MITSUBISHI
ELECTRIC

Programmable Controller

MELSEC iQ-R series

MELSEC iQ-R Serial Communication Module User's Manual (Application)

-RJ71C24
-RJ71C24-R2
-RJ71C24-R4

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.

The precautions given in this manual are concerned with this product only. For the safety precautions for the programmable controller system, refer to the user's manual for the module used and the MELSEC iQ-R Module Configuration Manual.

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.

Observe the precautions of both levels because they are important for personal and system safety.

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

[Design Precautions]

⚠WARNING

- Configure safety circuits external to the programmable controller to ensure that the entire system operates safely even when a fault occurs in the external power supply or the programmable controller. Failure to do so may result in an accident due to an incorrect output or malfunction.
  (1) Emergency stop circuits, protection circuits, and protective interlock circuits for conflicting operations (such as forward/reverse rotations or upper/lower limit positioning) must be configured external to the programmable controller.
  (2) When the programmable controller detects an abnormal condition, it stops the operation and all outputs are:
    - Turned off if the overcurrent or overvoltage protection of the power supply module is activated.
    - Held or turned off according to the parameter setting if the self-diagnostic function of the CPU module detects an error such as a watchdog timer error.

  (3) All outputs may be turned on if an error occurs in a part, such as an I/O control part, where the CPU module cannot detect any error. To ensure safety operation in such a case, provide a safety mechanism or a fail-safe circuit external to the programmable controller. For a fail-safe circuit example, refer to the MELSEC iQ-R Module Configuration Manual.
  (4) Outputs may remain on or off due to a failure of a component such as a relay and transistor in an output circuit. Configure an external circuit for monitoring output signals that could cause a serious accident.
- In an output circuit, when a load current exceeding the rated current or an overcurrent caused by a load short-circuit flows for a long time, it may cause smoke and fire. To prevent this, configure an external safety circuit, such as a fuse.
- Configure a circuit so that the programmable controller is turned on first and then the external power supply. If the external power supply is turned on first, an accident may occur due to an incorrect output or malfunction.
- Configure a circuit so that the external power supply is turned off first and then the programmable controller. If the programmable controller is turned off first, an accident may occur due to an incorrect output or malfunction.
- For the operating status of each station after a communication failure, refer to manuals for the network used. For the manuals, please consult your local Mitsubishi representative. Incorrect output or malfunction due to a communication failure may result in an accident.
- When connecting an external device with a CPU module or intelligent function module to modify data of a running programmable controller, configure an interlock circuit in the program to ensure that the entire system will always operate safely. For other forms of control (such as program modification, parameter change, forced output, or operating status change) of a running programmable controller, read the relevant manuals carefully and ensure that the operation is safe before proceeding. Improper operation may damage machines or cause accidents. When a Safety CPU is used, data cannot be modified while the Safety CPU is in SAFETY MODE.
- Especially, when a remote programmable controller is controlled by an external device, immediate action cannot be taken if a problem occurs in the programmable controller due to a communication failure. To prevent this, configure an interlock circuit in the program, and determine corrective actions to be taken between the external device and CPU module in case of a communication failure.

[Design Precautions]

⚠WARNING

- Do not write any data to the "system area" and "write-protect area" of the buffer memory in the module. Also, do not use any "use prohibited" signals as an output signal from the CPU module to each module. Doing so may cause malfunction of the programmable controller system. For the "system area", "write-protect area", and the "use prohibited" signals, refer to the user's manual for the module used. For areas used for safety communications, they are protected from being written by users, and thus safety communications failure caused by data writing does not occur.
- If a communication cable is disconnected, the network may be unstable, resulting in a communication failure of multiple stations. Configure an interlock circuit in the program to ensure that the entire system will always operate safely even if communications fail. Incorrect output or malfunction due to a communication failure may result in an accident. When safety communications are used, an interlock by the safety station interlock function protects the system from an incorrect output or malfunction.

[Design Precautions]

⚠CAUTION

- Do not install the control lines or communication cables together with the main circuit lines or power cables. Doing so may result in malfunction due to electromagnetic interference. Keep a distance of 100mm or more between those cables.
- During control of an inductive load such as a lamp, heater, or solenoid valve, a large current (approximately ten times greater than normal) may flow when the output is turned from off to on. Therefore, use a module that has a sufficient current rating.
- After the CPU module is powered on or is reset, the time taken to enter the RUN status varies depending on the system configuration, parameter settings, and/or program size. Design circuits so that the entire system will always operate safely, regardless of the time.
- Do not power off the programmable controller or do not reset the CPU module while the settings are being written. Doing so will make the data in the flash ROM or SD memory card undefined. The values need to be set in the buffer memory and written to the flash ROM or the SD memory card again. Doing so may cause malfunction or failure of the module.
- When changing the operating status of the CPU module from external devices (such as the remote RUN/STOP functions), select "Do Not Open by Program" for "Opening Method" of "Module Parameter". If "Open by Program" is selected, an execution of the remote STOP function causes the communication line to close. Consequently, the CPU module cannot reopen the line, and external devices cannot execute the remote RUN function.

[Security Precautions]

⚠WARNING

- To maintain the security (confidentiality, integrity, and availability) of the programmable controller and the system against unauthorized access, denial-of-service (DoS) attacks, computer viruses, and other cyberattacks from external devices via the network, take appropriate measures such as firewalls, virtual private networks (VPNs), and antivirus solutions.

[Installation Precautions]

⚠WARNING

- Shut off the external power supply (all phases) used in the system before mounting or removing the module. Failure to do so may result in electric shock or cause the module to fail or malfunction.

[Installation Precautions]

⚠CAUTION

- Use the programmable controller in an environment that meets the general specifications in the MELSEC iQ-R Module Configuration Manual. Failure to do so may result in electric shock, fire, malfunction, or damage to or deterioration of the product.
- To mount a module, place the concave part(s) located at the bottom onto the guide(s) of the base unit, and push in the module until the hook(s) located at the top snaps into place. Incorrect interconnection may cause malfunction, failure, or drop of the module.
- To mount a module with no module fixing hook, place the concave part(s) located at the bottom onto the guide(s) of the base unit, push in the module, and fix it with screw(s). Incorrect interconnection may cause malfunction, failure, or drop of the module.
- When using the programmable controller in an environment of frequent vibrations, fix the module with a screw.
- Tighten the screws within the specified torque range. Undertightening can cause drop of the component or wire, short circuit, or malfunction. Overtightening can damage the screw and/or module, resulting in drop, short circuit, or malfunction. For the specified torque range, refer to the MELSEC iQ-R Module Configuration Manual.
- When using an extension cable, connect it to the extension cable connector of the base unit securely. Check the connection for looseness. Poor contact may cause malfunction.
- When using an SD memory card, fully insert it into the SD memory card slot. Check that it is inserted completely. Poor contact may cause malfunction.

[Installation Precautions]

⚠CAUTION

- Securely insert an extended SRAM cassette or a battery-less option cassette into the cassette connector of the CPU module. After insertion, close the cassette cover and check that the cassette is inserted completely. Poor contact may cause malfunction.
- Beware that the module could be very hot while power is on and immediately after power-off.
- Do not directly touch any conductive parts and electronic components of the module, SD memory card, extended SRAM cassette, battery-less option cassette, or connector. Doing so can cause malfunction or failure of the module.

[Wiring Precautions]

⚠WARNING

- Shut off the external power supply (all phases) used in the system before installation and wiring. Failure to do so may result in electric shock or cause the module to fail or malfunction.
- After installation and wiring, attach a blank cover module (RG60) to each empty slot before powering on the system for operation. Also, attach an extension connector protective cover*1 to each unused extension cable connector as necessary. Directly touching any conductive parts of the connectors while power is on may result in electric shock.

  *1 For details, please consult your local Mitsubishi Electric representative.

[Wiring Precautions]

⚠CAUTION

- Individually ground the FG and LG terminals of the programmable controller with a ground resistance of 100 ohms or less. Failure to do so may result in electric shock or malfunction.
- Use applicable solderless terminals and tighten them within the specified torque range. If any spade solderless terminal is used, it may be disconnected when the terminal screw comes loose, resulting in failure.
- Check the rated voltage and signal layout before wiring to the module, and connect the cables correctly. Connecting a power supply with a different voltage rating or incorrect wiring may cause fire or failure.
- Connectors for external devices must be crimped or pressed with the tool specified by the manufacturer, or must be correctly soldered. Incomplete connections may cause short circuit, fire, or malfunction.
- Securely connect the connector to the module. Poor contact may cause malfunction.
- Do not install the control lines or communication cables together with the main circuit lines or power cables. Doing so may result in malfunction due to noise. Keep a distance of 100mm or more between those cables.
- Place the cables in a duct or clamp them. If not, dangling cables may swing or inadvertently be pulled, resulting in malfunction or damage to modules or cables.
  In addition, the weight of the cables may put stress on modules in an environment of strong vibrations and shocks.
  Do not clamp the extension cables with the jacket stripped. Doing so may change the characteristics of the cables, resulting in malfunction.
- Check the interface type and correctly connect the cable. Incorrect wiring (connecting the cable to an incorrect interface) may cause failure of the module and external device.
- Tighten the terminal screws or connector screws within the specified torque range. Undertightening can cause drop of the screw, short circuit, fire, or malfunction. Overtightening can damage the screw and/or module, resulting in drop, short circuit, fire, or malfunction.
- When disconnecting the cable from the module, do not pull the cable by the cable part. For the cable with connector, hold the connector part of the cable. For the cable connected to the terminal block, loosen the terminal screw. Pulling the cable connected to the module may result in malfunction or damage to the module or cable.
- Prevent foreign matter such as dust or wire chips from entering the module. Such foreign matter can cause a fire, failure, or malfunction.
- When a protective film is attached to the top of the module, remove it before system operation. If not, inadequate heat dissipation of the module may cause a fire, failure, or malfunction.

[Wiring Precautions]

⚠CAUTION

- Programmable controllers must be installed in control panels. Connect the main power supply to the power supply module in the control panel through a relay terminal block. Wiring and replacement of a power supply module must be performed by qualified maintenance personnel with knowledge of protection against electric shock. For wiring, refer to the MELSEC iQ-R Module Configuration Manual.
- For Ethernet cables to be used in the system, select the ones that meet the specifications in the user's manual for the module used. If not, normal data transmission is not guaranteed.

[Startup and Maintenance Precautions]

⚠WARNING

- Do not touch any terminal while power is on. Doing so will cause electric shock or malfunction.
- Correctly connect the battery connector. Do not charge, disassemble, heat, short-circuit, solder, or throw the battery into the fire. Also, do not expose it to liquid or strong shock. Doing so will cause the battery to produce heat, explode, ignite, or leak, resulting in injury or fire.
- Shut off the external power supply (all phases) used in the system before cleaning the module or retightening the terminal screws, connector screws, or module fixing screws. Failure to do so may result in electric shock.

[Startup and Maintenance Precautions]

⚠CAUTION

- When connecting an external device with a CPU module or intelligent function module to modify data of a running programmable controller, configure an interlock circuit in the program to ensure that the entire system will always operate safely. For other forms of control (such as program modification, parameter change, forced output, or operating status change) of a running programmable controller, read the relevant manuals carefully and ensure that the operation is safe before proceeding. Improper operation may damage machines or cause accidents.
- Especially, when a remote programmable controller is controlled by an external device, immediate action cannot be taken if a problem occurs in the programmable controller due to a communication failure. To prevent this, configure an interlock circuit in the program, and determine corrective actions to be taken between the external device and CPU module in case of a communication failure.
- Do not disassemble or modify the modules. Doing so may cause failure, malfunction, injury, or a fire.
- Use any radio communication device such as a cellular phone or PHS (Personal Handy-phone System) more than 25cm away in all directions from the programmable controller. Failure to do so may cause malfunction.
- Shut off the external power supply (all phases) used in the system before mounting or removing the module. Failure to do so may cause the module to fail or malfunction.
- Tighten the screws within the specified torque range. Undertightening can cause drop of the component or wire, short circuit, or malfunction. Overtightening can damage the screw and/or module, resulting in drop, short circuit, or malfunction.
- After the first use of the product, do not perform each of the following operations more than 50 times (IEC 61131-2/JIS B 3502 compliant).

  Exceeding the limit may cause malfunction.
  - Mounting/removing the module to/from the base unit
  - Inserting/removing the extended SRAM cassette or battery-less option cassette to/from the CPU module
  - Mounting/removing the terminal block to/from the module
  - Connecting/disconnecting the extension cable to/from the base unit
- After the first use of the product, do not insert/remove the SD memory card to/from the CPU module more than 500 times. Exceeding the limit may cause malfunction.
- Do not touch the metal terminals on the back side of the SD memory card. Doing so may cause malfunction or failure of the module.
- Do not touch the integrated circuits on the circuit board of an extended SRAM cassette or a battery-less option cassette. Doing so may cause malfunction or failure of the module.
- Do not drop or apply shock to the battery to be installed in the module. Doing so may damage the battery, causing the battery fluid to leak inside the battery. If the battery is dropped or any shock is applied to it, dispose of it without using.

[Startup and Maintenance Precautions]

⚠CAUTION

- Startup and maintenance of a control panel must be performed by qualified maintenance personnel with knowledge of protection against electric shock. Lock the control panel so that only qualified maintenance personnel can operate it.
- Before handling the module, touch a conducting object such as a grounded metal to discharge the static electricity from the human body. Wearing a grounded antistatic wrist strap is recommended. Failure to discharge the static electricity may cause the module to fail or malfunction.
- Use a clean and dry cloth to wipe off dirt on the module.

[Operation Precautions]

⚠CAUTION

- When changing data and operating status, and modifying program of the running programmable controller from an external device such as a personal computer connected to an intelligent function module, read relevant manuals carefully and ensure the safety before operation. Incorrect change or modification may cause system malfunction, damage to the machines, or accidents.
- Do not power off the programmable controller or reset the CPU module while the setting values in the buffer memory are being written to the flash ROM in the module. Doing so will make the data in the flash ROM or SD memory card undefined. The values need to be set in the buffer memory and written to the flash ROM or SD memory card again. Doing so can cause malfunction or failure of the module.

[Computer Connection Precautions]

⚠CAUTION

- When connecting a personal computer to a module having a USB interface, observe the following precautions as well as the instructions described in the manual for the personal computer used. Failure to do so may cause the module to fail.

  (1) When the personal computer is AC-powered

  When the personal computer has a 3-pin AC plug or an AC plug with a grounding wire, connect the plug to a grounding receptacle or ground the grounding wire. Ground the personal computer and the module with a ground resistance of 100 ohms or less.

  When the personal computer has a 2-pin AC plug without a grounding wire, connect the computer to the module by following the procedure below. For power supplied to the personal computer and the module, using the same power source is recommended.

  1. Unplug the personal computer from the AC receptacle.
  2. Check that the personal computer is unplugged. Then, connect the personal computer to the module with a USB cable.
  3. Plug the personal computer into the AC receptacle.

  (2) When the personal computer is battery-powered

  The personal computer can be connected to the module without taking specific measures.

  For details, refer to the following.

  Cautions When Using Mitsubishi Electric Programmable Controllers or GOTs Connected to a Personal Computer With the RS-232/USB Interface (FA-A-0298)

  When the USB cable used is the GT09-C30USB-5P manufactured by Mitsubishi Electric, specific measures are not required to connect the AC-powered personal computer to the module. However, note that the signal ground (SG) is common for the module and its USB interface. Therefore, if an SG potential difference occurs between the module and the connected devices, it causes failures of the module and the connected devices.

[Disposal Precautions]

⚠CAUTION

- When disposing of this product, treat it as industrial waste.
- When disposing of batteries, separate them from other wastes according to the local regulations. For details on battery regulations in EU member states, refer to the MELSEC iQ-R Module Configuration Manual.

[Transportation Precautions]

⚠CAUTION

- When transporting lithium batteries, follow the transportation regulations. For details on the regulated models, refer to the MELSEC iQ-R Module Configuration Manual.
- The halogens (such as fluorine, chlorine, bromine, and iodine), which are contained in a fumigant used for disinfection and pest control of wood packaging materials, may cause failure of the product. Prevent the entry of fumigant residues into the product or consider other methods (such as heat treatment) instead of fumigation. The disinfection and pest control measures must be applied to unprocessed raw wood.

CONDITIONS OF USE FOR THE PRODUCT

(1) MELSEC 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 ELECTRIC 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 ELECTRIC USER'S, 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 Electric 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 Electric 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 Electric representative in your region.

(3) Mitsubishi Electric shall have no responsibility or liability for any problems involving programmable controller trouble and system trouble caused by DoS attacks, unauthorized access, computer viruses, and other cyberattacks.

INTRODUCTION

Thank you for purchasing the Mitsubishi Electric MELSEC iQ-R series programmable controllers.

This manual describes the functions and programming to use the module listed below.

Before using the product, please read this manual and relevant manuals carefully and develop familiarity with the performance of MELSEC iQ-R series programmable controller to handle the product correctly.

When applying the example programs provided in this manual to an actual system, ensure the applicability and confirm that it will not cause system control problems.

Please make sure that the end users read this manual.

Point

The program examples shown in this manual are the examples in which C24 is assigned to the input/output No. X/Y0 to X/Y1F unless otherwise specified. To use the program examples shown in this manual, the input/output number assignment is required. For details on the assignment of input/output number, refer to the following manual.

📖MELSEC iQ-R Module Configuration Manual

Relevant products

RJ71C24, RJ71C24-R2, RJ71C24-R4

CONTENTS

PART 1 BASIC FUNCTIONS

CHAPTER 18 TRANSPARENT CODES AND ADDITIONAL CODES 358

CHAPTER 19 COMMUNICATING WITH ASCII CODE (ASCII-BIN CONVERSION) 379

CHAPTER 20 DATA COMMUNICATIONS BETWEEN TARGET DEVICE AND CPU MODULE IN M:N BASIS 398

APPENDIX 544

INDEX 687

RELEVANT MANUALS

Manual name [manual number]	Description	Available form
MELSEC iQ-R Serial Communication Module User's Manual(Application) [SH-081251ENG] (this manual)	Functions, input/output signals, buffer memory, parameter setting, and trouble shooting of Serial communication module	Print book e-Manual PDF
MELSEC iQ-R Serial Communication Module User's Manual(Startup) [SH-081250ENG]	Specifications, procedures prior to operation, system configurations, wring, and data communication examples of Serial communication module	Print book e-Manual PDF
MELSEC iQ-R Module Configuration Manual [SH-081262ENG]	The combination of the MELSEC iQ-R series modules, common information on the installation/wiring in the system, and specifications of the power supply module, base unit, SD memory card, and battery	e-Manual PDF
MELSEC iQ-R Programming Manual (Module Dedicated Instructions) [SH-081976ENG]	Dedicated instructions for an intelligent function module	e-Manual PDF
MELSEC iQ-R Serial Communication Module Function Block Reference [BCN-P5999-0379]	FBs of a Serial communication module	e-Manual PDF
GX Works3 Operating Manual [SH-081215ENG]	System configurations, parameter settings, and operation methods for the online function in GX Works3	e-Manual PDF
MELSEC Communication Protocol Reference Manual [SH-080008]	Specifications, accessible range, message protocols, and functions of MELSEC Communication Protocol	Print book e-Manual PDF

Point

e-Manual refers to the Mitsubishi Electric FA electronic book manuals that can be browsed using a dedicated tool.

e-Manual has the following features:

- Required information can be cross-searched in multiple manuals.
- Other manuals can be accessed from the links in the manual.
- The hardware specifications of each part can be found from the product figures.
- Pages that users often browse can be bookmarked.
- Sample programs can be copied to an engineering tool.

TERMS

Unless otherwise specified, this manual uses the following terms.

Terminology	Description
3C frame (Formats 1 to 4) 4C frame (Formats 1 to 4)	A message format for C24 for data communications with ASCII code data using the MC protocol This is the same message format as the communication frame using the dedicated protocol for the QnA series Serial Communication Module. • QnA compatible 3C frame (Formats 1 to 4): QnA frame (Formats 1 to 4) • QnA compatible 4C frame (Formats 1 to 4): QnA extension frame (Formats 1 to 4)
4C frame (Format 5)	A message format for C24 for data communication with binary code data using the MC protocol This is the same message format as the communication frame using the dedicated protocol for the QnA series Serial Communication Module. • QnA compatible 4C frame (Format 5): QnA extension frame (Format 5)
Bidirectional protocol	A data communication function to communicate any data between a CPU module and the target device using the communication procedure for C24
Buffer memory	A memory of an Intelligent function module to store data (including setting value and monitored value) sent to/receive from CPU module
BUFRCVS	An abbreviation for Z.BUFRCVS
C24	Another term for the MELSEC iQ-R series Serial Communication Module
CPRTCL	An abbreviation for G.CPRTCL or GP.CPRTCL
CPU module	A generic term for MELSEC iQ-R series CPU module
CSET	An abbreviation for ZP.CSET
Data communication function	A generic term for data communication functions using MC protocol, predefined protocol, nonprocedural protocol, bidirectional protocol, MODBUS, and simple CPU communication
Engineering tool	A tool for setting, programming, debugging, and maintaining programmable controllers For the supported tools, refer to the following manual. 📖MELSEC iQ-R Module Configuration Manual
GETE	An abbreviation for G.GETE or GP.GETE
Independent operation	A mode of interface operation to communicate data with the target device with a function specified in each communication protocol setting without interaction between two interfaces of C24
INPUT	An abbreviation for G.INPUT
Intelligent function module	A generic term for modules which has functions other than input and output, such as A/D converter module and D/A converter module
Interlink operation	A mode of interface operation to communicate data with the target device with the two interfaces linked one another when two interfaces of C24 are connected to target devices respectively The two interfaces communicate data with the same data communication function (MC protocol (identical format) or nonprocedural protocol) and the same transmission specifications. (Interlink operation of the predefined protocol and the bidirectional protocol is not allowed.)
MC protocol	An abbreviation for MELSEC communication protocol, which is a name of communication method to access a CPU module form a target device using the communication procedure for Serial communication modules or Ethernet communication modules
Multidrop connection	A name of connection when more than one target devices or other C24s are connected on a 1:n basis or an m:n basis using the RS-422/485 interface of C24
Nonprocedural protocol	A data communication function to communicate any data between a CPU module and the target device using user's communication procedure
ONDEMAND	An abbreviation for G.ONDEMAND or GP.ONDEMAND
OUTPUT	An abbreviation for G.OUTPUT or GP.OUTPUT
Packet	A data string used for communication with a target device by predefined protocol
Predefined protocol	A data communication function to send/receive data using a protocol of the target device for data communication between C24 and the target device This setting is configured with GX Works3 (the predefined protocol support function).
Predefined protocol support function	A function available using GX Works3 (the predefined protocol support function) The function includes the following: • Setting the protocol appropriate to respective target devices • Writing or reading the protocol setting data to/from CPU module or C24 • Debug support function
PRR	An abbreviation for G.PRR or GP.PRR
PUTE	An abbreviation for G.PUTE or GP.PUTE
Redundant system with redundant extension base unit	A redundant system that is configured using extension base unit(s)
Remote head module	An abbreviation for RJ72GF15-T2 CC-Link IE Field Network remote head module.

Terminology	Description
SPBUSY	An abbreviation for G.SPBUSY or GP.SPBUSY
Target device	A generic term for GOT, measuring instruments, ID modules, barcode readers, regulators, and other C24s connected to this C24 for data communication
UINI	An abbreviation for ZP.UINI
User frame	A data name used to send/receive data by registering the fixed format part of messages to be sent/received between C24 and the target device to the module (The contents of a user frame data should conform to the specifications of the target device.) The data array of the head and tail sections in messages to be sent/received (transmission control code, C24 station number, sum check, fixed data, etc.) are registered to C24 respectively before use. This is used for the on-demand function of MC protocol and the data transmit/receive function by nonprocedural protocol.

MEMO

PART 1 BASIC FUNCTIONS

This part explains the basic functions of C24.

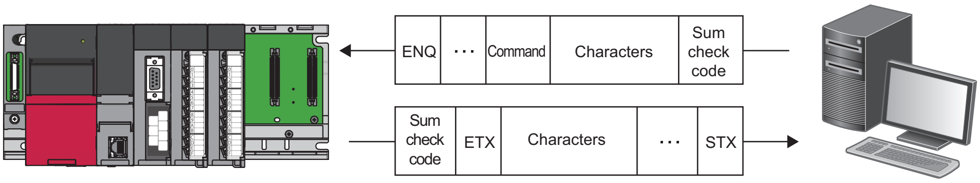
1 DATA COMMUNICATION USING MC PROTOCOL

MC protocol is a communication method for programmable controllers, which enable device data and programs in the CPU module to be read from/written to target devices via C24.

C24 returns a response message automatically based on requests from the target devices. Thus, programs for response messages are not required.

For details of the MC protocol, refer to the following manual.

📖MELSEC Communication Protocol Reference Manual



1.1 Accessing to CPU Module Using MC Protocol

This section explains how to access to a CPU module using MC protocol.

Reading/writing device memory of CPU module

Read/write data from/to the device memory of the access target CPU or intelligent function module buffer memory. For the accessible range, refer to the following manual.

📖MELSEC Communication Protocol Reference Manual

Reading/writing programs and parameters

This function reads and writes program files and parameter files stored in the CPU module.

Remote control of CPU module

This function performs remote RUN/STOP/PAUSE/latch clear/reset operations.

Monitoring of CPU module

This function monitors the operation status and data in the device memory of the CPU module.

Transmitting data to a target device from a CPU module

This function requests data transmission from a CPU module to a target device, and transmits on-demand data to the target device.

File access method

To access to a file, which the password setting is configured, from external devices using the MC protocol, specifying the command for file password is required.

1. Specify the password with a request message.
2. Perform the password authentication by using the specified password.

For command requiring the password authentication, specify "0040" with the setting value of "subcommand" in the request message.

Function	Command (subcommand)
Deleting file	1822 (0040)
Copying file	1824 (0040)
Changing file attribute	1825 (0040)
Opening file	1827 (0040)

For more details on the commands, refer to the following manual.
📖MELSEC Communication Protocol Reference Manual

1.2 Data Communication

MC protocol uses the following communication frames for data communication.
Transmit/receive data from the target device in the data format corresponding to the MC protocol.
For details on the communication frames format, refer to the following manual.
📖MELSEC Communication Protocol Reference Manual

Formats and codes

There are five formats for the data communication frames:

Setting value	format	Code of communication data	Remarks	Reference
1	Format 1	ASCII code	—	MELSEC Communication Protocol Reference Manual
2	Format 2	ASCII code	A format that a block number is added	
3	Format 3	ASCII code	A format enclosed with STX and ETX	
4	Format 4	ASCII code	A format that CR and LF are added at the end	
5	Format 5	Binary code	This can be used only for 4C frame.	

Set the format in the Communication protocol setting of an Engineering tool.

Point

Communication using binary code shortens the communication time since the amount of communication data is about a half of the one in data communications using ASCII code.

Frames

The following explains the types of frame (data communication message) used for a target device to access a supported equipment using the MC protocol, and the purpose of each frame.
The data communication frames are as shown below.

Frame	Feature and purpose	Compatible message format	Format
4C frame	Accessing from a target device to the maximum access range is available.	Dedicated protocol for C24 (QnA extension frame)	Formats 1 to 5
3C frame	The message format of access route is more simple than 4C frame. Data communication software for MELSEC-QnA series programmable controller can be used.	Dedicated protocols for C24 (QnA frame)	Formats 1 to 4
2C frame		Dedicated protocols for C24 (QnA simplified frame)	
1C frame	1C frame is the same message structure as when accessing using MELSEC-A series computer link modules. Data communication software for MELSEC-A series programmable controller can be used.	Dedicated protocols for A series Computer link modules.	

Parameter settings for MC protocol communication

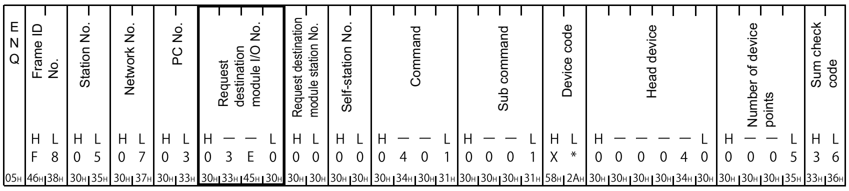
Set the transmission setting and the communication protocol, etc. for a target device.
Select from "MC protocol (Format 1)" to "MC protocol (Format 5)" for "Communication protocol setting".

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

Configure other settings according to the system being used.
For details of the settings, refer to the following chapter.
☞ Page 454 PARAMETER SETTING

Applicability of multiple CPU systems

When a target device accesses to RCPU, which is a multiple CPU system, accessing to the control CPU or Non-control CPU of the multiple CPU system is available by specifying the target RCPU in "Request destination module I/O number" of MC protocol 4C frame.



ENQ	Frame ID No.	Station No.	Network No.	PC No.	Request destination module I/O No.	Request destination module station No.	Self-station No.	Command	Sub command	Device code	Head device	Number of device points	Sum check code
	H L	H L	H L	H L	H — — L	H L	H L	H — — L	H — — L	H L	H — — — — L	H — — L	H L
	F 8	0 5	0 7	0 3	0 3 E 0	0 0	0 0	0 4 0 1	0 0 0 1	X *	0 0 0 0 4 0	0 0 0 5	3 6
05H	46H 38H	30H 35H	30H 37H	30H 33H	30H 33H 45H 30H	30H 30H	30H 30H	30H 34H 30H 31H	30H 30H 30H 31H	58H 2AH	30H 30H 30H 30H 34H 30H	30H 30H 30H 35H	33H 36H

Applicability of remote password function

The remote password function prevents unauthorized access to a CPU module from a remote location.
By setting a remote password to the CPU module, this function can be applicable.
For the method for setting remote password, refer to the following manual.
📖GX Works3 Operating Manual

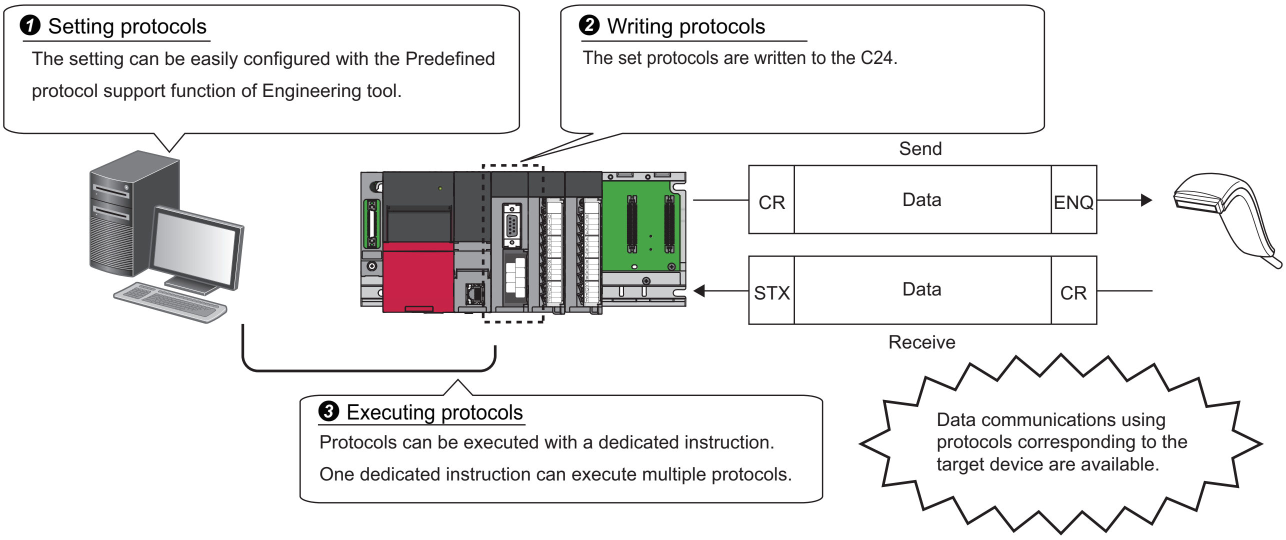
Using MX Component

MX Component is an ActiveX control library that supports various types of communication paths between a personal computer and CPU module.
By using MX Component, an arbitrary system can be configured. Furthermore, the communication program for the target device can be created without considering the content of MC protocol (transmission/reception procedures).
For details, refer to the following manuals.
📖MX Component Version 4 Operating Manual
📖MX Component Version 4 Programming Manual

2 DATA COMMUNICATION USING PREDEFINED PROTOCOL

Data can be transmitted and received between the CPU module and an target device with a protocol appropriate to the target device (such as a measuring instrument and a barcode reader).
Since devices or buffer memory data can be taken into communication packets, this protocol communication is suitable for data that may change every transmission.
The protocol setting required to communicate with the target device is configured in Engineering tool.
Protocols can be selected from the predefined protocol library offered or can be arbitrarily created and edited.
In addition, since the registered setting data is written to the CPU built-in memory, SD memory card, or flash ROM, the program size in the CPU module can be reduced.
For an example of data communication using the predefined protocol, refer to the following manual.
📖MELSEC iQ-R Serial Communication Module User's Manual(Startup)

Data communication using the predefined protocol



❶ Setting protocols

The setting can be easily configured with the Predefined protocol support function of Engineering tool.

❷ Writing protocols

The set protocols are written to the C24.

❸ Executing protocols

Protocols can be executed with a dedicated instruction.
One dedicated instruction can execute multiple protocols.

Data communications using protocols corresponding to the target device are available.

