Bits corresponding to the MELSECNET stations or unconnected become "0."
<table>
<tr><th rowspan="2">Device number</th><th colspan="16">Bit</th></tr>
<tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>SD1202</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr>
<tr><td>SD1203</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr>
<tr><td>SD1241</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr>
<tr><td>SD1242</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr>
</table>
• If a local station goes down during the operation, the contents before going down are retained.
Contents of SD1224 to SD1227 and SD1228 to SD1231 are ORed.
If the corresponding bit is "0", the corresponding bit of the special register above becomes valid.
• If the host (master) station goes down, the contents before going down are also retained.</td><td rowspan="2">QnA</td></tr>
<tr><td>D9242</td><td>SD1242</td><td>Stores conditions for up to numbers 49 to 64</td></tr>
<tr><td>D9243</td><td>SD1243</td><td>–</td><td>Station number information for host station</td><td>Stores station number (0 to 64)</td><td>Allows a local station to confirm its own station number</td><td>QnA</td></tr>
<tr><td>D9244</td><td>SD1244</td><td>–</td><td>Number of link device stations</td><td>Stores number of slave stations</td><td>Indicates the number of slave stations in one loop.</td><td>QnA</td></tr>
<tr><td>D9245</td><td>SD1245</td><td>–</td><td>Receive error detection count</td><td>Stores cumulative total of receive errors</td><td>Stores the number of times the following transmission errors have been detected: CRC, OVER, AB. IF
Count is made to a maximum of FFFFH. To return the value to "0", perform reset operation.</td><td>QnA</td></tr>
</table>

Table App. 3.12.Special register

<table>
<tr><th>ACPU Special Register</th><th>Special Register after Conversion</th><th>Special Register for Modification</th><th>Name</th><th>Meaning</th><th>Details</th><th>Corresponding CPU</th></tr>
<tr><td>D9248</td><td>SD1248</td><td>–</td><td>Local station operation status</td><td>Stores conditions for up to numbers 1 to 16</td><td rowspan="4">Stores the local station number which is in STOP or PAUSE mode.
<table>
<tr><th rowspan="2">Device number</th><th colspan="16">Bit</th></tr>
<tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>SD1248</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr>
<tr><td>SD1249</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr>
<tr><td>SD1250</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr>
<tr><td>SD1251</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr>
</table>
The bit corresponding to the station number which is in STOP or PAUSE mode, becomes "1".
Example: When local stations 7 and 15 are in STOP mode, b6 and b14 of SD1248 become "1", and when SD1248 is monitored, its value is "16448 (4040H)".</td><td rowspan="4">QnA</td></tr>
<tr><td>D9249</td><td>SD1249</td><td>–</td><td>Local station operation status</td><td>Stores conditions for up to numbers 17 to 32</td></tr>
<tr><td>D9250</td><td>SD1250</td><td>–</td><td>Local station operation status</td><td>Stores conditions for up to numbers 33 to 48</td></tr>
<tr><td>D9251</td><td>SD1251</td><td>–</td><td>Local station operation status</td><td>Stores conditions for up to numbers 49 to 64</td></tr>
<tr><td>D9252</td><td>SD1252</td><td>–</td><td>Local station error conditions</td><td>Stores conditions for up to numbers 1 to 16</td><td rowspan="4">Stores the local station number other than the host, which is in error.
<table>
<tr><th rowspan="2">Device number</th><th colspan="16">Bit</th></tr>
<tr><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>SD1252</td><td>L16</td><td>L15</td><td>L14</td><td>L13</td><td>L12</td><td>L11</td><td>L10</td><td>L9</td><td>L8</td><td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td></tr>
<tr><td>SD1253</td><td>L32</td><td>L31</td><td>L30</td><td>L29</td><td>L28</td><td>L27</td><td>L26</td><td>L25</td><td>L24</td><td>L23</td><td>L22</td><td>L21</td><td>L20</td><td>L19</td><td>L18</td><td>L17</td></tr>
<tr><td>SD1254</td><td>L48</td><td>L47</td><td>L46</td><td>L45</td><td>L44</td><td>L43</td><td>L42</td><td>L41</td><td>L40</td><td>L39</td><td>L38</td><td>L37</td><td>L36</td><td>L35</td><td>L34</td><td>L33</td></tr>
<tr><td>SD1255</td><td>L64</td><td>L63</td><td>L62</td><td>L61</td><td>L60</td><td>L59</td><td>L58</td><td>L57</td><td>L56</td><td>L55</td><td>L54</td><td>L53</td><td>L52</td><td>L51</td><td>L50</td><td>L49</td></tr>
</table>
The bit corresponding to the station number which is in error, becomes "1".
Example: When local station 12 is in error, b11 of SD1252 becomes "1", and when SD1252 is monitored, its value is "2048 (800H) ".</td><td rowspan="4">QnA</td></tr>
<tr><td>D9253</td><td>SD1253</td><td>–</td><td>Local station error conditions</td><td>Stores conditions for up to numbers 17 to 32</td></tr>
<tr><td>D9254</td><td>SD1254</td><td>–</td><td>Local station error conditions</td><td>Stores conditions for up to numbers 33 to 48</td></tr>
<tr><td>D9255</td><td>SD1255</td><td>–</td><td>Local station error conditions</td><td>Stores conditions for up to numbers 49 to 64</td></tr>
</table>

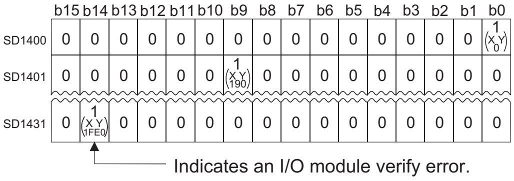
(11) Fuse blown module

Table App. 3.13. Special register

<table>
<tr><th>Number</th><th>Name</th><th>Meaning</th><th>Explanation</th><th>Set by (When Set)</th><th>Corres-ponding ACPU D9□□□</th><th>Corresponding CPU</th></tr>
<tr><td>SD1300</td><td rowspan="11">Fuse blown module</td><td rowspan="11">Bit pattern in units of 16 points, indicating the modules whose fuses have blown
0 : No blown fuse
1 : Blown fuse present</td><td rowspan="11">• The numbers of output modules whose fuses have blown are input as a bit pattern (in units of 16 points).
(If the module numbers are set by parameter, the parameter-set numbers are stored.)
• Also detects blown fuse condition at remote station output modules
<table>
<tr><th></th><th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th></tr>
<tr><td>SD1300</td><td>0</td><td>0</td><td>0</td><td>1 (YC0)</td><td>0</td><td>0</td><td>0</td><td>1 (Y80)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>SD1301</td><td>1 (Y1F0)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (Y1A0)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>SD1331</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (Y1FB0)</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1 (Y1F30)</td><td>0</td><td>0</td><td>0</td></tr>
</table>
Indicates fuse blow.
For a module whose number of output points exceeds 16 points, all bits corresponding to output module numbers within the number of output points occupied by the module (in increments of 16 points) turn on.
(Example) When a 64-point module is mounted on the slot 0, b0 to b3 turn on when the fuse has blown.
• Not cleared even if the blown fuse is replaced with a new one. This flag is cleared by error resetting operation.</td><td rowspan="11">S (Error)</td><td>D9100</td><td rowspan="11">QnA</td></tr>
<tr><td>SD1301</td><td>D9101</td></tr>
<tr><td>SD1302</td><td>D9102</td></tr>
<tr><td>SD1303</td><td>D9103</td></tr>
<tr><td>SD1304</td><td>D9104</td></tr>
<tr><td>SD1305</td><td>D9105</td></tr>
<tr><td>SD1306</td><td>D9106</td></tr>
<tr><td>SD1307</td><td>D9107</td></tr>
<tr><td>SD1308</td><td>New</td></tr>
<tr><td>SD1309
to
SD1330</td><td>New</td></tr>
<tr><td>SD1331</td><td>New</td></tr>
</table>

(12)I/O module verification

Table App. 3.14. Special register

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD1400	I/O module verify error	Bit pattern, in units of 16 points, indicating the modules with verification errors. 0 : No I/O verification errors 1 : I/O verification error present	• When the I/O modules whose I/O module information differs from that registered at power-ON are detected, the numbers of those I/O modules are entered in bit pattern. (If the I/O numbers are set by parameter, the parameter-set numbers are stored.) • Also detects I/O module information.  Indicates an I/O module verify error. For a module whose number of I/O points exceeds 16 points, all bits corresponding to I/O module numbers within the number of I/O points occupied by the module (in increments of 16 points) turn on. (Example) When a 64-point module is mounted on the slot 0, b0 to b3 turn on when an error is detected. • Not cleared even if the blown fuse is replaced with a new one. This flag is cleared by error resetting operation.	S (Error)	D9116	QnA
SD1401					D9117	
SD1402					D9118	
SD1403					D9119	
SD1404					D9120	
SD1405					D9121	
SD1406					D9122	
SD1407					D9123	
SD1408					New	
SD1409 to SD1430					New	
SD1431					New	

(13)Process control instructions

Table App. 3.15. Special register

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD1500 SD1501	Basic period	Basic period tome	• Set the basic period (1 second units) use for the process control instruction using floating point data. Floating point data = SD1501 | SD1500	U	New	Q4AR
SD1502	Process control instruction detail error code	Process control instruction detail error code	• Shows the detailed error contents for the error that occurred in the process control instruction.	S (Error)	New	Q4AR
SD1503	Process control instruction generated error location	Process control instruction generated error location	• Shows the error process block that occurred in the process control instruction.	S (Error)	New	Q4AR

(14)For redundant systems (Host system CPU information [*1])

SD1510 to SD1599 are only valid for redundant systems.

They are all set to 0 for stand-alone systems.

Table App. 3.16. Special register

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD1512	Operation mode during CPU module start up	Hot start switch power out time	• Shows the power out time (S) during the automatic switch from hot start to initial start in the operation mode when the CPU module is started up.	S (Initial)	New	Q4AR
SD1590	Switch request network No.	Request source network No.	• Stores the request source at work No. when the SM1590 is turned on.	S (Error)	New	Q4AR

(15)For redundant systems (Other system CPU information [*1])

SD1600 to SD1659 is only valid during the back up mode for redundant systems, and refresh cannot be done when in the separate mode.

SD1651 to SD1699 are valid in either the backup mode or separate mode.

When a stand-alone system SD1600 to SD1699 are all 0.

Table App. 3.17. Special register

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU SD□□*2	Corresponding CPU
SD1600	Diagnosis error	Diagnosis error No.	• Stores as BIN code the error No. of the error that occurred during the other system CPU module diagnosis. • Stores the latest error currently occurring.	S (Each END)	SD0	Q4AR
SD1601	Diagnosis error occurrence time	Diagnosis error occurrence time	• SD1600 stores the updated date and time. • Stores each of the BCD two digits. • Refer to SD1 to SD3 for the storage status. (SD1 → SD1601, SD2 → SD1602, SD3 → SD1603)	S (Each END)	SD1 to SD3	Q4AR
SD1602						
SD1603						
SD1604	Error information classification	Error information classification	• Stores the error comment information/individual information classification code. • Refer to SD4 for the storage status.	S (Each END)	SD4	Q4AR
SD1605	Error common information	Error common information	• Stores the common information for the error code. • Refer to SD5 to SD15 for the storage status. (SD5 → SD1605, SD6 → SD1606, SD7 → SD1607, SD8 → SD1608, SD9 → SD1609, SD10 → SD1610, SD11 → SD1611, SD12 → SD1612, SD13 → SD1613, SD14 → SD1614, SD15 → SD1615)	S (Each END)	SD5 to SD15	Q4AR
SD1606						
SD1607						
SD1608						
SD1609						
SD1610						
SD1611						
SD1612						
SD1613						
SD1614						
SD1615						
SD1616	Error individual information	Error individual information	• Stores the individual information for the error code. • Refer to SD16 to SD26 for the storage status. (SD16 → SD1616, SD17 → SD1617, SD18 → SD1618, SD19 → SD1619, SD20 → SD1620, SD21 → SD1621, SD22 → SD1622, SD23 → SD1623, SD24 → SD1624, SD25 → SD1625, SD26 → SD1626)	S (Each END)	SD16 to SD26	Q4AR
SD1617						
SD1618						
SD1619						
SD1620						
SD1621						
SD1622						
SD1623						
SD1624						
SD1625						
SD1626						
SD1650	Switch status	CPU module switch status	• Stores the CPU module switch status. • Refer to SD200 for the storage status. (SD1650 → SD200)	S (Each END)	SD200	Q4AR
SD1651	LED status	CPU module -LED status	• Stores the CPU module's LED status. • Shows 0 when turned off, 1 when turned on, and 2 when flicking. • Refer to SD201 for the storage status. (SD1651 → SD201)	S (Each END)	SD201	Q4AR
SD1653	CPU module operation status	CPU module operation status	• Stores the CPU module operation status. • Refer to SD203 for the storage status. (SD1653 → SD203)	S (Each END)	SD203	Q4AR

[*1] : Stores other system CPU module diagnostics information and system information.

[*2] : Shows the special register (SD□□) for the host system CPU module.

(16)For redundant systems (Trucking)

SD1700 to SD1779 is valid only for redundant systems.

These are all 0 for stand-alone systems.

Table App. 3.18. Special register

Number	Name	Meaning	Explanation	Set by (When Set)	Corres-ponding ACPU D9□□□	Corresponding CPU
SD1700	Tracking error detection count	Tracking error detection count	• When the tracking error is detected, count is added by one.	S(Error)	New	Q4AR

APPENDIX 4 PRECAUTIONS FOR UTILIZING THE EXISTING MELSEC-A SERIES PROGRAM FOR Q2ASCPU

To utilize a sequence program, created for AnNCPU, AnACPU, or AnUCPU, for Q2ASCPU, convert it using the "A→QnA Conversion" option of the "Option" menu in the file maintenance mode of the GPP function.
For details on the GPP function operations, refer to the GX Developer Operating Manual or SW□IVD-GPPQ Operating Manual (Offline).
For details on instructions and devices, refer to the QCPU (Q mode)/QnACPU Programming Manual (Common Instructions).
The instructions, devices, and comments, etc. indicated below may require modification in each mode after conversion.