Point

The number of protocols and packets that can be registered are as follows:

- Protocols: Up to 128
- Packets: Up to 256

Depending on the packet to be registered, the maximum numbers of packets above may not be registered.
The usage rate of the packet data area differs depending on the types of protocol.
It can be confirmed on the status bar of the "Protocol Setting" screen displayed by following the procedure as follows:

- Select [Tool] ⇨ [Predefined Protocol Support Function] of Engineering tool, and set "Module Type" and "Connection Destination CPU" on the "Predefined Protocol Support Function" window. And, add the protocol on the "Predefined Protocol Support Function" screen.

2.1 Data Communication Procedure

By using the Predefined protocol support function, data can be communicated with a target device in the following procedure.

Operating procedure

1. Select a protocol in "Predefined Protocol Support Function", and write the protocol setting data.

☞ Page 32 Creating protocol setting data

2. Set the module parameter to "Predefined protocol".

☞ Page 37 Setting method

3. Execute the protocol written with the dedicated instruction (CPRTCL).

📖MELSEC iQ-R Serial Communication Module User's Manual(Startup)

4. Monitor the executed protocol in the Protocol execution log.

The transmitted/received data and the communication control signal status can also be checked using the Circuit trace function.

📖MELSEC iQ-R Serial Communication Module User's Manual(Startup)

Restriction

Communication using the modem function is not available for the predefined protocol.

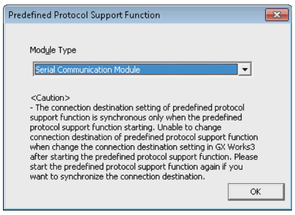
Creating protocol setting data

Create a protocol setting data using the Predefined protocol support function.

■Selecting target module of a protocol setting data

Select a target module of the protocol setting data.

Engineering tool [Tool] ⇨ [Predefined Protocol Support Function]



Predefined Protocol Support Function

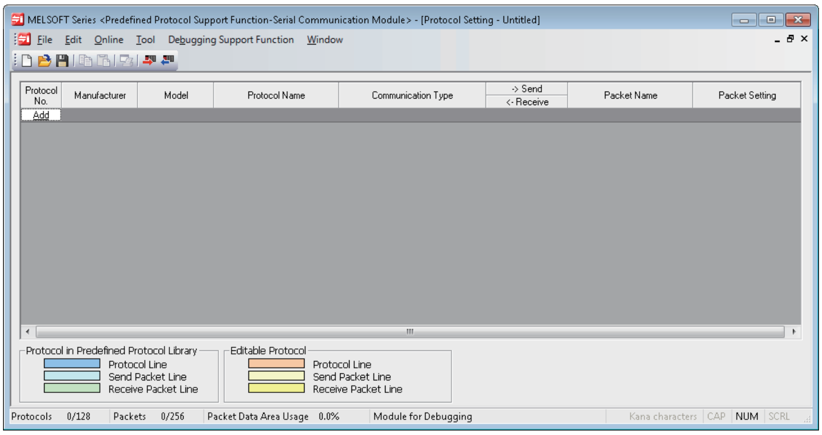
Module Type: Serial Communication Module

<Caution>
- The connection destination setting of predefined protocol support function is synchronous only when the predefined protocol support function starting. Unable to change connection destination of predefined protocol support function when change the connection destination setting in GX Works3 after starting the predefined protocol support function. Please start the predefined protocol support function again if you want to synchronize the connection destination.

■Creating new protocol setting data

Create new protocol setting data.

[File]⇨[New]



MELSOFT Series <Predefined Protocol Support Function-Serial Communication Module> - [Protocol Setting - Untitled]

Protocol No.	Manufacturer	Model	Protocol Name	Communication Type	-> Send <- Receive	Packet Name	Packet Setting
Add							

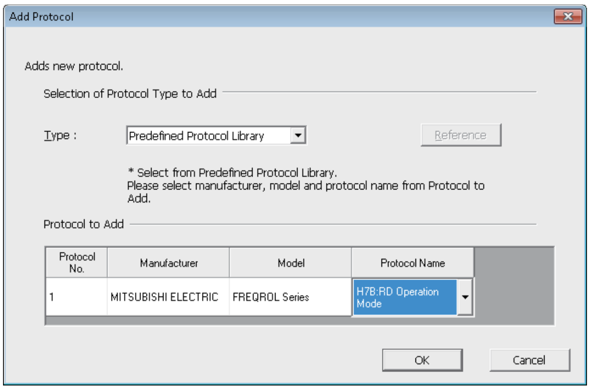
Protocols 0/128 Packets 0/256 Packet Data Area Usage 0.0% Module for Debugging

Item	Description
Protocol No.	Displays the protocol number to be used for a predefined protocol dedicated instruction in programs.
Manufacturer	Displays the manufacture's name of a device to which a protocol to be set is applied.
Model	Displays the model name of a device to which a protocol to be set is applied.
Protocol Name	Displays the name of a protocol to be set.
Communication Type	Displays the communication type in a protocol to be set. • Send Only: Send one send packet once. • Receive Only: Receive a packet if it matches any of up-to-16 registered receive packets. • Send&Receive: Send one send packet, and receives a packet if it matches any of up-to-16 registered receive packets.
→Send/←Receive	Displays the packet direction. • →: For 'Send' • ←(1) to (16): For 'Receive', a receive packet number is displayed in ().
Packet Name	Displays the packet name.
Packet Setting	Displays the existence or non-existence of variables in an element, and set or unset of the variables. With 'Variable Unset', 'Element Unset', or 'Element error', the settings cannot be written to a CPU module or C24. • No Variable: There is no variable in the element. • Variable Set: All variables are set. • Variable Unset: There are one or more unset variables. • Element Unset: There is no element in the editable protocol. • Element error: The element does not meet requirement.

■Adding protocols

Add a protocol.

[Edit] ⇨ [Add Protocol]



Add Protocol

Adds new protocol.

Selection of Protocol Type to Add

Type : Predefined Protocol Library

* Select from Predefined Protocol Library.
Please select manufacturer, model and protocol name from Protocol to Add.

Protocol to Add

Protocol No.	Manufacturer	Model	Protocol Name
1	MITSUBISHI ELECTRIC	FREQROL Series	H7B:RD Operation Mode

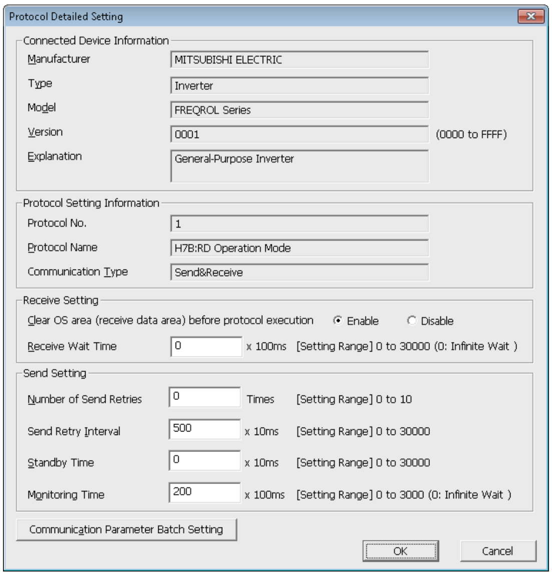
Item	Description	Setting range
Type	Select a protocol type to be added.	• Predefined Protocol Library • User Protocol Library • Add New
Protocol No.	Set the number of the protocol to be added.	1 to 128
Manufacturer[*1]	Set a manufacturer's name of the protocol to be added.	—
Model[*1]	Set a model of the protocol to be added.	—
Protocol Name[*1]	Set a name of the protocol to be added.	—

[*1] These items can be set only when "Predefined Protocol Library" is selected in "Type".

■Configuring detailed setting of protocols

Set the transmission/reception parameters of the protocol.

Select a row of any protocol on the "Protocol Setting" screen ⇨ [Edit] ⇨ [Protocol Detailed Setting].



Protocol Detailed Setting

Connected Device Information

Manufacturer: MITSUBISHI ELECTRIC

Type: Inverter

Model: FREQROL Series

Version: 0001 (0000 to FFFF)

Explanation: General-Purpose Inverter

Protocol Setting Information

Protocol No.: 1

Protocol Name: H7B:RD Operation Mode

Communication Type: Send&Receive

Receive Setting

Clear OS area (receive data area) before protocol execution ● Enable ○ Disable

Receive Wait Time: 0 x 100ms [Setting Range] 0 to 30000 (0: Infinite Wait)

Send Setting

Number of Send Retries: 0 Times [Setting Range] 0 to 10

Send Retry Interval: 500 x 10ms [Setting Range] 0 to 30000

Standby Time: 0 x 10ms [Setting Range] 0 to 30000

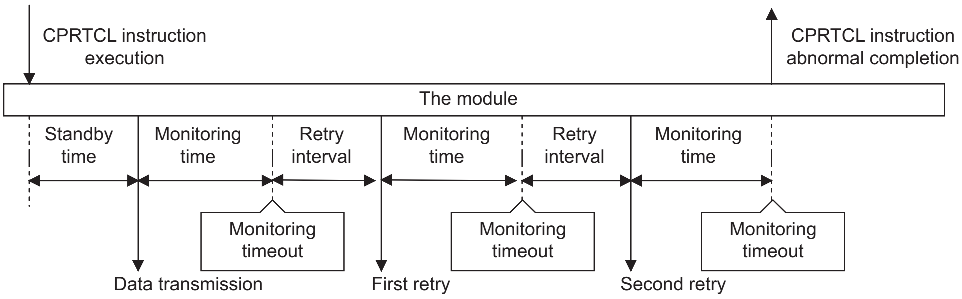
Monitoring Time: 200 x 100ms [Setting Range] 0 to 3000 (0: Infinite Wait)

Item		Description
Connected Device Information[*1]	Manufacturer	Set a manufacturer's name of the connected device.
	Type	Set a device type of the connected device.
	Model	Set a model of the connected device.
	Version	Set a device version of the connected device.
	Explanation	Set equipment description for the connected device.
Protocol Setting Information[*1]	Protocol No.	Displays a protocol number of the selected protocol.
	Protocol Name	Set a protocol name of the protocol.
	Communication Type	Select a communication type of the protocol.
Receive Setting	Clear OS area (receive data area) before protocol execution	Select whether to clear the OS area (receive data area) of the module before the protocol execution. If 'Disable' is selected, the data received in the module before the protocol execution also become a receive target of the protocol.
	Receive Wait Time	Set waiting time after the module turns to the 'waiting for reception' status. If the communication with target devices is not available due to such as a cable disconnection and no matched packet can be received within the set time, the module determines an error and cancels the 'waiting for reception' status.
Send Setting	Number of Send Retries	Set the number of times the module retries to send when the sending from the module has not been completed within the set time of "Monitoring Time". The module determines an error if the sending has not been completed despite the specified number of times of sending retries.
	Send Retry Interval	Set the interval between the failure of sending from the module and the retry when the sending from the module has not been completed within the set time of "Monitoring Time".
	Standby Time	Set standby time between when a protocol set to the module turns to the execution status and when it actually sends the data. By setting this item, the send timing of the module can be adjusted to readiness of target devices to receive data.
	Monitoring Time	Set waiting time between when the module turns to the 'sending' status and when the sending is completed. If the communication with target devices is not available due to such as a cable disconnection and the sending cannot be completed within the set time, the module determines an error and cancels the 'sending' status.

[*1] For a protocol selected from the predefined protocol library, "Connected Device Information" and "Protocol Setting Information" cannot be modified.

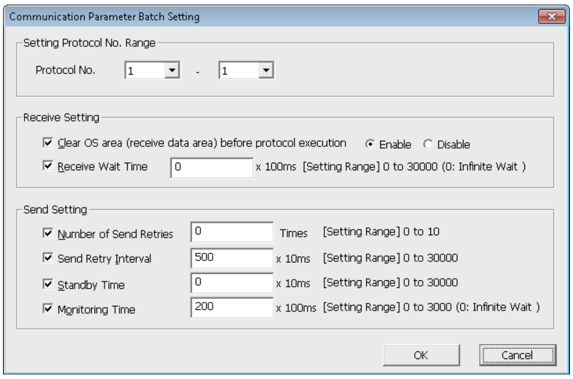
●Processing when the data transmitted error occurred (when the number of send retries is set to '2')

When the set value of "Number of Retries" is 2, the module determines an error at the following timing if it cannot send the data.



Point

The communication parameters can be set for multiple protocols by clicking the [Communication Parameter Batch Setting] button and setting "Setting Protocol No. Range", "Receive Setting", or "Send Setting".



Communication Parameter Batch Setting

Setting Protocol No. Range

Protocol No. 1 - 1

Receive Setting

☑ Clear OS area (receive data area) before protocol execution ● Enable ○ Disable

☑ Receive Wait Time 0 x 100ms [Setting Range] 0 to 30000 (0: Infinite Wait)

Send Setting

☑ Number of Send Retries 0 Times [Setting Range] 0 to 10

☑ Send Retry Interval 500 x 10ms [Setting Range] 0 to 30000

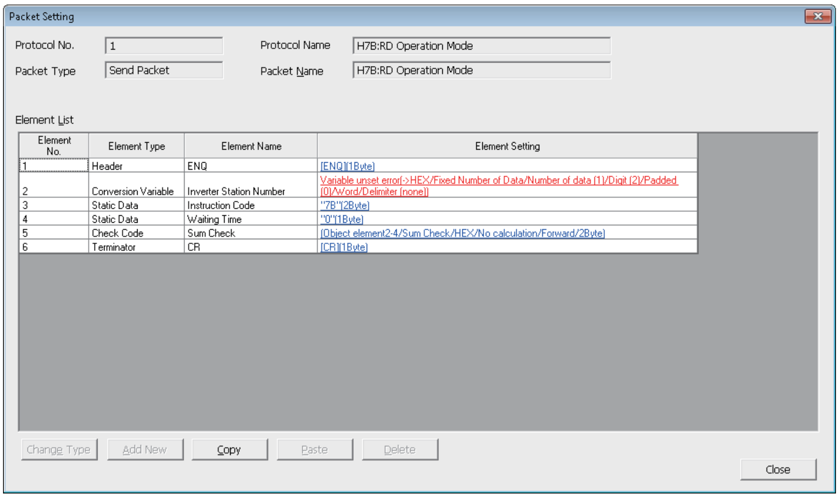
☑ Standby Time 0 x 10ms [Setting Range] 0 to 30000

☑ Monitoring Time 200 x 100ms [Setting Range] 0 to 3000 (0: Infinite Wait)

■Setting packets

Set the packet elements to be transmitted/received on the "Packet Setting" screen.

"Protocol setting" screen ⇨ desired packet setting



Protocol No. 1

Protocol Name H7B:RD Operation Mode

Packet Type Send Packet

Packet Name H7B:RD Operation Mode

Element List

Element No.	Element Type	Element Name	Element Setting
1	Header	ENQ	[ENQ][1Byte]
2	Conversion Variable	Inverter Station Number	Variable unset error[->HEX/Fixed Number of Data/Number of data [1]/Digit [2]/Padded [0]/Word/Delimiter [none]]
3	Static Data	Instruction Code	"7B"[2Byte]
4	Static Data	Waiting Time	"0"[1Byte]
5	Check Code	Sum Check	[Object element2-4/Sum Check/HEX/No calculation/Forward/2Byte]
6	Terminator	CR	[CR][1Byte]

The screen above is an example when "Predefined Protocol Library" is selected on the "Add protocol" screen.

When "Add New" or "User Protocol Library" is selected, press the [Change Type] button or [Add New] button to configure the packet.

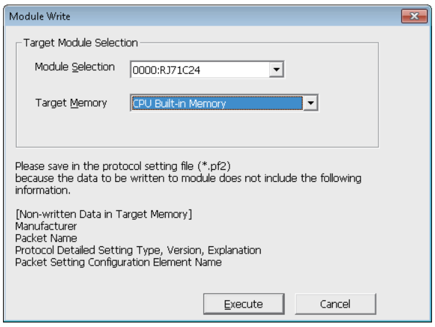
For details of packet elements, refer to the following section.

☞ Page 39 Packet

■Writing protocol setting data

Write data of the protocol settings to a CPU module or C24.

[Online] ⇨ [Write to Module]



Module Write

Target Module Selection

Module Selection: 0000:RJ71C24

Target Memory: CPU Built-in Memory

Please save in the protocol setting file (*.pf2)
because the data to be written to module does not include the following
information.

[Non-written Data in Target Memory]
Manufacturer
Packet Name
Protocol Detailed Setting Type, Version, Explanation
Packet Setting Configuration Element Name

Write the protocol setting data after selecting the module to be written and memory.

When writing data to the CPU module, the protocol setting data is written to "Module Extended Parameter".

Point

The following data is not displayed even when reading from CPU module because it will not be written as a protocol setting data. However, the protocol selected from the predefined library is displayed.

- Manufacturer
- Packet name
- Type, Version, Explanation in the protocol detailed setting
- Element name in the packet setting

The operation when the predefined protocol setting is written to a multiple target memory is as follows:

● When the predefined protocol setting is written to both CPU built-in memory and an SD memory card

The CPU module operates according to the settings of "Memory Card Parameter". For details on the "Memory Card Parameter", refer to the following manual.

MELSEC iQ-R CPU Module User's Manual (Application)

● When the predefined protocol setting is written to a CPU built-in memory or an SD memory card, and C24

The predefined protocol settings of the CPU built-in memory or SD memory card is enabled in the following timing. (If the predefined protocol settings are written both CPU built-in memory and SD memory card, the CPU module operates according to the settings of "Memory Card Parameter".

When the predefined protocol is enabled, the predefined protocol settings of C24 will be overwritten with the predefined protocol of CPU built -in memory or SD memory card.

- When the power is turned OFF→ON
- When the CPU module is reset
- When the CPU module is switched STOP→RUN

Setting method

Set the predefined protocol settings in "Various control specification" of "Basic Settings". (☞ Page 454 PARAMETER SETTING)

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

1. Set the "Communication protocol setting" in "Various control specification" to "Predefined protocol".
2. Set the required parameters other than above.

Dedicated instruction to be used

Use the CPRTCL instruction.

2.2 Communication Type of Protocols

A send packet to the target devices and a receive packet from the target devices at the time of process execution are registered in a protocol.

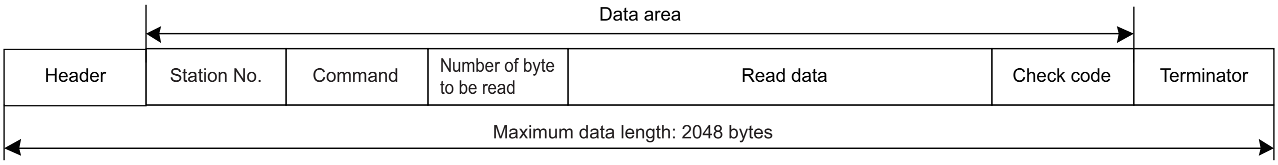
The packet elements which are set to the Predefined protocol support function is applied to the actual data portion of the packet to be transmitted.

For details of packet elements, refer to the following section.

☞ Page 39 Packet elements

Ex.

Example of a packet configuration



With the Predefined protocol support function, communications with target devices are performed using the following procedures (communication types).

Communication type name	Processing
Send Only	Sends a send packet once. One send packet is required.
Receive Only	Receives a packet if it matches any of up to 16 defined receive packets. One or more receive packets are required.
Send & Receive	Sends a send packet, and then receives a packet if it matches any of up to 16 defined receive packets. One send packet and one or more receive packets are required.

For the operation of each operation type, refer to the following section.

☞ Page 581 Operation Image and Data Structure of Predefined Protocol

2.3 Packet

A packet consists of packet elements.

Up to 32 elements can be set in a packet, and the maximum data length is 2048 bytes per packet.

Packet elements

The following explains the details of the packet elements.

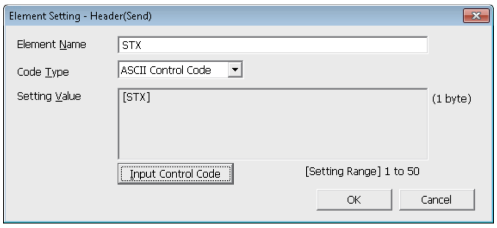
For data examples on packet elements, refer to the following section.

☞ Page 587 Data examples of packet elements

Header

Use this element when a specific code or character string exists at the head of a packet.

- When sending: Sends a specified code and character string.
- When receiving: Verifies a header and received data.



Element Setting - Header(Send)

Element Name: STX

Code Type: ASCII Control Code

Setting Value: [STX] (1 byte)

[Setting Range] 1 to 50

Item	Description	Remarks
Element Name	Set a name of the element.	—
Code Type	Select a data type of the setting value. ASCII String/ASCII Control Code/HEX	—
Setting Value	Set data within 1 to 50 bytes. The setting ranges that can be set for each code type are as follows: • ASCII String: 20H to 7EH • ASCII Control Code: 00H to 1FH, and control code of 7FH • HEX: hexadecimal data of 00H to FFH	Setting example • ASCII String: 'ABC' • ASCII Control Code: STX • HEX : FFFF

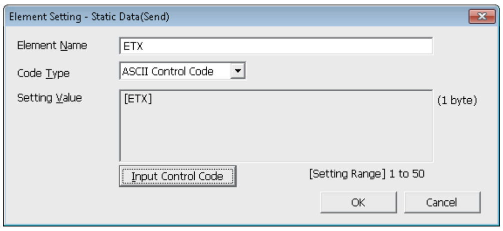
Point

- Only one header can be used in a packet.
- A header can be set only at the head of a packet.

Static Data

Use this element when a specific code or character string such as command exists in a packet.

- When sending: Sends a specified code and character string.
- When receiving: Verifies the received data.


Element Setting - Static Data(Send)

Element Name: ETX

Code Type: ASCII Control Code

Setting Value: [ETX] (1 byte)

[Setting Range] 1 to 50

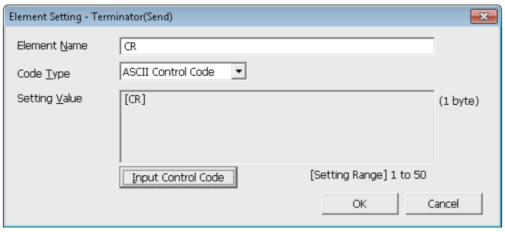
Item	Description	Remarks
Element Name	Set a name of the element.	—
Code Type	Select a data type of the setting value. ASCII String/ASCII Control Code/HEX	—
Setting Value	Set data within 1 to 50 bytes. The setting ranges that can be set for each code type are as follows: • ASCII String: 20H to 7EH • ASCII Control Code: 00H to 1FH, and control code of 7FH • HEX: hexadecimal data of 00H to FFH	Setting example • ASCII String: 'ABC' • ASCII Control Code: US • HEX : FFFF

Point

Multiple static data elements can be placed to desired positions in the data portion.

Terminator

Use this element when a code or character string indicating the end of the packet is included.


Element Setting - Terminator(Send)

Element Name: CR

Code Type: ASCII Control Code

Setting Value: [CR] (1 byte)

[Setting Range] 1 to 50

Item	Description	Remarks
Element Name	Set a name of the element.	—
Code Type	Select a data type of the setting value. ASCII String/ASCII Control Code/HEX	—
Setting Value	Set data within 1 to 50 bytes. The setting ranges that can be set for each code type are as follows: • ASCII String: 20H to 7EH • ASCII Control Code: 00H to 1FH, and control code of 7FH • HEX: hexadecimal data of 00H to FFH	Setting example • ASCII String: 'ABC' • ASCII Control Code: ETX • HEX : FFFF

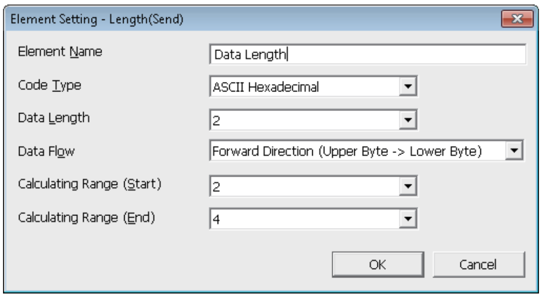
Point

- Only one terminator can be used in a packet.
- Only a check code and static data can be placed after a terminator.

Length

Use this element when an element indicating the data length is included in a packet.

- When sending: Calculates the data length of a specified range, and adds the result to a send packet.
- When receiving: Verifies the data (value) corresponds to the length in the received data as the data length of a specified range.



Element Setting - Length(Send)

Element Name: Data Length

Code Type: ASCII Hexadecimal

Data Length: 2

Data Flow: Forward Direction (Upper Byte -> Lower Byte)

Calculating Range (Start): 2

Calculating Range (End): 4

Item		Description		Remarks
Element Name		Set a name of the element.		—
Code Type		Select a format of the data length. ASCII Hexadecimal/ASCII Decimal/HEX		—
Data Length		Select the data length on the line. The range is 1 to 4.		—
Data Flow		Forward Direction (Upper Byte → Lower Byte)	• When sending: Sends a calculated length from the upper byte. • When receiving: Receives data from the upper byte.	Cannot be set when "Data Length" is 1 byte.
		Reverse Direction (Lower Byte → Upper Byte)	• When sending: Sends a calculated length from the lower byte. • When receiving: Receives data from the lower byte.	
		Byte Swap (by Word)	• When sending: Sends a calculated length swapping the upper byte and lower byte by word. • When receiving: Receives data swapping the upper byte and lower byte by word.	
Calculating Range	Start	Select the start packet element number for the calculating range. The range is 1 to 32.		—
	End	Select the end packet element number for the calculating range. The range is 1 to 32.		

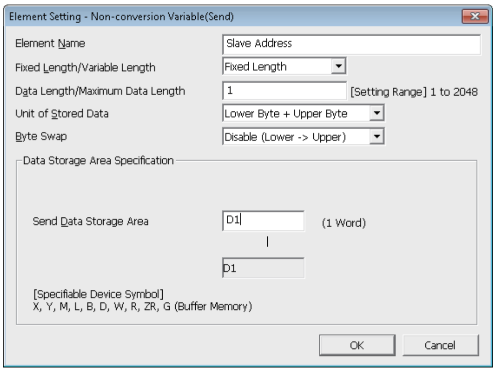
Point

- Only one length can be used in a packet.
- When there is no element other than a length, an element error occurs.
- When the number of digits of calculation result is greater than that specified in "Data Length", digits greater than the specified digit are omitted (ignored).
  Example) When "2 bytes" is specified in "Data Length" and the calculation result is "123 bytes", the data length is considered as "23".
- When any of a Non-conversion variable (Variable length), Conversion variable (Variable number of data), Conversion variable (Fixed number of data/Variable number of digits[*1]), and Non-verified reception (Variable number of characters) is placed behind a Length and they are not included in the calculating range of the Length, place any of the following data immediate after the Non-conversion variable (Variable length), Conversion variable (Variable length) or Non-verified reception.
  Static Data
  Terminator
  Check code + Static Data
  Check code + Terminator
- When "Code Type" is "ASCII Hexadecimal", a corresponding packet is regarded as a mismatch packet if a string except for '0' to '9', 'A' to 'F', and 'a' to 'f' is received.
- When "Code Type" is "ASCII Decimal", a corresponding packet is regarded as a mismatch packet if a string except for '0' to '9' is received.

[*1] Excluding a case where "Number of Send Data" is '1' and "Delimiter" is not 'No delimiter'.

Non-conversion Variable

Use this element to send the data in the device memory of a CPU module or buffer memory as a part of a send packet, or store a part of a receive packet to the device memory of a CPU module or buffer memory.



Element Setting - Non-conversion Variable(Send)

Element Name: Slave Address

Fixed Length/Variable Length: Fixed Length

Data Length/Maximum Data Length: 1 [Setting Range] 1 to 2048

Unit of Stored Data: Lower Byte + Upper Byte

Byte Swap: Disable (Lower -> Upper)

Data Storage Area Specification

Send Data Storage Area: D1 (1 Word)

|

D1

[Specifiable Device Symbol]
X, Y, M, L, B, D, W, R, ZR, G (Buffer Memory)

Item	Description	
Element Name	Set a name of the element.	
Fixed Length/Variable Length	Fixed Length	Sends and receives the data whose length is fixed.
	Variable Length	• When sending: Specifies the data length at the time of the protocol execution and sends data. • When receiving: Receives data whose data length is variable.
Data Length/ Maximum Data Length	Set the length of data to be transmitted and received. (For the variable length data, set the maximum data length that can be specified to the data length storage area.) The range is 1 to 2048.	
Unit of Stored Data	Lower Byte + Upper Byte	• When sending: Sends each one word (2 bytes) data in the data storage area in the order of the lower byte to the upper byte. • When receiving: Stores the received data to the data storage area in the order of the lower byte to the upper byte.
	Lower Bytes Only	• When sending: Sends only lower byte of data in the data storage area. C24 ignores data in the upper byte. • When receiving: Stores the received data in the only lower byte of the data storage area. C24 stores 00H in the upper byte.
Byte Swap	Disable (lower -> upper)/ Enable (upper -> lower)	• When sending: When "Enable" is selected, sends data swapping the upper byte and lower byte by word (2 bytes). When "Unit of Stored Data" is "Lower Byte + Upper Byte" and "Data Length" is an odd number of bytes, sends the upper byte at transmission of the last byte. When "Unit of Stored Data" is "Lower Bytes Only" and "Data Length" is an odd number of bytes, sends data without any byte swap at transmission of the last byte. • When receiving: When 'Enable' is selected, receives data swapping the upper byte and lower byte by word (2 bytes). When "Unit of Stored Data" is "Lower Byte + Upper Byte" and "Data Length" is an odd number of bytes, stores the last byte to the upper byte. When "Unit of Stored Data" is "Lower Bytes Only" and "Data Length" is an odd number of bytes, stores the last byte without any byte swap.
Data Storage Area Specification	Specify a start device to store variable value. ■Settable devices Internal user[*1][*2] • Input (X) • Output (Y) • Internal relay (M) • Latch relay (L) • Link relay (B) • Data register (D) • Link register (W) File register [*2] • File register (R,ZR) Buffer memory • G device (G)	

[*1] Do not set local devices.

[*2] Set within the device range specified in the following screen.
Project window ⇨ [Parameter] ⇨ CPU module type ⇨ [CPU Parameter] ⇨ [Memory/Device Setting]

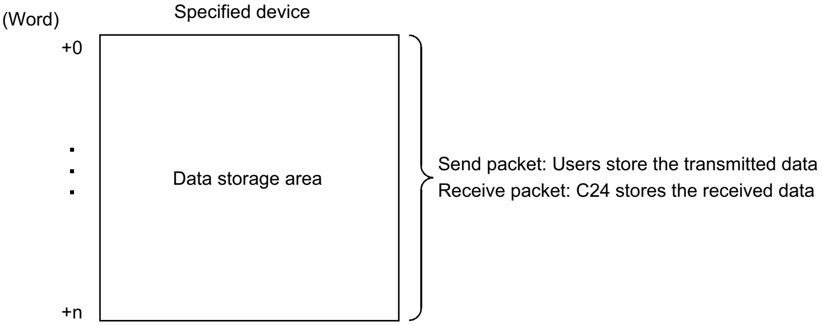
The following explains the configuration of the data storage area.

■When "Fixed Length/Variable Length" is "Fixed Length"

An area starting from the device number which is specified on the "Element Setting" screen is considered as the data storage area.

The data storage area to be occupied varies depending on the setting of "Unit of Stored Data".

- When "Lower Byte + Upper Byte" is selected, the same size as the data length is occupied.
  (However, when the data length of a send packet is an odd number, the upper byte (lower byte for "Byte Swap") of the last device is not sent. When the data length of a receive packet is an odd number, the last data is stored with one byte of 00H added.)
- When "Lower Bytes Only" is selected, twice the size of the data length is occupied.

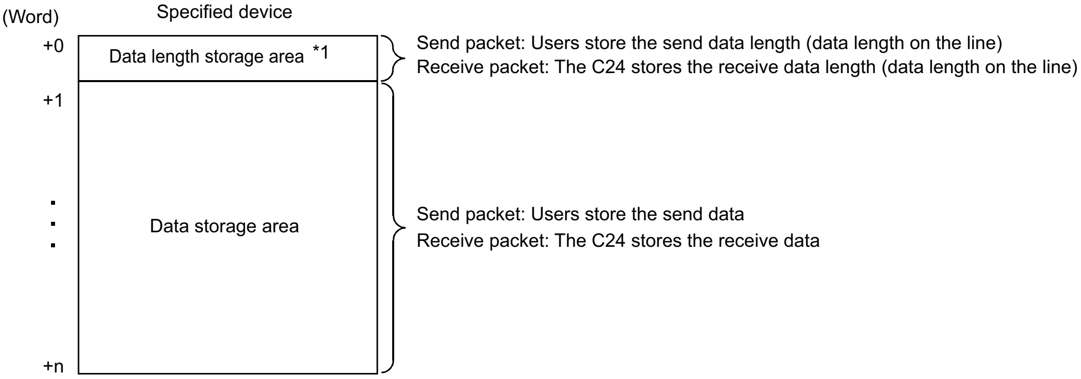

(Word)
Specified device
+0
Data storage area
+n
Send packet: Users store the transmitted data
Receive packet: C24 stores the received data

■When "Fixed Length/Variable Length" is "Variable Length"

An area starting from the device number which is specified on the "Element Setting" screen +1 is considered as the data storage area.

The data storage area to be occupied varies depending on the setting of "Unit of Stored Data".