Appendix 4.1 Instructions

An□CPU Instruction	Instruction after A→QnA Conversion	Corrective Action
BMOVR instruction Program example: LEDA BMOVR LEDC D10 LEDC D100 SUB K10 LEDR	OUT SM1255 LEDC D10 LEDC D100 OUT SM1255 LEDR	Modify the instruction to a BMOV instruction. BMOV ZR100 ZR1000 K10
BXCHR instruction Program example: LEDA BXCHR LEDC D10 LEDC D100 SUB K10 LEDR	OUT SM1255 LEDC D10 LEDC D100 OUT SM1255 LEDR	Modify the instruction to a BXCH instruction. BXCH ZR100 ZR1000 K10
CHG instruction	OUT SM1255	Since the Q2ASCPU does not have the main/subsequence program system, it has no CHG instructions.Delete OUT SM1255 as it is not necessary.Modify the main/subsequence program after conversion and set new parameters. (Refer to Appendix 4.5)
CHK instruction Program example: CHK M10 X100	CHK	Refer to Appendix 4.12
CLC instruction Program example: CLC	RST SM1012	Modify the instruction to SM700, special relay for carry flag. RST SM700

An□CPU Instruction	Instruction after A→QnA Conversion	Corrective Action
AnA/AnUCPU dedicated instruction IX instruction	OUT SM1255	Refer to Appendix 4.12
LEDA instruction (excluding dedicated instructions for AnACPU, AnUCPU) Program example: LEDA ABCDEFGH	OUT SM1255	Modify the instruction to an LED instruction. $MOV "ABCDEFGH" D0 $MOV "IJKLMNOP" D10 $+ D0 D10 D20 LED D20 LED display is performed after adding the right 8 characters and the left 8 characters.
LEDB instruction (excluding dedicated instructions for AnACPU, AnUCPU) Program example: LEDB IJKLMNOP	OUT SM1255	
LRDP instruction Program example: LRDP K3 D10 D100 K10	OUT SM1255	Modify the instruction to a ZNRD instruction. J.ZNRD J0 K3 D10 D100 K10 M0
LWTP instruction Program example: LWTP K3 D10 D100 K10	OUT SM1255	Modify the instruction to a ZNWR instruction. J.ZNWR J0 K3 D10 D100 K10 M0
OUT instruction Program example: The number of counter points or the device by which the set value is used is set by parameter. Number of counter points: 512 Setting val. stored dev. start: D3000 OUT C0 K10 OUT C256 D3000	OUT C0 K10 OUT C256 D3000	After conversion, the parameters will be set as defaults, so they must be set again if using an interrupt counter.
RFRP instruction Program example: RFRP H100 K10 W100 K10	OUT SM1255	Modify the instruction to an RFRP instruction for QnACPU. U.RFRP U10 K10 W100 K10 M0
RTOP instruction Program example: RTOP H100 K10 W100 K10	OUT SM1255	Modify the instruction to an RTOP instruction for Q2ASCPU. U.RTOP U10 K10 W100 K10 M0
SCMP instruction Program example: LEDA SCAP LEDC D10 LEDC D100 LEDC M0 LEDR	OUT SM1255 LEDC D10 LEDC D100 LEDC M0	Modify the instruction to an instruction using AND$= and OUT instructions. AND$= D0 D100 OUT M0

An□CPU Instruction	Instruction after A→QnA Conversion	Corrective Action
SEG instruction (When used as a partial refresh instruction) Program example: SET M9052 SEG K4Y10 K4B1	SET SM1052 SEG K4Y10 K4B1	Modify the instruction to an RFS instruction. RFS Y10 H8
STC instruction Program example: STC	SET SM1012	Modify the instruction to SM700, special relay for carry flag. SET SM700
SUB instruction	OUT SM1255	As the Q2ASCPU cannot store any microcomputer program, it has no SUB instructions. Delete OUT SM1255 as it is not necessary. Change the microcomputer program for the AnNCPU or A3HCPU to the sequence program using Q2ASCPU instructions. (Refer to Appendix 4.6)
ZRRD instruction Program example: DMOV K8000 D9036 LEDA ZRRD	DMOV K8000 SD1036 OUT SM1255	Modify the instruction to an MOV instruction. MOV ZR8000 SD718 SD718 is the device resulting from converting accumulator A0.
ZRWR instruction Program example: DMOV K8000 D9036 LEDA ZRWR	DMOV K8000 SD1036 OUT SM1255	Modify the instruction to an MOV instruction. MOV SD718 ZR8000 SD718 is the device resulting from converting accumulator A0.

(a) Instructions for which program modification is unnecessary after conversion

An□CPU Instruction	Instruction after A→QnA Conversion
ASC instruction Program example: ASC ABCDEFGH D10	$MOV ABCDEFGH D10 Note:Since the $MOV instruction has 00H appended at the end, 5 data register words (for 9 characters) must be secured.
DFLOAT instruction Program example: LEDA DFLOAT LEDC D10 LEDC D100 LEDR	DFLT D10 D100
DOUT instruction Program example: LEDA DOUT LEDC Y10 LEDR	OUT DY10
DRCL instruction Program example: DRCL K8	DRCL SD718 K8 SD718 is the device resulting from converting accumulator A0.
DRCR instruction Program example: DRCR K8	DRCR SD718 K8 SD718 is the device resulting from converting accumulator A0.
DROL instruction Program example: DROL K8	DROL SD718 K8 SD718 is the device resulting from converting accumulator A0.
DROR instruction Program example: DROR K8	DROR SD718 K8 SD718 is the device resulting from converting accumulator A0.
DRST instruction Program example: LEDA DRST LEDC Y10 LEDR	RST DY10

An□CPU Instruction	Instruction after A→QnA Conversion
DSUM instruction Program example: DSUM D10	DSUM D10 SD718 SD718 is the device resulting from converting accumulator A0.
DSET instruction Program example: LEDA DSET LEDC Y10 LEDR	SET DY10
FLOAT instruction Program example: LEDA FLOAT LEDC D10 LEDC D100 LEDR	FLT D10 D100
OUT instruction Program example: Set head numbers with parameters. Low speed : 0 High speed: 200 Retentive : 224 Extension timer Low speed : 256 High speed: 512 Retentive : 768 Setting val. stored dev. start: D5000 OUT T0 K10 OUT T200 K10 OUT T225 K10 OUT T256 D5000 OUT T512 D5256 OUT T768 D5512	OUT T0 K10 OUTH T200 K10 OUT ST225 K10 OUT T256 D5000 OUTH T512 D5256 OUT ST768 D5512
RCL instruction Program example: RCL K8	RCL SD718 K8 SD718 is the device resulting from converting accumulator A0.
RCR instruction Program example: RCR K8	RCR SD718 K8 SD718 is the device resulting from converting accumulator A0.
ROL instruction Program example: ROL K8	ROL SD718 K8 SD718 is the device resulting from converting accumulator A0.

An□CPU Instruction	Instruction after A→QnA Conversion
ROR instruction Program example: ROR K8	ROR SD718 K8 SD718 is the device resulting from converting accumulator A0.
SADD instruction Program example: LEDA SADD LEDC D10 LEDC D100 LEDC D200 LEDR	$+ D10 D100 D200
SER instruction Program example: SER D10 D100 K5	SER D10 D100 SD718 K5 SD718 is the device resulting from converting accumulator A0.
SMOV instruction Program example: LEDA SMOV LEDC D10 LEDC D100 LEDR	$MOV D10 D100
SUM instruction Program example: SUM D10	SUM D10 SD718 SD718 is the device resulting from converting accumulator A0.
ZRRDB instruction Program example: DMOV K8000 D9036 LEDA ZRRDB	DMOV K8000 SD1036 ZRRDB SD1036 SD718 SD1036 is the device resulting from converting the special register D9036. SD718 is the device resulting from converting accumulator A0.
ZRWRB instruction Program example: DMOV K8000 D9036 LEDA ZRWRB	DMOV K8000 SD1036 ZRWRB SD1036 SD718 SD1036 is the device resulting from converting the special register D9036. SD718 is the device resulting from converting accumulator A0.

An□CPU Instruction	Instruction after A→QnA Conversion
AnA/AnUCPU dedicated instruction LEDA/LEDB instruction name SUB/LEDC device 1 ... ... SUB/LEDC device n LEDR	Instruction name device 1 .... device n
Program example 1: SIN instruction LEDA SIN LEDC D10 LEDC D100 LEDR	SIN D10 D100
Program example 2: DSER instruction LEDA DSER LEDC D10 LEDC D100 SUB K5 LEDR	DSER D10 D100 SD718 K5 SD718 is the device resulting from converting accumulator A0.
AnA/AnUCPU special function module dedicated instruction LEDA/LEDB instruction name SUB/LEDC device 1 ... ... SUB/LEDC device n LEDR	Enter "G." before the instruction. G. instruction name device Un .... device n
Program example: LEDA SVWR1 SUB H2 LEDC D10 LEDR	G.SVWR1 U2 D10
AnA/AnUCPU data link dedicated instruction LEDA/LEDB instruction name SUB/LEDC device 1 ... ... SUB/LEDC device n LEDR	Enter "J." before the instruction. J. instruction name J0 device 1 .... device n ↑ Network for using MELSECNET II
Program example: LEDA LRDP SUB K12 LEDC D10 LEDC D100 SUB K5 LEDC M0 LEDR	OUT SM1255 J.ZNRD J0 K12 D10 D100 K5M0

An□CPU Instruction	Instruction after A→QnA Conversion
Index register Z, Z1 to Z6, V, V1 to V6 Index register double word $\underset{\text{Upper}}{\underline{Vn}} \cdot \underset{\text{Lower}}{\underline{Zn}}$	Z→Z0 Z1 to Z6→Z1 to Z6 V→Z7 V1 to V6→Z8 to Z13 $\underset{\text{Upper}}{\underline{Zn+1}} \cdot \underset{\text{Lower}}{\underline{Zn}}$ If an index register is used for destination of double word operation or single word multiplication/division, the relation of upper and lower levels may be broken, causing a problem.

Appendix 4.2 Device

(a) Only devices within the Q2ASCPU range are converted.

An□CPU Device		Device after A→QnA Conversion
X□□□		Same as to left
Y□□□		Same as to left
M□□□	M/L/S is determined by the parameter settings.	Same as to left
L□□□		Same as to left
S□□□		Same as to left (Correct to M□□□.)
M9000 to M9255		SM1000 to SM1255
B□□□		Same as to left
T (low-speed timer)	Low-speed/high-speed/ retentive is determined by parameter setting.	Same as to left
T (high-speed timer)		Same as to left
T (retentive timer)		ST□□□
C□□□		Same as to left
F□□□		Same as to left
D□□□		Same as to left
D9000 to D9255		SD1000 to SD1255
W□□□		Same as to left
R□□□		Same as to left
Z□		Z →Z0 Z1 to Z6→Z1 to Z6
V□		V →Z7 V1 to V6→Z8 to Z13
A0,A1		SD718,SD719
P□□□		Same as to left*
I□□		Same as to left
N□		Same as to left
K□□□		Same as to left
H□□□		Same as to left

REMARK

* When P254 is used as the CHK instruction pointer, P254 can be converted to P254 as is. (Refer to Appendix 4.12)

(b) Devices that are outside the Q2ASCPU range are converted to SM1255 if they are bit devices and to SD1255 if they are word devices.

Appendix 4.3 Parameters

The following parameter settings only are converted to Q2ASCPU use.

- Latch range setting
  Converted to the "latch clear key valid" range.
  The latch clear key invalid range is made blank (no setting).
- MELSECNET (II, /10) setting
  For the MELSECNET setting when the ACPU is an AnN or AnA, the number of modules are stored after conversion, but the network refresh parameters are not converted.
- I/O assignment
  Only the head I/O No. is made blank; all other items are converted.
- MELSECNET/MINI auto refresh setting
  If only I/O assignment was set in the parameters and MELSECNET/MINI auto refresh settings have not been made, the MELSECNET/MINI data link operates with the default values.

The following items are set for the Q2ASCPU default. If settings have been made, make the settings again.

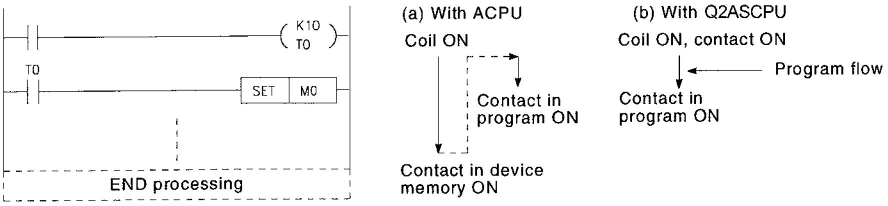
- RUN-PAUSE contacts :No setting
- Output at STOP → RUN :Before operation
- Interrupt counter No. :No setting
- WDT setting :200ms
- Operation mode when there is an error :STOP (All items)

Appendix 4.4 Timer and Interrupt Counter Operations

(1) Timer

(a) The ACPU turns timer coils ON/OFF on execution of the OUT instruction, and updates timer current values and turns contacts ON/OFF on execution of the END instruction. In contrast, the Q2ASCPU turns timer coils ON/OFF, updates current values, and turns contacts ON/OFF on execution of the OUT instruction. Note that after conversion, the turning of contacts ON/OFF may be up to one scan faster.

Example: Timing for turning contact ON



In the case of ACPU, a timer contact will turn ON quickly if it is located in the first step.In the case of Q2ASCPU, it will turn ON quickly if it is located in the step following OUT T.

(b) Note that processing differs as follows when the set value of a timer is set to K0:

- For ACPU, count is in infinite units (timer does not count up).
- For Q2ASCPU, the timer counts up instantaneously.

(2) Interrupt counter

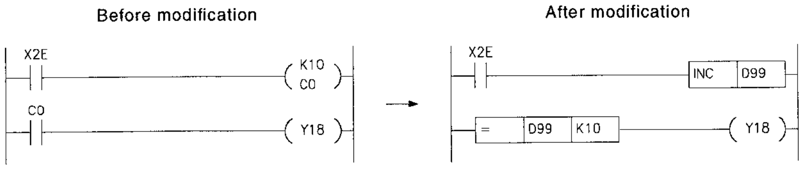
Interrupt counters for Q2ASCPU count the number of interrupt occurrences.However, the counter contact does not turn ON even when the count has reached the set value.

The operation of interrupt counters for ACPU differs according to the CPU type.

(a) Interrupt counters for A3HCPU, AnACPU, or AnUCPU count the number of interrupts occurrences.When the count reaches the set value, the counter contact turns ON.

In order to achieve the same operation as with interrupt counters for A3HCPU, AnACPU, and AnUCPU when using a Q2ASCPU, the program must be modified after conversion.

An example modification is shown below.



(b) Interrupt counters for AnCPU and AnNCPU operate as counters used in interrupt programs.

To achieve the same operation as with interrupt counters for AnCPU or AnNCPU when using a Q2ASCPU, the program modification is not needed after conversion.

When ordinary counters are used in an interrupt program with Q2ASCPU, they operate in the same way as with AnNCPU.

Appendix 4.5 Sequence Programs, Statements, Notes

After conversion by A → QnA conversion, sequence programs are stored in the set file. If a subsequence program is included, the main/subsequence program must be modified. There are two types of modification, as indicated below:

(a) When executing the main sequence program and subsequence program alternately, modify the parameters and programs as follows.

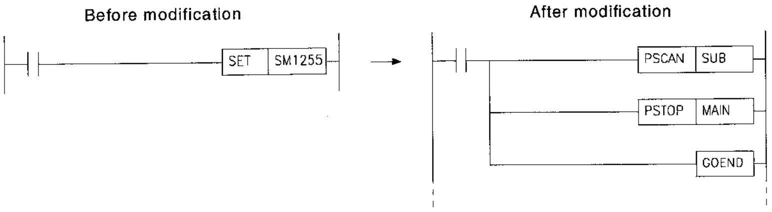
1) Modification of parameters

Set file names of the main sequence program and subsequence program in program setting in "Auxiliary setting" in the parameter mode.Select scan execution for the main sequence program and standby execution for the subsequence program.

2) Modification of the sequence program

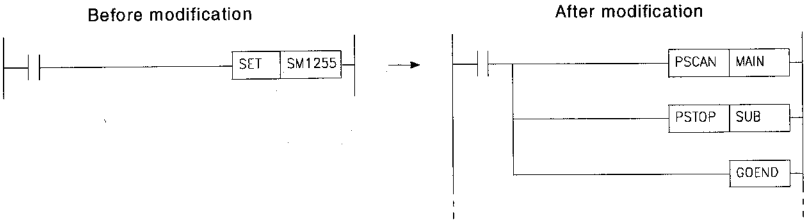
- The CHG instruction that switches between main sequence and subsequence programs is converted to OUT SM1255 after A → QnA conversion. Modify this OUT SM1255 to the PSCAN instruction which converts another sequence program to an scan execution type program.
- Next, add the GOEND instruction that executes a jump to the END instruction to the following step.
- Next, add the PSTOP instruction, which converts another sequence program to a standby execution type program, to the first step of the sequence program.

  This enables execution of the subsequence program from the main sequence program, and disables execution of the main sequence program when the subsequence program is executed.
- Main sequence program



* "SUB" is the subsequence program file name.
* "MAIN" is the main sequence program file name.

- Subsequence program



* "MAIN" is the main sequence program file name.
* "SUB" is the subsequence program file name.

(b) To execute the main sequence program and subsequence program serially as one program, modify the parameters and program as follows.

1) Modification of parameters
   Set the file names in the order of main sequence program and subsequence program in program setting in "Auxiliary setting" in the parameter mode. Select scan execution as the execution type for both the main sequence program and the subsequence program.
2) Modification of the sequence program
   - The CHG instruction that switches between main sequence and subsequence programs is converted to OUT SM1255 after A→QnA conversion.Delete it as it is not required for Q2ASCPU.
   - If the same interrupt program or pointer is used for the main sequence program or subsequence program, use only one interrupt program or pointer.

REMARK

AnACPU executes END processing on switching from execution of the main sequence program to execution of the subsequence program, and also executes END processing after execution of the subsequence program.
Note that END processing is executed only after execution of the second program when a Q2ASCPU executes two programs consecutively.

Statements and notes are entered in the sequence program file after A→QnA conversion.
No modification is required after conversion.

Appendix 4.6 Microcomputer programs

Microcomputer programs and utility software packages cannot be converted as the Q2ASCPU has no microcomputer mode.
When a microcomputer program or utility software package is used with the ACPU, a SUB instruction (microcomputer program call instruction) is written in the sequence program to execute it.The SUB instruction is converted to OUT SM1255 after A→QnA conversion; delete it as it is not necessary.
In the case of user-created microcomputer programs, convert processing contents of the microcomputer programs to sequence programs using operation instructions added for Q2ASCPU.
When using a utility software package of the following, convert processing contents of the utility software package to a sequence program using operation instructions added for Q2ASCPU.

• SW□□□-AD57P ・・・・・・	Refer to the QnACPU Programming Manual (AD57 Instructions).
• SW□□□-UTLP-FN0 ・・・・	Refer to the QCPU (Q mode)/QnACPU Programming Manual (Common Instructions).
• SW□□□-UTLP-FN1 ・・・・	Refer to the QCPU (Q mode)/QnACPU Programming Manual (Common Instructions).
• SW□□□-UTLP-PID ・・・・	Refer to the QCPU (Q mode)/QnACPU Programming Manual (PID Control Instructions).
• SW□□□-SIMA • SW□□□-UTLP-FD1 • SW□□□-SAPA	Unusable

Appendix 4.7 Comments

Conversions are made for the device range of Q2ASCPU. Devices outside the range are not converted.

Appendix 4.8 Constant Scan Function, Error Check Function

When using the constant scan function or error check function for ACPUs, special registers or special relays are set.
In contrast, for Q2ASCPUs, these functions are set with parameters. To use these functions after conversion, make settings in "PLC RAS" in the parameter mode.

Appendix 4.9 I/O control mode

The I/O control mode for Q2ASCPU is refresh mode (direct I/O is enabled depending on the device).

- As the I/O control mode for A2US is the refresh mode as with Q2ASCPU, there are no problems with the input timing of inputs (X) or the output timing of outputs (Y).
- In the case of A1SCPU, and A2SCPU, I/O control mode is fixed or selected to direct mode, and the input timing for inputs (X) and output timing for outputs (Y) differs from that for the refresh mode.
- Modifying programs that generate pulses from SET/RST instructions by using direct devices
  Modify programs which, in the direct mode, output pulse output to the external using SET/RST instructions to programs that use direct output devices for Q2ASCPU.

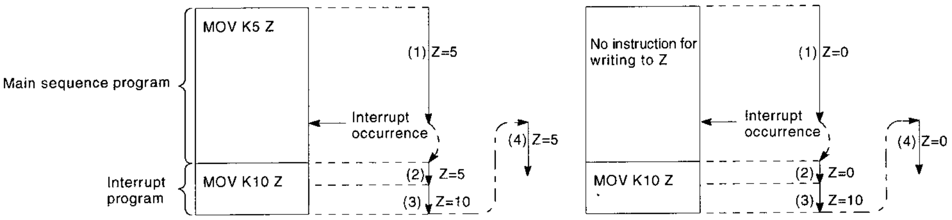
Appendix 4.10 Data Link System

- AnUCPU data link systems
  The network settings in the AnUCPU parameters can be converted by A → QnA conversion. Parameter modifications after conversion are not needed.
- CPU modules other than AnUCPU
  The link settings in the CPU module parameters cannot be converted by A → QnA conversion. Link settings must be made in the parameters after conversion.

Appendix 4.11 Index Register Processing

For Q2ASCPU, the contents of index registers change when program processing transfers between the main sequence program and interrupt programs.

- Transfer of program processing from main sequence program to interrupt program
  The contents of the index registers of the main sequence program are saved, and then these contents are passed to the interrupt program.
- Transfer of program processing from interrupt program to main sequence program
  The index registers in the interrupt program are cleared, and the saved main sequence program contents are written to them.



For ACPU, processing differs according to the CPU module type.

- The processing for A2USCPU is the same as for Q2ASCPU, and no program modification is required after conversion.
- In the case of A1SCPU, and A2SCPU, when program processing is transferred from an interrupt program to the main program, the data updated in the interrupt program are passed on to the main program.
  When passing a value written to the index register in an interrupt program on to the main sequence program, for example, modify the program so that the value is passed on via a data register.

Appendix 4.12 CHK Instruction, IX Instruction

(1) CHK instruction

The CHK instruction operates as a fault check instruction for Q2ASCPU.
For ACPU, there are two types of processing depending on the CPU type.

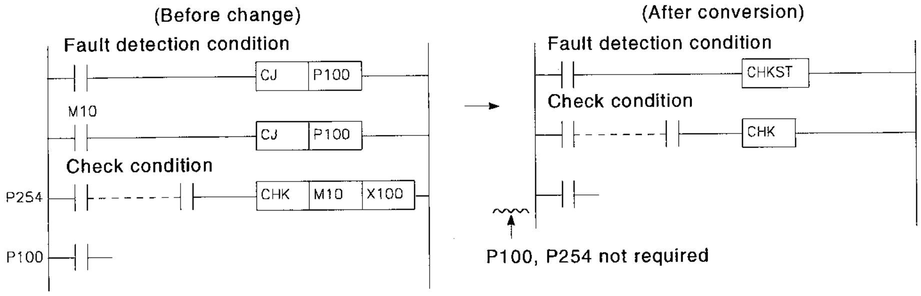
- Fault check............................ AnCPU, AnNCPU (direct I/O control mode), A3HCPU, AnACPU, AnUCPU
- Bit device output inversion..... AnNCPU (refresh I/O control mode)

After conversion, program modification is required for each processing.

[For fault check]

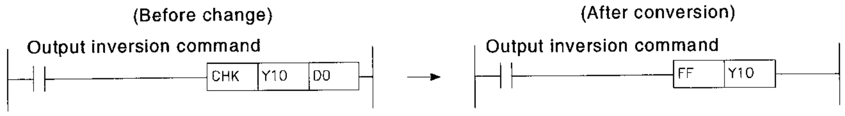
Modify the CJ instruction in the step before the CHK instruction to a CHKST instruction.

The CHK instruction pointer (P254) and the CJ instruction destination pointer are converted to pointers with the same number. As the above pointers are not used for Q2ASCPU, delete them.



[For bit device output inversion]

The Q2ASCPU has the FF instruction for inverting bit device outputs.
Modify the CHK instruction to the FF instruction.



(2) IX instruction

The IX instruction is converted, but not executed. Modify the program so that all the devices that are objects of the IX instruction are subject to indexing.

Appendix 4.13 Accessing File Register R with Instructions

For Q2ASCPU, no error will occur even if an instruction for accessing file registers outside the setting range is executed.
When reading data, FFFFH is stored to the storage device. When writing data, the instruction is executed but no data is stored in the file register.
For ACPU, execution of such an instruction causes an error.

The capacity of file register R is set by parameter. It is therefore necessary to check the capacities of file registers before executing instructions that access file registers, such as the MOV instruction and + instruction.

APPENDIX 5 ERROR CODES RETURNED TO THE REQUEST SOURCE IN GENERAL DATA PROCESSING

With the Q2ASCPU, when an error occurs while general data processing is requested from a peripheral device, a special function module, or a network system, the error code is returned to the source of the general data processing request.

POINT

Since this error code is not an error detected by the Q2ASCPU self-diagnostics function, it is not stored to special relay SD0.
If the request source is a peripheral device, the message or the error code is displayed.
If the request source is a special function module or a network system, the error code corresponding to the requested processing is returned.

Appendix 5.1 Error Codes

The error code's numbers depends on the location where the error has been detected.
The correspondences between the locations where errors are detected and the error codes are indicated in the table below.

Location Where an Error is Detected	Error Code	Reference for Error Contents
CPU module	4000H to 4FFFH	App Appendix 5.2
Serial communication module, etc.	7000H to 7FFFH	Serial Communication Module User's Manual, etc.
CC-Link module	B000H to BFFFH	Control & Communication Link System Master/ Local Module User's Manual
Ethernet module	C000H to CFFFH	Ethernet Interface Module User's Manual
MELSECNET/10 network module	F000H to FFFFH	For QnA/Q4AR MELSECNET/10 Network System Reference Manual

Appendix 5.2 Error Contents of Error Codes Detected by the CPU Module (4000H to 4FFFH)

The error contents of error codes detected by the CPU module (4000H to 4FFFH), and the messages displayed on the peripheral device are indicated in the table below.

Error Code (Hexadecimal)	Error	Error Contents	Message Displayed at Peripheral Device	Corrective Action
4000H	CPU module-related error	Sum check error	Message (1) is displayed.	Check the connection between the CPU module and connection cable.
4001H 4002H		Remote request that cannot be handled is performed.	Message (1) is displayed.	Check the requested remote operation.
4003H		Command to which a global request is not allowed is performed.	Message (1) is displayed.	Check the requested command.
4004H		Since the Q2ASCPUsystem is protected, the request contents cannot be performed.	Execution is not allowed during system protection.	Turn the Q2ASCPU system protect switch OFF.
4005H		The data volume is too large for the specified request.	Cannot execute in excess of capacity.	Reduce the data volume so that it can be handled with the request.
4006H		Password has not been cancelled.	Password has not been cancelled.	Cancel the set password.
4007H		CID is different from the Q2ASCPU data.	Message (1) is displayed.	Check the CID.
4008H		The Q2ASCPU is not BUSY. (Buffer is not empty.)	Message (1) is displayed.	Re-perform the request after the elapse of an arbitrary time period.
4010H	CPU module mode error	The request contents cannot be performed because the Q2ASCPU is in RUN.	Cannot execute when PLC is in RUN mode.	Perform the request after setting the Q2ASCPU to STOP.
4011H		The request contents cannot be performed because the Q2ASCPU is not in STEP-RUN.	Cannot execut while PLC is not in STEP RUN mode.	Perform the request after setting the Q2ASCPU to STEP-RUN.
4012H		The request contents cannot be performed because the Q2ASCPU is in STEP-RUN.	Not executed due to STEP-RUN of PLC.	Perform the request after setting the Q2ASCPU to RUN/STOP.