- When "Lower Byte + Upper Byte" is selected, the same size as the data length + one word (length for the data length storage area) are occupied.
  (However, when the data length of a send packet is an odd number, the upper byte (lower byte for "Byte Swap") of the last device is not transmitted. When the data length of a receive packet is an odd number, one byte of 00H is added to the last data and stored.)
- When "Lower Bytes Only" is selected, twice size of the data length + one word (length for the data length storage area) are occupied.


(Word)
Specified device
+0
Data length storage area *1
+1
Data storage area
+n
Send packet: Users store the send data length (data length on the line)
Receive packet: The C24 stores the receive data length (data length on the line)
Send packet: Users store the send data
Receive packet: The C24 stores the receive data

*1 The unit of data length is byte.

Precautions

● When receiving variable length data whose length exceeds the "Maximum data length", C24 stores data as long as the maximum data length and omits the rest. (A protocol completes successfully.)

● Out of packet data received from target devices, the data corresponding to variable needs to be discriminated from a terminator or a static data immediately after non-conversion variable. The reception processing may not be performed normally if they cannot discriminated.

Ex.

When value of a terminator or value of a static data following a non-conversion variable is used in a non-conversion variable, C24 recognizes it as data of a terminator or a static data following a non-conversion variable and performs the verification/ reception processing.

● Multiple Non-conversion variable (Fixed length) elements can be placed in a packet, and multiple Non-conversion variable (Variable length) elements can also be placed in a send packet. However, only one Non-conversion variable (Variable length) can be placed in a receive packet, and any of the following requirements need to be met.

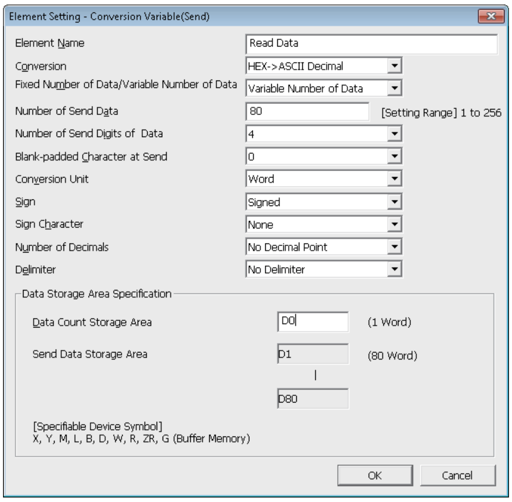
- Place any of the following items immediate after the non-conversion variable.
  Static Data
  Terminator
  Check code + Static Data
  Check code + Terminator
- Place a Length before a Non-conversion variable and include the Non-conversion variable in the calculating range.

In addition, two or more of the following four elements cannot be placed in the same packet.

- Variable number of data [Conversion variable]
- Fixed number of data and variable number of digits [Conversion variable] (Excluding a case where "Number of Send Data" is '1' and "Delimiter" is not 'No delimiter'.)
- Variable length [Non-conversion variable]
- Variable number of characters [Non-verified reception]

Conversion Variable

This element converts the numerical data in the device memory of a CPU module or buffer memory to an ASCII string and sends it, or converts the received data (ASCII string) to the numerical data and stores it to the device memory of a CPU module or buffer memory.



Element Setting - Conversion Variable(Send)

Element Name: Read Data
Conversion: HEX->ASCII Decimal
Fixed Number of Data/Variable Number of Data: Variable Number of Data
Number of Send Data: 80 [Setting Range] 1 to 256
Number of Send Digits of Data: 4
Blank-padded Character at Send: 0
Conversion Unit: Word
Sign: Signed
Sign Character: None
Number of Decimals: No Decimal Point
Delimiter: No Delimiter

Data Storage Area Specification
Data Count Storage Area: D0 (1 Word)
Send Data Storage Area: D1 (80 Word)
|
D80

[Specifiable Device Symbol]
X, Y, M, L, B, D, W, R, ZR, G (Buffer Memory)

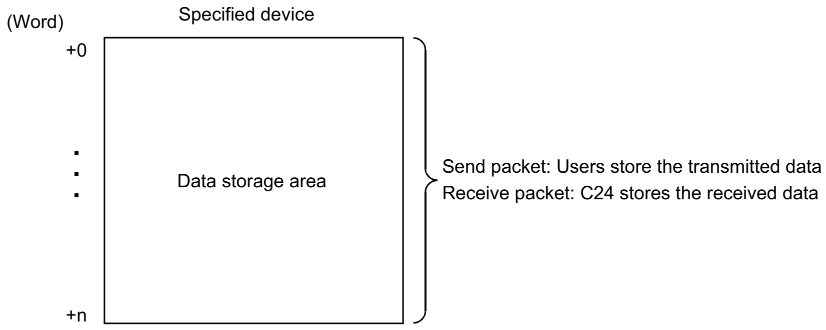
Item	Description		
Element Name	Set a name of the element.		
Conversion	When sending	HEX → ASCII Decimal	Converts numeric value stored in the data storage area to ASCII decimal.
		HEX → ASCII Hexadecimal	Converts numeric value stored in the data storage area to ASCII hexadecimal.
	When receiving	ASCII Decimal → HEX	Treats received data as ASCII decimal, converts it to numeric value, and stores it to the data storage area.
		ASCII Hexadecimal → HEX	Treats received data as ASCII hexadecimal, converts it to numeric value, and stores it to the data storage area.
Fixed Number of Data/ Variable Number of Data	Fixed Number of Data		Fixes the number of data to be sent and received.
	Variable Number of Data		• When sending: Specifies the number of data to be sent at the time of the protocol execution and sends the data. • When receiving: Receives data of which the number is variable. For Variable number of digits, delimiters are required.
Number of Send Data/ Maximum Number of Data	Set the number of data to be transmitted and received. (For "Variable Number of Data", set the maximum number of data that can be specified to the data quantity storage area.) The range is 1 to 256.		
Number of Send Digits of Data	1 to 10		Set the number of digits per one transmitted and received data. When the number of send digits of data is less than the specified number of digits of data, upper digits are filled with blank-padded characters.
	Variable Number of Digits		• When sending: Sends only the data portion converted to an ASCII string in variable length. • When receiving: Receives only an ASCII string of the data portion in variable length. When the number of send data/maximum data length is set to 2 or greater, blank-padded characters are required.
Blank-padded Character at Send	0/Space		Select a character used to fill upper digits when the number of digits of data is not "Variable Number of Digits" and the number of digits of transmitted/received data is less than the specified number of send digits of data.
Conversion Unit	Select how many words are manipulated as one data in the data storage area. For 'Word', converts data manipulating one word as one data. For 'Double word', converts data manipulating two words as one data.		
Sign	Unsigned/Signed		Select whether to add signs to date in the data storage area. This item can be set when "Conversion" is "HEX → ASCII Decimal" or "ASCII Decimal → HEX".
Sign Character	None/+/0/Space		Select the sign character for positive value on line. This item can be set when "Conversion" is "HEX → ASCII Decimal" or "ASCII Decimal → HEX", and "Sign" is "Signed". The sign character for negative value is fixed to "-".
Number of Decimals	No Decimal Point/Variable Point/1 to 9		Select the decimal point position of data on line. This item can be set when "Conversion" is "HEX → ASCII Decimal" or "ASCII Decimal → HEX".

Item	Description	
Delimiter	No Delimiter/Comma/Space	Select the delimiter inserted after one data. A delimiter is not added to the end of data when the number of data is 2 or more.
Data Storage Area Specification	Specify a start device to store variable value.	

The following explains the configuration of the data storage area.

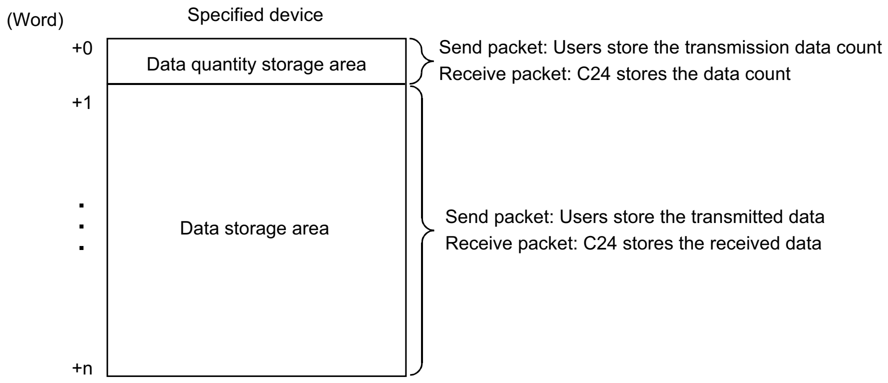
■When "Fixed Number of Data/Variable Number of Data" is "Fixed Number of Data"

An area starting from the device number which is specified on the "Element Setting" screen is considered as the data storage area.


(Word)
Specified device
+0
Data storage area
+n
Send packet: Users store the transmitted data
Receive packet: C24 stores the received data

■When "Fixed Number of Data/Variable Number of Data" is "Variable Number of Data"

An area starting from the device number which is specified on the "Element Setting" screen +1 is considered as the data storage area.


(Word)
Specified device
+0
Data quantity storage area
Send packet: Users store the transmission data count
Receive packet: C24 stores the data count
+1
Data storage area
+n
Send packet: Users store the transmitted data
Receive packet: C24 stores the received data

■Occupied size in data storage area

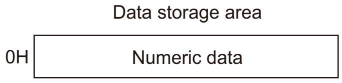
The size occupied in the data storage area varies depending on the settings of "Conversion Unit" and "Number of Decimals".

Setting item		Occupied size in data storage area per one data
Conversion Unit	Number of Decimals	
Word	No Decimal point/Fixed point	1 word
	Variable point	2 words
Double Word	No Decimal point/Fixed point	2 words
	Variable point	4 words

■Data storage area configuration per one data

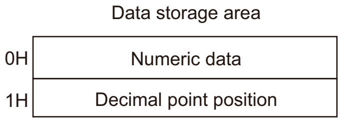
The following shows the data storage area configuration per one data.

●"Conversion Unit": Word, "Number of Decimals": No decimal Point/Fixed point



●"Conversion Unit": Word, "Number of Decimals": Variable point

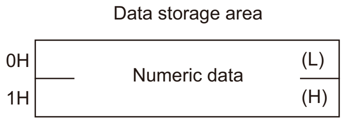
For 'Variable Point', the decimal point position is set in the data storage area.



In the decimal point position area, the decimal point position is set as follows.

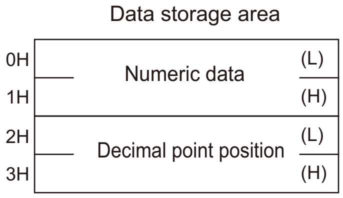
Send/receive data (number of digits of data is 5)	Numeric data	Decimal point position
12345	12345 (3039H)	1 (1H)
1234.5	12345 (3039H)	10 (0AH)
123.45	12345 (3039H)	100 (64H)
12.345	12345 (3039H)	1000 (3E8H)
1.2345	12345 (3039H)	10000 (2710H)

●"Conversion Unit": Double word, "Number of Decimals": No Decimal Point/Fixed point



●"Conversion Unit": Double word, "Number of Decimals": Variable Point

For 'Variable Point', the decimal point position is set in the data storage area.



In the decimal point position area, the decimal point position is set as follows.

Send/receive data (number of digits of data is 10)	Numeric data	Decimal point position
1234567890	1234567890 (499602D2H)	1 (1H)
123456789.0	1234567890 (499602D2H)	10 (0AH)
12345678.90	1234567890 (499602D2H)	100 (64H)
1234567.890	1234567890 (499602D2H)	1000 (3E8H)
:	:	:
1.234567890	1234567890 (499602D2H)	1000000000 (3B9ACA00H)

■Range of value that can be used in the data storage area

The following table shows the range of value that can be used in the data storage area.

Conversion	Sign	Conversion unit	Range of value
HEX →ASCII decimal ASCII decimal → HEX	Unsigned	Word	0 to 65535 (0H to FFFFH)
		Double word	0 to 4294967295 (0H to FFFFFFFFH)
	Signed	Word	-32768 to 32767 (8000H to FFFFH, 0H to 7FFFH)
		Double word	-2147483648 to 2147483647 (80000000H to FFFFFFFFH, 0H to 7FFFFFFFH)
HEX → ASCII hexadecimal ASCII hexadecimal → HEX	—	Word	0H to FFFFH
		Double word	0H to FFFFFFFFH

■Considerations for configuring packet element

To place a Conversion variable in a packet, the following requirements need to be met.

- To place Conversion variable in send packet

  Multiple Conversion variable elements can be placed in one packet, and they can be placed in desired positions in the data portion.
- To place Conversion variable in receive packet

  Multiple Conversion variable elements can be placed in one packet in the cases other than "Variable number of data" and (3) of "Fixed number of data".

Variable number of data (Only one Conversion variable can be placed in one packet, and either of the following (1) or (2) need to be met.)		(1)	■For discriminating the data length of a Conversion variable, any of the following items is placed immediate after the Conversion variable. • Static data • Terminator • Check code + Static data • Check code + Terminator
		(2)	■A "Length" is placed before a Conversion variable. (The Conversion variable needs to be included in the calculating range.)
Fixed number of data	Variable number of digits	(3)	■When the number of data is 2 or more, or the number of data is 1 with no delimiter, only one Conversion variable can be placed in a packet and it needs to be placed in the order mentioned in [Variable number of data].
		(4)	■When a Conversion variable (the number of data is 1, with delimiter, and variable number of digits) and the following four elements are placed in the same packet, these four elements need to be placed behind the Conversion variable (the number of data is 1, with delimiter, and variable number of digits). • "Conversion Variable" (variable number of data) • Conversion variable (fixed number of data and variable number of digits) (The case of (3). Excluding a case where "Number of Send Data" is '1' and "Delimiter" is not "No Delimiter".) • Non-conversion variable (variable length) • Non-verified reception (variable number of characters) Note that two or more of these four elements cannot be placed in the same packet.
	Fixed number of digits (1 to 10)	(5)	■When a Conversion variable (variable decimal point) and the four elements mentioned in (4) are placed in the same packet, these elements need to be placed behind the Conversion variable (variable decimal point).
		(6)	■When a Conversion variable (unsigned) and the four elements mentioned in (4) are placed in the same packet, these elements need to be placed behind the Conversion variable (unsigned).

Precautions

An error may occur on C24 in any of the following cases.

- A string other than '0'-'9', 'A'-'F', or 'a'-'f' is received

  When "Conversion" is "ASCII Hexadecimal→HEX", an ASCII → BIN conversion error (7F20H) may occur if a string other than '0'-'9', 'A'-'F', or 'a'-'f' is received.
- A string other than '0'-'9' is received

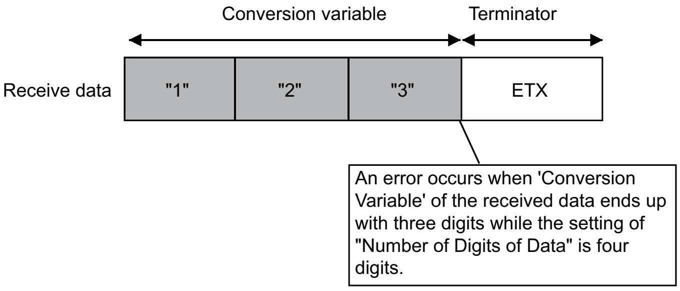
  When "Conversion" is "ASCII Decimal→HEX", an ASCII → BIN conversion error (7F20H) may occur if a string other than '0'-'9' is received. However, the error does not occur in the following cases.

Item	C24 operation
Sign/Sign Character	When "Signed" is selected, a signed character can be received. However, when a signed character which is not at the head of data is received, an ASCII → BIN conversion error (7F20H) may occur.
Number of Decimals	When any other than "No decimal point" is selected, '. ' (a period) can be received. However an ASCII → BIN conversion error (7F20H) may occur when '. ' (a period) which is not at the specified number of digits of data is received. An ASCII → BIN conversion error (7F20H) may also occur when "Variable point" is selected and '. ' (a period) which is at the head or end of a unit of data is received.
Delimiter	When any other than "No delimiter" is selected, a delimiter can be received. However an ASCII → BIN conversion error (7F20H) may occur when a delimiter which is not at a boundary of data is received.

- A data of which number of digits exceeds the upper limit is received
  When "Number of Send Digits of Data" is "Variable Number of Digits", an abnormal digit number error (7D19H) may occur if the number of digits of received data exceeds the upper limit shown below.

Conversion unit	Conversion	Upper limit of number data to be received
Word	ASCII Decimal → HEX	5 digits
	ASCII Hexadecimal → HEX	4 digits
Double word	ASCII Decimal → HEX	10 digits
	ASCII Hexadecimal → HEX	8 digits

- A data whose decimal point position is greater than the number of digits is specified
  When "Number of Decimals" is 'Variable Point' in a send packet, a decimal point place specification error (7D21H) may occur if the decimal point position is greater than the number of digits.
- A data whose number of digits is '0' is received
  When "Number of Send Digits of Data" is "Variable Number of Digits", an abnormal digit number error (7D19H) may occur if the number of digits of received data is '0'.
- A data whose number of digits are fewer than the digits specified in "Number of Send Digits of Data" are received
  An insufficient digit number error (7D18H) may occur if data whose number of digits are fewer than the digits specified in "Number of Send Digits of Data" are received.
  Example) When "Number of Send Digits of Data" is set to 4 digits



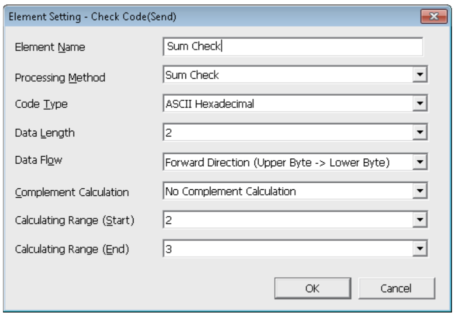
An error occurs when 'Conversion Variable' of the received data ends up with three digits while the setting of "Number of Digits of Data" is four digits.

- A data whose number exceeds "Maximum number of data" is received
  When receiving data of which the number exceeds the "Maximum number of data", C24 stores data as many as the maximum number of data and omits the rest. (A protocol completes successfully.)
- A data that cannot be distinguished between Terminator or a Static data
  In received packet data from target devices, C24 needs to be able to discriminate data corresponding to a Conversion variable from those of a Terminator or a Static data following a Conversion variable.
  The reception processing may not be performed normally if they cannot be distinguished.
  Example) When the value of Terminator or a Static data following the Conversion variable is used as a Conversion variable, C24 recognized it as a fixed data following the Conversion variable, verify, and perform reception processing.

Check Code

Use this element when an element indicating check code data is included in a packet.

C24 automatically calculates a specified check code at timing of sending/receiving, and adds it to a send packet or detects an error of a receive packet.



Element Setting - Check Code(Send)

Element Name: Sum Check

Processing Method: Sum Check

Code Type: ASCII Hexadecimal

Data Length: 2

Data Flow: Forward Direction (Upper Byte -> Lower Byte)

Complement Calculation: No Complement Calculation

Calculating Range (Start): 2

Calculating Range (End): 3

Item		Description		Remarks
Element Name		Set a name of the element.		—
Processing Method		Select a calculating method. Horizontal parity/Sum check/16-bit CRC (for MODBUS)		—
Code Type		ASCII Hexadecimal ASCII Decimal HEX	• When sending: Select a format in which a calculated check code is sent. • When receiving: Select a format in which data are received.	Not settable when "Processing Method" is "16-bit CRC (for MODBUS)"
Data Length		Select the data length on the line (byte). The range is 1 to 4.		Not settable when "Processing Method" is "16-bit CRC (for MODBUS)"
Data Flow		Forward Direction (Upper byte → Lower byte)	• When sending: Sends a calculated check code, from the upper byte. • When receiving: Handles received data as a check code, from the upper byte.	Not settable when "Processing Method" is "16-bit CRC (for MODBUS)", or when the "Data Length" is set to "1 (byte)".
		Reverse Direction (Lower byte → Upper byte)	• When sending: Sends a calculated check code, from the upper byte. • When receiving: Receives data from the lower byte. Effective when the data length is 2 to 4 (bytes).	
		Byte Swap (by word)	• When sending: Sends a calculated check code by swapping the upper byte and lower byte by word. When receiving: Receives data by swapping the upper byte and lower byte by word, and handles as a check code. Effective when the data length is 4 bytes.	
Complement Calculation		Select the complement calculation. No complement calculation/One's complement/Two's complement		Not settable when "Processing Method" is "16-bit CRC (for MODBUS)"
Calculating Range	Start	Select the start packet element number for the calculating range. The range is 1 to 32.		—
	End	Select the end packet element number for the calculating range. The range is 1 to 32.		

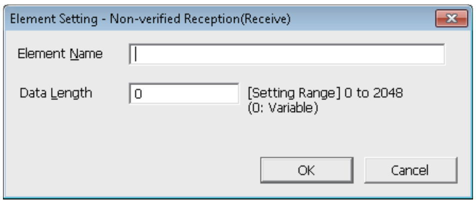
Precautions

- When "Code Type" is "ASCII Hexadecimal", an ASCII → BIN conversion error (7F20H) may occur if a string except for '0' to '9' 'A' to 'F' 'a' to 'f' is received.
- When "Code Type" is "ASCII Decimal", an ASCII → BIN conversion error (7F20H) may occur if a string except for '0' to '9' is received.
- Only one check code can be placed in a packet.
- Packet elements placed behind a check code cannot be included in the calculating range.
- A check code cannot be included in the calculated range.
- When a calculated check code (Sum check/Horizontal parity/16-bit CRC) does not match a received check code, a sum check error (7F24H) may occur.
- One or more elements are required before a check code.

Non-verified reception

Use this element when received data includes unnecessary data while receiving data.

C24 skips characters as many as the specified number if a receive packet includes a non-verified reception.



Element Setting - Non-verified Reception(Receive)

Element Name

Data Length: 0

[Setting Range] 0 to 2048
(0: Variable)

Item	Description	
Element Name	Set a name of the element.	
Data Length	0 (variable number of characters)	Set this item when the number of characters not to be verified varies in each of communication.
	1 to 2048 (specified number of characters)	Set the number of characters not to be verified.

Precautions

- When "Data Length" is '0', the following requirements need to be met.

■Only one Non-verified reception can be placed in a packet.

■Include a Non-verified reception in the calculating range of a Length. Otherwise, place any of the following data immediate after a Non-verified reception.

- Static Data
- Terminator
- Check code + Static Data
- Check code + Terminator

■Two or more of the following four elements cannot be placed in the same packet.

- "Conversion Variable" (variable number of data)
- Conversion variable (Fixed number of data and Variable number of digits) (Excluding a case when the number of data is '1' with delimiter.)
- Non-conversion variable (Variable length)
- Non-verified reception (Variable number of characters)

- In received packet data from target devices, C24 needs to be able to discriminate data corresponding to a Non-verified reception from those of a Terminator or a Static Data following a Non-verified reception. The receive processing may not be performed normally if they cannot be distinguished.

Ex.) When value of a Terminator or value of a Static Data following a Non-verified reception is used in a Non-verified reception, C24 recognizes it as data of a Terminator or a Static Data following a Non-verified reception and performs the verify/receive process.

2.4 Execution Condition of Predefined Protocol Communication

The predefined protocol communication can be performed when the 'predefined protocol ready' (X1D) is ON.

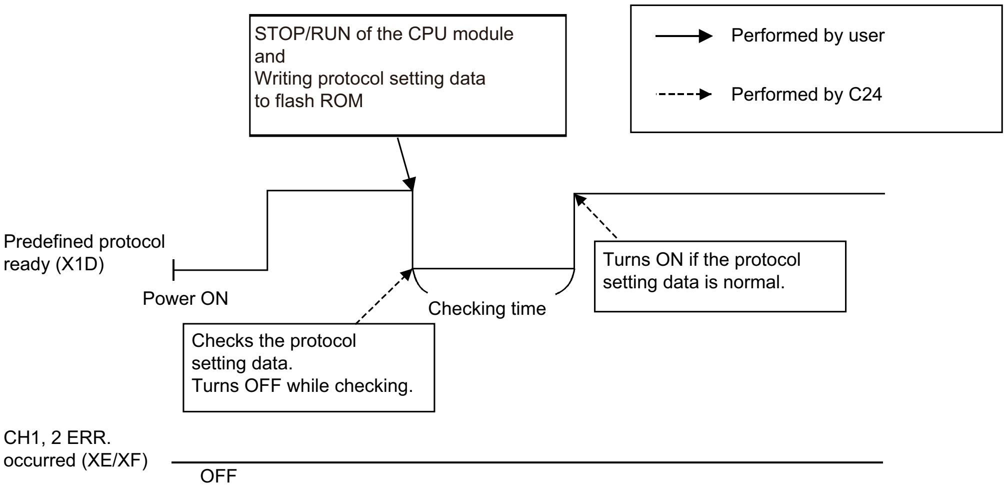
When executing the predefined protocol communication in the program, use the 'predefined protocol ready' (X1D) as an interlock signal.

This signal turns ON only when the communication protocol setting is set to predefined protocol

ON/OFF timing of predefined protocol ready (X1D)

After the protocol setting data is written to a flash ROM, 'predefined protocol ready' (X1D) turns OFF while checking the written data.

When the written data is normal, 'predefined protocol ready' (X1D) turns ON.



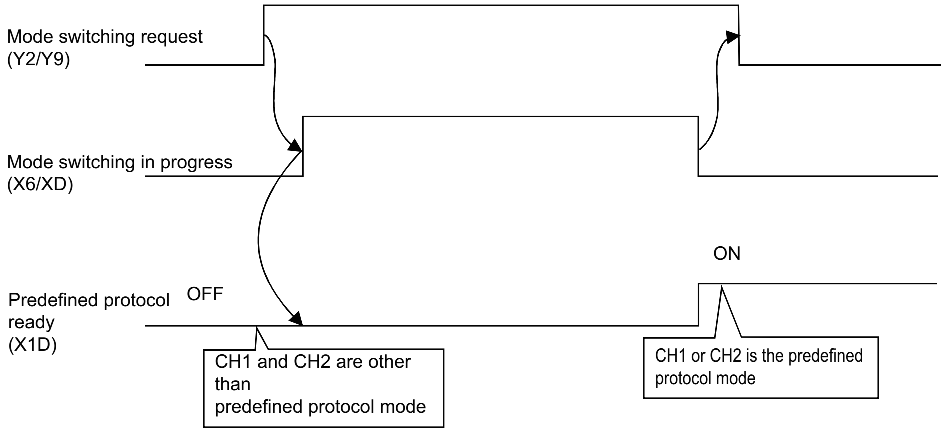
Execution timing for mode switching request signal (Y2/Y9) or UINI instruction

It is assumed that the protocol setting data written to the flash ROM is normal, or the protocol setting data is not written to the flash ROM.

■Execution timing for mode switching request signal (Y2/Y9)

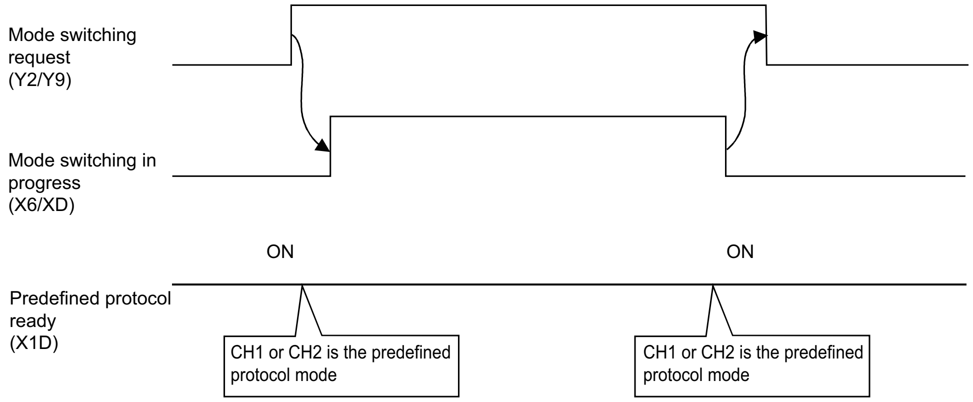
Communication protocol before change: CH1 and CH2 are other than the predefined protocol mode.

Communication protocol after change: CH1 or CH2 is the predefined protocol mode.


Mode switching request
(Y2/Y9)
Mode switching in progress
(X6/XD)
ON
Predefined protocol
ready
(X1D)
OFF
CH1 and CH2 are other than predefined protocol mode
CH1 or CH2 is the predefined protocol mode

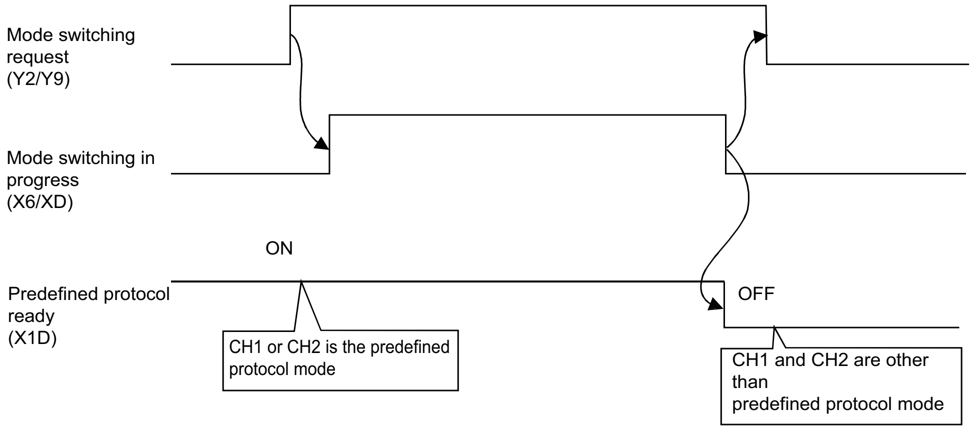
Communication protocol before change: CH1 or CH2 is the predefined protocol mode.

Communication protocol after change: CH1 or CH2 is the predefined protocol mode.


Mode switching request
(Y2/Y9)
Mode switching in progress
(X6/XD)
ON
ON
Predefined protocol ready
(X1D)
CH1 or CH2 is the predefined protocol mode
CH1 or CH2 is the predefined protocol mode

Communication protocol before change: CH1 or CH2 is the predefined protocol mode.

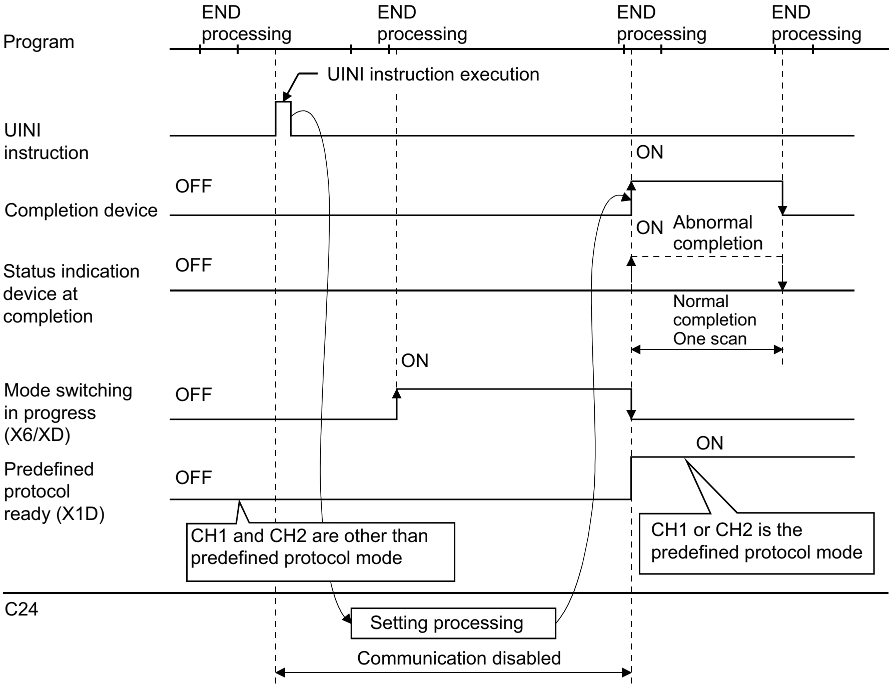
Communication protocol after change: CH1 and CH2 are other than the predefined protocol mode.


Mode switching request
(Y2/Y9)
Mode switching in progress
(X6/XD)
ON
Predefined protocol ready
(X1D)
OFF
CH1 or CH2 is the predefined protocol mode
CH1 and CH2 are other than predefined protocol mode

■Execution timing for UINI instruction

Communication protocol before change: CH1 and CH2 are other than the predefined protocol mode.

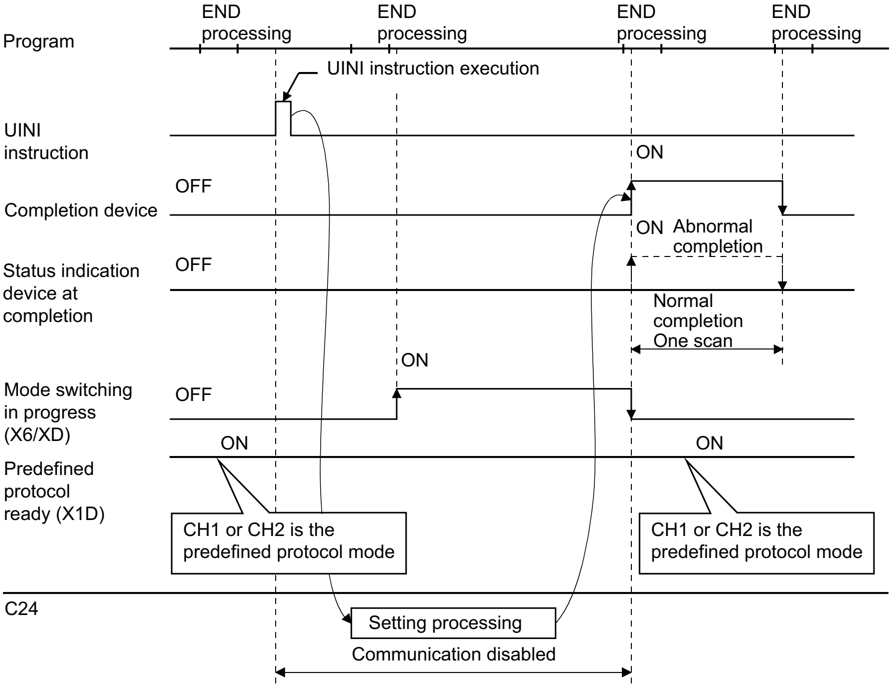
Communication protocol after change: CH1 or CH2 is the predefined protocol mode.



Program
END processing
END processing
END processing
END processing
UINI instruction execution
UINI instruction
ON
OFF
Completion device
ON
Abnormal completion
Status indication device at completion
OFF
Normal completion
One scan
ON
Mode switching in progress (X6/XD)
OFF
ON
Predefined protocol ready (X1D)
OFF
CH1 and CH2 are other than predefined protocol mode
CH1 or CH2 is the predefined protocol mode
C24
Setting processing
Communication disabled

Communication protocol before change: CH1 or CH2 is the predefined protocol mode.

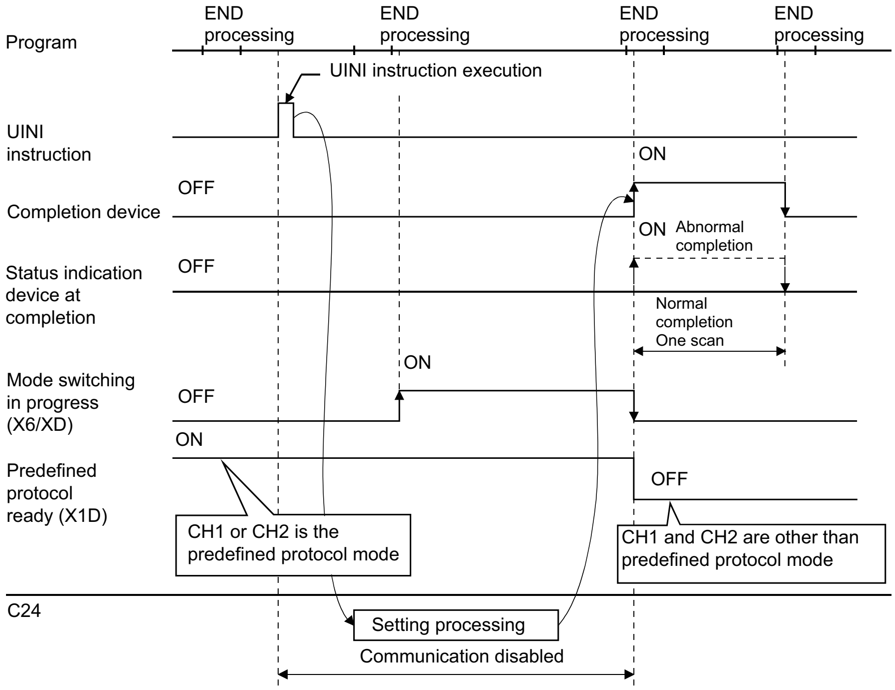
Communication protocol after change: CH1 or CH2 is the predefined protocol mode.



Program
END processing
END processing
END processing
END processing
UINI instruction execution
UINI instruction
ON
OFF
Completion device
ON
Abnormal completion
Status indication device at completion
OFF
Normal completion
One scan
ON
Mode switching in progress (X6/XD)
OFF
ON
ON
Predefined protocol ready (X1D)
CH1 or CH2 is the predefined protocol mode
CH1 or CH2 is the predefined protocol mode
C24
Setting processing
Communication disabled

Communication protocol before change: CH1 or CH2 is the predefined protocol mode.

Communication protocol after the change: CH1 and CH2 are other than the predefined protocol mode.

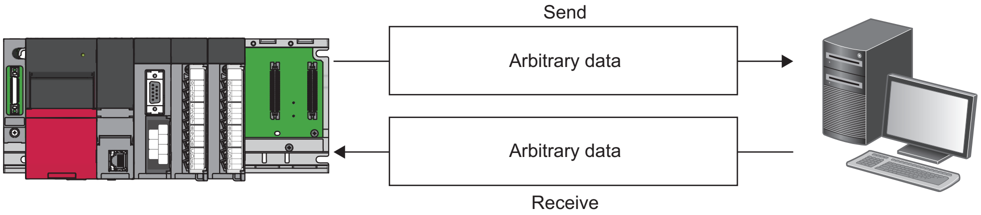

Program
END processing
END processing
END processing
END processing
UINI instruction execution
UINI instruction
Completion device
OFF
ON
Status indication device at completion
OFF
ON
Abnormal completion
Normal completion
One scan
Mode switching in progress (X6/XD)
OFF
ON
Predefined protocol ready (X1D)
ON
OFF
CH1 or CH2 is the predefined protocol mode
CH1 and CH2 are other than predefined protocol mode
C24
Setting processing
Communication disabled

3 DATA COMMUNICATION USING NONPROCEDURAL PROTOCOL

Nonprocedural protocol enables to transmit and receive data between a CPU module and target device by setting a data format and transmission control procedure arbitrarily.

A program for data communication is required for the CPU module.

Use this protocol for data communication with any given data format.



Point

The predefined protocol is recommended for data communication appropriate to the protocol of the target device, such as a measuring instrument or barcode reader.

Communication using the nonprocedural protocol can be replaced with that of the predefined protocol.

The predefined protocol support function of Engineering tool allows the easy replacement.

With the predefined protocol, the required program is only for starting communication according to the specified protocol, and this can reduce the number of steps considerably.

For the communication using the predefined protocol, refer to the following chapter.

☞ Page 30 DATA COMMUNICATION USING PREDEFINED PROTOCOL

Parameter settings of data communication using nonprocedural protocol

Set the transmission setting and the communication protocol, etc. for a target device.

Set the "Nonprocedural protocol" for "Communication protocol setting".

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

Configure other settings according to the system being used.

For details of the settings, refer to the following chapter.

☞ Page 454 PARAMETER SETTING

3.1 Receiving Data from Target Device

This section explains data reception from a target device.

Receiving methods

The following shows the methods for receiving data in any format using the nonprocedural protocol.

There are two methods for receiving data: the 'reception by receive end code' for receiving variable length messages, and the 'reception by receive end data quantity' for receiving fixed length messages. If the receive end code and the receive end data quantity are not determined, the 'reception by time out for the no-reception monitoring time (timer 0)' is also available.

The receive end code and receive end data quantity used for data reception can be changed to any setting values by the user using Engineering tool.

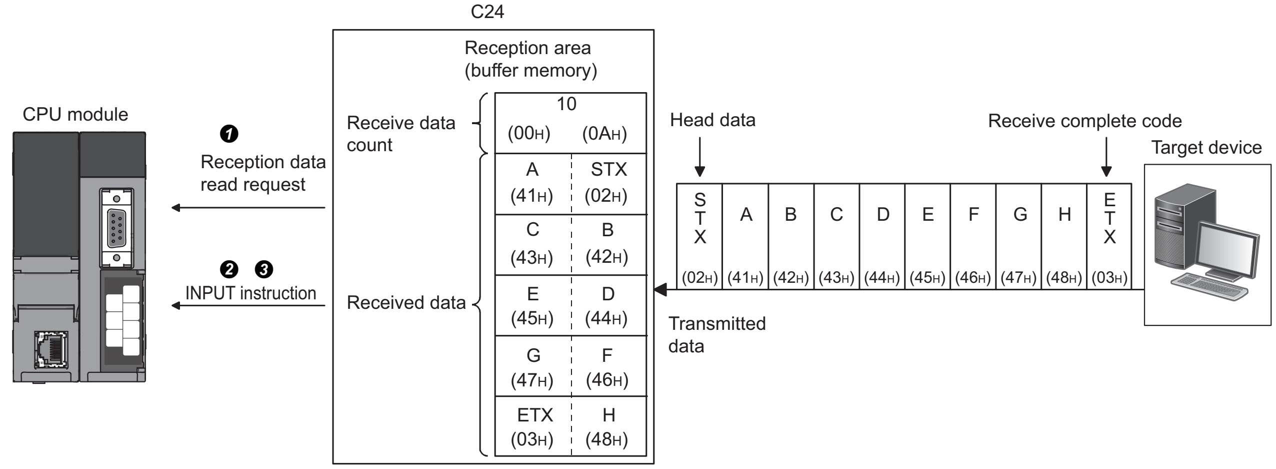
C24 turns the 'CH□ reception data read request' (X3/XA) ON when it receives the receive end code before data equivalent to the receive end data quantity has been received.

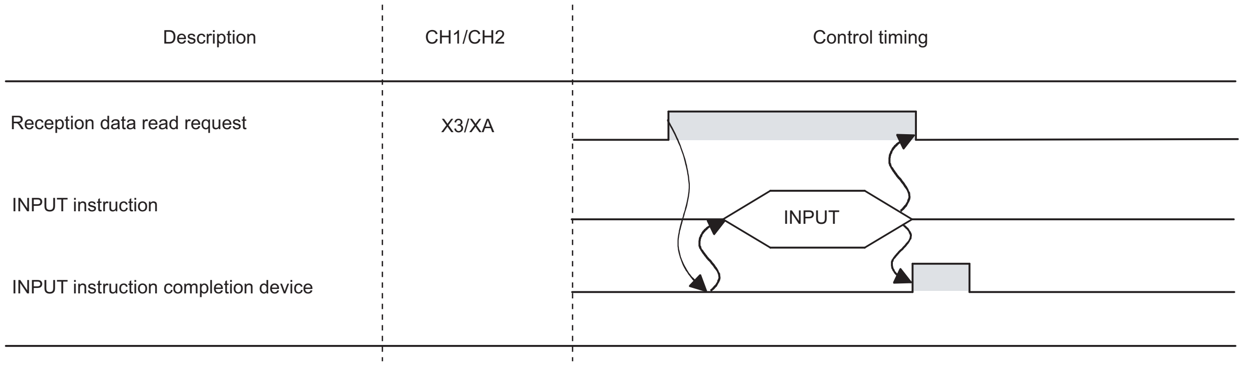
Therefore, when receiving data using the receive end code, set a value which is bigger than the maximum number of data sent from the target device in the receive end data quantity designation.

For the changing methods for the receive end code and receive end data quantity, refer to the following section.

☞ Page 77 Receive end data quantity and receive end code settings

Method	Application
Reception by the receive end code	Receives variable length messages
Reception by receive end data quantity	Receives fixed length messages.
Reception by time out for the no-reception monitoring time (timer 0)	Receives messages the receive end code and the receive end data quantity of which are not determined. For specifying a no-reception monitoring time (timer 0), refer to the following section. ☞ Page 260 No-reception Monitoring Time (timer 0) Setting




Description
CH1/CH2
Control timing
Reception data read request
X3/XA
INPUT instruction
INPUT
INPUT instruction completion device

❶ When data is received from the target device using either 'reception by receive end code' or 'reception by receive end data quantity' method, the CH□ reception data read request' (X3/XA) turns ON.

❷ Control data is stored in the device designated with the INPUT instruction.

❸ When the INPUT instruction is executed, received data is read from the receive data storage area of the buffer memory.

Data reception by receive end code (for variable length reception)

This method is used to transmit data by adding the receive end code set in C24 at the end of a message transmitted from a target device.

The receive end code can be changed according to the specifications of the target device. Arbitrary 1-character (1-byte) code within the range of 00H to FFH can be designated.

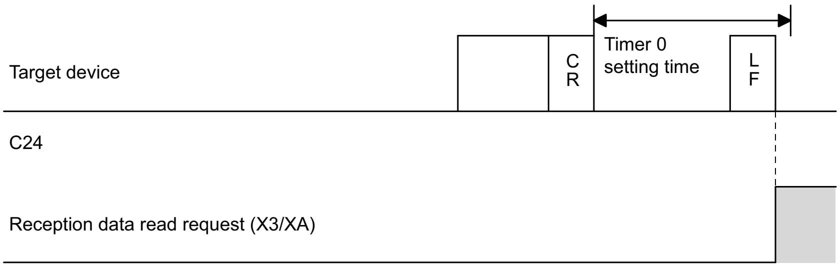
Setting	Description	Remarks
0D0AH (default)	The receive end code is not changed.	CR, LF is added to the end of the message and sent from the target device.
□□H	Arbitrary code is designated for the receive end code. (00H to FFH)	The designated receive end code data is added to the end of the message and sent from the target device.
FFFFH	No receive end code	Data reception by receive end data quantity is available.

- When C24 receives the receive end code data preset with Engineering tool by the user, it issues a reception data read request to the program.
  The read request from C24 allows the program to read the received data up to the receive end code received from the target device.
- C24 issues a reception data read request to the CPU module in the cases below, according to the setting value of the receive end code.

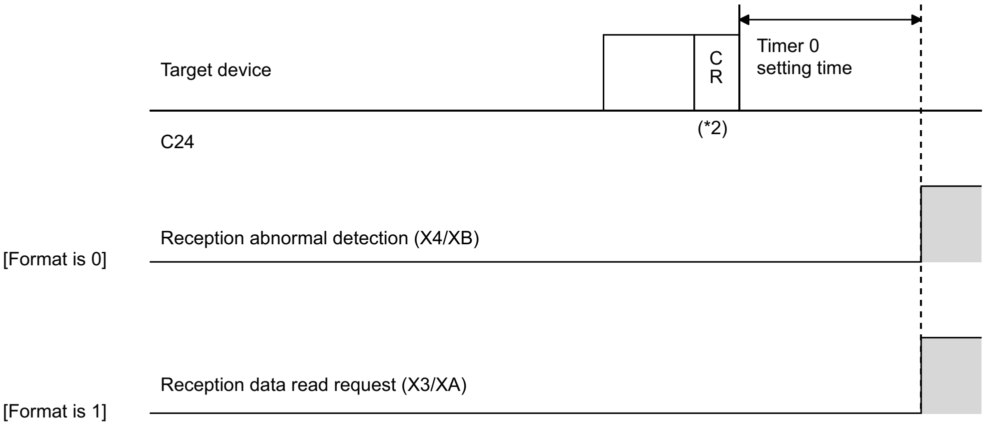
■When receive end code is not changed (Default value: 0D0AH)

(CR+LF is added to the end of the message and sent from the target device.)

- When LF is received within the time set in the no-reception monitoring time (timer 0) after CR is received, C24 stores the receive data up to CR+LF to the receive data storage area in the buffer memory and turns ON the 'CH□ reception data read request' (X3/XA) to the CPU module.



- If LF is not received within the time set in the no-reception monitoring time (timer 0) after CR is received, C24 stores the received data up to CR to the receive data storage area on the buffer memory, and turns ON the following signals to the CPU module.
  Nonprocedural and no-reception monitoring time
  Method 0 :'CH□ reception abnormal detection' (X4/XB)
  Method 1 :'CH□ reception data read request' (X3/XA)[*1]

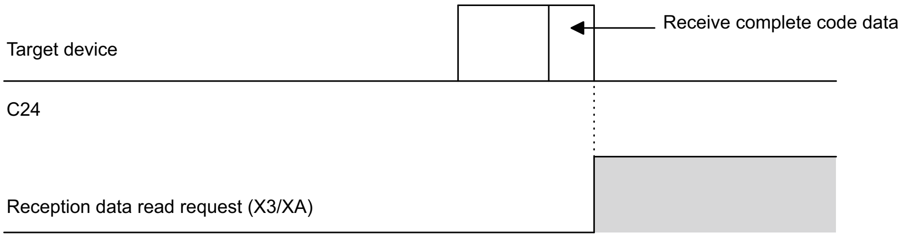

Target device
C
R
Timer 0
setting time
C24
(*2)
Reception abnormal detection (X4/XB)
[Format is 0]
Reception data read request (X3/XA)
[Format is 1]

*1 It is used when a message of which receive end code and receive end data quantity have not been determined due to the non-reception monitoring time (timer 0) time out in the nonprocedural protocol. (☞ Page 261 Nonprocedural no-reception monitoring time method designation (Method 0/1))

*2 CR is handled as 1-byte data in the message.

■When the receive end code is changed and an arbitrary code is designated (designate from 0000H to 00FFH)

(User defined receive end code data is added to the end of the message and transmitted from the target device).
When C24 receives the receive end code data changed by the user, it stores the received data up to the receive end code in the receive area on the buffer memory and turns ON the 'CH□ reception data read request' (X3/XA) to the CPU module.


Receive complete code data
Target device
C24
Reception data read request (X3/XA)

■When the receive end code is not designated (FFFFH designated)

Reading by the receive end data quantity is enabled.
For more information on reading by the receive end data quantity, refer to the following section.
☞ Page 59 Data reception by receive end data quantity (for fixed length reception)

Data reception by receive end data quantity (for fixed length reception)

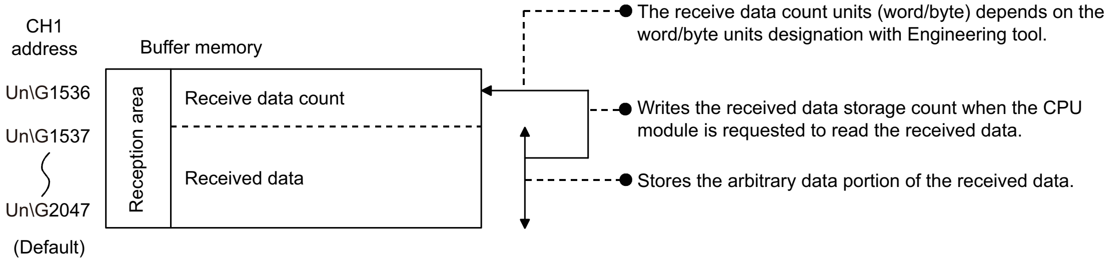
- This method is used to receive messages of the same length (size) from the target device every time.
- When C24 receives data equivalent to the receive end data quantity preset with Engineering tool by the user, it issues a reception data read request to the program.
  Upon the read request by the C24, the program reads the data equivalent to the receive end data quantity from the target device.
- The receive end data quantity default value is set to 511 (words). However, it can be changed within the receive data storage area depending on the data contents exchanged with the target device.

Receive area and receive data arrangement

The following shows the receive area and the receive data arrangement for performing data reception using the nonprocedural protocol.

Receive area

The receive area is a memory area for storing the data received from the target device, and the receive data quantity to read the received data to the CPU module.
By default, the receive area is assigned to the addresses, Un\G1536 to 2047 (CH1 side) and Un\G2560 to 3071 (CH2 side).


CH1
address
Buffer memory
Un\G1536
Un\G1537
Un\G2047
(Default)
Reception area
Receive data count
Received data
The receive data count units (word/byte) depends on the word/byte units designation with Engineering tool.
Writes the received data storage count when the CPU module is requested to read the received data.
Stores the arbitrary data portion of the received data.

The position and size of the receive area in the buffer memory can be changed with Engineering tool in accordance with the specifications of the target device and the receive data length. (☞ Page 454 PARAMETER SETTING)
Configure the settings as follows:

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

- Designate the start address for the area to be used as receive area in the user setting area (Un\G1024 to 6911, Un\G9728 to 16383) to "Receiving buffer memory start address designation".
- Designate the length of area to be used as the receive area in the user setting area (Un\G1024 to 6911, Un\G9728 to 16383) by address count (0001H to 1A00H) to "Receiving buffer memory size specification".

When changing the position and size of the receive area in the buffer memory, if any of the functions below are used concurrently, make sure that the addresses of the receive area do not overlap with the buffer memory addresses assigned for storing send/receive data with the functions.

- MC protocol buffer memory read/write function
- MC protocol on-demand function
- Predefined protocol function (Send/receive data storage area)
- Nonprocedural protocol transmission/reception function
- Bidirectional protocol transmission/reception function
- Send/receive data monitoring function

Set the size of data per data transmission from the target device to C24 to be equal to or smaller than the size of the receive data storage area in the buffer memory.

- (Receive data storage area) ≥ (Amount of data in any data portion to be transmitted from the target device)

To transmit data whose size is larger than the receive data storage area, increase the receive area, or divide the send data before sending.

Point

The position and size of the receive area can be changed in the buffer memory.

- 'Receiving buffer memory start address designation' (Un\G166/326)
- 'Receive buffer memory length designation' (Un\G167/327)

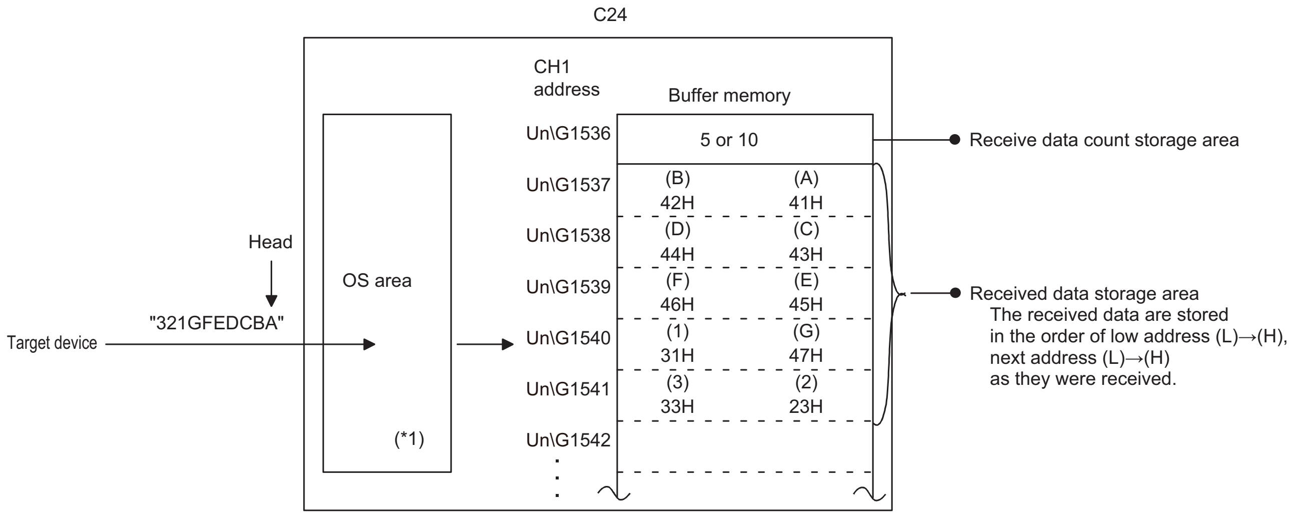
Receive data arrangement

The following explains the data arrangement when data received from a target device is stored to the receive area.

- The received message is stored to the C24 buffer memory (Receive data storage area).
- The data is stored to the receive data storage area in the order of data reception from (L)→(H) of the low address to (L)→(H) of the next address.
  Under the following conditions, when the receive end data quantity is an odd byte, 00H is stored in the upper byte of the last data storage position:
  When the unit for the receive end data quantity is designated in bytes.
  When data reception is performed using the receive end code.

Ex.

When receive data "ABCDEFG123" was stored (The receive area is the default value.)



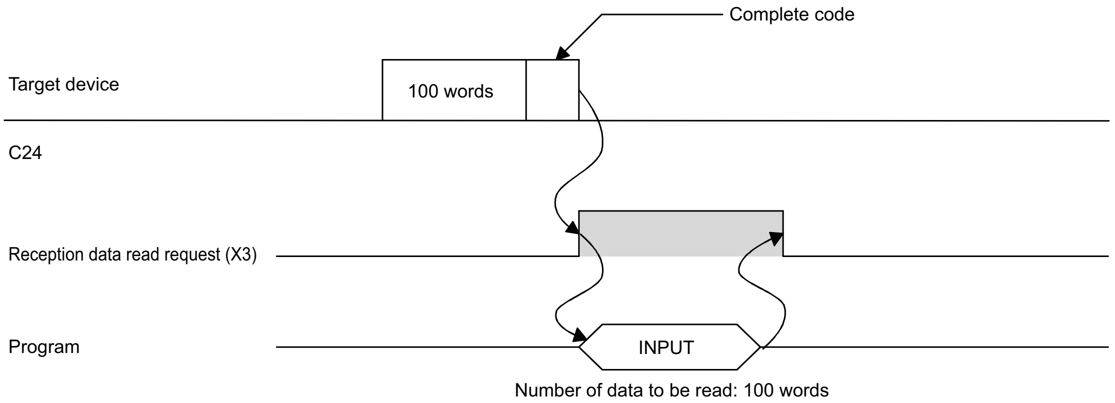
*1 The OS area of C24 is the memory (8448 bytes) that temporarily stores the data to be received while a reception data read request is being issued to the CPU module. (Reading received data in the OS area cannot be performed by the user.)
When the program finishes reading the received data in the buffer memory in response to the current read request, the received data in the OS area, and any succeeding received data, is stored successively in the receive area in the buffer memory when the next read request is issued.
Also, when the free OS area, which stores the received data, is reduced to 64 bytes (default value) or less, a request to discontinue data transmission from the target device is issued by the following transmission control (The RS (RTS) signal does not turn OFF):
·When DTR control is set, the ER(DTR) signal turns OFF.
·When DC1/DC3 control is set, DC3 is sent.
When there is no more free OS area to store received data, SIO error occurs and the SIO information bit of the 'CH□ side LED lighting status and communication error status' (Un\G513/514) turns ON. In that case, succeeding received data is discarded until a free area becomes available in the OS area.
For more details on the transmission control, refer to the following section.
☞ Page 268 DATA COMMUNICATIONS USING DC CODE TRANSMISSION CONTROL

Reception operation

■Data reception when the receive data storage area > receive data length

Ex.

When data of 511 words or less is received by the CH1 side interface on the condition that the receive area for the CH1 interface is Un\G1536 to 2047 (default value)

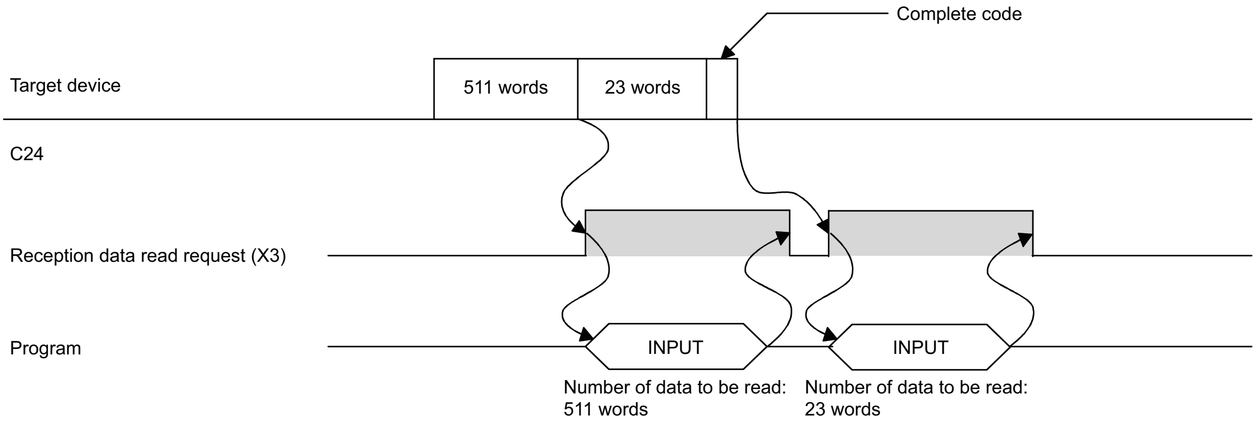

Complete code
Target device
100 words
C24
Reception data read request (X3)
Program
INPUT
Number of data to be read: 100 words

■Data reception when the receive data storage area < receive data length (Reception using the receive end code)

The data link processing is required in the program.

Ex.

When data of 534 words is received by the CH1 side interface on the condition that the receive area for the CH1 interface is Un\G1536 to 2047 (default value).

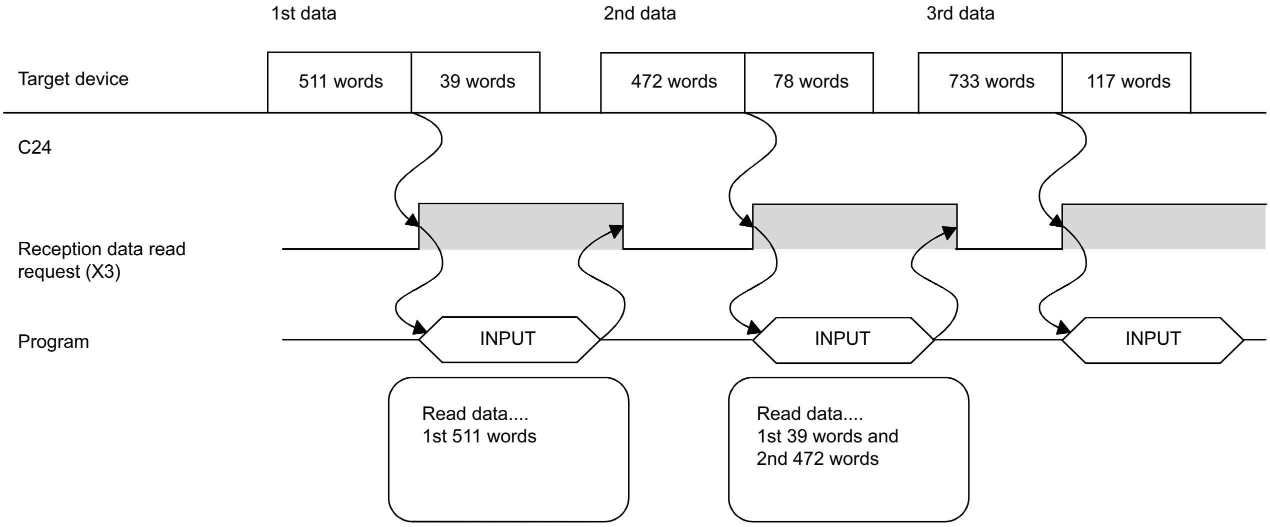

Complete code
Target device
511 words
23 words
C24
Reception data read request (X3)
Program
INPUT
INPUT
Number of data to be read: 511 words
Number of data to be read: 23 words

■Data reception when the receive data storage area < receive data length (Reception by receive end data quantity)

If a specified receive end data quantity is bigger than the receive data storage area, the 'receive buffer memory length designation' (Un\G167/327, default: 512) minus 1 is used as the receive end data quantity instead.
(Receive data storage area) > (Data length transmitted from the target device)

Ex.

When 550 words of data is received by the CH1 interface three times on the condition that the receive area for the CH1 interface is Un\G1536 to 2047 (default value).