Error Code (Hexadecimal)	Error	Error Contents	Message Displayed at Peripheral Device	Corrective Action
4021H	CPU module file-related error	Designated drive memory does not exist or is abnormal.	The target dirve contains a fault.	Check the status of the designated drive memory.
4022H		The file with designated file name, and file No. does not exist.	The file name does not exist.	Check the designated file name and file No.
4023H		The file name and file No. of the designated file do not match.	Cannot access files.	Delete the file and create a new one.
4024H		The designated file cannot be accessed by the user.	This file cannot be handled.	Do not access the designated file.
4025H		The designated file is processing a request from another source.	Alert (1) is displayed.	Forcibly perform the request. Or perform the request again after other processing has completed.
4026H		The keyword set for target drive memory has to be designated.	Keyword doesn't match.	Access by designating the keyword set for the target drive memory.
4027H		The designated range exceeds the file range.	File capacity is not enough.	Check the designated range, and access within the permissible range.
4028H		The same file has already existed.	Alert (2) is displayed.	Forcibly perform the request. Or change the file name and then perform the request.
4029H		The capacity of the designated file is not secured.	File capacity is not enough.	Review the capacity of the designated file. Or sort the designated drive memory and re-perform.
402AH		The designated cluster No. does not exist.	Cannot access files.	Check the designated cluster No., and access by designating a cluster No. within the number of clusters of the designated drive memory.
402BH		The request contents cannot be performed with the designated drive memory.	Cannot access files.	Do not make requests which caused an error to the designated drive memory.
402CH		The request contents cannot be currently performed.	Cannot access files.	Re-perform after the elapse of an arbitrary time period.
4030H	CPU module device designation error	The designated device name cannot be handled.	Device is invalid.	Check the designated device name.
4031H		The designated device No. is out of range.	Device No. is out of range.	Check the designated device No.
4032H		A mistake in the designated device qualification.	Device is invalid.	Check the method for qualification of the designated device.
4033H		The designated device is for system use and cannot be written to.	Device is invalid.	Do not write data to the designated device, or turn it ON/OFF.

Error Code (Hexadecimal)	Error	Error Contents	Message Displayed at Peripheral Device	Corrective Action
4040H	Special function module designation error	The designated special function module cannot perform the request contents.	The unit does not exist.	Do not make requests which caused an error to the designated special function module.
4041H		Access range exceeds the buffer memory range of the designated special function module.	The # of devices is too large.	Check the head address and number of accessed points, and access within the actual ranges at the special function module.
4042H		Access to the designated special function module is not possible.	The corresponding unit is faulty.	Check if the designated special function module normally operates.
4043H		The special function module is not at the designated position.	The unit does not exist.	Check the head I/O No. of the designated special function module.
4044H		A control bus error has occurred.	The corresponding unit is faulty.	Check if there is a fault in the hardware of the special function module or other modules.
4045H		Setting required for simulation has not been made.	Data error	Make settings for the simulation.
4046H		The head number of the device or the number of device points designated for simulation is not in 16-point units.	Device No. is not in 16 units.	Check the head number and number of device points and then modify them to 16-point units.
4050H	Protect error	Request contents cannot be performed because the write protect switch of the memory card is ON.	Cannot execute as the memory protect switch is ON.	Turn the write protect switch of the memory card OFF.
4051H		The designated device memory cannot be accessed.	Wrong ROM	Check the following and take corrective action. • Whether the memory is usable • Whether the designated drive memory correctly installed
4052H		Data cannot be written to the designated file because its attribute is read only.	Write is prohibited.	Do not write data to the designated file. Or change the file attribute.
4053H		An error occurred when writing data to the designated drive memory.	Cannot write correctly in ROM.	Check the designated drive memory. Or replace the target drive memory and then rewrite the data.
4054H		An error occurred when deleting data from the designated drive memory.	Cannot erase ROM correctly.	Check the designated drive memory. Or replace the target drive memory and then delete the data again.

Error Code (Hexadecimal)	Error	Error Contents	Message Displayed at Peripheral Device	Corrective Action
4060H	Online registration error	The CPU module system area for registering monitor conditions is being used by another device.	Alert (1) is displayed.	When monitoring of the other device has completed, perform the monitoring again. Or increase the system area of the buil-in RAM using a format with an option.
4061H		Communications failed.	Not registered.	Re-perform communications.
4062H		Another device is monitoring using the detailed condition for monitoring.	Alert (1) is displayed.	Do not use the detailed condition for monitoring from the designated device. Or cancel the monitoring detailed condition for other device and perform the monitoring again.
4063H		The number of registrations for file lock is greater than 16.	Cannot access files.	Reduce the number of registrations to 16 or less.
4064H		Incorrect setting contents.	Setting is incorrect.	Check the set contents.
4065H		Device I/O information differs from parameters.	Does not match the parameter.	Check the parameters. Or check the data.
4066H		A keyword that differs from the one set for the designated drive memory was specified.	Keyword doesn't match.	Check the designated keyword.
4067H		The designated monitor file has not been secured.	File capacity is not enough.	Secure the monitor file, then perform monitoring.
4068H		The designated command cannot be registered or cancelled since it is in execution.	Unable to execute due to on going process.	Re-perform the command after requests from other devices has been completed.
4069H		Condition has already satisfied at device.	Setting is incorrect.	Check the monitor condition. Or perform monitor registration again and then monitor.
406AH		Drive other than No.1 to 3 has been designated.	Drive specification is incorrect.	Check the designated drive and specifies a correct drive.
4070H	Ladder verification	The program before modification differs from the registered program.	Program does not match.	Check the registered program and match the program to it.

Error Code (Hexadecimal)	Error	Error Contents	Message Displayed at Peripheral Device	Corrective Action
4080H	Other error	Data error	Data is faulty.	Check the requested data contents.
4081H		The searched target cannot be detected.	Cannot find the find target.	Check the data to be searched.
4082H		The designated command cannot be performed since it is in execution.	Unable to execute due to on going process.	Re-perform the command after requests from other devices has been completed.
4083H		An attempt was made to perform a program not registered in the parameters.	Not registered.	Register the program to be performed to the parameters.
4084H		The designated pointer P, I cannot be detected.	Cannot find the find target.	Check the data to be searched.
4085H		Pointer P, I designation is not possible because the program is not registered in the parameters.	Not registered.	Register the program to be performed in the parameters, then designate the pointer P, I.
4086H		An attempt was made to add a pointer P, I that have already existed.	Device ranges are duplicated.	Check the pointer No. to be added and change it.
4087H		The number of pointers designated is too great.	No pointer exists.	Check and correct the designated pointer.
4088H		The designated step No. is not at the head of the instruction.	Execution position is incorrect.	Check and correct the designated step No.
4089H		The END instruction was inserted/ deleted while the CPU module had been in RUN.	Setting is incorrect.	Insert/Delete the instruction after setting the CPU module to STOP.
408AH		The file capacity has been exceeded by performing write during RUN.	File capacity is not enough.	Set the CPU module to STOP and then write the program.
408BH		Cannot perform a remote request.	Data error.	Set the CPU module to the state for performing a remote request, then reissue the request.

Error Code (Hexadecimal)	Error	Error Contents	Message Displayed at Peripheral Device	Corrective Action
4090H	Online registration error during SFC STEP RUN	Too many block break points.	Setting is out of range.	Check and correct the set number.
4091H		The number of registered block break points is incorrect.	Setting is out of range.	Check and correct the set number.
4092H		Too many step break points.	Setting is out of range.	Check and correct the set number.
4093H		The number of registered step break points is incorrect.	Setting is out of range.	Check and correct the set number.
4094H		An attempt was made to perform a request during block continuous processing.	Unable to execute due to on going process.	Reissue the request after the processing has been completed.
4095H		An attempt was made to perform a request during block forced execution processing.	Unable to execute due to on going process.	Reissue the request after the processing has been completed.
4096H		An attempt was made to perform a request during step continuous processing.	Unable to execute due to on going process.	Reissue the request after the processing has been completed.
4097H		An attempt was made to perform a request during step forced execution processing.	Unable to execute due to on going process.	Reissue the request after the processing has been completed.
4098H		An attempt was made to perform a request during one step continuous processing.	Unable to execute due to on going process.	Reissue the request after the processing has been completed.
4099H		An attempt was made to perform a request during one step forced execution processing.	Unable to execute due to on going process.	Reissue the request after the processing has been completed.
409AH		An attempt was made to perform a request during block forced end processing.	Unable to execute due to on going process.	Reissue the request after the processing has been completed.
409BH		An attempt was made to perform a request during step forced end processing.	Unable to execute due to on going process.	Reissue the request after the processing has been completed.
409CH		An attempt was made to perform a request during holding step reset processing.	Unable to execute due to on going process.	Reissue the request after the processing has been completed.
409DH		A block No. with no created block or out-of-range block No. has been designated.	Setting is incorrect.	Check and correct the set contents.
409EH		A step No. for which no step has been created was designated.	Setting is incorrect.	Check and correct the set contents.
409FH		The designated number of cycles is out of range.	Setting is out of range.	Check and correct the set number.

Error Code (Hexadecimal)	Error	Error Contents	Message Displayed at Peripheral Device	Corrective Action
40A0H	SFC device designation error	Out-of-range block No. is designated.	Setting is incorrect.	Check and correct the set contents.
40A1H		Designation exceeds the range for the number of blocks.	Setting is out of range.	Check and correct the set number.
40A2H		Out-of-range step No. is designated.	Setting is incorrect.	Check and correct the set contents.
40A3H		Designation exceeds the range for number of steps.	Setting is out of range.	Check and correct the set number.
40A4H		Out-of-range sequence step No. is designated.	Setting is incorrect.	Check and correct the set contents.
40A5H		The designated device is out of range.	Setting is out of range.	Check and correct the set number.
40A6H		The block designation pattern or step designation pattern was incorrect.	Setting is incorrect.	Check and correct the set contents.
40B0H	SFC file-related error	The designated drive is incorrect.	Setting is incorrect.	Check and correct the set contents.
40B1H		The designated program does not exist.	The file name does not exist.	Check and correct the designated file name.
40B2H		The designated program was not an SFC program.	This file cannot be handled.	Check and correct the designated file name.
40B3H		The SFC dedicated instruction exists in the write during RUN area.	Setting is incorrect.	Check and correct the set contents.
4A00H	Link-related error	The designated station cannot be accessed because no routing parameters have been set to the relevant station.	Routing parameter does not exist.	Set the routing parameters for accessing to the designated station in the relevant station.
4A01H		No network with the No. set in the routing parameters exists.	The network I/O does not exist.	Check and correct the routing parameters set at the relevant station.
4A02H		Cannot access to the designated station.	Link unit error.	Check if an error has occurred at the network module/link module, or if the online state has not been established.
4B00H	Target-related error	An error occurred at the access target station or the relay station.	The corresponding unit is faulty.	Check and correct the error at the designated access target station or the relay station for the access station.

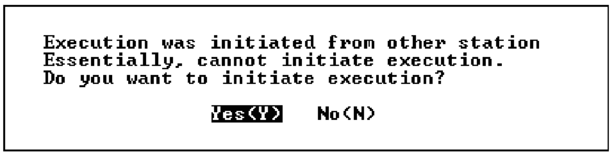
REMARK

(1) Message (1)

```
Cannot communicate with PC. Error ## = ****
```

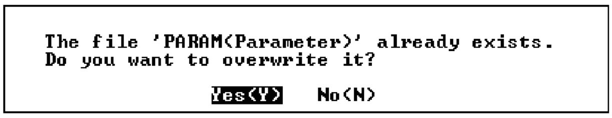
An error code is displayed in ****.

(2) Alert (1)



```
Execution was initiated from other station
Essentially, cannot initiate execution.
Do you want to initiate execution?
```

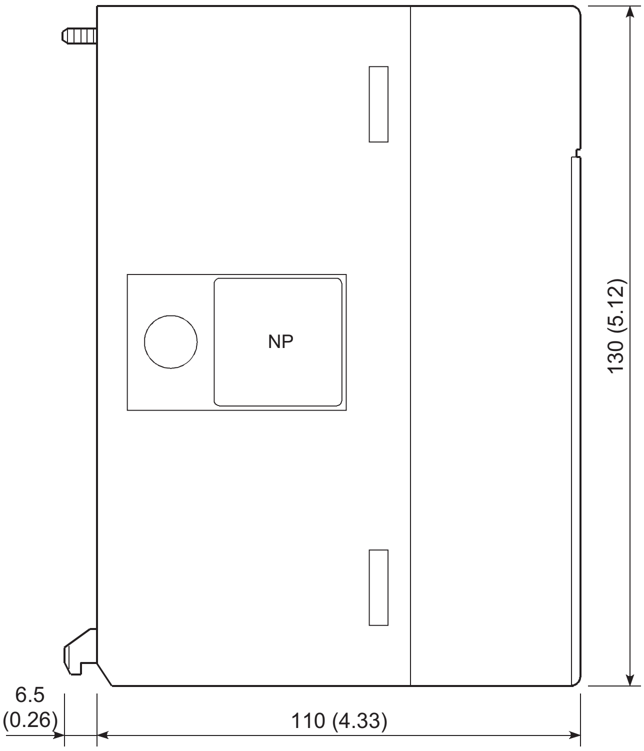
(3) Alert (2)

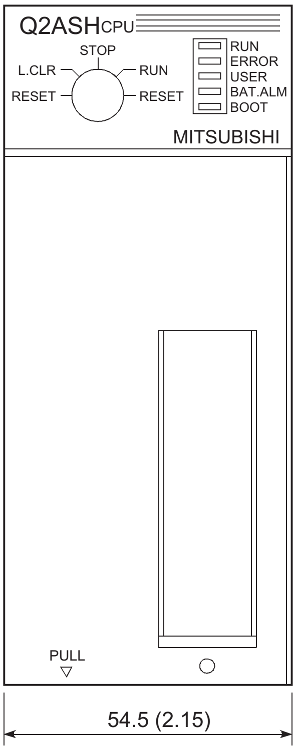


```
The file 'PARAM(Parameter)' already exists.
Do you want to overwrite it?
```