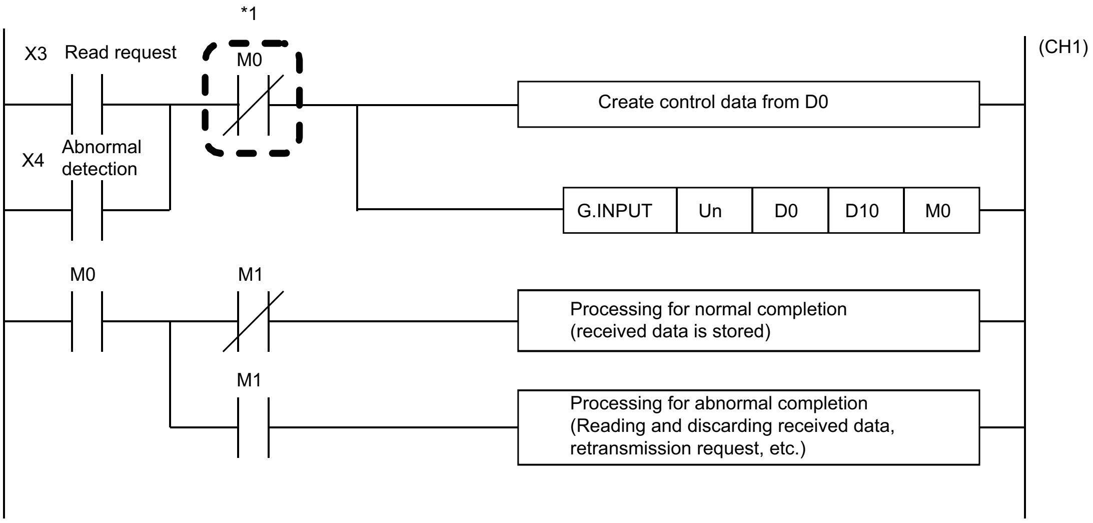


Program for data reception

Program example for data reception

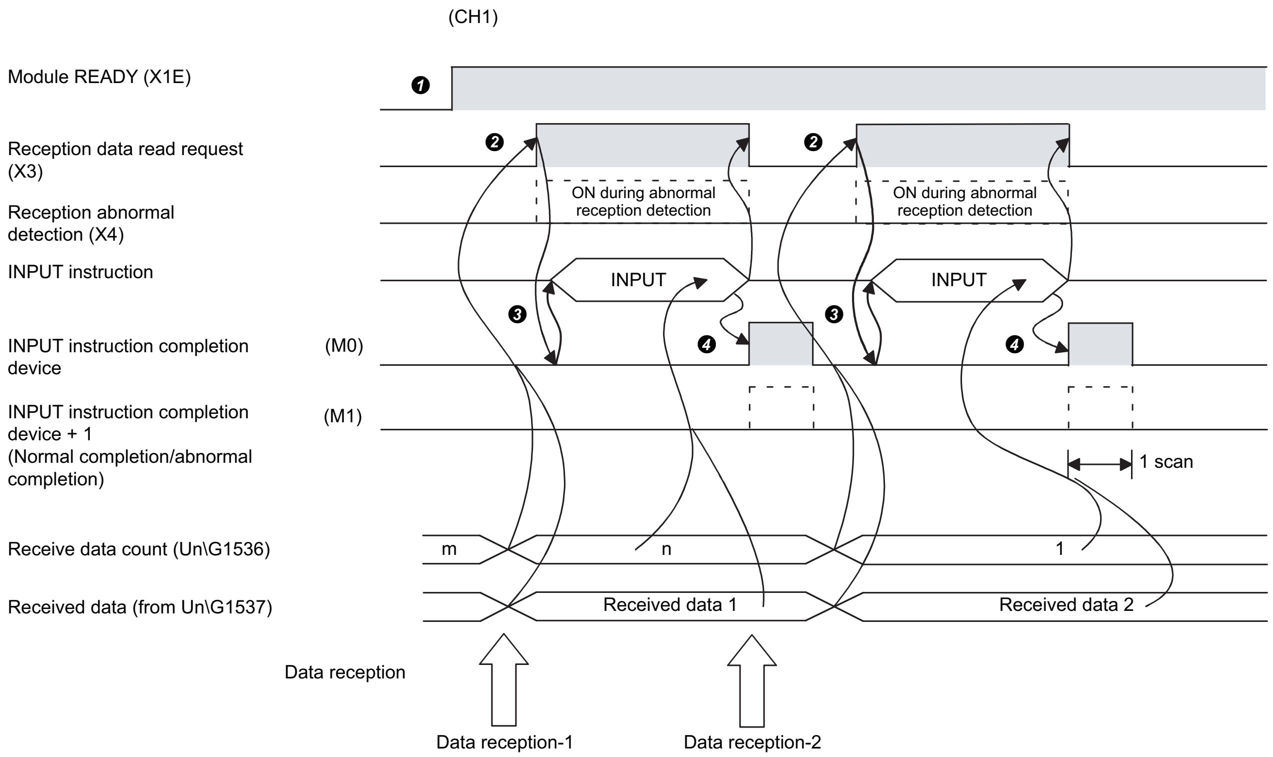
The following shows the program for data reception using the nonprocedural protocol and its example.

For details on the INPUT instruction, refer to the dedicated instructions.(☞ Page 580 Dedicated Instructions)



*1
X3 Read request
M0
(CH1)
Create control data from D0
X4 Abnormal detection
G.INPUT
Un
D0
D10
M0
M0
M1
Processing for normal completion (received data is stored)
M1
Processing for abnormal completion (Reading and discarding received data, retransmission request, etc.)

*1 When the completion device is OFF, execute the INPUT instruction.
When INPUT instruction is executed when the completion device is ON, data may not be received properly.


(CH1)
Module READY (X1E)
Reception data read request (X3)
Reception abnormal detection (X4)
INPUT instruction
INPUT instruction completion device
(M0)
INPUT instruction completion device + 1 (Normal completion/abnormal completion)
(M1)
Receive data count (Un\G1536)
Received data (from Un\G1537)
Data reception
❶
❷
❷
ON during abnormal reception detection
ON during abnormal reception detection
INPUT
INPUT
❸
❸
❹
❹
1 scan
m
n
1
Received data 1
Received data 2
Data reception-1
Data reception-2

❶ Start the programmable controller of the host station. The setting values specified with an Engineering tool are stored in C24.

❷ When the amount of data specified by the receive end data quantity or data containing the receive end code is received from the target device, the 'CH1 reception data read request' (X3) turns ON.

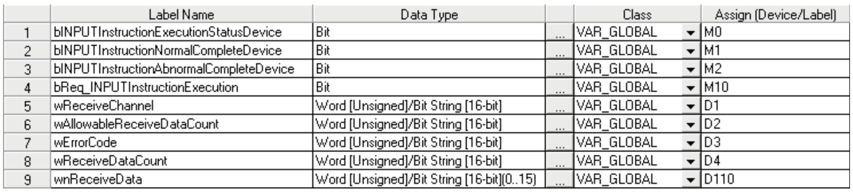
❸ After the control data for the INPUT instruction is stored in the device, the program executes the INPUT instruction and reads the received data.

Before executing the INPUT instruction, check that the INPUT instruction completion device (M0) is OFF.

❹ Once reading of the received data is completed, the INPUT instruction completion device turns ON.

When the completion device + 1 (abnormal completion signal) turns ON, the error code is stored in the control data completion status (S1 + 1).

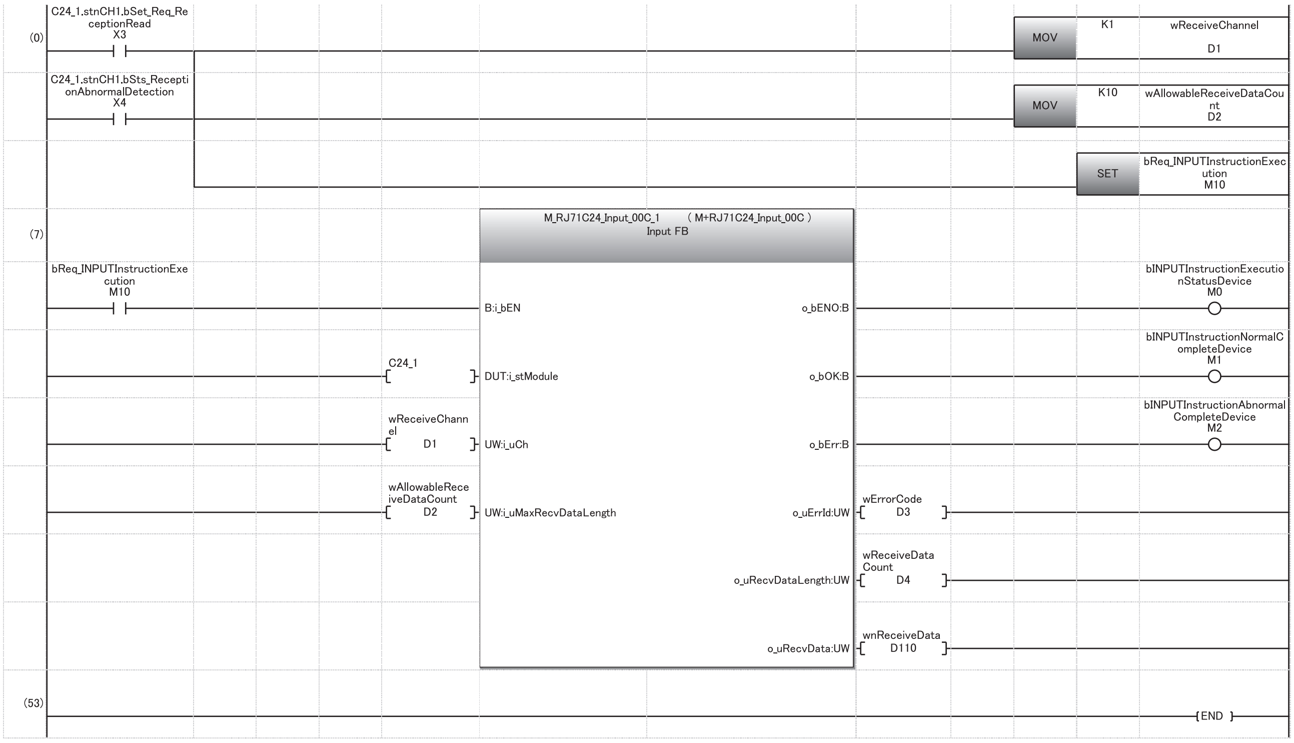
■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection	CH1 Reception abnormal detection	X4
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_Input	Reads the received data.	—

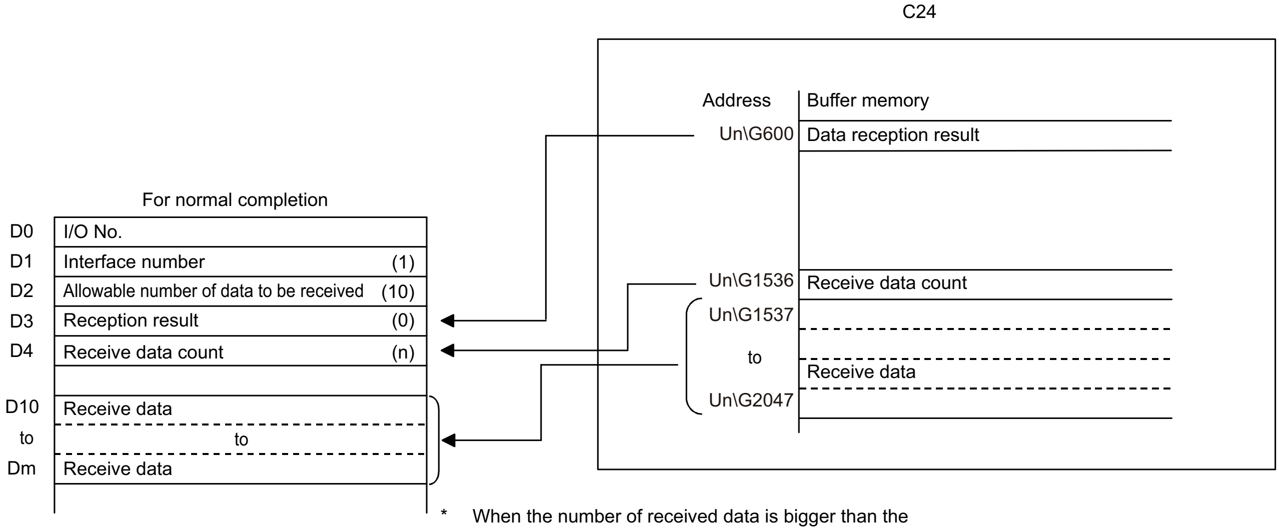
	Label Name	Data Type		Class	Assign (Device/Label)
1	bINPUTInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M0
2	bINPUTInstructionNormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
3	bINPUTInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M2
4	bReq_INPUTInstructionExecution	Bit	...	VAR_GLOBAL	M10
5	wReceiveChannel	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
6	wAllowableReceiveDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2
7	wErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D3
8	wReceiveDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D4
9	wnReceiveData	Word [Unsigned]/Bit String [16-bit](0..15)	...	VAR_GLOBAL	D110

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
C24_1.stnCH1.bSet_Req_ReceptionRead X3
MOV K1 wReceiveChannel D1
C24_1.stnCH1.bSts_ReceptionAbnormalDetection X4
MOV K10 wAllowableReceiveDataCount D2
SET bReq_INPUTInstructionExecution M10
(7)
M_RJ71C24_Input_00C_1 (M+RJ71C24_Input_00C)
Input FB
bReq_INPUTInstructionExecution M10
B:i_bEN
o_bENO:B
bINPUTInstructionExecutionStatusDevice M0
C24_1
DUT:i_stModule
o_bOK:B
bINPUTInstructionNormalCompleteDevice M1
wReceiveChannel D1
UW:i_uCh
o_bErr:B
bINPUTInstructionAbnormalCompleteDevice M2
wAllowableReceiveDataCount D2
UW:i_uMaxRecvDataLength
o_uErrId:UW
wErrorCode D3
o_uRecvDataLength:UW
wReceiveData Count D4
o_uRecvData:UW
wnReceiveData D110
(53)
END

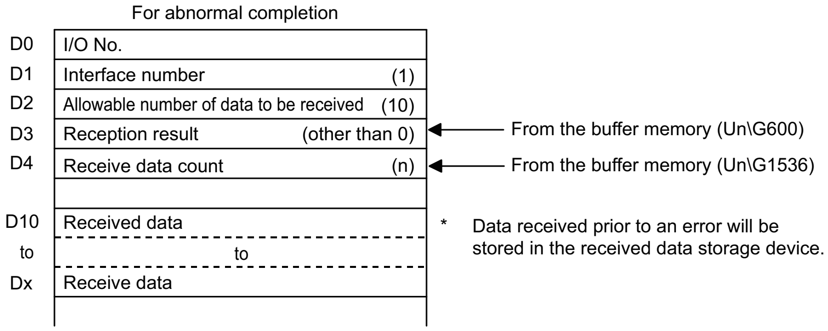
(0) Specify the reception channel.
Specify the allowable number of data to be received.
Specify the execution instruction.



For normal completion

D0	I/O No.	
D1	Interface number	(1)
D2	Allowable number of data to be received	(10)
D3	Reception result	(0)
D4	Receive data count	(n)
D10	Receive data	
to	to	
Dm	Receive data	

* When the number of received data is bigger than the allowable number of data to be received, only the data equivalent to the allowable number will be stored and the rest of the data will be discarded.


From the buffer memory (Un\G600)
From the buffer memory (Un\G1536)

For abnormal completion

D0	I/O No.	
D1	Interface number	(1)
D2	Allowable number of data to be received	(10)
D3	Reception result	(other than 0)
D4	Receive data count	(n)
D10	Received data	
to	to	
Dx	Receive data	

* Data received prior to an error will be stored in the received data storage device.

Point

- Received data can also be read by using an interrupt programs. (☞ Page 251 RECEIVING DATA WITH AN INTERRUPT PROGRAM)
  Note that if the reading of data received from the same interface is performed, it is not possible to combine reading data by the main program and that by the interrupt program.
  Use either of the above programs to read the data received.
- The SPBUSY instruction is used to read the execution status when using a dedicated instruction. (☞ Page 580 Dedicated Instructions)

Restriction

- More than one INPUT instruction cannot be executed simultaneously on the same interface.
  Execute the next INPUT instruction only after the execution of the first INPUT instruction is completed.
- When the send/receive data quantity is set in word units and the receive data quantity is an odd byte when requesting the reading of received data to the CPU module, the receive data quantity is processed as follows:
  Receive data quantity = Receive data byte count/2...Round up after the decimal point.
  Also, 00H is stored in the higher byte of the last data storage position in the receive area where received data is stored.

Reception error detection and check methods

The following explains how to detect and confirm errors that may occur when receiving data from target devices.

The following items are considered as the primary causes of errors that may occur during data reception.

Cause of reception error	Reference
A transmission error occurred due to noise.	MELSEC iQ-R Serial Communication Module User's Manual(Startup)
A timeout occurred for the no-reception monitoring time (timer 0).	Page 260 No-reception Monitoring Time (timer 0) Setting
Data that could not be converted using the ASCII-BIN conversion is included.	Page 380 ASCII-BIN Conversion for Data Communicated with Nonprocedural Protocol
Data larger than the size that could be stored in the OS area of C24 is received.	Page 61 Receive data arrangement

Detection/checking of reception error by a program

■Detecting a reception error

The following device and input signals turn ON.

- INPUT instruction completion device + 1
- 'CH□ reception abnormal detection' (X4/XB)
- 'CH□ Error occurrence' (XE/XF)

■Checking a reception error

The reception error code can be checked using the INPUT instruction control data ((S1) + 1).

Or, it can be checked by reading the 'data reception result' (Un\G600/616).

For details on how to check the error code contents and corrective actions, refer to the following section.

☞ Page 519 Error Code List

'CH□ Error occurrence' (XE/XF) turns OFF when the data transmission/reception processing is performed normally.

Detection/checking of reception error with module and Engineering tool

■Confirmation using the indicator LED

When C24 detects an error, including a transmission error, the ERR LED turns ON.

■Confirmation using Engineering tool

Monitor the 'data reception result' (Un\G600/616) to check the error code.

Considerations

The following describes the considerations for received data in case of errors.

■Storing received data

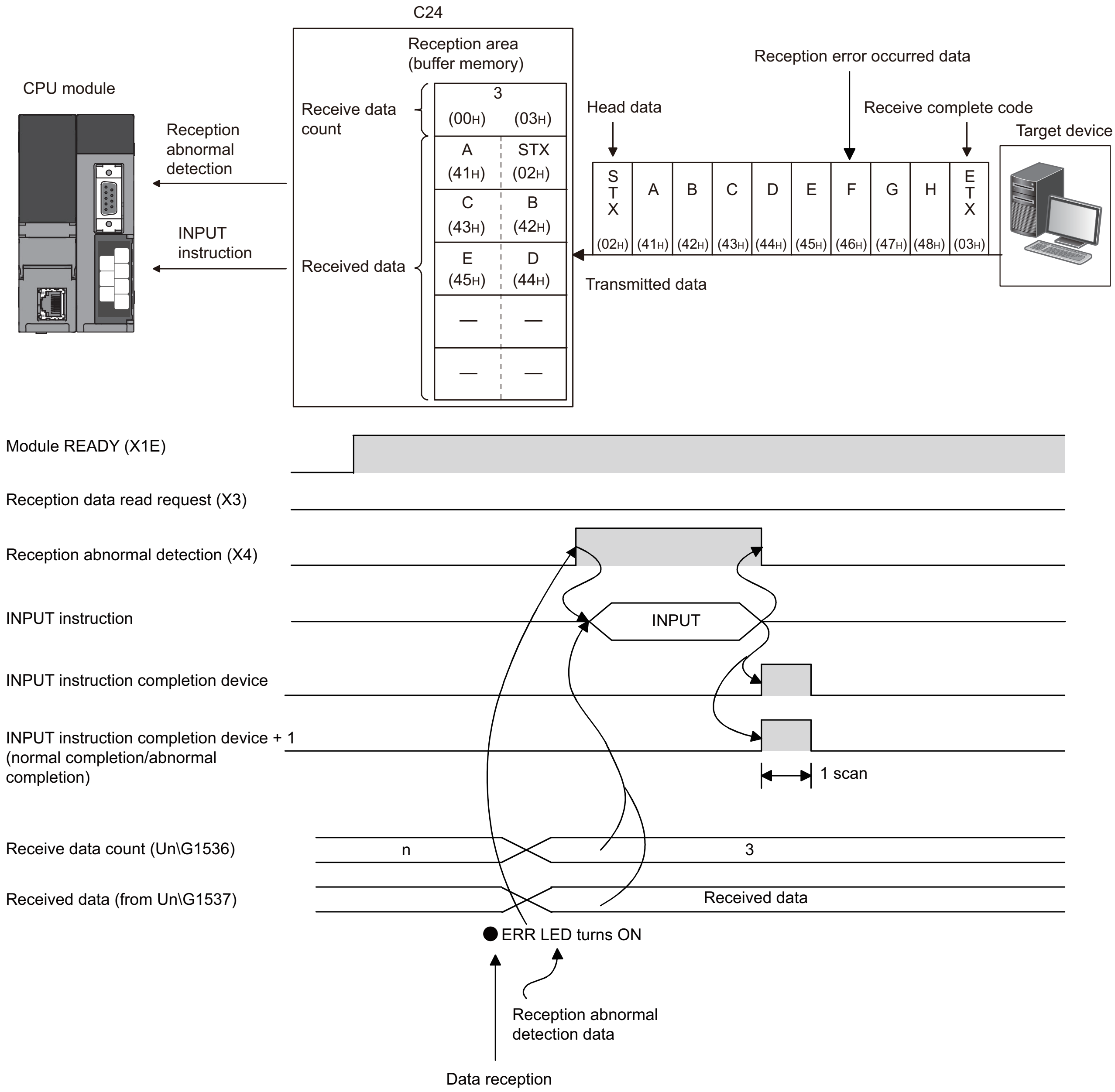
The received data for which an error was detected is discarded and only the normally received data is stored in C24.

Thus, there may be missing data among the data that has been read to the CPU module when a reception error occurs.

■Reading received data

When receive data is read to the CPU module by turning the 'CH□ reception abnormal detection' (X4/XB) ON, the data that was normally received prior to the error detection is read.

Read and discard received data as needed.


C24
Reception area (buffer memory)
CPU module
Reception abnormal detection
INPUT instruction
Receive data count
Received data
3
(00H) (03H)
A (41H)
STX (02H)
C (43H)
B (42H)
E (45H)
D (44H)
Reception error occurred data
Head data
Receive complete code
Target device
STX A B C D E F G H ETX
(02H) (41H) (42H) (43H) (44H) (45H) (46H) (47H) (48H) (03H)
Transmitted data
Module READY (X1E)
Reception data read request (X3)
Reception abnormal detection (X4)
INPUT instruction
INPUT
INPUT instruction completion device
INPUT instruction completion device + 1 (normal completion/abnormal completion)
1 scan
Receive data count (Un\G1536)
n
3
Received data (from Un\G1537)
Received data
● ERR LED turns ON
Reception abnormal detection data
Data reception

When the data of "G" or later in the reception message shown in the above figure is received normally, the data is stored in the C24 OS area.

The received data stored in the OS area will be read to the CPU module in the next read operation.

Receive data clear

For the data reception using the nonprocedural protocol, if the transmission of data from a transmitting device is interrupted due to any trouble, clearing the data received up to the present time on the receiving device side and receiving again from the first data may be required.

The methods for clearing received data in the C24 are shown below.

Use any of the methods to clear received data as necessary.

- Receive data clear request by the dedicated instruction "CSET" (☞ Page 70 Receive data clear with dedicated instruction "CSET")
- Write to/read from the 'receive data clear request' (Un\G168/328). (MOV instruction) (☞ Page 72 Receive data clear with MOV instruction)
- Receive data clear request with Engineering tool. (☞ Page 76 Receive data clear request procedure using Engineering tool)

Precautions

When transmitting data to a target device from C24, do not execute a clear request of the receive data with Engineering tool. If a receive data clear request is executed when transmitting data to a target device, C24 stops data transmission and terminates the dedicated instruction for transmission abnormally. (The transmission completion signal does not turn ON.) If a receive data clear request is executed during data reception, the data received up to that point are cleared.

Receive data clear processing

C24 performs the following processing when the receive data clear is performed.

- C24 regards the received data up to the present time as invalid data.
- C24 clears the receive data quantity up to the present time to "0". (The receive area in the buffer memory is not initialized).
- C24 sets the status in the one prior to the start of data reception.

Procedure for clearing the received data by the program

Using the following instructions, clear the received data by a program.

- Dedicated instruction "CSET"
- MOV instruction

For the procedure to clear received data by a program, refer to the following.

Receive data clear with dedicated instruction "CSET"

This instruction clears the received data up to the present time without interrupting the data transmission processing.

For details on CSET instruction, refer to the dedicated instruction. (☞ Page 580 Dedicated Instructions)

■Procedure for clearing (For the CH1 side interface)

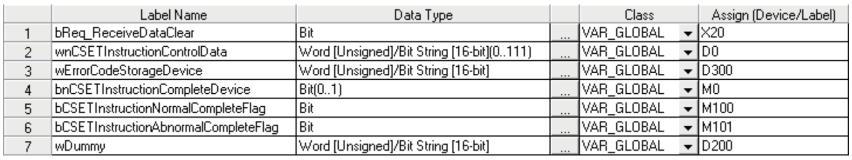
❶ If the CSET instruction is executed while the 'CH□ reception data read request' (X3/XA) or 'CH□ reception abnormal detection signal' (X4/XB) is ON, the CSET instruction is kept waiting until the signal is turned OFF.

❷ The receive data clear processing is executed to the OS area of the C24. (☞ Page 61 Receive data arrangement)

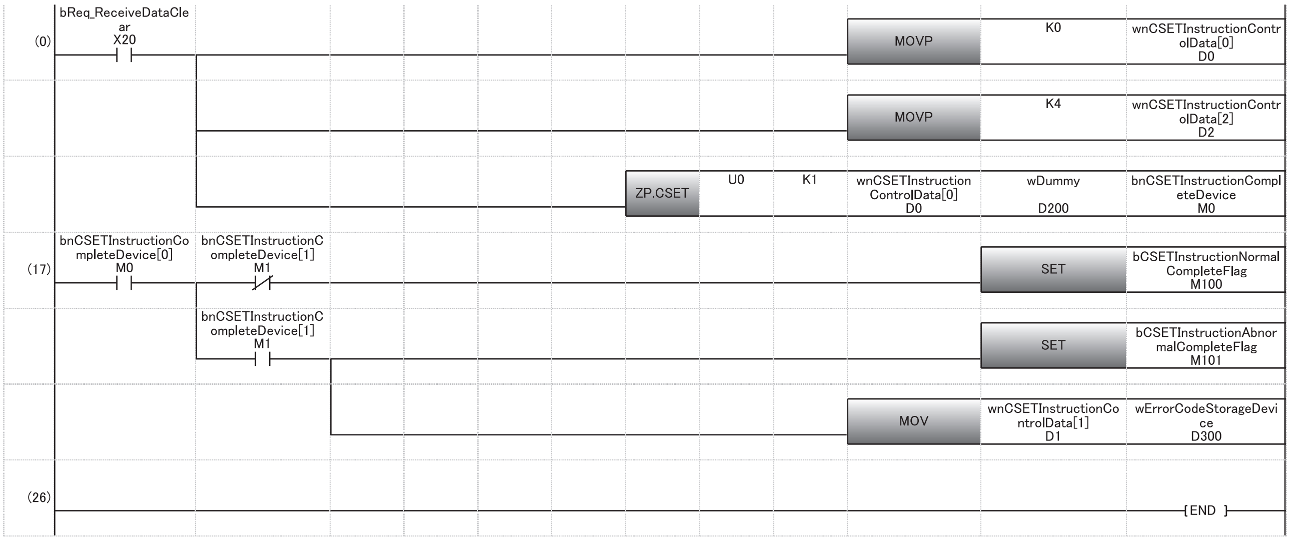
■Program example

The following shows the CH1 receive data clear program example using CSET instruction.

Category	Setting content
Label to be defined	Define the global label as shown below.



	Label Name	Data Type		Class		Assign (Device/Label)
1	bReq_ReceiveDataClear	Bit	...	VAR_GLOBAL	▼	X20
2	wnCSETInstructionControlData	Word [Unsigned]/Bit String [16-bit](0..111)	...	VAR_GLOBAL	▼	D0
3	wErrorCodeStorageDevice	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	▼	D300
4	bnCSETInstructionCompleteDevice	Bit(0..1)	...	VAR_GLOBAL	▼	M0
5	bCSETInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	▼	M100
6	bCSETInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	▼	M101
7	wDummy	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	▼	D200


(0)
bReq_ReceiveDataClear
X20
MOVP
K0
wnCSETInstructionControlData[0]
D0
MOVP
K4
wnCSETInstructionControlData[2]
D2
ZP.CSET
U0
K1
wnCSETInstructionControlData[0]
D0
wDummy
D200
bnCSETInstructionCompleteDevice
M0
(17)
bnCSETInstructionCompleteDevice[0]
M0
bnCSETInstructionCompleteDevice[1]
M1
SET
bCSETInstructionNormalCompleteFlag
M100
bnCSETInstructionCompleteDevice[1]
M1
SET
bCSETInstructionAbnormalCompleteFlag
M101
MOV
wnCSETInstructionControlData[1]
D1
wErrorCodeStorageDevice
D300
(26)
END

(0) Set the execution type.
Set the request type (receive data clear).
Execute receive data clear.

(17) Processing for normal completion and abnormal completion

Receive data clear with MOV instruction

Write to/read from the 'receive data clear request' (Un\G168/328) in the procedure below with the MOV instruction.

When resuming data communication with a target device, resume it after completing the receive data clear processing with C24.

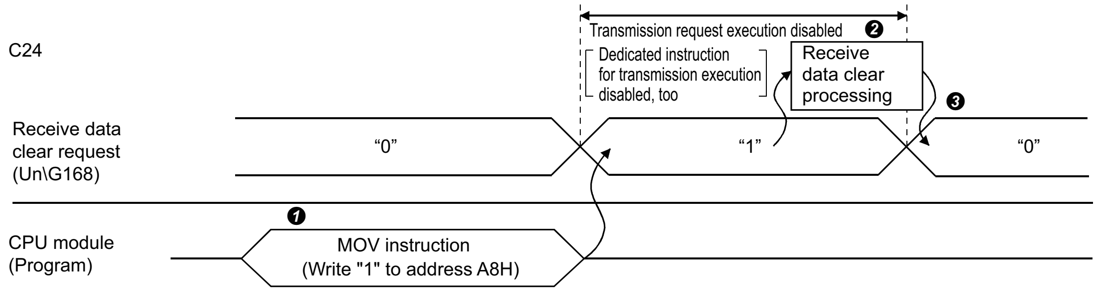
Depending on the data communication system, it may be necessary to restart data communications with the target device after notifying the target device from the CPU module that resumption of communications is possible.

Restriction

Create a program including contacts indicating the following status (such as flag for the user) in MOV instruction.

- Contacts indicating that the 'CH□ reception data read request' (X3/XA) and 'CH□ reception abnormal detection' (X4/XB) are OFF
- Contact indicating that no dedicated instruction (OUTPUT/PRR/INPUT) is currently being executed
- Contact indicating that receive data clear processing is not currently being performed

■Clear Procedure (When the CH1 side interface is used)


C24
Transmission request execution disabled ❷
Dedicated instruction for transmission execution disabled, too
Receive data clear processing
❸
Receive data clear request (Un\G168)
"0"
"1"
"0"
❶
CPU module (Program)
MOV instruction (Write "1" to address A8H)

❶ Write "1" to the 'receive data clear request' (Un\G168) with the MOV instruction when no data reception, data transmission, or receive data clearing processing is being performed.

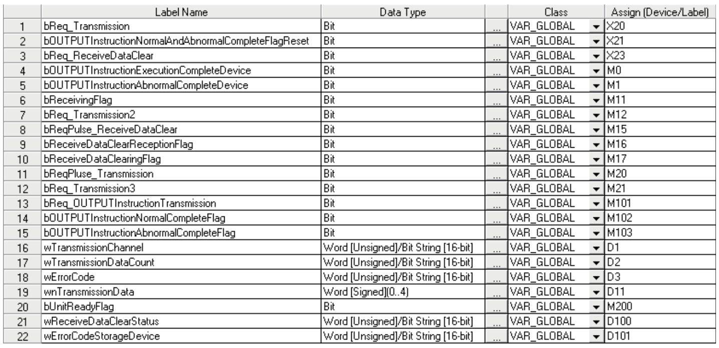
❷ The receive data clear processing is executed to the OS area of the C24.(☞ Page 61 Receive data arrangement)

❸ When receive data clear processing is completed, the 'receive data clear request' (Un\G168) becomes "0".

■Program example

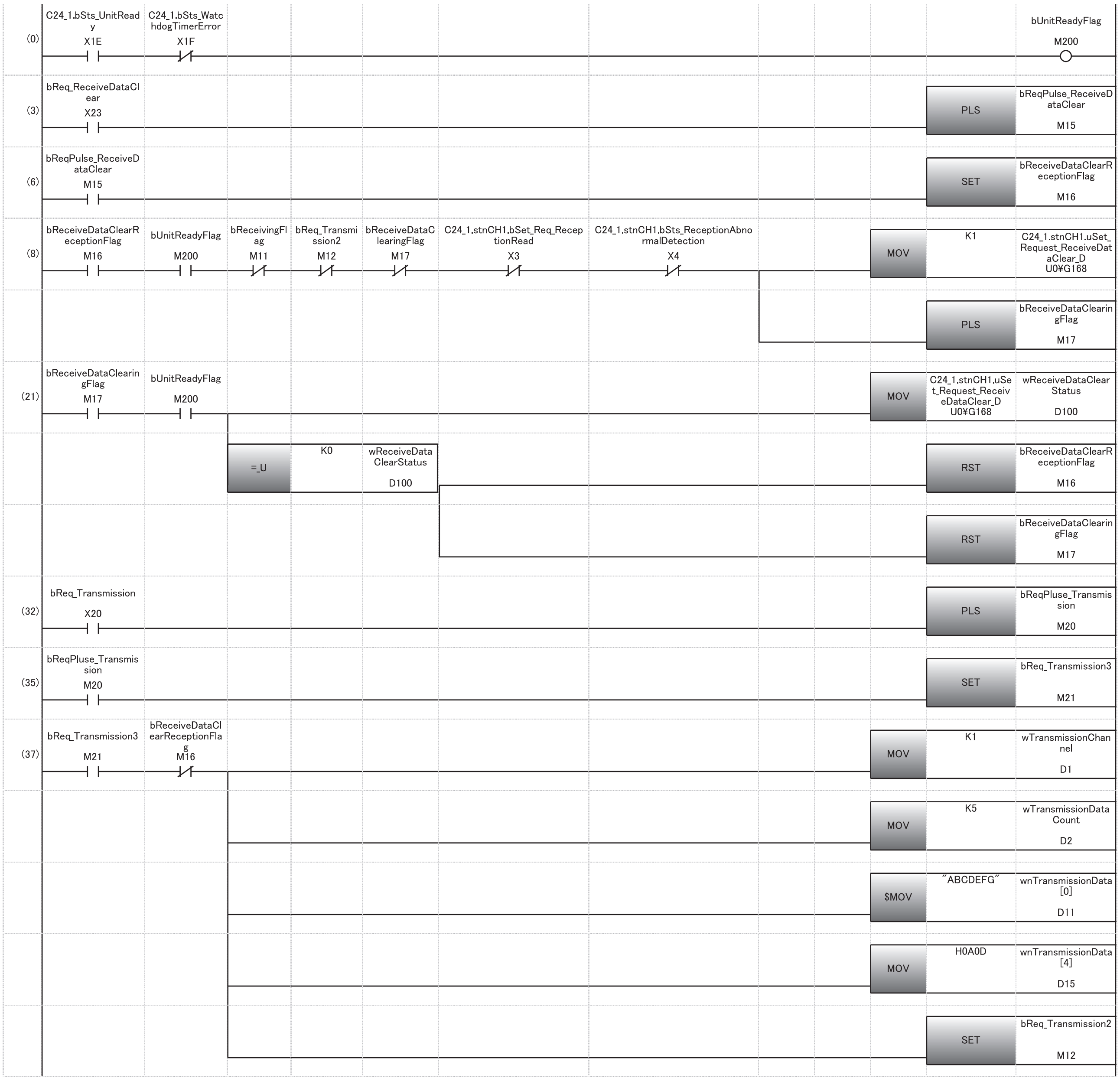
The following shows the CH1 receive data clear program example using MOV instruction.

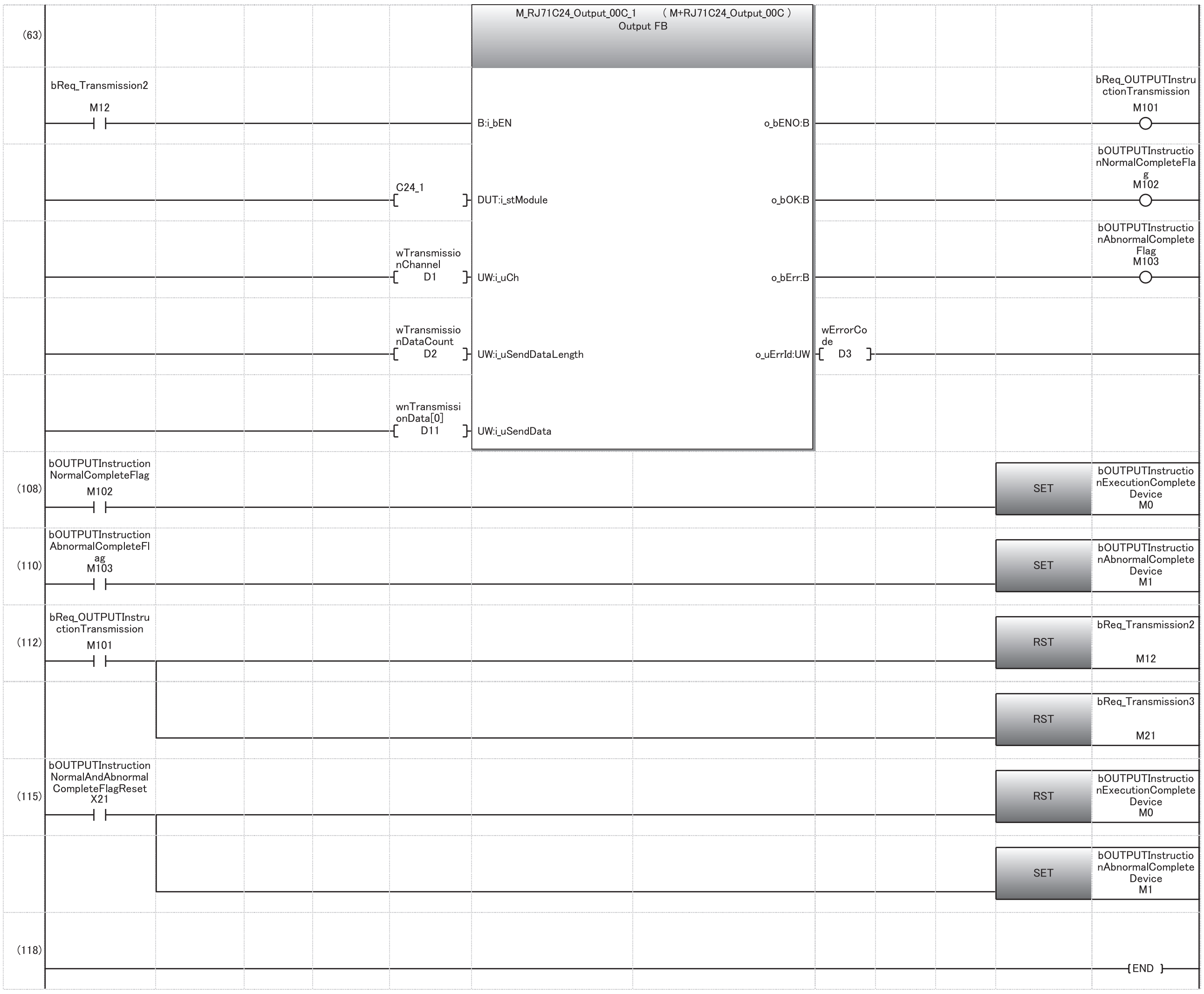
The dedicated instructions (OUTPUT/INPUT) is used for data transmission/reception.(☞ Page 580 Dedicated Instructions)

<table>
<thead>
<tr><th>Category</th><th>Label name/FB name</th><th>Description</th><th>Device</th></tr>
</thead>
<tbody>
<tr><td>Module label</td><td>C24_1</td><td>Target module</td><td>—</td></tr>
<tr><td></td><td>C24_1.bSts_UnitReady</td><td>Module READY</td><td>X1E</td></tr>
<tr><td></td><td>C24_1.bSts_WatchdogTimerError</td><td>Watchdog timer error</td><td>X1F</td></tr>
<tr><td></td><td>C24_1.stnCH1.bSet_Req_ReceptionRead</td><td>CH1 Reception data read request</td><td>X3</td></tr>
<tr><td></td><td>C24_1.stnCH1.bSts_ReceptionAbnormalDetection</td><td>CH1 Reception abnormal detection</td><td>X4</td></tr>
<tr><td></td><td>C24_1.stnCH1.uSet_Request_ReceiveDataClear_D</td><td>CH1_Receive data clear request</td><td>U0\G168</td></tr>
<tr><td>Label to be defined</td><td colspan="3">Define the global label as shown below.

<table>
<thead>
<tr><th></th><th>Label Name</th><th>Data Type</th><th></th><th>Class</th><th>Assign (Device/Label)</th></tr>
</thead>
<tbody>
<tr><td>1</td><td>bReq_Transmission</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>X20</td></tr>
<tr><td>2</td><td>bOUTPUTInstructionNormalAndAbnormalCompleteFlagReset</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>X21</td></tr>
<tr><td>3</td><td>bReq_ReceiveDataClear</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>X23</td></tr>
<tr><td>4</td><td>bOUTPUTInstructionExecutionCompleteDevice</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M0</td></tr>
<tr><td>5</td><td>bOUTPUTInstructionAbnormalCompleteDevice</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M1</td></tr>
<tr><td>6</td><td>bReceivingFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M11</td></tr>
<tr><td>7</td><td>bReq_Transmission2</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M12</td></tr>
<tr><td>8</td><td>bReqPulse_ReceiveDataClear</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M15</td></tr>
<tr><td>9</td><td>bReceiveDataClearReceptionFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M16</td></tr>
<tr><td>10</td><td>bReceiveDataClearingFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M17</td></tr>
<tr><td>11</td><td>bReqPluse_Transmission</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M20</td></tr>
<tr><td>12</td><td>bReq_Transmission3</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M21</td></tr>
<tr><td>13</td><td>bReq_OUTPUTInstructionTransmission</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M101</td></tr>
<tr><td>14</td><td>bOUTPUTInstructionNormalCompleteFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M102</td></tr>
<tr><td>15</td><td>bOUTPUTInstructionAbnormalCompleteFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M103</td></tr>
<tr><td>16</td><td>wTransmissionChannel</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D1</td></tr>
<tr><td>17</td><td>wTransmissionDataCount</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D2</td></tr>
<tr><td>18</td><td>wErrorCode</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D3</td></tr>
<tr><td>19</td><td>wnTransmissionData</td><td>Word [Signed](0..4)</td><td>...</td><td>VAR_GLOBAL</td><td>D11</td></tr>
<tr><td>20</td><td>bUnitReadyFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M200</td></tr>
<tr><td>21</td><td>wReceiveDataClearStatus</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D100</td></tr>
<tr><td>22</td><td>wErrorCodeStorageDevice</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D101</td></tr>
</tbody>
</table>
</td></tr>
<tr><td>FB</td><td>M+RJ71C24_Output</td><td>Sends data for specified data points.</td><td>—</td></tr>
</tbody>
</table>

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
C24_1.bSts_UnitReady
X1E
C24_1.bSts_WatchdogTimerError
X1F
bUnitReadyFlag
M200
(3)
bReq_ReceiveDataClear
X23
PLS
bReqPulse_ReceiveDataClear
M15
(6)
bReqPulse_ReceiveDataClear
M15
SET
bReceiveDataClearReceptionFlag
M16
(8)
bReceiveDataClearReceptionFlag
M16
bUnitReadyFlag
M200
bReceivingFlag
M11
bReq_Transmission2
M12
bReceiveDataClearingFlag
M17
C24_1.stnCH1.bSet_Req_ReceptionRead
X3
C24_1.stnCH1.bSts_ReceptionAbnormalDetection
X4
MOV
K1
C24_1.stnCH1.uSet_Request_ReceiveDataClear_D
U0¥G168
PLS
bReceiveDataClearingFlag
M17
(21)
bReceiveDataClearingFlag
M17
bUnitReadyFlag
M200
MOV
C24_1.stnCH1.uSet_Request_ReceiveDataClear_D
U0¥G168
wReceiveDataClearStatus
D100
=_U
K0
wReceiveDataClearStatus
D100
RST
bReceiveDataClearReceptionFlag
M16
RST
bReceiveDataClearingFlag
M17
(32)
bReq_Transmission
X20
PLS
bReqPluse_Transmission
M20
(35)
bReqPluse_Transmission
M20
SET
bReq_Transmission3
M21
(37)
bReq_Transmission3
M21
bReceiveDataClearReceptionFlag
M16
MOV
K1
wTransmissionChannel
D1
MOV
K5
wTransmissionDataCount
D2
$MOV
"ABCDEFG"
wnTransmissionData[0]
D11
MOV
H0A0D
wnTransmissionData[4]
D15
SET
bReq_Transmission2
M12


(63)
M_RJ71C24_Output_00C_1 (M+RJ71C24_Output_00C)
Output FB
bReq_Transmission2
M12
B:i_bEN
o_bENO:B
bReq_OUTPUTInstructionTransmission
M101
C24_1
DUT:i_stModule
o_bOK:B
bOUTPUTInstructionNormalCompleteFlag
M102
wTransmissionChannel
D1
UW:i_uCh
o_bErr:B
bOUTPUTInstructionAbnormalCompleteFlag
M103
wTransmissionDataCount
D2
UW:i_uSendDataLength
o_uErrId:UW
wErrorCode
D3
wnTransmissionData[0]
D11
UW:i_uSendData
(108)
bOUTPUTInstructionNormalCompleteFlag
M102
SET
bOUTPUTInstructionExecutionCompleteDevice
M0
(110)
bOUTPUTInstructionAbnormalCompleteFlag
M103
SET
bOUTPUTInstructionAbnormalCompleteDevice
M1
(112)
bReq_OUTPUTInstructionTransmission
M101
RST
bReq_Transmission2
M12
RST
bReq_Transmission3
M21
(115)
bOUTPUTInstructionNormalAndAbnormalCompleteFlagReset
X21
RST
bOUTPUTInstructionExecutionCompleteDevice
M0
SET
bOUTPUTInstructionAbnormalCompleteDevice
M1
(118)
END

(6) Set the receive data clear acknowledgement flag.

(8) Execute receive data clear processing.

(21) Read receive data clear status.
Reset the receive data clear acknowledgement flag.
Reset the receive data clear in progress flag.

(35) Set the transmission request.

(37) Designate the transmission channel.
Set the send data quantity.
Set the send data (string).
Set the send data (numeric value).
Set the transmission request.

(108) OUTPUT instruction normal completion

(110) OUTPUT instruction abnormal completion

(112) Reset the transmission processing request.
Reset the transmission request.

(115) Reset the OUTPUT instruction execution completion flag.
Reset the OUTPUT instruction abnormal completion flag.

Receive data clear request procedure using Engineering tool

The following shows the method which uses "Intelligent Function Module Monitor" of Engineering tool.

For the operation method of the Intelligent Function Module Monitor function, refer to the following manual.

📖GX Works3 Operating Manual

Operating procedure

1. Register C24 in "Intelligent Function Module Monitor".
2. Set "CH□ Receive data clear request" of "Receiving end specification" to "Requested".
3. The received data is cleared.

Receive end data quantity and receive end code settings

Setting method before receiving data

The following default values have been set for the receive end data quantity and receive end code that are used for data reception with the nonprocedural protocol.

Setting data	Default value	Allowable range	Buffer memory address where the setting value is stored	Remarks
Receive end data quantity	511 (1FFH)	Less than the size of the receive data storage area	Un\G164/324	Unit is in accordance with the word/byte units designation.
Receive end code	0D0AH (CR+LF)	0000H to 00FFH	Un\G165/325	When any code is set.
		FFFFH		When there is no receive end code.

The receive end data quantity and the receive end code setting value can be changed by the program before data reception processing.

The default value can be changed in the following unit parameter of Engineering tool.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

For details, refer to the following section.

☞ Page 454 PARAMETER SETTING

Setting method after receiving data

Other than through the setting using Engineering tool or the MOV instruction at the startup of the C24, the receive end data quantity and receive end code setting values can be changed even after data communication has started if the timing is as shown below.

The following describes the restriction of the module, and the timing and procedure for changing setting values to continue the data receive processing by changing the setting values for data reception using the nonprocedural protocol after the start of data communication.

■How to receive data

Reception using an interrupt program [*1] cannot be performed. (Transmission using a dedicated instruction (OUTPUT) can be performed.)

Data must be received by the main program.

The following shows the I/O signals.

[*1] For details on the data reception using interrupt programs, refer to the following section.
☞ Page 251 RECEIVING DATA WITH AN INTERRUPT PROGRAM

Type	Input/Output Signals		Signal name	Device that turns ON/OFF	
	CH1 side	CH2 side		CPU module	C24
Reception	X3	XA	Reception data read request		○
	X4	XB	Reception abnormal detection		○

Restriction

When the 'CH□ reception abnormal detection' (X4/XB) turns ON, read the error code from the buffer memory below, and then check the type of error and take corrective actions.(☞ Page 519 Error Code List)

- 'Data reception result' (Un\G600/616)

■Setting values that can be changed

Of the initial settings from Engineering tool stored in the buffer memory, only the following setting values can be changed after data communication has started.

- 'Receive end data quantity designation' (Un\G164/324)
- 'Receive end code designation' (Un\G165/325)

■Timing and procedure for changing the setting values

● Timing of change

To change the receive end code and receive end data quantity after data communication has started, change the setting when the status of the I/O signals relating to the reading of received data is as shown below.

- 'CH□ Reception data read request' (X3/XA) or 'CH□ reception abnormal detection' (X4/XB): ON

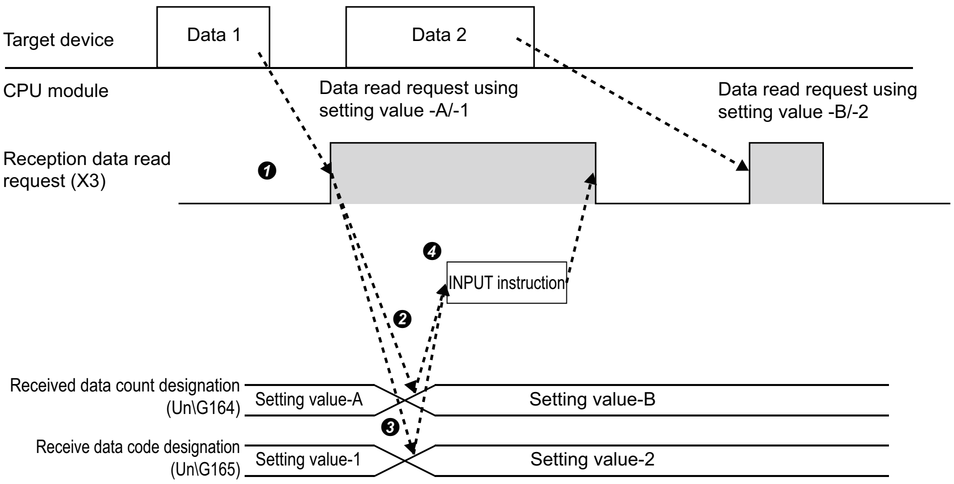
● Change procedure

When the 'CH□ reception data read request' (X3/XA)/'CH□ reception abnormal detection' (X4/XB) turns ON upon data reception from the target device, perform the following and execute the INPUT instruction.

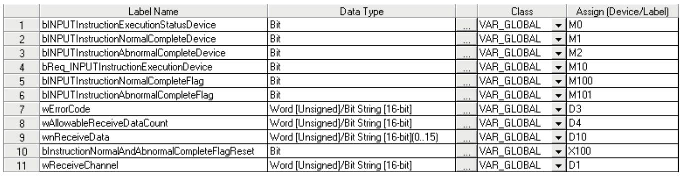
- Change the setting values for the receive end code and the receive end data quantity.

Ex.

When reading data received in the CH1 side interface


Target device
Data 1
Data 2
CPU module
Data read request using setting value -A/-1
Data read request using setting value -B/-2
Reception data read request (X3)
❶
❷
❸
❹
INPUT instruction
Received data count designation (Un\G164)
Setting value-A
Setting value-B
Receive data code designation (Un\G165)
Setting value-1
Setting value-2

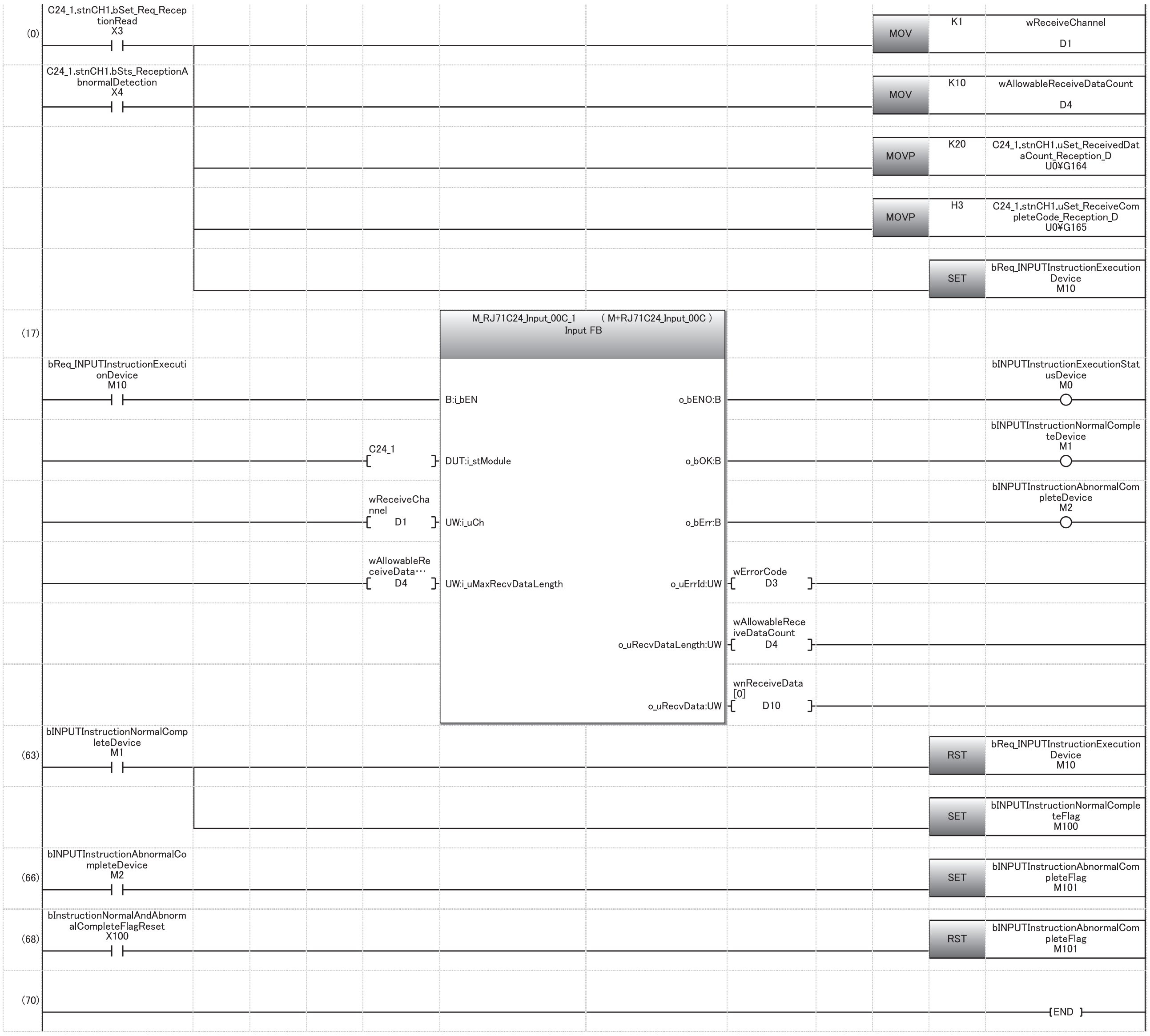
■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection	CH1 Reception abnormal detection	X4
	C24_1.stnCH1.uSet_ReceivedDataCount_Reception_D	CH1_'Receive end data quantity designation	U0\G164
	C24_1.stnCH1.uSet_ReceiveCompleteCode_Reception_D	CH1_Receive end code designation	U0\G165
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_Input	Reads the received data.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bINPUTInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M0
2	bINPUTInstructionNormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
3	bINPUTInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M2
4	bReq_INPUTInstructionExecutionDevice	Bit	...	VAR_GLOBAL	M10
5	bINPUTInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
6	bINPUTInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101
7	wErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D3
8	wAllowableReceiveDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D4
9	wnReceiveData	Word [Unsigned]/Bit String [16-bit](0..15)	...	VAR_GLOBAL	D10
10	bInstructionNormalAndAbnormalCompleteFlagReset	Bit	...	VAR_GLOBAL	X100
11	wReceiveChannel	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
C24_1.stnCH1.bSet_Req_ReceptionRead
X3
MOV
K1
wReceiveChannel
D1
C24_1.stnCH1.bSts_ReceptionAbnormalDetection
X4
MOV
K10
wAllowableReceiveDataCount
D4
MOVP
K20
C24_1.stnCH1.uSet_ReceivedDataCount_Reception_D
U0¥G164
MOVP
H3
C24_1.stnCH1.uSet_ReceiveCompleteCode_Reception_D
U0¥G165
SET
bReq_INPUTInstructionExecutionDevice
M10
(17)
M_RJ71C24_Input_00C_1 (M+RJ71C24_Input_00C)
Input FB
bReq_INPUTInstructionExecutionDevice
M10
B:i_bEN
o_bENO:B
bINPUTInstructionExecutionStatusDevice
M0
C24_1
DUT:i_stModule
o_bOK:B
bINPUTInstructionNormalCompleteDevice
M1
wReceiveChannel
D1
UW:i_uCh
o_bErr:B
bINPUTInstructionAbnormalCompleteDevice
M2
wAllowableReceiveData⋯
D4
UW:i_uMaxRecvDataLength
o_uErrId:UW
wErrorCode
D3
o_uRecvDataLength:UW
wAllowableReceiveDataCount
D4
o_uRecvData:UW
wnReceiveData[0]
D10
(63)
bINPUTInstructionNormalCompleteDevice
M1
RST
bReq_INPUTInstructionExecutionDevice
M10
SET
bINPUTInstructionNormalCompleteFlag
M100
(66)
bINPUTInstructionAbnormalCompleteDevice
M2
SET
bINPUTInstructionAbnormalCompleteFlag
M101
(68)
bInstructionNormalAndAbnormalCompleteFlagReset
X100
RST
bINPUTInstructionAbnormalCompleteFlag
M101
(70)
END

(0) Specify the reception channel.
Specify the allowable number of data to be received.
Set the receive end data quantity.
Set the receive end code.

(63) Normal completion

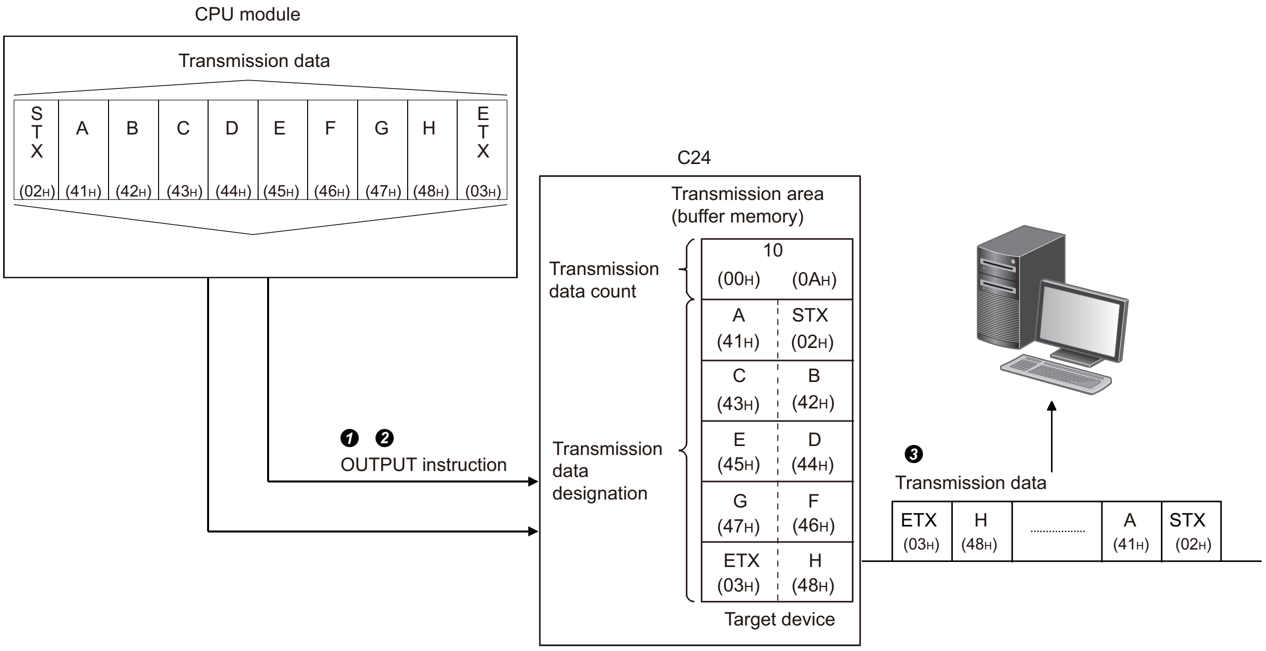
(66) Abnormal completion

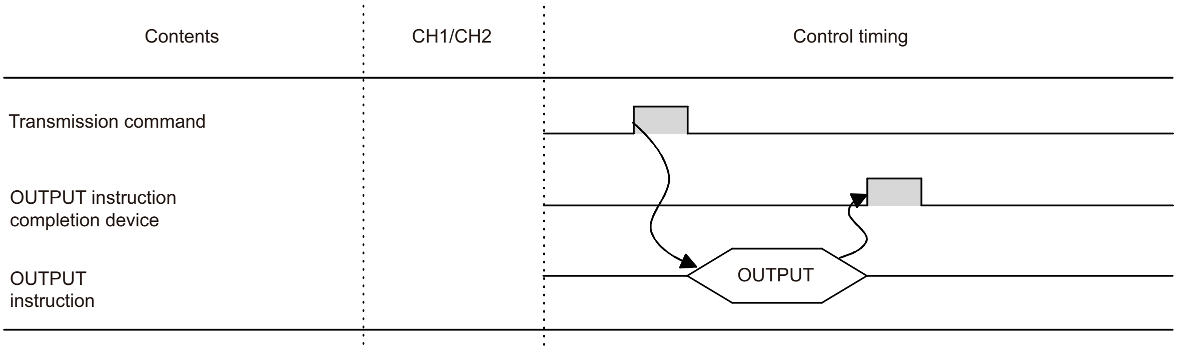
3.2 Transmitting Data to Target Device

This section explains data transmission from the CPU module to a target device.

Sending methods

The following shows how to send data in any format using the nonprocedural protocol.





❶ Send data is stored in the device designated with the OUTPUT instruction.

❷ Control data is stored in the device designated with the OUTPUT instruction.

❸ When the OUTPUT instruction is executed, C24 transmits the amount of data designated by the send data quantity from the send data area in ascending order of the send data area addresses.

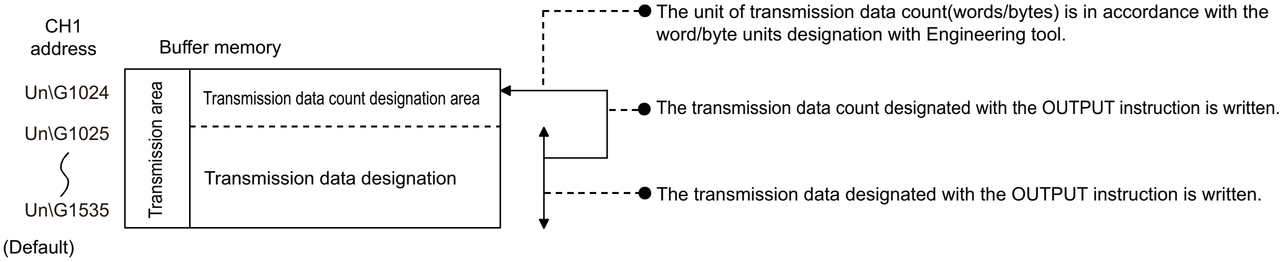
Send area and send data arrangement

This section explains the send area and the send data arrangement for performing data transmission using the nonprocedural protocol.

Send area

The send area is a memory to write data and the data quantity that are transmitted from the CPU module to a target device via C24.

The send area is assigned to the address, Un\G1024 to 1535 (CH1 side), Un\G2048 to 2559 (CH2 side) by default.



CH1 address
Buffer memory
Un\G1024
Un\G1025
Un\G1535
(Default)
Transmission area
Transmission data count designation area
Transmission data designation
The unit of transmission data count(words/bytes) is in accordance with the word/byte units designation with Engineering tool.
The transmission data count designated with the OUTPUT instruction is written.
The transmission data designated with the OUTPUT instruction is written.

The position and size of the send area in the buffer memory can be changed with Engineering tool in accordance with the specifications of the target device and the send data length. (☞ Page 454 PARAMETER SETTING)

Configure the settings as follows:

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

- Designate the start address for the area to be used as the send area in the user setting area (Un\G1024 to 6911, Un\G9728 to 16383) to "Send buffer memory start address designation".
- Designate the length of the area to be used as the send area in the user setting area (Un\G1024 to 6911, Un\G9728 to 16383) by address count (0001H to 1A00H) to "Send buffer memory size specification".

When changing the position and size of the send area in the buffer memory, if any of the functions below are used concurrently, make sure not to overlap with the buffer memory addresses assigned for storing send/receive data handled with the functions.

- MC protocol buffer memory read/write function
- MC protocol on-demand function
- Predefined protocol function (Send/receive data storage area)
- Nonprocedural protocol transmission/receive function
- Bidirectional protocol transmission/receive function
- Send/receive data monitoring function

Set the size of data per data transmission from the target device to C24 to be equal to or smaller than the size of the send data storage area in the buffer memory.

- (Send data storage area) ≥ (Amount of data in any data portion to be transmitted from C24)

To transmit data whose size is larger than the send data storage area, increase the send area, or divide the send data before sending.

Point

The position and size of send area can be changed in the buffer memory.

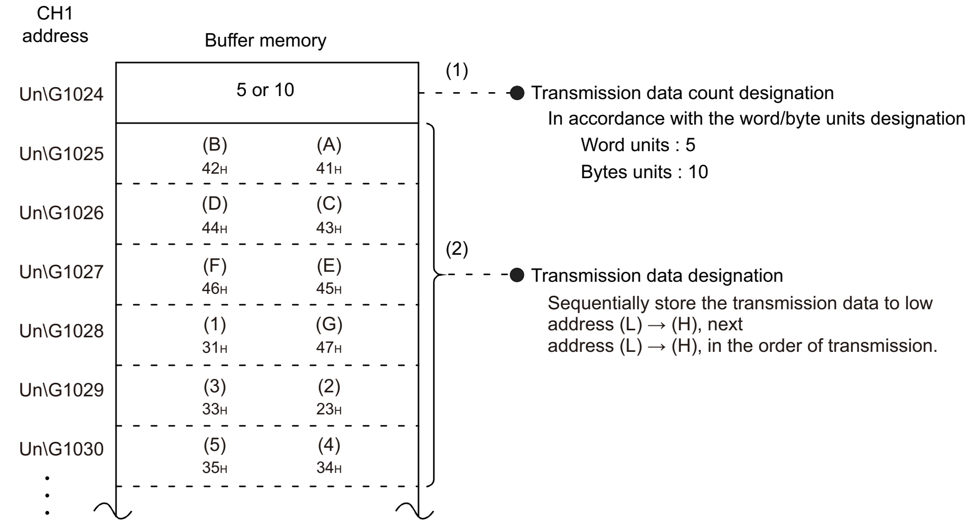
- 'Send buffer memory start address designation' (Un\G162/322)
- 'Send buffer memory length designation' (Un\G163/323)

Send data arrangement

The following example shows an arrangement of the send data to be sent to the target device when storing it in the send area.