APPENDIX 6 EXTERNAL DIMENSIONS

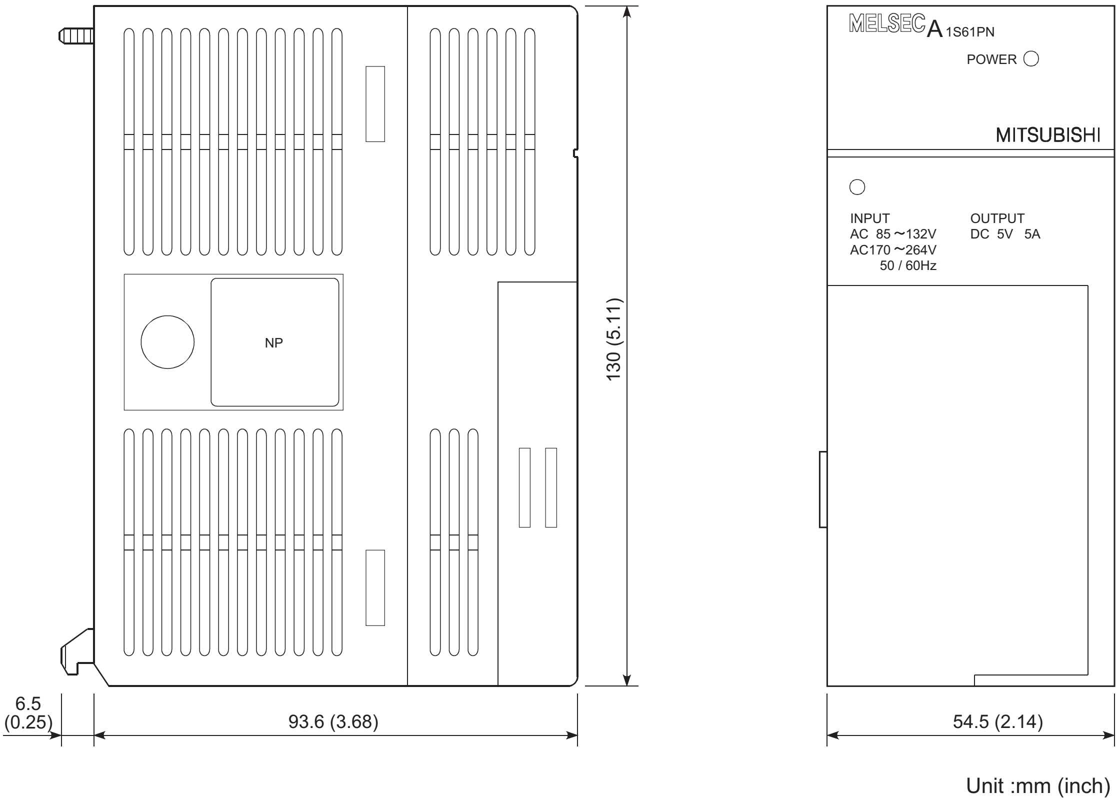
Appendix 6.1 Q2AS(H)CPU(S1) module





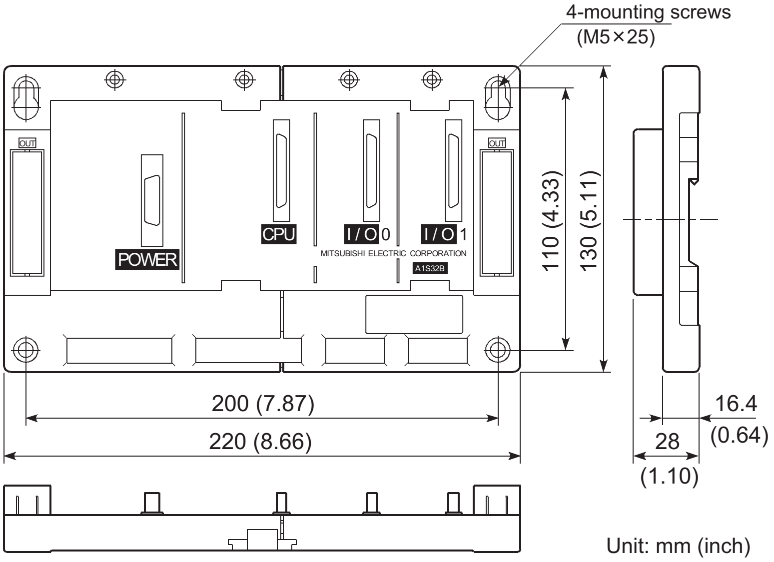
Unit : mm (inch)

Appendix 6.2 A1S61PN, A1S62PN and A1S63P power supply modules



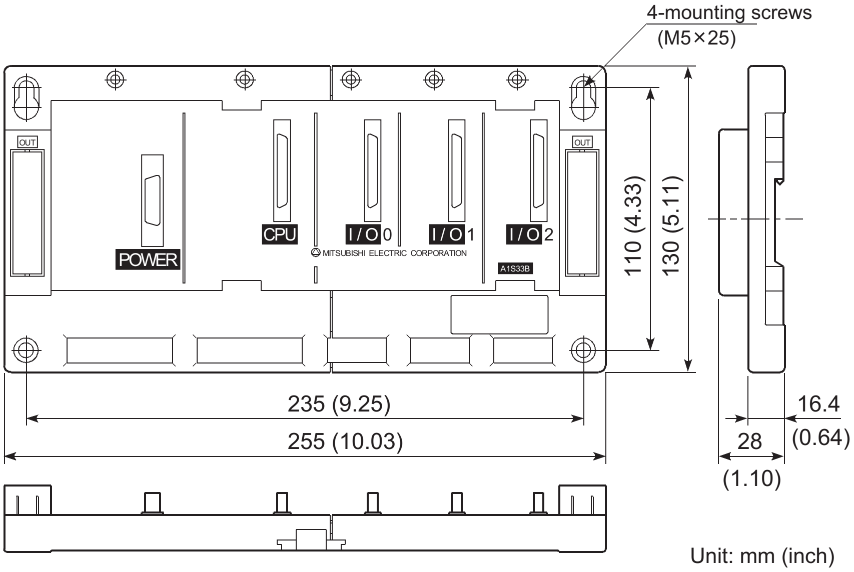
Appendix 6.3 Main Base Unit

(1) A1S32B main base unit



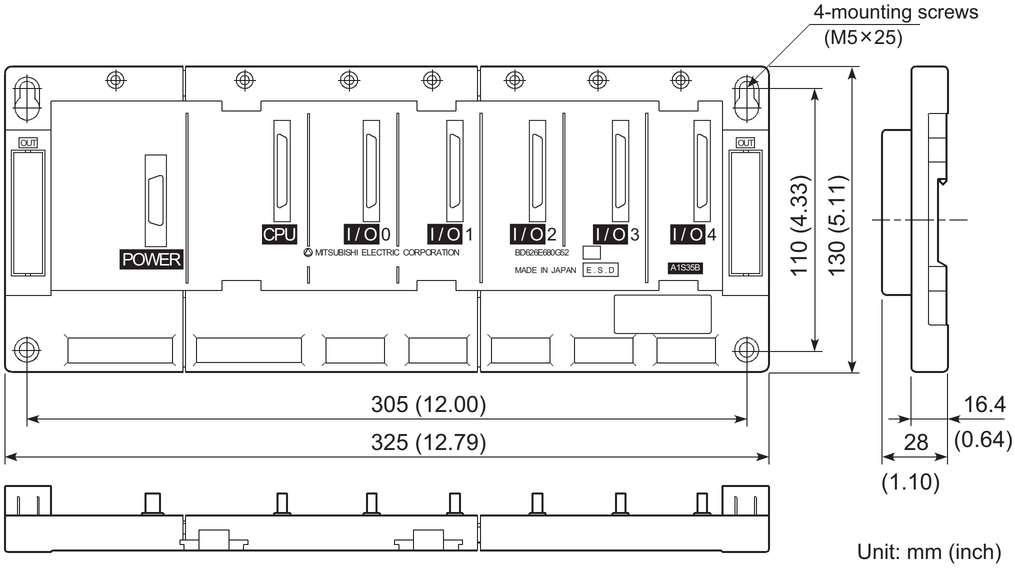
Unit: mm (inch)

(2) A1S33B main base unit

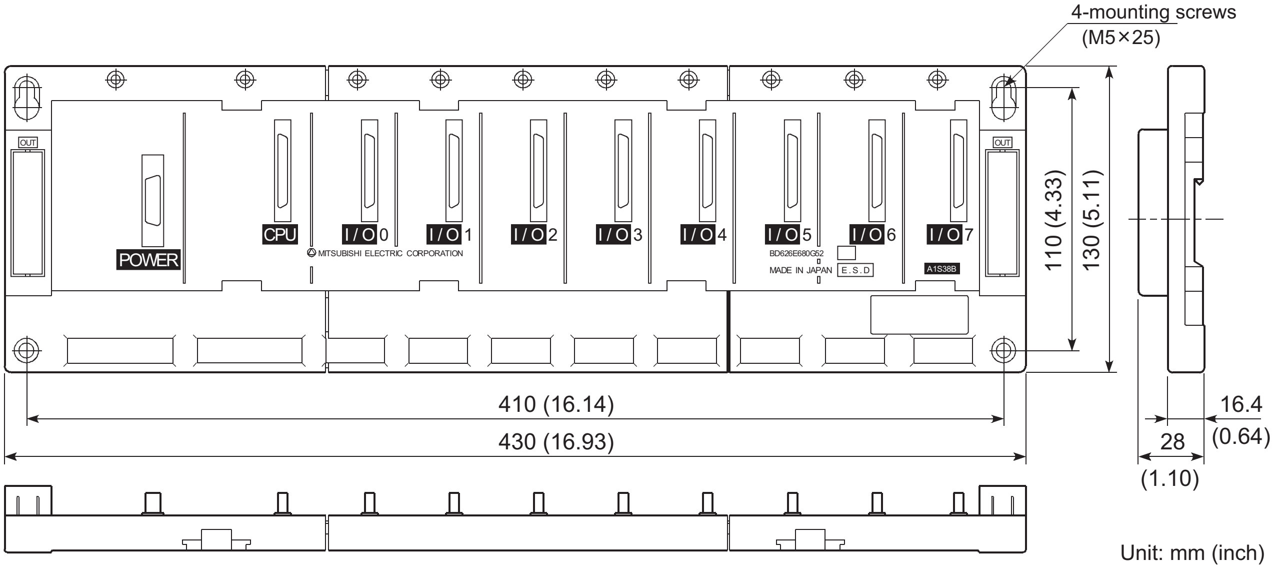


Unit: mm (inch)