Ex.

When transmitting 'ABCDEFG123' (The send area is the default value.)



CH1 address	Buffer memory	
Un\G1024	5 or 10	
Un\G1025	(B) 42H	(A) 41H
Un\G1026	(D) 44H	(C) 43H
Un\G1027	(F) 46H	(E) 45H
Un\G1028	(1) 31H	(G) 47H
Un\G1029	(3) 33H	(2) 23H
Un\G1030	(5) 35H	(4) 34H
⋮		

(1) Transmission data count designation
In accordance with the word/byte units designation
Word units : 5
Bytes units : 10

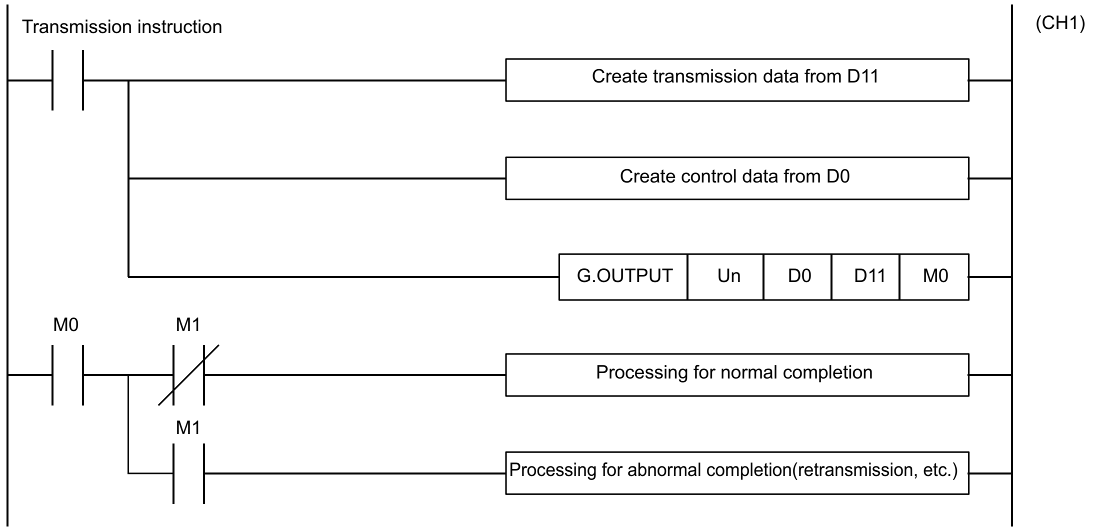
(2) Transmission data designation
Sequentially store the transmission data to low address (L) → (H), next address (L) → (H), in the order of transmission.

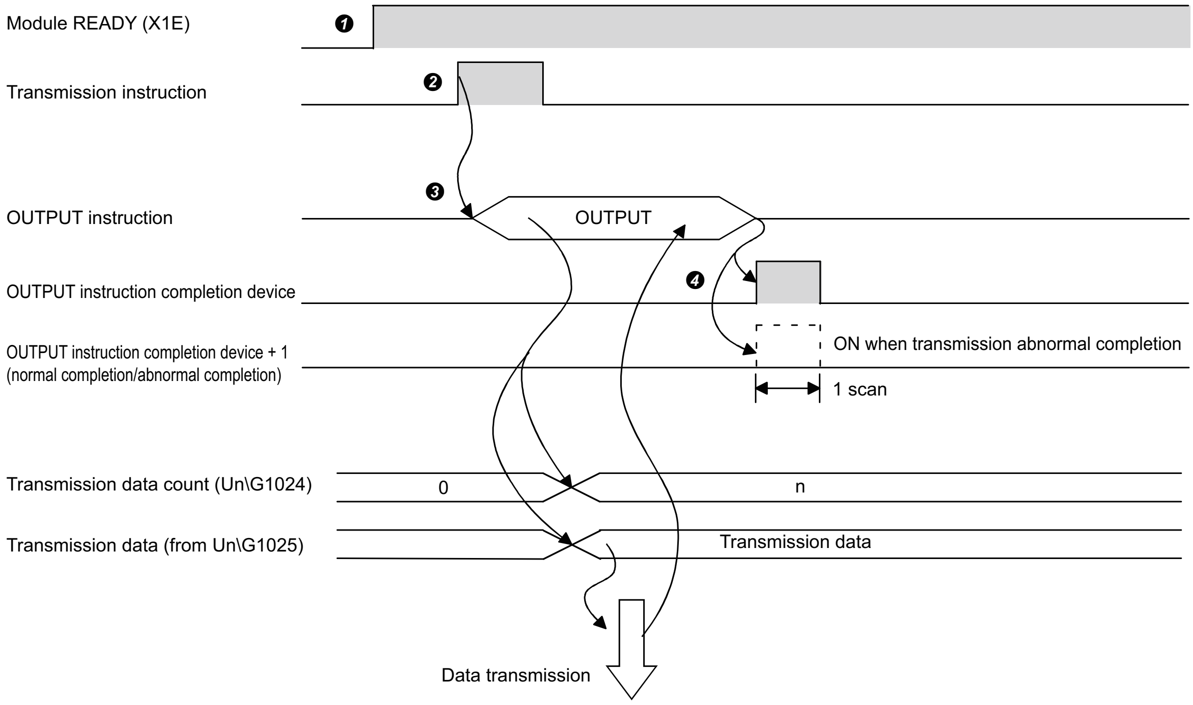
Program for data transmission

Program example for data transmission

The following shows the program for data transmission using nonprocedural protocol and its example.

For details on OUTPUT instructions, refer to the dedicated instruction. (☞ Page 580 Dedicated Instructions)


Transmission instruction
(CH1)
Create transmission data from D11
Create control data from D0
G.OUTPUT
Un
D0
D11
M0
M0
M1
Processing for normal completion
M1
Processing for abnormal completion(retransmission, etc.)


Module READY (X1E)
❶
Transmission instruction
❷
❸
OUTPUT instruction
OUTPUT
OUTPUT instruction completion device
❹
OUTPUT instruction completion device + 1
(normal completion/abnormal completion)
ON when transmission abnormal completion
1 scan
Transmission data count (Un\G1024)
0
n
Transmission data (from Un\G1025)
Transmission data
Data transmission

❶Start the programmable controller on the host station.

The value set with Engineering tool is stored in the C24.

❷ Input the user data transmission command signal.

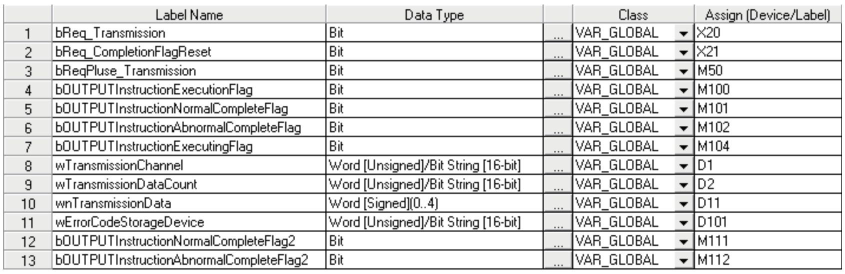
❸ Executes the OUTPUT instruction after storing the send data and the control data for the OUTPUT instruction in the device.

Data is sent when the OUTPUT instruction is executed.

❹ C24 transmission processing completes and the OUTPUT instruction completion device turns ON.

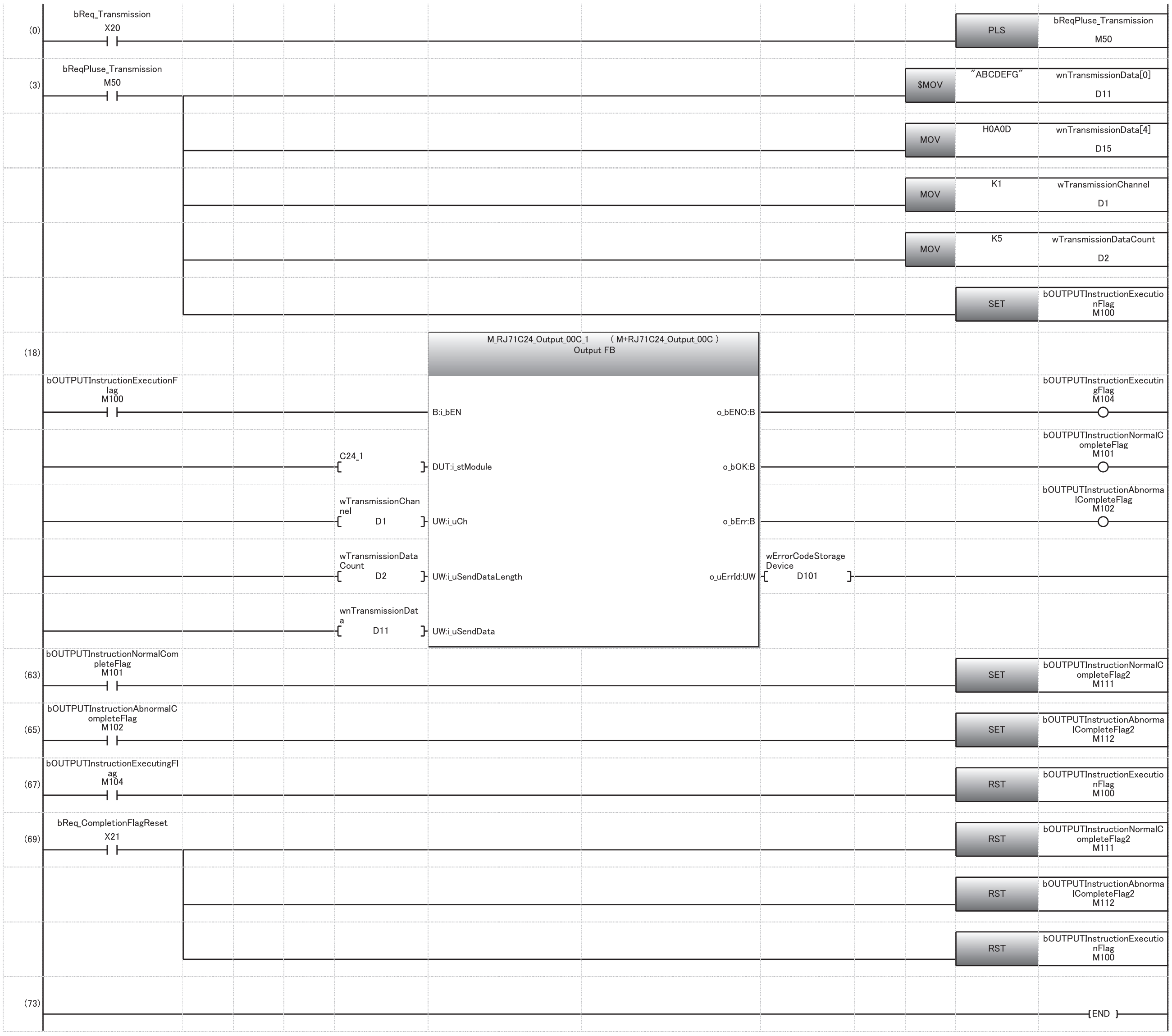
When the completion device +1 (abnormal completion signal) turns ON, the error code is stored in the control data completion status (S1 + 1).

■Program example

<table>
<tr><th>Category</th><th>Label name/FB name</th><th>Description</th><th>Device</th></tr>
<tr><td>Module label</td><td>C24_1</td><td>Target module</td><td>—</td></tr>
<tr><td>Label to be defined</td><td colspan="3">Define the global label as shown below.

<table>
<tr><th></th><th>Label Name</th><th>Data Type</th><th></th><th>Class</th><th>Assign (Device/Label)</th></tr>
<tr><td>1</td><td>bReq_Transmission</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>X20</td></tr>
<tr><td>2</td><td>bReq_CompletionFlagReset</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>X21</td></tr>
<tr><td>3</td><td>bReqPluse_Transmission</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M50</td></tr>
<tr><td>4</td><td>bOUTPUTInstructionExecutionFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M100</td></tr>
<tr><td>5</td><td>bOUTPUTInstructionNormalCompleteFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M101</td></tr>
<tr><td>6</td><td>bOUTPUTInstructionAbnormalCompleteFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M102</td></tr>
<tr><td>7</td><td>bOUTPUTInstructionExecutingFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M104</td></tr>
<tr><td>8</td><td>wTransmissionChannel</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D1</td></tr>
<tr><td>9</td><td>wTransmissionDataCount</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D2</td></tr>
<tr><td>10</td><td>wnTransmissionData</td><td>Word [Signed](0..4)</td><td>...</td><td>VAR_GLOBAL</td><td>D11</td></tr>
<tr><td>11</td><td>wErrorCodeStorageDevice</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D101</td></tr>
<tr><td>12</td><td>bOUTPUTInstructionNormalCompleteFlag2</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M111</td></tr>
<tr><td>13</td><td>bOUTPUTInstructionAbnormalCompleteFlag2</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M112</td></tr>
</table>
</td></tr>
<tr><td>FB</td><td>M+RJ71C24_Output</td><td>Sends data for specified data points.</td><td>—</td></tr>
</table>

For information on the module FB to be used, refer to the following manual.

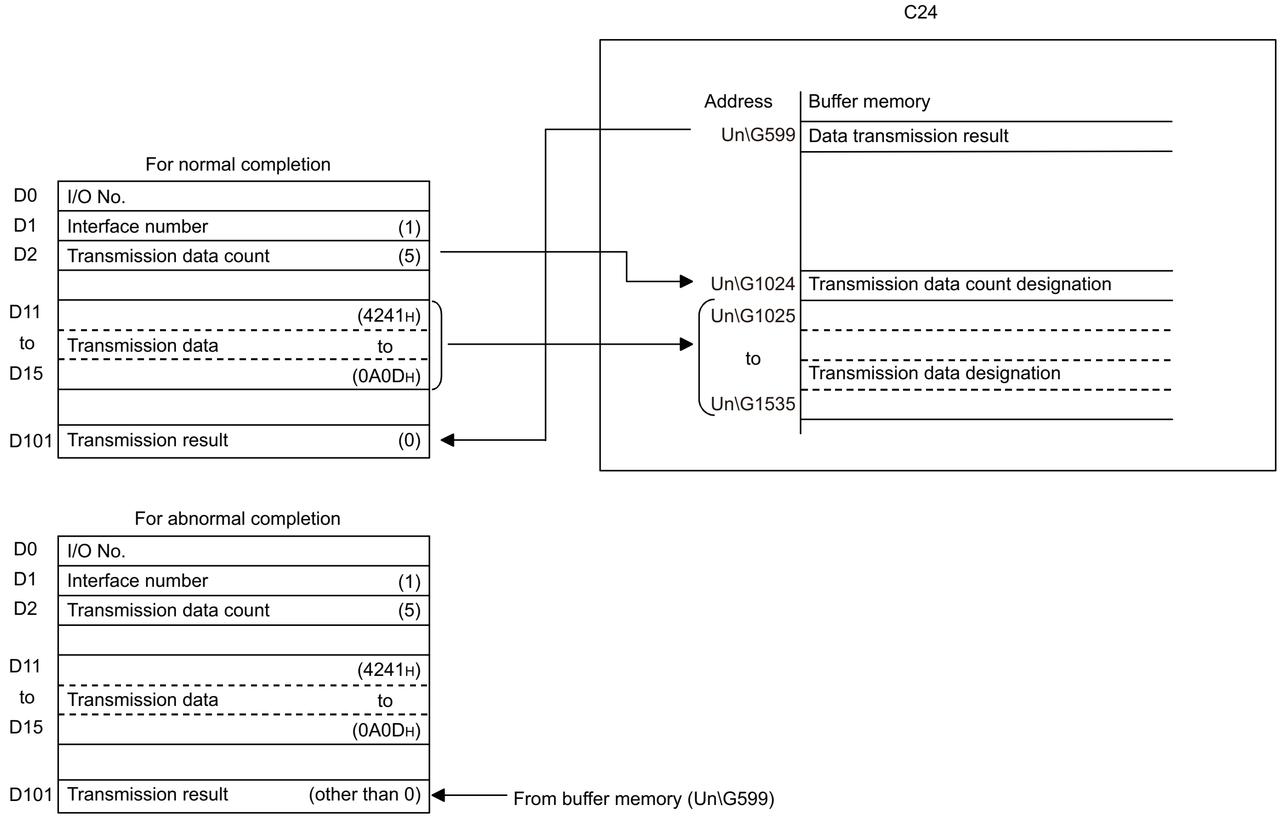
📖MELSEC iQ-R Serial Communication Module Function Block Reference



(0) Convert the transmission command into pulse.

(3) Store the data to be transmitted.
Designate the transmission channel.
Designate the send data quantity.
Set the instruction execution flag.

(69) Reset the completion flag by an external command.



Point

The SPBUSY instruction is used to read the execution status when using a dedicated instruction. (☞ Page 580 Dedicated Instructions)

Restriction

More than one OUTPUT instruction cannot be executed simultaneously on the same interface.
Execute the next OUTPUT instruction only after the execution of the first OUTPUT instruction is completed.

Transmission error detection and check methods

The following explains how to detect and confirm errors that may occur when sending data to target devices.

The following items are considered as the primary causes of errors that may occur during data transmission.

Cause of transmission error	Reference
A timeout occurred for the send monitoring time (timer 2).	Page 265 Send Monitoring Time (timer 2) Setting
The send data quantity exceeding the size that can be stored in the send area is designated.	Page 81 Send area and send data arrangement

Detection/checking of transmission error by a program

■Detecting a transmission error

The following device and input signal turn ON.

- OUTPUT instruction completion device + 1
- 'CH□ Error occurrence' (XE/XF)

■Checking a transmission error

The transmission error code can be checked using the OUTPUT instruction control data ((S1) + 1).

Or, it can be checked by reading the 'data transmission result' (Un\G599/615).

For details on how to check the error code contents and corrective actions, refer to the following section.

☞ Page 519 Error Code List

'CH□ Error occurrence' (XE/XF) turns OFF when the data transmission/reception processing is performed normally.

Detection/checking of transmission error with module and Engineering tool

■Confirmation using the indicator LED

When C24 detects an error, including a transmission error, the ERR LED turns ON.

■Confirmation using Engineering tool

Monitor the 'data transmission result' (Un\G599/615) to check the error code.

Point

When a transmission error occurs, the CPU module may not transmit all the designated data.

Establishing any communication procedure by the user so that normal transmission can be checked by receiving a response to data transmission is recommended.

3.3 Considerations for Data Communication

The following describes considerations for data communications using the nonprocedural protocol.

Factors that initialize the C24 transmission sequence

The initial status means the status where transmission has stopped and received data has been discarded.

C24 transmission sequence initialization factors are as follows.

- When the power is turned ON, the CPU module is reset, or the C24 mode is switched.
- When the received data was cleared.

Framing error on the target device side

When nothing is transmitted from C24 to a target device through the RS-422/485 interface, a framing error may be generated by the noise etc, on the target device side.

Send an arbitrary code first so that the target device identifies the head of the data transmitted from C24.

Check the C24 interface specifications before data communication.

Data communications with target device using the multidrop connection

When communicating data by the multidrop connection between the target device and the CPU module in a 1:n basis, data transmitted by the target device is received by C24.

For the multidrop connection, data reception using a user frame should be performed. (☞ Page 310 Data Reception)

If data is not received by user frame, create a program that ignores (reads and discards) the received data other than the received data addressed to the host station, including the data indicating the target CPU module in the message.

Ex.

STX	Space	Station No. 0	Station No. 2	Data length (BIN data)	Data	CR	LF
(02H)	(20H)	(30H)	(32H)			(0DH)	(0AH)

Retry processing for data reception errors

The data when reception error occurred is discarded and only the data for the normal reception is taken into the C24.

For this reason, the received message may be missing data if reception error occurs.

To assure the correct transmission and reception of data, it is recommended that a normal/error response message is received and a response message reception timeout check is conducted for the data transmitted between the CPU module and the target device, and that measures such as retransmitting the data (transmission retry) is taken when an abnormal response is received or when an error timeout occurs.

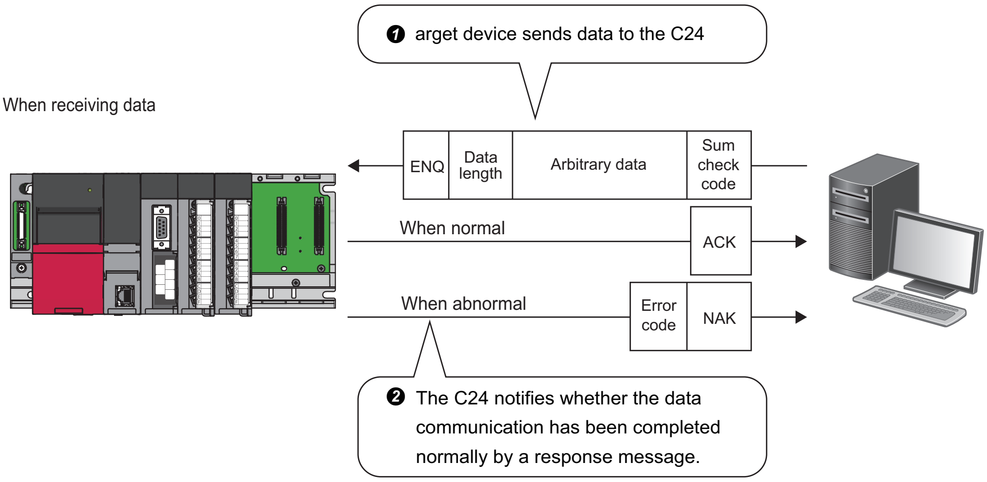
4 DATA COMMUNICATION USING BIDIRECTIONAL PROTOCOL

This communication enables data transfer between a target device and a CPU module using the data format and transmission control procedure for the bidirectional protocol.

The feature of this data communication is that a response message is sent from the data receiving side to the data sending side to inform whether the communication has been completed.

Use this to check if the data communication has been completed normally.

A program for data communication is required for the CPU module.


❶ arget device sends data to the C24
When receiving data
ENQ
Data length
Arbitrary data
Sum check code
When normal
ACK
When abnormal
Error code
NAK
❷ The C24 notifies whether the data communication has been completed normally by a response message.

Point

Communication using the bidirectional protocol can be replaced with those of the predefined protocol.

The Predefined protocol support function of Engineering tool allows the easy replacement.

With the predefined protocol, the required program is only for starting communication according to the specified protocol, and this can reduce the number of steps considerably.

For the communication using the predefined protocol, refer to the following chapter.

☞ Page 30 DATA COMMUNICATION USING PREDEFINED PROTOCOL

Parameter setting for data communication using bidirectional protocol

Set the transmission setting and the communication protocol, etc. for a target device.

Set "Bidirectional protocol" for "Communication protocol setting".

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

Configure other settings according to the system being used.

For details of the settings, refer to the following chapter.

☞ Page 454 PARAMETER SETTING

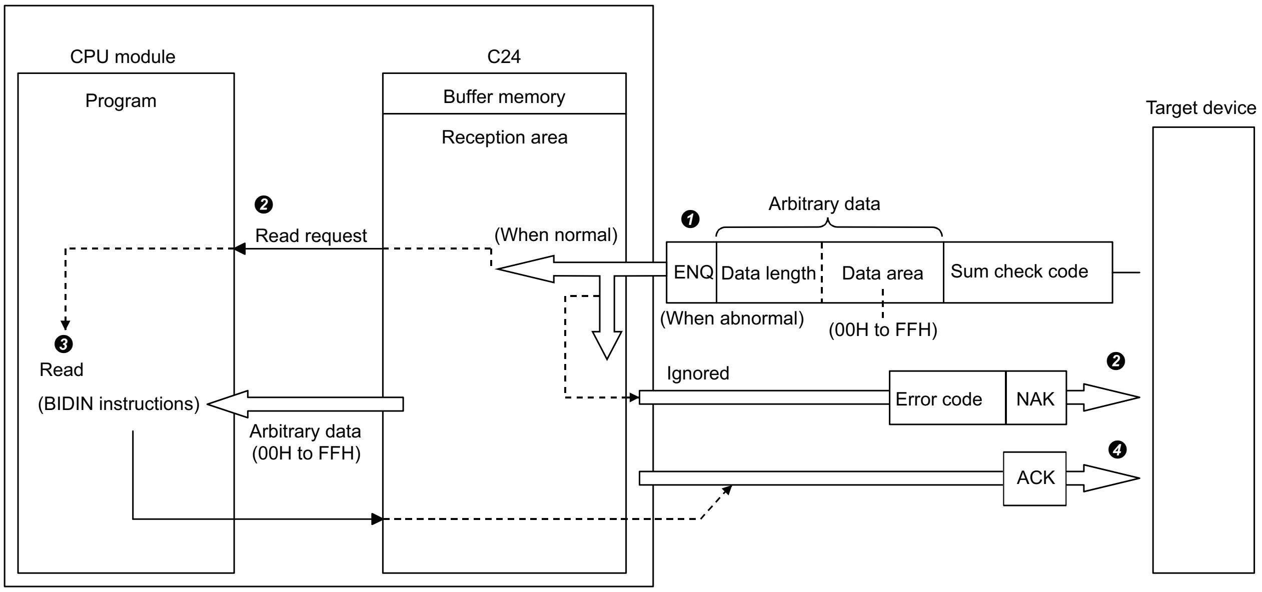
4.1 Receiving Data from Target Device

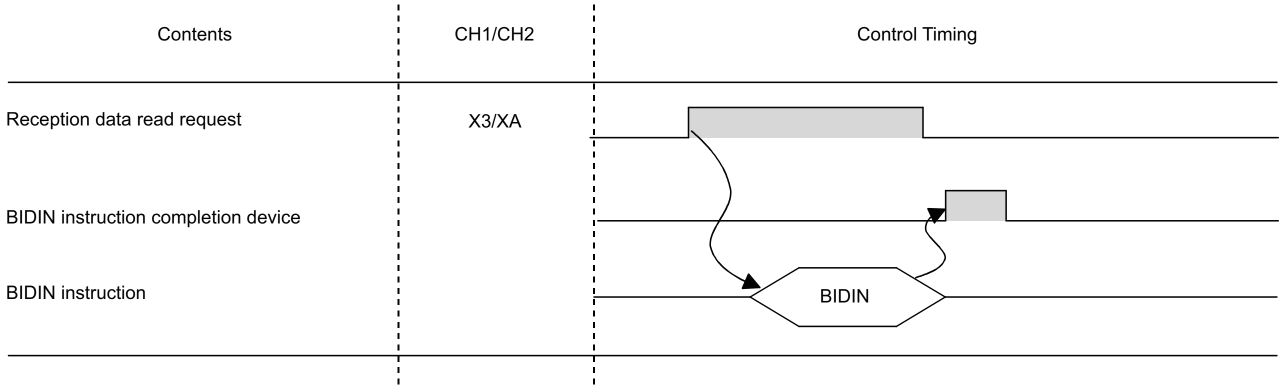
This section explains data reception from a target device.

Receiving methods

The following shows how to receive data sent from the target device using the bidirectional protocol.

Since the size of each message to be received is recognized using the "data length" information contained in the message, messages of any size can be sent from the target device.




Contents
CH1/CH2
Control Timing
Reception data read request
X3/XA
BIDIN instruction completion device
BIDIN instruction
BIDIN

❶When the control code ENQ is received, C24 starts data reception processing.

When data portion for the data length is received, C24 ends the reception processing of arbitrary data portion.

- When sum check code is enabled in the transmission setting of Engineering tool, C24 checks whether the message is normal or abnormal using the sum check code received immediately after the message and the sum check code calculated by internal processing.
- When sum check code is disabled in the transmission setting of Engineering tool, messages are not checked using the sum check codes.

❷ When the message has normally been received, the 'CH□ reception data read request' (X3/XA) turns ON.

- If an error has been detected when receiving the message, C24 transmits an abnormal completion response message (NAK message) to the target device.
- The reception data read request will not be sent to the CPU module. (The 'CH□ reception abnormal detection' (X4/XB) does not turn ON.)

❸ C24 stores the control data in the device designated with the BIDIN instruction, and then executes the BIDIN instruction.

- Received data is read from the receive area in the buffer memory.

❹ When the BIDIN instruction has been executed, C24 transmits a normal completion response (ACK message) to the target device.

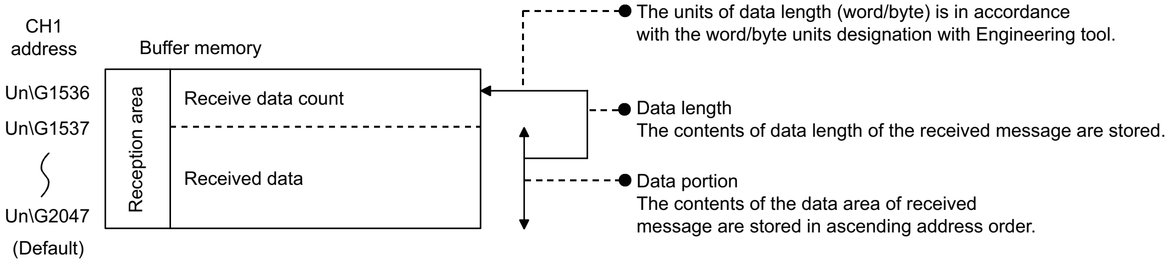
Receive area, receive data arrangement and contents

This section shows the receive area and the receive data arrangement for performing data reception using the bidirectional protocol.

Receive area

The receive area is a memory that stores the data length (receive data quantity) and data portion received from a target device.

By default, the receive area is assigned to the addresses, Un\G1536 to 2047 (CH1 side) and Un\G2560 to 3071 (CH2 side).



CH1 address
Buffer memory
Un\G1536
Un\G1537
Un\G2047
(Default)
Reception area
Receive data count
Received data
The units of data length (word/byte) is in accordance with the word/byte units designation with Engineering tool.
Data length
The contents of data length of the received message are stored.
Data portion
The contents of the data area of received message are stored in ascending address order.

The position and size of the receive area in the buffer memory can be changed with Engineering tool in accordance with the specifications of the target device and the receive data length. (☞ Page 454 PARAMETER SETTING)

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

Configure the settings as follows:

- Designate the start address for the area to be used as receive area in the user setting area (Un\G1024 to 6911, Un\G9728 to 16383) to "Receiving buffer memory start address designation".
- Designate the length of area to be used as the receive area in the user setting area (Un\G1024 to 6911, Un\G9728 to 16383) by address count (0001H to 1A00H) to "Receiving buffer memory size specification".

When changing the position and size of the receive area in the buffer memory, if any of the functions below are used concurrently, make sure that the addresses of the receive area do not overlap with the buffer memory addresses assigned for storing send/receive data with the functions.

- MC protocol buffer memory read/write function
- MC protocol on-demand function
- Predefined protocol function (Send/receive data storage area)
- Nonprocedural protocol transmission/receive function
- Bidirectional protocol transmission/receive function
- Send/receive data monitoring function

Set the size of data per data transmission from the target device to C24 to be equal to or smaller than the size of the receive data storage area in the buffer memory.

- (Receive data storage area) ≥ (Amount of data in any data portion to be transmitted from the target device)

To transmit data whose size is larger than the receive data storage area, increase the receive area, or divide the send data before sending.

Point

The position and size of the receive area can be changed in the buffer memory.

- 'Receiving buffer memory start address designation' (Un\G166/326)
- 'Receive buffer memory length designation' (Un\G167/327)

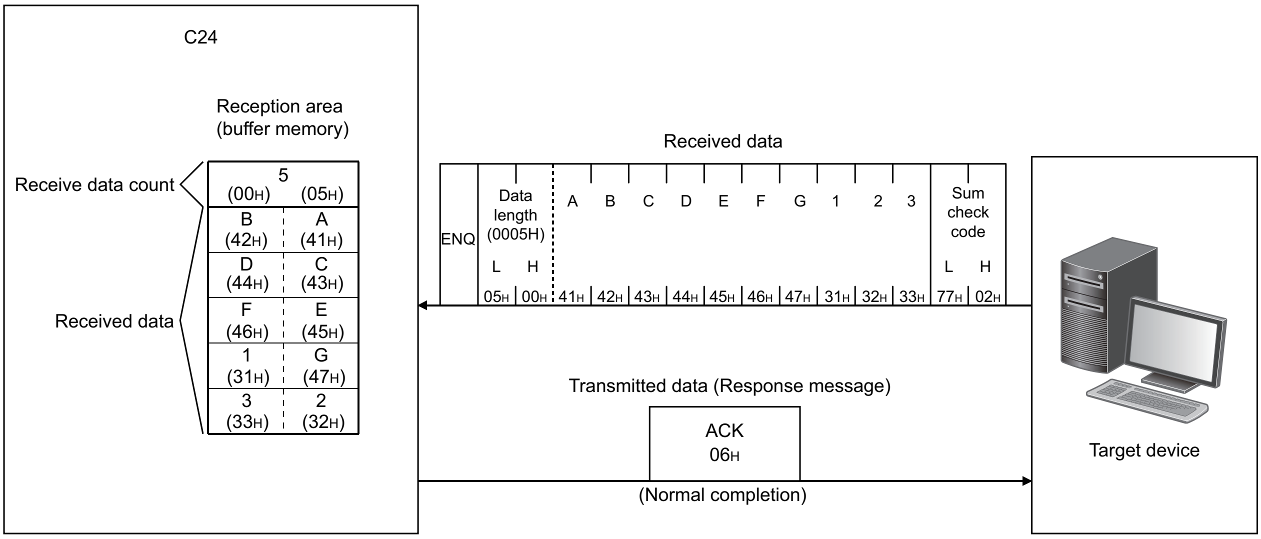
Receive data arrangement

The following shows the example of the arrangement of data received from the target device when it is stored in the receive area.