(3) A1S35B main base unit

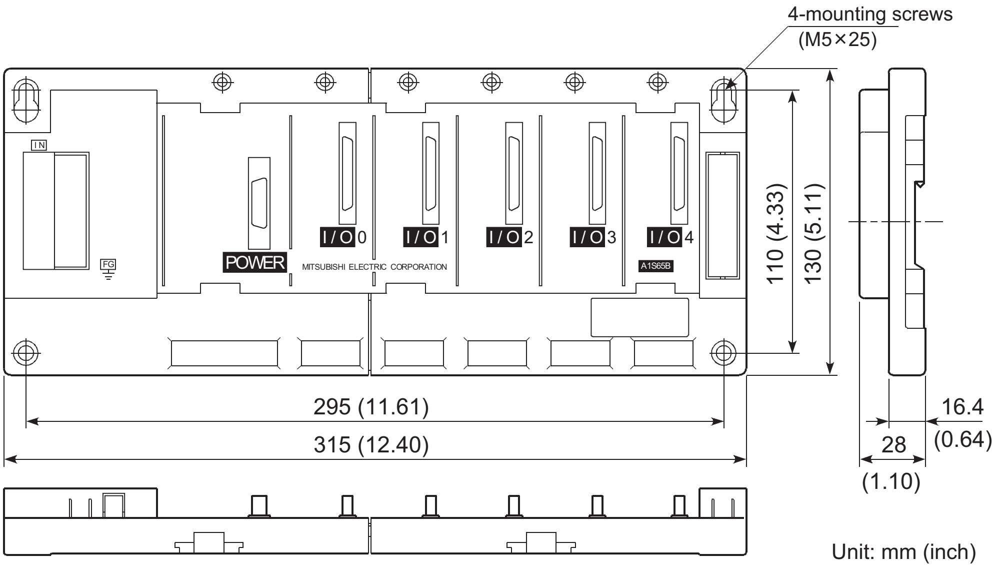


(4) A1S38B, A1S38HB main base units

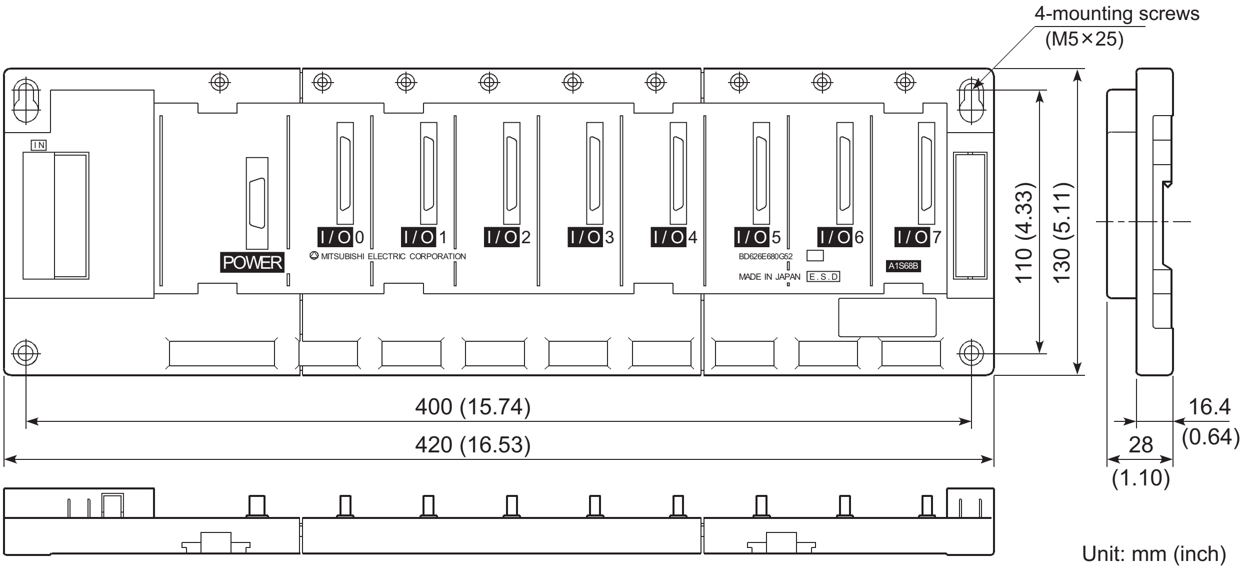


Appendix 6.4 Extension Base Unit

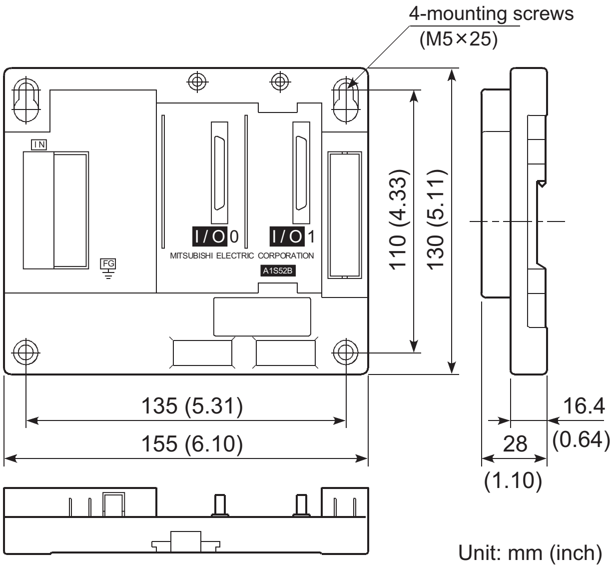
(1) A1S65B extension base unit



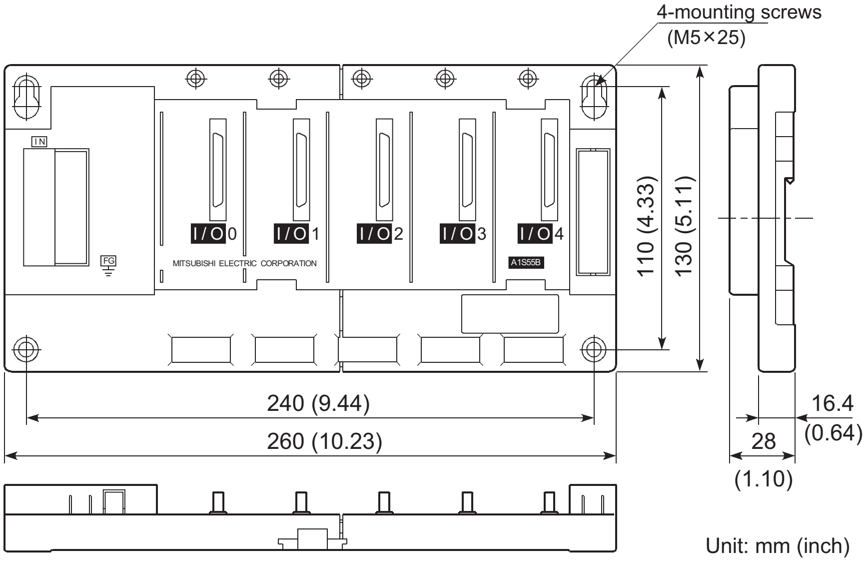
(2) A1S68B extension base unit



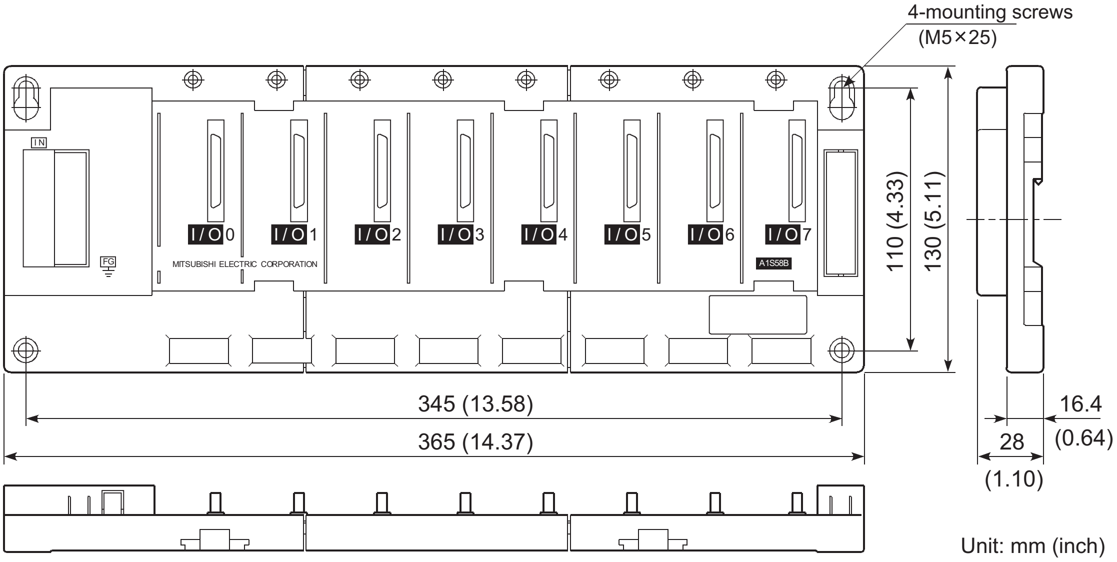
(3) A1S52B extension base unit



(4) A1S55B extension base unit

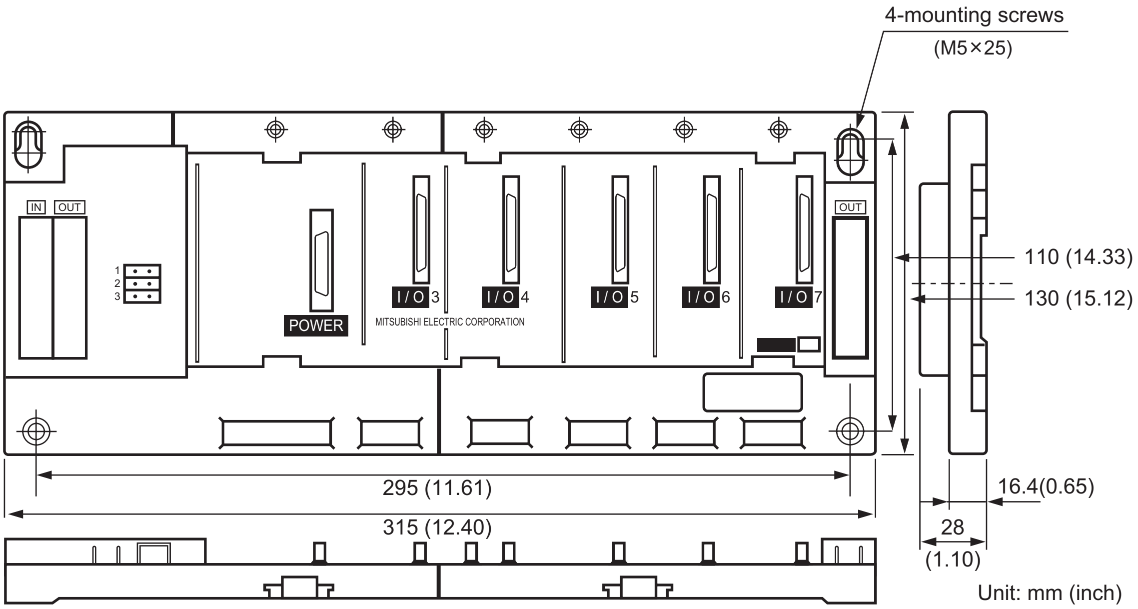


(5) A1S58B extension base unit

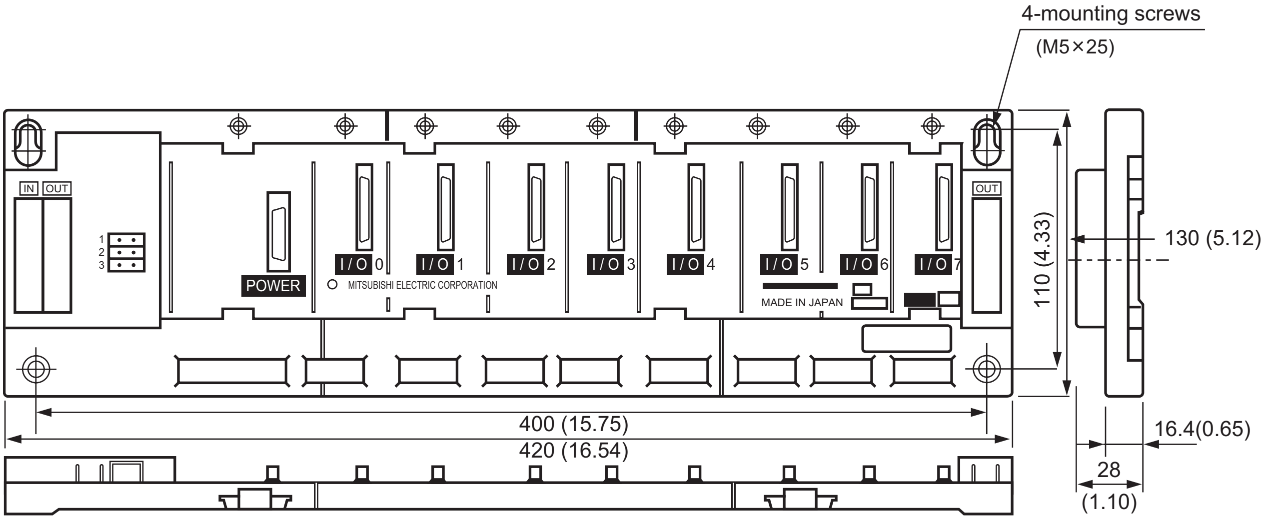


Unit: mm (inch)

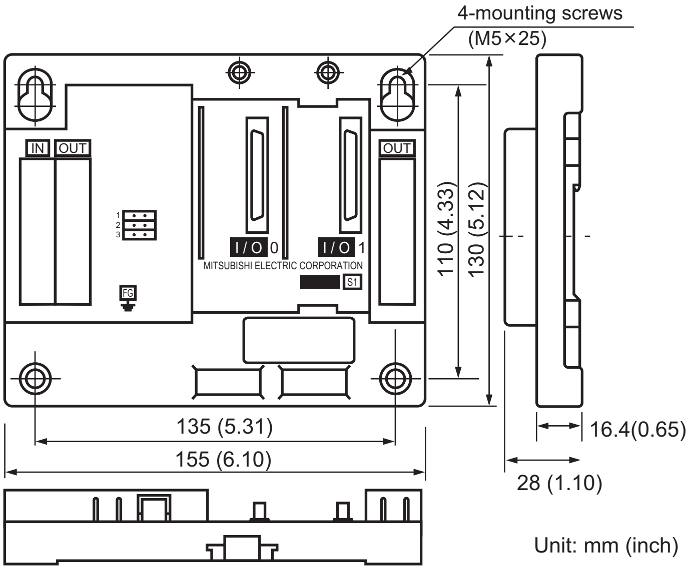
(6) A1S65B-S1 extension base unit



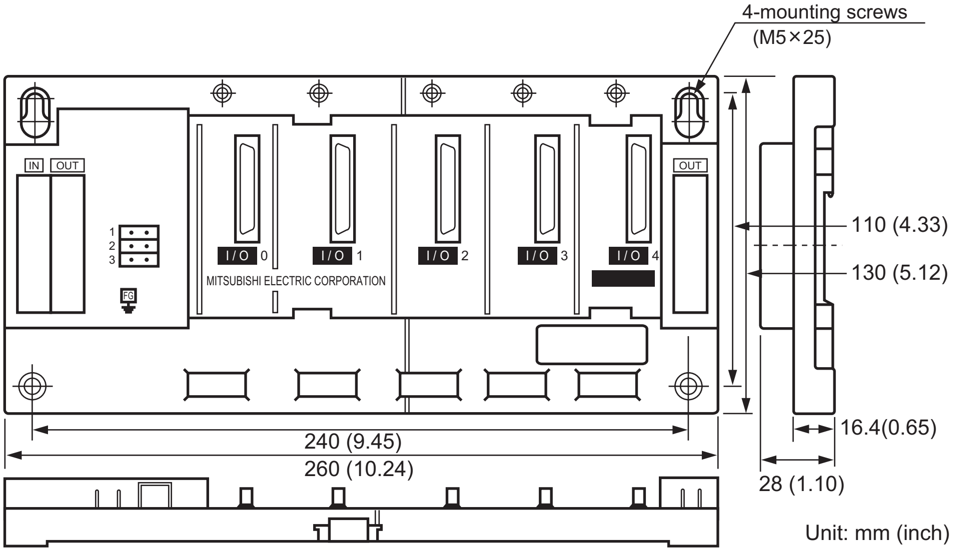
(7) A1S68B-S1 extension base unit



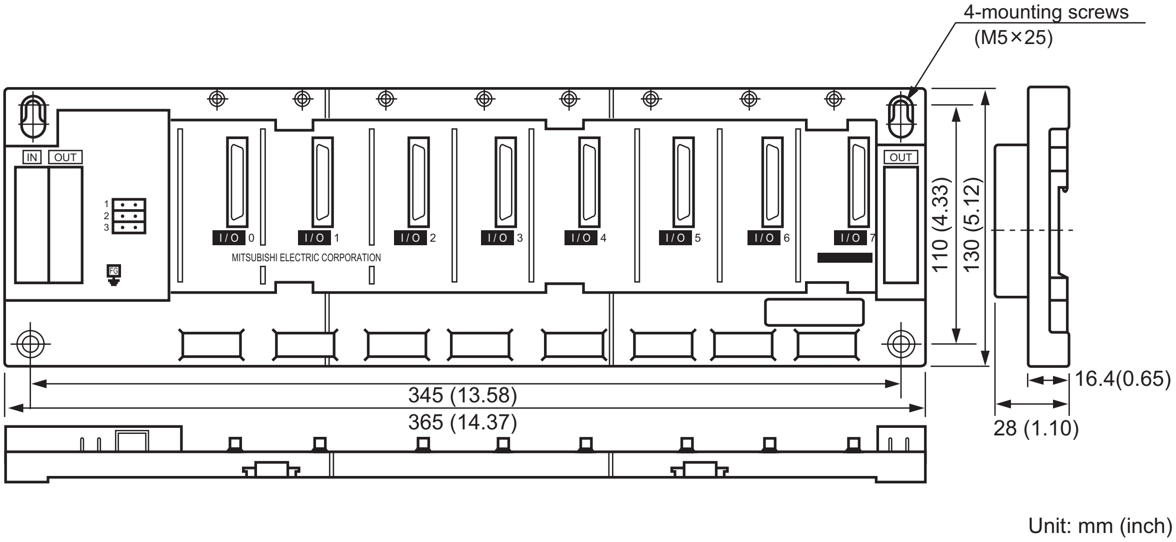
(8) A1S52B-S1 extension base unit



(9) A1S55B-S1 extension base unit



(10) A1S58B-S1 extension base unit



APPENDIX 7 USE OF LOCAL DEVICE FOR SUBROUTINE/INTERRUPT PROGRAM STORAGE FILE (FUNCTION VERSION B OR LATER)

When the subroutine/interrupt program is executed, the local device for the subroutine/ interrupt program storage files can be used.
To use the local device in the storage destination file for the subroutine/interrupt program, set the special relaies below:

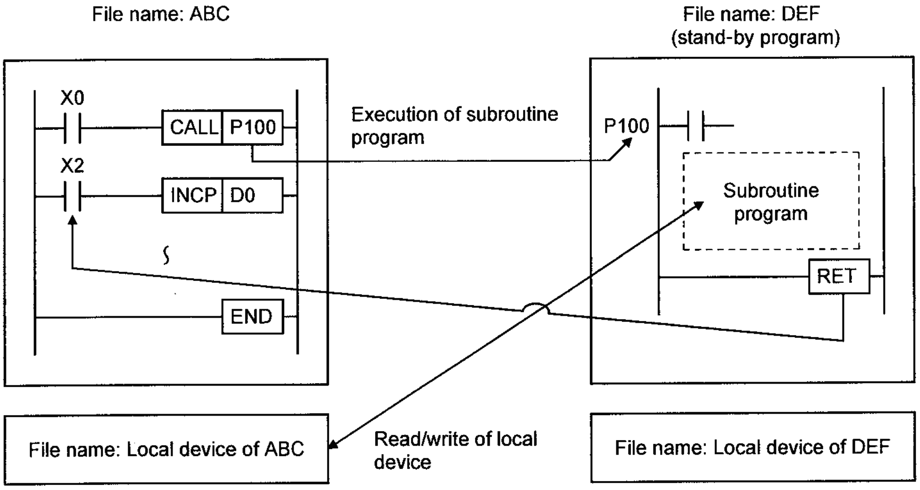
- Subroutine program : SM776
- Interrupt program : SM777

(1) Switching of local device with special relay ON/OFF

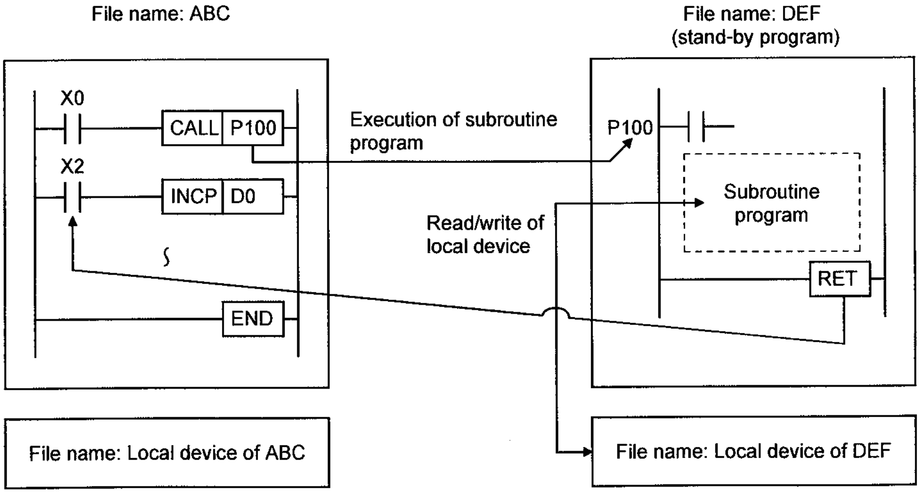
	SM776	SM777
OFF	Operation is performed at the local device of the call source file of the subroutine program.	Operation is performed at the local device of the file executed before execution of the interrupt program.
ON	Operation is performed at the local device of the file where the subroutine program is stored.	Operation is performed at the local device of the file where the interrupt program is stored.

(a) Operation for subroutine program

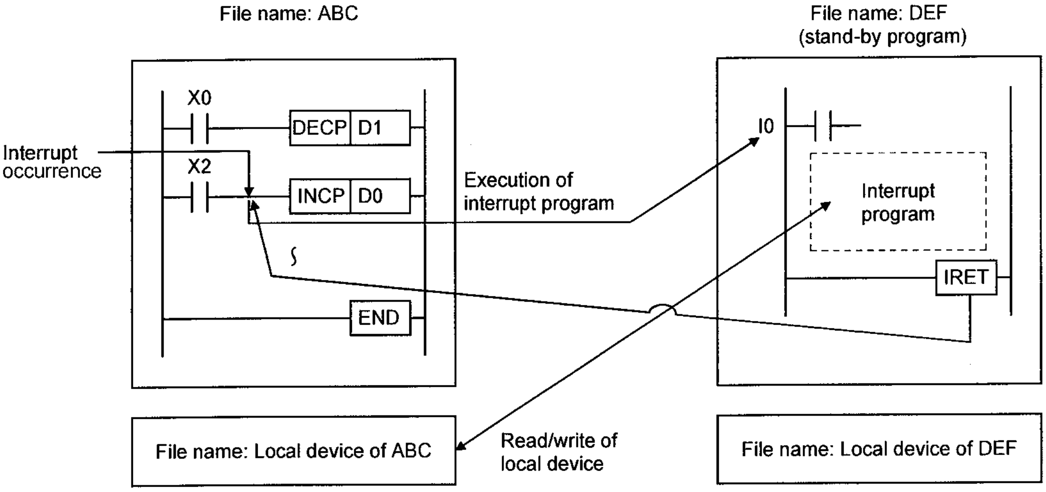
[SM776 operation: OFF without function version B or with function version B]



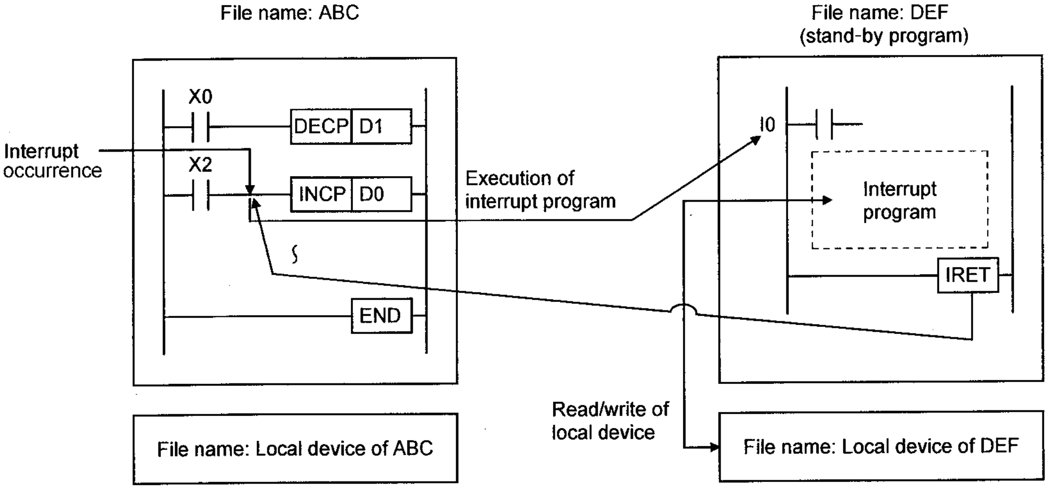
[SM776 operation: ON with function version B]



(b) Operation for interrupt program
[SM776 operation: OFF without function version B or with function version B]



[SM776 operation: ON with function version B]



(2) Precautions

(a) When the SM776 is ON, the local device data can be read while the subroutine program is called. Furthermore, the data will be escaped after performing the RET instruction.
When the SM777 is ON, the local device data is read before performing the interrupt program. The data will be escaped after performing the IRET instruction.
Therefore, when SM776 and SM777 are ON, the scan time is extended by the time below after the subroutine program/interrupt program is executed once.

・Q2ASCPU(S1)	F560{1.3×(Number of words in the local device) [μs]
・Q2ASHCPU(S1)	F220{0.8×(Number of words in the local device) [μs]

(b) ON/OFF of SM776 and SM777 is set for each CPU module.
It cannot be set for each file.

(c) When ON/OFF of SM776 and SM777 is changed during execution of the sequence program, the control is performed with the changed information.

APPENDIX 8 NETWORK RELAY FROM ETHERNET MODULE (FUNCTION VERSION B OR LATER)

This is the network system that mixes Ethernet with MELSECNET/10. The network allows communicating data with the Q2ASCPU in other station via many Ethernet or MELSECNET/10.
To perform the network relay from the Ethernet module, the Ethernet module with function version B or later is required.

(1) Access range
Table 8.1 shows the access range of the network relay ftom the computer/peripheral device with the system:

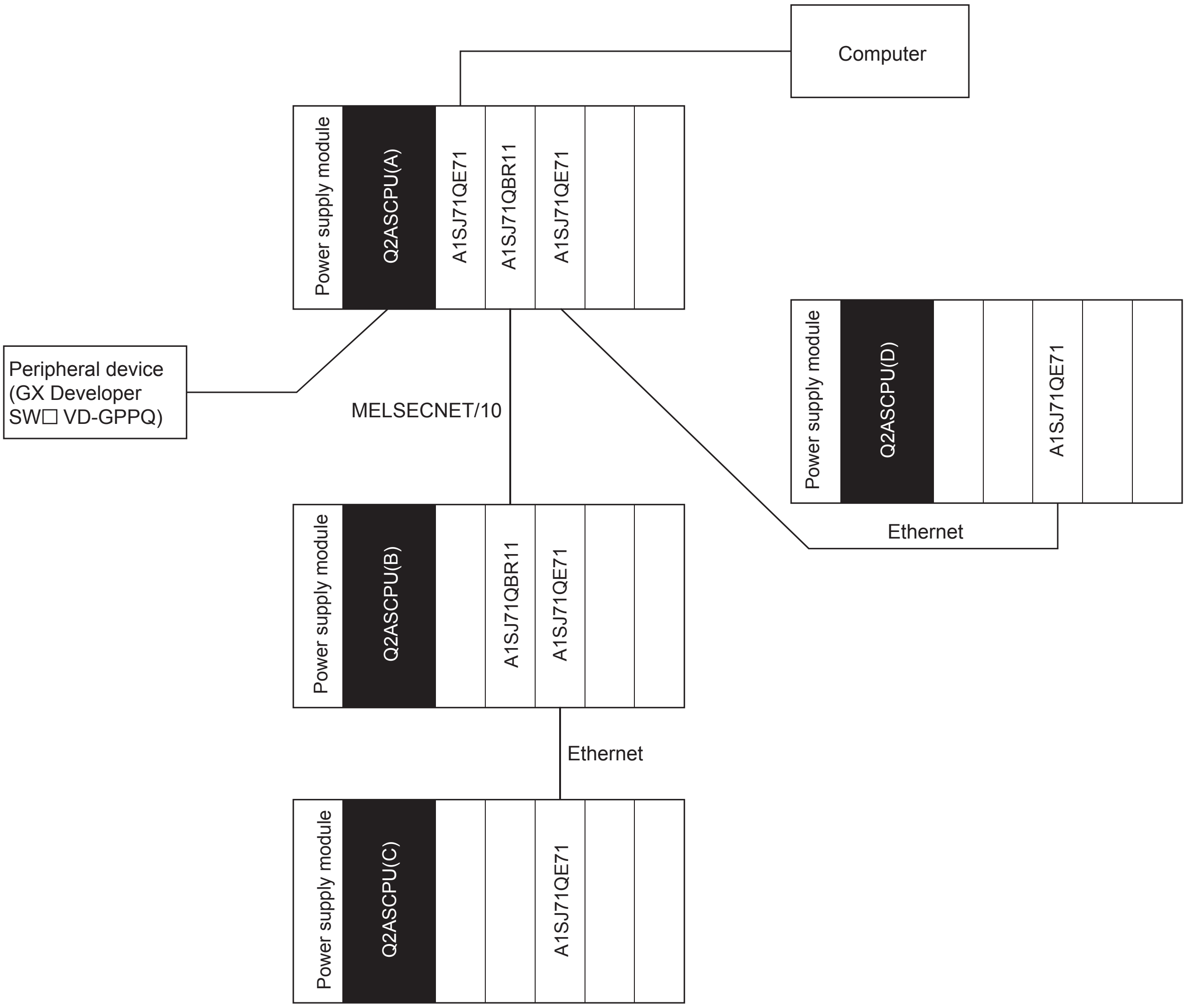


Table 8.1 Comparison table of access range from Ethernet module

Access to	Route	Q2ASCPU "with" Function Version		Q2ASCPU "without" Function Version	
		A	B	A	B
Host access	(Computer)→ Q2ASCPU(A)	○	○	○	○
Other station access in host network (MELSECNET/10)	(Computer)→ Q2ASCPU(B)	○	○	○	○
Other station access of other network (From MELSECNET/10 to Ethernet)	(Computer)→ Q2ASCPU(C)	○	×	×	×
Other station access in host network (Ethernet)	(Computer)→ Q2ASCPU(D)	○	×	×	×
Host access	(Peripheral device) → Q2ASCPU(A)	○	○	○	○
Other station access in host network (MELSECNET/10)	(Peripheral device) → Q2ASCPU(B)	○	○	○	○
Other station access of other network (From MELSECNET/10 to Ethernet)	(Peripheral device) → Q2ASCPU(C)	○	×	×	×
Other station access in host network (Ethernet)	(Peripheral device) → Q2ASCPU(D)	○	×	×	×

○ : Access allowed, × : Access not allowed
A : Ethernet module has indication of the function version.
B : Ethernet module does not have indication of the function version.

(2) Precautions

(a) With combination of Ethernet module and MELSECNET/10, maximum 7 relays can be performed.

(b) The following shows other station access with or without setting for other station access:

- When other station access valid module is set, the set module is used for relay.
- When other station access valid module is not set, the relay is as follows:
  When MELSECNET/10 is available: 1st of MELSECNET/10 is relayed.
  When MELSECNET/10 is not available: 1st of Ethernet is relayed.

(c) When parameters are not registered in the Ethernet module, the Q2ASCPU stores the default parameters in all AJ71QE71.
When multiple Ethernet modules are installed, settings are made in the order of 1st station and 2nd station and so on counting from the Q2ASCPU side.

(d) Table 8.2 shows operation of the Q2ASCPU for online/offline of the Ethernet module.

Table 8.2 Operation of Q2ASCPU for online/offline of Ethernet module

Ethernet Parameter	Ethernet Module Status	Q2ASCPUOperation
With	Online	Communication with external device is performed with the specified parameter.
	Offline	The Q2ASCPU does not show an error, but communication with external device is not performed.
Without	Online	Communication with external device is performed with the default parameter.
	Offline	The Q2ASCPU does not show an error, but communication with external device is not performed.

(e) Set the Ethernet module and MELSECNET/10 not to overlap their Network No.s each other.Same network No. cannot be set for them.
The following shows the number of the Ethernet modules and the MELSECNET(/10, /II) modules that can be mounted on one Q2ASCPU:
- (Ethernet module)≦4
- [(MELSECNET/10) + (MELSECNET/II)]≦4

(f) When the Ethernet parameters are set for the Ethernet module without function version B, error code "3103" (No Ethernet module in the I/O number set with the parameter) appears and the system stops due to an error.

APPENDIX 9 Q2AS(H)CPU(S1) PROCESSING TIME

The Q2AS(H)CPU(S1) processing time is explained below.

Appendix 9.1 Overview of the Q2AS(H)CPU(S1) Scan Time

The Q2AS(H)CPU(S1) scan time comes to the total of the following values.

- I/O refresh processing
- Total values of instruction execution time
- END processing

(1) I/O refresh time

(a) I/O data refresh time between the following modules, which is mounted in the Q2AS(H)CPU(S1) main base unit, extension base unit.

- Input module
- Output module
- Special function module

(b) I/O refresh time can be calculated in the following formula.

(I/O refresh time) = (I/O points÷16)×N1 + (Output points÷16)×N2

For N1 and N2, refer to the following table.

CPU module	N1	N2
Q2ASCPU(S1)	5.2μs	5.0μs
Q2ASHCPU(S1)	4.4μs	4.3μs

(2) Instruction execution time

(a) The processing time of each instruction used for the Q2AS(H)CPU(S1) program. For the processing time of each instruction, refer to the following manual.

- QCPU (Q mode)/QnACPU Programming Manual (Common Instructions)

(b) Since interrupt/fixed-cycle execution type program have overhead time, add the overhead time to the instruction execution time.

(3) END processing

(a) The Q2AS(H)CPU(S1) common processing time except for above (1) ,(2).

(b) The following table shows values of the END processing time.

	CPU module	END Proc Time
With error check (SM1084 = OFF)	Q2ASCPU(S1)	1.7ms
	Q2ASHCPU(S1)	0.7ms
Without error check (SM1084 = ON)	Q2ASCPU(S1)	1.2ms
	Q2ASHCPU(S1)	0.5ms

Appendix 9.2 Causes of Increasing Scan Time

The following shows the functions that increase the Q2AS(H)CPU(S1) scan time.
When using the following functions, add the values calculated in Appendix 9.1 to the following values.

- MELSECNET/10 refresh
- MELSECNET/MINI-S3 refresh
- CC-Link auto refresh
- Sampling trace
- Monitor using GX Developer
- Local device
- Multiple program execution
- Installation/removal of memory card
- File register whose file name is the same as the program

(1) MELSECNET/10 refresh
Refresh time between the Q2AS(H)CPU(S1) and MELSECNET/10 network module.
For MELSECNET/10 refresh time, refer to the following manual.
- QnA/Q4AR MELSECNET/10 Network System Reference Manual

(2) MELSECNET/MINI-S3 refresh
Refresh time between the Q2AS(H)CPU(S1) and MELSECNET/MINI(S3) network module.
For MELSECNET/MINI (S3) refresh time, refer to the following manual.
- MELSECNET/MINI-S3 Master Module User's Manual

(3) CC-Link auto refresh
Refresh time between the Q2AS(H)CPU(S1) and CC-Link master/local module.
For the auto refesh processing time of CC-Link, refer to the following manual.
- Control & Communication Link System Master/Local Module type AJ61QBT11/ A1SJ61QBT11 User's Manual

(4) Sampling trace
(a) Processing time in the case of sampling trace execution
Sampling trace data are set using GX Developer, and the processing time is added when the sampling trace is executed.
(b) The following table shows the processing time when internal relay 50 points as a bit device, data register 50 points as a word device are set for sampling trace data.

CPU module	Processing Time
Q2ASCPU(S1)	3.2ms
Q2ASHCPU(S1)	1.2ms

(5) Monitor using GX Developer

Processing time in the case of monitoring by GX Developer

The processing time is added when monitoring by GX Developer.

(a) The following table shows the processing time when data register 64 points are set for registration monitor:

CPU module	Processing Time
Q2ASCPU(S1)	0.46ms
Q2ASHCPU(S1)	0.18ms

(b) The following shows the processing time when monitor conditions are set.

CPU module	Processing Time	
	Agreement in Designated Step	Agreement in Designated Device
Q2ASCPU(S1)	0.38ms	0.38ms
Q2ASHCPU(S1)	0.15ms	0.15ms

(6) Local device

Processing time when the local device is used

The processing time is added when the local device is used.

CPU module	Processing Time
Q2ASCPU(S1)	3.0 × (n - 1) + 2.8ms
Q2ASHCPU(S1)	1.1 × (n - 1) + 1.1ms

Condition: loca device setting: 1 k point, n: number of program files

(7) Multiple program execution

Overhead time of each program execution when the Q2AS(H)CPU(S1) performs multiple programs. The processing time is added when several programs are executed..

CPU module	Processing Time
Q2ASCPU(S1)	0.21 × n ms
Q2ASHCPU(S1)	0.08 × n ms

Condition: n: number of program files

(8) File register

Processing time when the file register is used

The processing time is added when the file register is used.

CPU module	Processing Time
Q2ASCPU(S1)	0.87 × (n - 1) + 0.74ms
Q2ASHCPU(S1)	0.32 × (n - 1) + 0.28ms

Condition: n: number of program files

APPENDIX 10 TRANSPORTATION PRECAUTIONS

When transporting lithium batteries, make sure to treat them based on the transportation regulations.

Appendix 10.1 Relevant Models

The batteries used for Q2ASCPU are classified as shown in the table below:

Product Name	Model Name	Description	Handled as
QnA series battery	A6BAT	Lithium battery alone	Non-dangerous goods
QnA series memory card	Q1MEM-128S, Q1MEM-128SE, Q1MEM-1MS, Q1MEM-1MSE, Q1MEM-256S, Q1MEM-256SE, Q1MEM-2MS, Q1MEM-512S, Q1MEM-512SE, Q1MEM-64S, Q1MEM-64SE	Packed with lithium coin battery (BR2325)	

Appendix 10.2 Transportation Guidelines

Products are packed properly in compliance with the transportation regulations prior to shipment. When repacking any of the unpacked products to transport it to another location, make sure to observe the IATA Dangerous Goods Regulations, IMDG Code and other local transportation regulations.
For details, please consult your transportation company.

APPENDIX 11 Handling of Batteries and Devices with Built-in Batteries in EU Countries

This section describes the precautions for disposing of used batteries in EU countries and exporting batteries and/or devices with built-in batteries to EU countries.

Appendix 11.1 Disposal precautions

In EU countries, there is a separate collection system for used batteries. Dispose of batteries properly at the local community waste collection/recycling center.

The following symbol is printed on the batteries and packaging of batteries and devices with built-in batteries used for Mitsubishi programmable controllers.

Symbol mark

Note: This symbol mark is for EU countries only.
The symbol mark is specified in the EU directive 2006/66/EC Article 20 AgInformation for end usersAh and Annex II.

The symbol mark indicates that batteries need to be disposed of separately from other wastes.

Appendix 11.2 Exportation precautions

In accordance with the enforcement of the new EU Battery Directive (2006/66/EC), the following must be required when marketing or exporting batteries and/or devices with builtin batteries to EU coutries.

- To print the symbol mark on batteries, devices, or their packaging
- To explain the symbol mark in the manuals of the products

(1) Printing the symbol mark
To market or export batteries and/or devices with built-in batteries, which have no symbol, to EU member states on September 26, 2008 or later, print the symbol shownon the previous page on the batteries, devices, or their packaging.

(2) Explaining the symbol mark in the manuals
To export devices incorporating Mitsubishi programmable controller to EU countries on September 26, 2008 or later, provide the latest manuals that include the explanation of the symbol mark.
If no Mitsubishi manuals or any old manuals without the explanation of the symbol mark are provided, separately attach an explanatory note regarding the symbol mark to each manual of the devices.

POINT

- The requirements apply to batteries and/or devices with built-in batteries manufactured before the enforcement date of the new EU Battery Directive.

INDEX

WARRANTY

Please confirm the following product warranty details before using this product.

1. Gratis Warranty Term and Gratis Warranty Range

If any faults or defects (hereinafter "Failure") found to be the responsibility of Mitsubishi occurs during use of the product within the gratis warranty term, the product shall be repaired at no cost via the sales representative or Mitsubishi Service Company.
However, if repairs are required onsite at domestic or overseas location, expenses to send an engineer will be solely at the customer's discretion. Mitsubishi shall not be held responsible for any re-commissioning, maintenance, or testing on-site that involves replacement of the failed module.

[Gratis Warranty Term]

The gratis warranty term of the product shall be for one year after the date of purchase or delivery to a designated place.
Note that after manufacture and shipment from Mitsubishi, the maximum distribution period shall be six (6) months, and the longest gratis warranty term after manufacturing shall be eighteen (18) months. The gratis warranty term of repair parts shall not exceed the gratis warranty term before repairs.

[Gratis Warranty Range]

(1) The range shall be limited to normal use within the usage state, usage methods and usage environment, etc., which follow the conditions and precautions, etc., given in the instruction manual, user's manual and caution labels on the product.

(2) Even within the gratis warranty term, repairs shall be charged for in the following cases.

1. Failure occurring from inappropriate storage or handling, carelessness or negligence by the user. Failure caused by the user's hardware or software design.
2. Failure caused by unapproved modifications, etc., to the product by the user.
3. When the Mitsubishi product is assembled into a user's device, Failure that could have been avoided if functions or structures, judged as necessary in the legal safety measures the user's device is subject to or as necessary by industry standards, had been provided.
4. Failure that could have been avoided if consumable parts (battery, backlight, fuse, etc.) designated in the instruction manual had been correctly serviced or replaced.
5. Failure caused by external irresistible forces such as fires or abnormal voltages, and Failure caused by force majeure such as earthquakes, lightning, wind and water damage.
6. Failure caused by reasons unpredictable by scientific technology standards at time of shipment from Mitsubishi.
7. Any other failure found not to be the responsibility of Mitsubishi or that admitted not to be so by the user.

2. Onerous repair term after discontinuation of production

(1) Mitsubishi shall accept onerous product repairs for seven (7) years after production of the product is discontinued.
Discontinuation of production shall be notified with Mitsubishi Technical Bulletins, etc.

(2) Product supply (including repair parts) is not available after production is discontinued.

3. Overseas service

Overseas, repairs shall be accepted by Mitsubishi's local overseas FA Center. Note that the repair conditions at each FA Center may differ.

4. Exclusion of loss in opportunity and secondary loss from warranty liability

Regardless of the gratis warranty term, Mitsubishi shall not be liable for compensation of damages caused by any cause found not to be the responsibility of Mitsubishi, loss in opportunity, lost profits incurred to the user by Failures of Mitsubishi products, special damages and secondary damages whether foreseeable or not, compensation for accidents, and compensation for damages to products other than Mitsubishi products, replacement by the user, maintenance of on-site equipment, start-up test run and other tasks.

5. Changes in product specifications

The specifications given in the catalogs, manuals or technical documents are subject to change without prior notice.

Model Q2AS (H) CPU (S1) User's Manual

MODEL	Q2ASCPU-U-E
MODEL CODE	13J858
SH(NA)-3599-K(1101)MEE	

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE : TOKYO BUILDING, 2-7-3 MARUNOUCHI, CHIYODA-KU, TOKYO 100-8310, JAPAN
NAGOYA WORKS : 1-14 , YADA-MINAMI 5-CHOME , HIGASHI-KU, NAGOYA , JAPAN

When exported from Japan, this manual does not require application to the Ministry of Economy, Trade and Industry for service transaction permission.