- The received message is stored to the C24 buffer memory (Receive data storage area).
- The data is stored in the receive data storage area in the order of the data reception from low address, (L) → (H), to next address, (L) → (H).

Ex.

When received data "ABCDEFG123" was stored (for the CH1 side)



When the data length unit is in bytes, 00H is stored in the higher byte of the last data storage position in the receive area when the data length of the message is an odd byte.

Contents of received data

The following shows the contents of data to be received, including data to be transmitted. (☞ Page 100 Send area and send data arrangement)

■Control codes

The types of control codes are shown in the table below.

Signal name	Code (hexadecimal)	Description	Application
ENQ	05H	Enquiry	Code for starting data transmission
ACK	06H	Acknowledge	Response code to the target side when data was received normally
NAK	15H	Negative Acknowledge	Response code to the target side when data could not be received normally (Error code is sent immediately after the response code.)

- Target device → C24

  C24 checks and processes the control code received.

  It cannot be read from the program.
- C24 → target device

  C24 adds the control code to be transmitted.

■Data length

This indicates the number of bytes or words for the data portion of the message.

The unit of data length (words/bytes) is in accordance with the word/byte units designation with Engineering tool.

- Target device → C24

  C24 checks the length of the data received.

  When the reception is normal, C24 stores the received data in the receive data storage area in the C24 buffer memory, assigning the first byte as the lower byte (L).
- C24 → target device

  The length of data to be transmitted is the send data quantity designated by the program using the BIDOUT instruction.

  C24 transmits this value as is from the lower byte (L).

■Data portion

This consists of an array of original one-byte data to be transmitted to the target device side, which can handle 00H to FFH code data.

- Target device→C24

  If the receive data portion is normal, C24 takes the codes as is and stores them in the receive data storage area in sequence, beginning with the lowest address.

  The storage size is in accordance with the data length in the message and the word/byte units designation with Engineering tool.
- C24 → target device

  The portion of data to be transmitted is the send data designated by the program using the BIDOUT instruction.

  C24 transmits the data for the send data quantity according to the word/byte units designation with Engineering tool from the lowest address in the send data specification area using the code as is.

■Sum check code

The sum check code expresses the numeric value for the lower two bytes (16 bits) of the results (sum) from the addition of the data length and the data portion in the message as binary code data.

When sum check code is enabled in the transmission setting of Engineering tool, it is necessary to add a sum check code at the end of the message.

- Target device→ C24

  C24 checks and processes the sum check code received. It cannot be read from the program.

  When "None" is specified for "Sum check code", once data portion of message corresponding to the data length is received, the succeeding data received will be ignored up to the control code that follows.

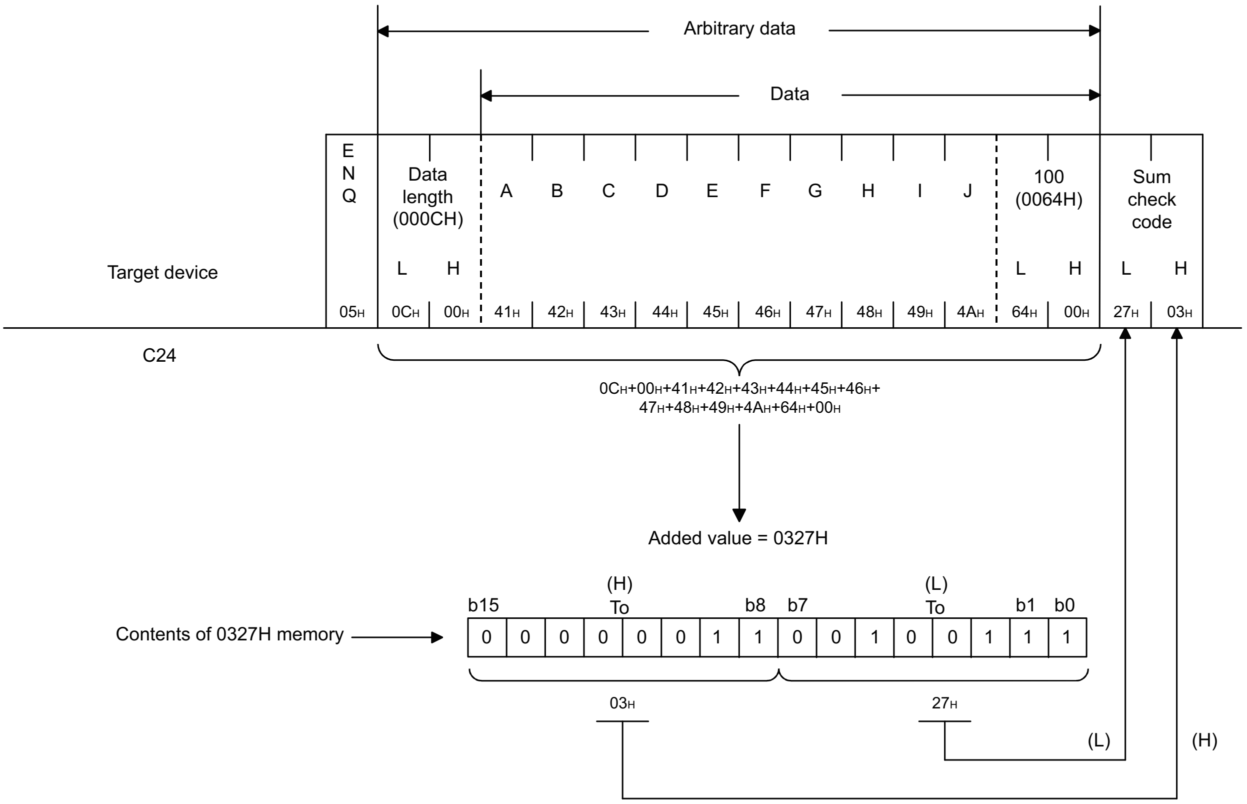
- C24 → target device

  C24 calculates and adds the sum check code to be transmitted.

  When "None" is specified, no sum check code is transmitted.

Ex.

Sum check code when "ABCDEFGHIJ " and the numeric value 100 are sent as data (when the unit is in bytes.)



■Error code

The error code indicates the definition of the error during NAK response.

For details on the error codes, refer to the following section.

☞Page 519 Error Code List

- Target device→ C24

  For error codes from the target device, transmit the codes specified by the user.

  The error codes (0022H to 005FH) that are not used by C24 can be used.

  C24 stores the received error codes as a completion status in the control code for the BIDOUT instruction. They are also stored in the 'data transmission result' (Un\G599).

- C24 → target device

  C24 adds an error code.

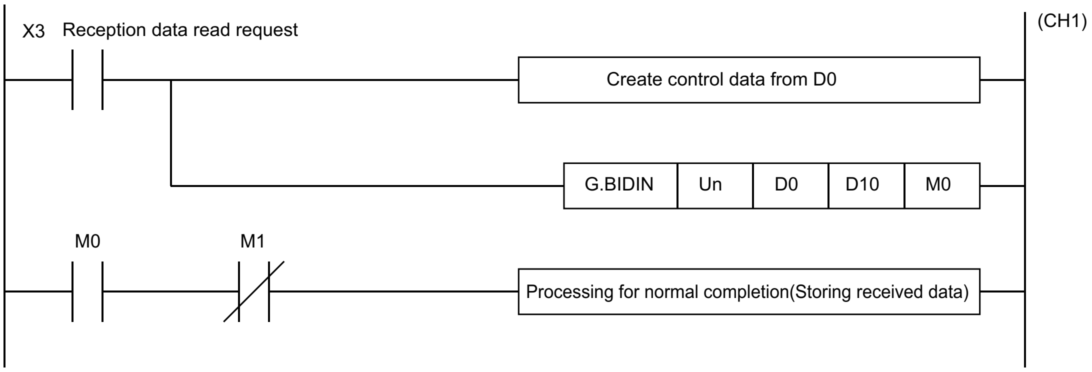
  When an error code is transmitted, C24 writes the same error code in the 'data reception result' (Un\G600).

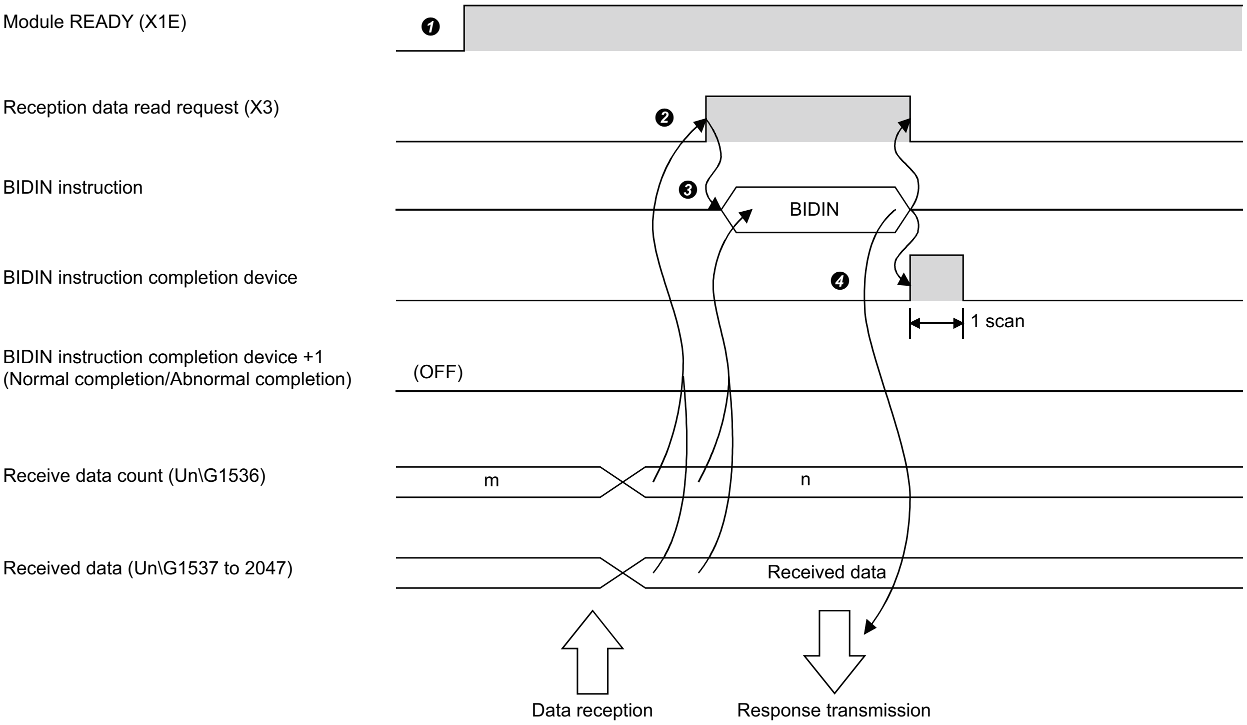
Program for data reception

Program example for data reception

The following shows the program for data reception using the bidirectional protocol and program example.

For details on the BIDIN instruction, refer to the dedicated instructions. (☞ Page 580 Dedicated Instructions)


X3 Reception data read request
(CH1)
Create control data from D0
G.BIDIN
Un
D0
D10
M0
M0
M1
Processing for normal completion(Storing received data)


Module READY (X1E)
Reception data read request (X3)
BIDIN instruction
BIDIN instruction completion device
BIDIN instruction completion device +1
(Normal completion/Abnormal completion)
Receive data count (Un\G1536)
Received data (Un\G1537 to 2047)
❶
❷
❸
❹
BIDIN
1 scan
(OFF)
m
n
Received data
Data reception
Response transmission

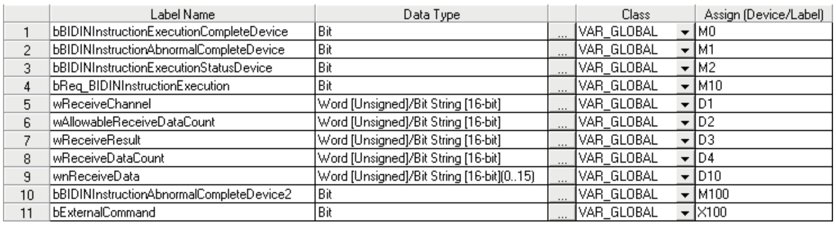
❶ Start the programmable controller of the host station. The setting values specified with an Engineering tool are stored in C24.

❷ When data is received from the target device, the 'CH1 reception data read request' (X3) turns ON.

❸ After the control data for the BIDIN instruction is stored in the device, the sequence program executes the BIDIN instruction and reads the receive data.

❹ When the reading of receive data is completed, a response message (ACK message when the operation was completed normally) is sent and the device that has completed the BIDIN instruction turns ON.

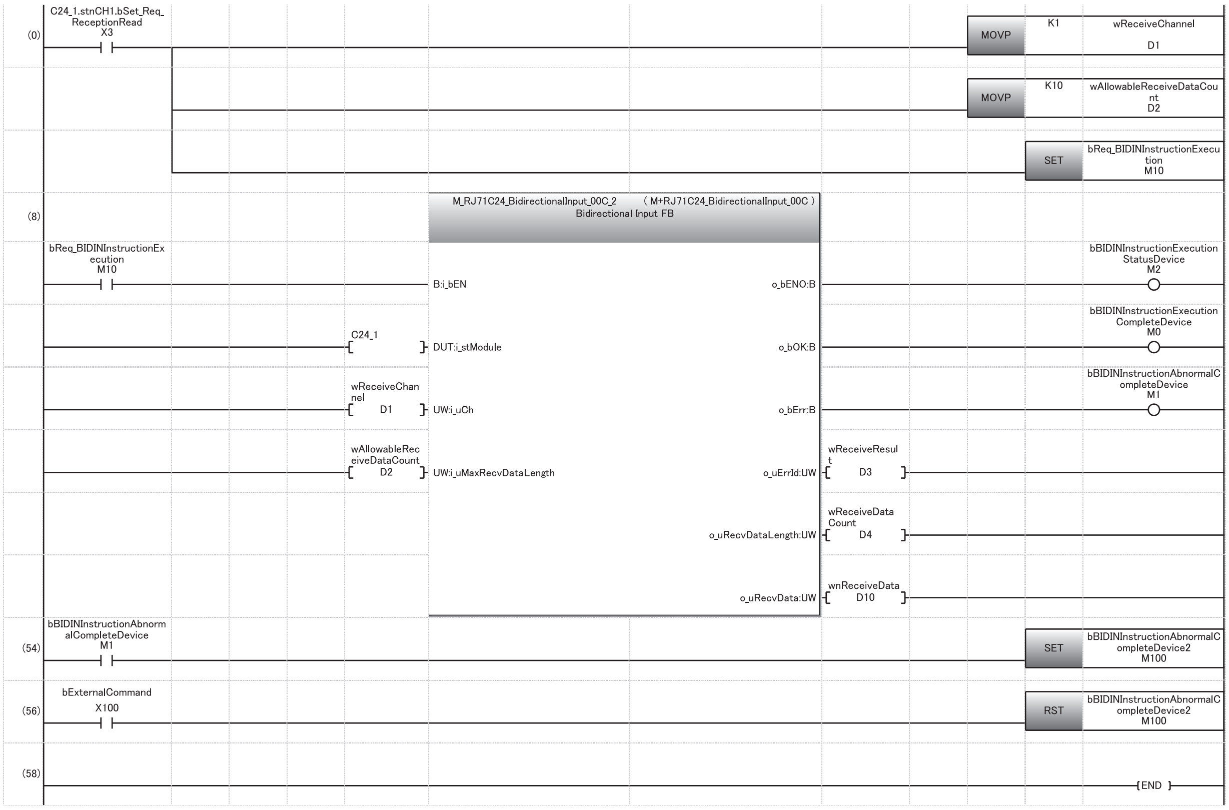
■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_BidirectionalInput	Reads the received data.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bBIDINInstructionExecutionCompleteDevice	Bit	...	VAR_GLOBAL	M0
2	bBIDINInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
3	bBIDINInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M2
4	bReq_BIDINInstructionExecution	Bit	...	VAR_GLOBAL	M10
5	wReceiveChannel	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
6	wAllowableReceiveDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2
7	wReceiveResult	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D3
8	wReceiveDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D4
9	wnReceiveData	Word [Unsigned]/Bit String [16-bit](0..15)	...	VAR_GLOBAL	D10
10	bBIDINInstructionAbnormalCompleteDevice2	Bit	...	VAR_GLOBAL	M100
11	bExternalCommand	Bit	...	VAR_GLOBAL	X100

For information on the module FB to be used, refer to the following manual.

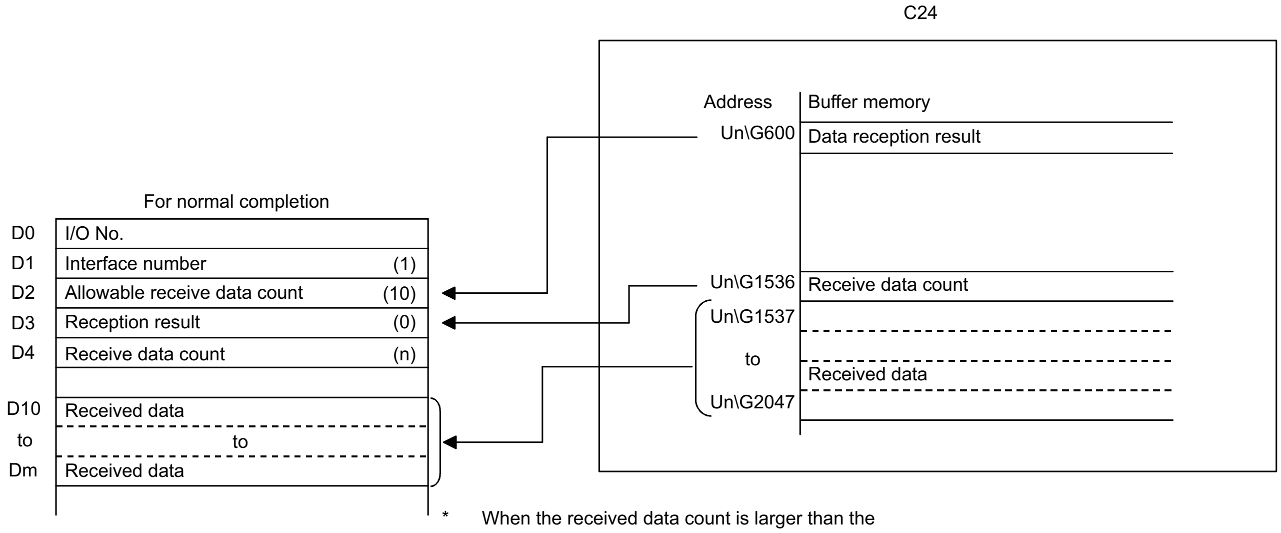
MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
C24_1.stnCH1.bSet_Req_ReceptionRead X3
MOVP K1 wReceiveChannel D1
MOVP K10 wAllowableReceiveDataCount D2
SET bReq_BIDINInstructionExecution M10
(8)
M_RJ71C24_BidirectionalInput_00C_2 (M+RJ71C24_BidirectionalInput_00C)
Bidirectional Input FB
bReq_BIDINInstructionExecution M10
B:i_bEN
o_bENO:B
bBIDINInstructionExecutionStatusDevice M2
C24_1
DUT:i_stModule
o_bOK:B
bBIDINInstructionExecutionCompleteDevice M0
wReceiveChannel D1
UW:i_uCh
o_bErr:B
bBIDINInstructionAbnormalCompleteDevice M1
wAllowableReceiveDataCount D2
UW:i_uMaxRecvDataLength
o_uErrId:UW
wReceiveResult D3
o_uRecvDataLength:UW
wReceiveDataCount D4
o_uRecvData:UW
wnReceiveData D10
(54)
bBIDINInstructionAbnormalCompleteDevice M1
SET bBIDINInstructionAbnormalCompleteDevice2 M100
(56)
bExternalCommand X100
RST bBIDINInstructionAbnormalCompleteDevice2 M100
(58)
END

(0) Specify the reception channel.
Specify the receive data quantity.
Set the BIDIN instruction execution direction.

(54) Abnormal completion

(56) Reset the abnormal completion flag by an external command.



When the received data count is larger than the allowable received data count, only the data up to the allowable received data count will be stored and the rest of the data will be discarded.

Point

- Received data can also be read by using an interrupt programs. (☞ Page 251 RECEIVING DATA WITH AN INTERRUPT PROGRAM)
  Note that if the reading of data received from the same interface is performed, it is not possible to combine reading data by the main program and that by the interrupt program.
  Use either of the above programs to read the data received.
- The SPBUSY instruction is used to read the execution status when using a dedicated instruction. (☞ Page 580 Dedicated Instructions)

Restriction

- More than one BIDIN instruction cannot be executed simultaneously on the same interface.
  Execute the next BIDIN instruction only after the execution of the first BIDIN instruction is completed.
- When the send/receive data quantity is set in byte units and the receive data quantity is an odd byte when requesting the reading of received data to the CPU module, 00H is stored in the higher byte of the last data storage position in the receive area where received data is stored.

Reception error detection and check methods

The following explains how to detect and confirm errors that may occur when receiving data from target devices.

The following items are considered as the primary causes of errors that may occur during data reception.

Cause of reception error	Reference
A transmission error occurred due to noise.	MELSEC iQ-R Serial Communication Module User's Manual(Startup)
A timeout occurred for the no-reception monitoring time (timer 0).	Page 260 No-reception Monitoring Time (timer 0) Setting
A timeout occurred for the send monitoring time (timer 2).	Page 265 Send Monitoring Time (timer 2) Setting
Data that could not be converted with the ASCII-BIN conversion was included.	Page 391 ASCII-BIN Conversion for Data Communicated with Bidirectional Protocol
The receive data length was too long to be stored in the receive area.	Page 91 Receive area, receive data arrangement and contents
Simultaneous transmissions occurred.	Page 107 Processing when Simultaneous Transmission is Performed during Full-Duplex Communications

Confirmation using a program

■Detecting a reception error

'CH□ Error occurrence' (XE/XF) turns ON.

■Checking a reception error

Read the 'data reception result' (Un\G600/616), and check it.

For details on how to check the error code contents and corrective actions, refer to the following section.

☞ Page 519 Error Code List

'CH□ Error occurrence' (XE/XF) turns OFF when the data transmission/reception processing is performed normally.

Confirmation using the module and Engineering tool

■Confirmation using the indicator LED

When C24 detects an error, including a transmission error, the ERR LED turns ON.

■Confirmation using Engineering tool

Read the 'data reception result' (Un\G600/616), and check the error code.

Considerations

- All of the received data for which an error was detected is ignored and a response message indicating an abnormal completion (NAK message) is transmitted to the target device. The 'CH□ reception abnormal detection' (X4/XB) does not turn ON.
  When an error is detected during a message reception, C24 does not request the CPU module to read the received data.
- The receive area in the buffer memory will contain the data received normally just before the error occurrence. (Data will not be rewritten.)

Receive data clear

Data communications using the bidirectional protocol must be performed after a response message is received in reply to the preceding data transmission.

If C24 detects an error while receiving data, it transmits an NAK message (response message) to the target device after the completion of the data reception and ignores the data being received when the error was detected.

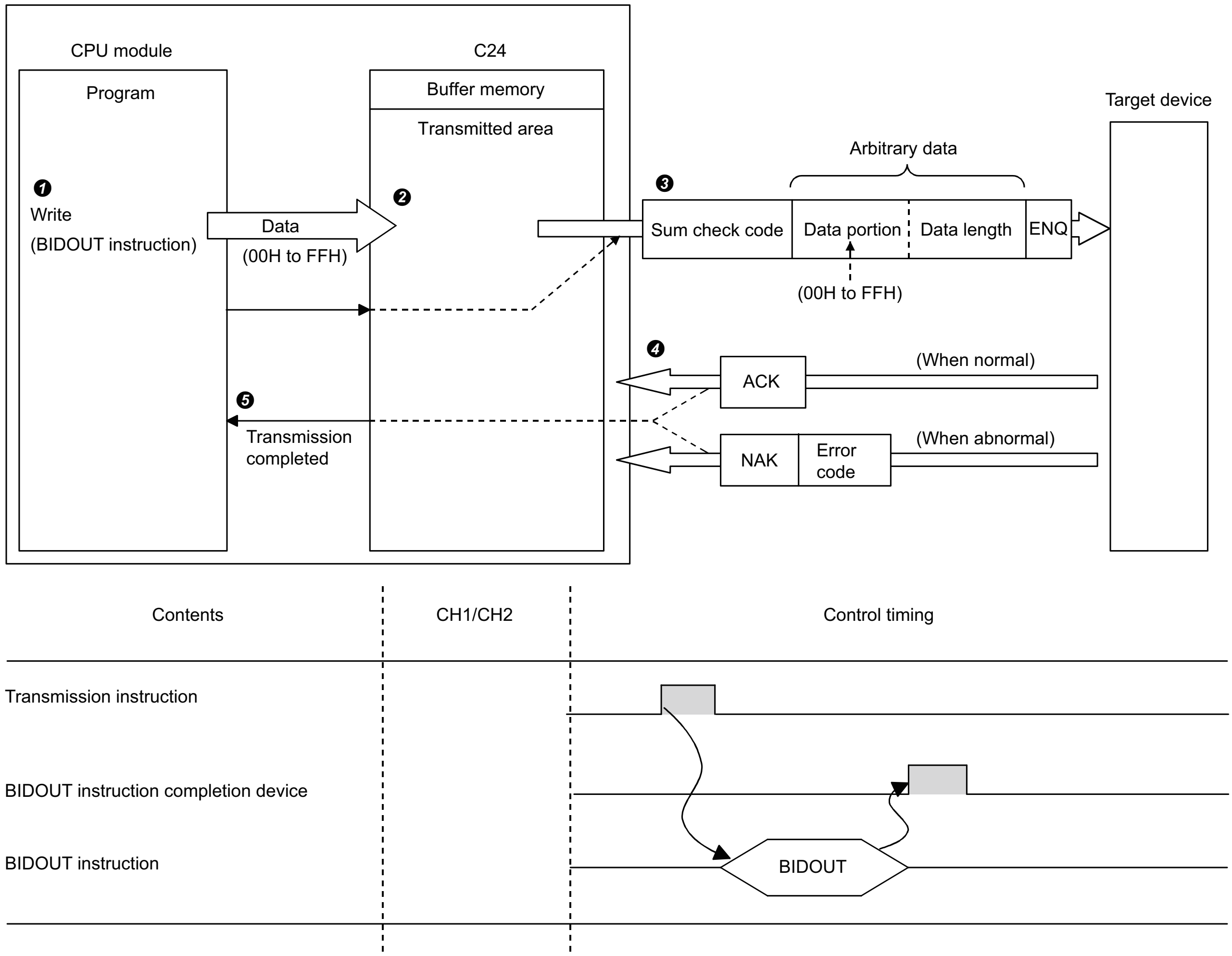
Therefore, the received data does not have to be cleared.

4.2 Transmitting Data to Target Device

This section explains the data transmission from the CPU module to a target device.

Sending methods

The following shows the method of transmitting data to the target device through data communication using the bidirectional protocol.



❶ Data such as control data and send data are stored in the device designated with the BIDOUT instruction, and then the BIDOUT instruction is executed.

❷ The send data quantity and send data are written into the send area in the buffer memory.

❸ C24 transmits data by adding the control code ENQ to the head of the data.

- When sum check code is enabled in the transmission setting with Engineering tool, C24 adds the sum check code calculated by internal processing to the end of the message and then transmits the message.
- When sum check code is disabled in the transmission setting with Engineering tool, sum check code is not transmitted.

❹ A response for the data transmission (for normal completion: ACK message; for abnormal completion: NAK message) is received.

❺ Once the transmission processing of C24 is completed, the execution of the BIDOUT instruction is completed.

Send area and send data arrangement

This section explains the send area and the send data arrangement for performing data transmission using the bidirectional protocol.

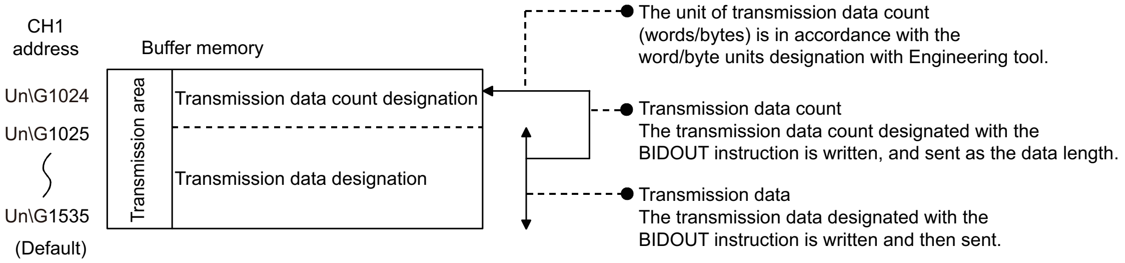
For details on the contents of send data, refer to the following section.

☞ Page 93 Contents of received data

Send area

The send area is a memory area for storing the data and the data quantity that are transmitted from the CPU module to a target device via C24.

The send area is assigned to the address, Un\G1024 to 1535 (CH1 side), Un\G2048 to 2559 (CH2 side) by default.



CH1 address
Buffer memory
Un\G1024
Un\G1025
Un\G1535
(Default)
Transmission area
Transmission data count designation
Transmission data designation
● The unit of transmission data count (words/bytes) is in accordance with the word/byte units designation with Engineering tool.
● Transmission data count
The transmission data count designated with the BIDOUT instruction is written, and sent as the data length.
● Transmission data
The transmission data designated with the BIDOUT instruction is written and then sent.

The position and size of the send area in the buffer memory can be changed with Engineering tool in accordance with the specifications of the target device and the send data length. (☞ Page 454 PARAMETER SETTING)

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

Configure the settings as follows:

- Designate the start address for the area to be used as the send area in the user setting area (Un\G1024 to 6911, Un\G9728 to 16383) to "Send buffer memory start address designation".
- Designate the length of the area to be used as the send area in the user setting area (Un\G1024 to 6911, Un\G9728 to 16383) by address count (0001H to 1A00H) to "Send buffer memory size specification".

When changing the position and size of the send area in the buffer memory, if any of the functions below are used concurrently, make sure not to overlap with the buffer memory addresses assigned for storing send/receive data handled with the functions.

- MC protocol buffer memory read/write function
- MC protocol on-demand function
- Predefined protocol function (Send/receive data storage area)
- Nonprocedural protocol transmission/receive function
- Bidirectional protocol transmission/receive function
- Send/receive data monitoring function

Set the size of data per data transmission from the target device to C24 to be equal to or smaller than the size of the send data storage area in the buffer memory.

- (Send data storage area) ≥ (Amount of arbitrary data portion to be transmitted from the CPU module)

To transmit data whose size is larger than the send data storage area, increase the send area, or divide the send data before sending.

Point

The position and size of send area can be changed in the buffer memory.

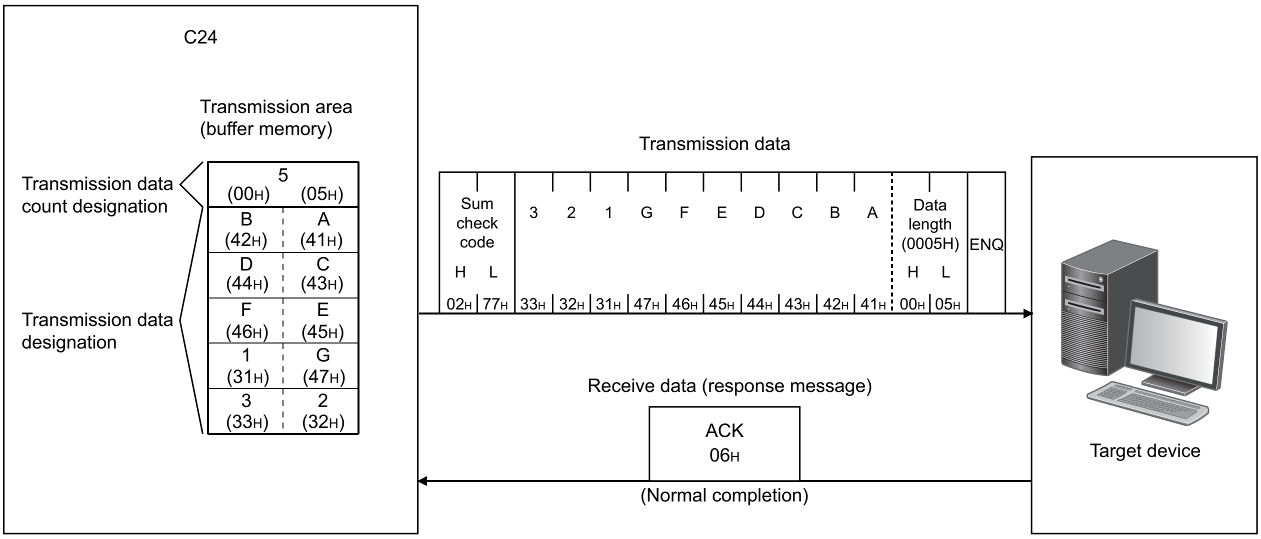
- 'Send buffer memory start address designation' (Un\G162/322)
- Send buffer memory size specification (Un\G163/323)

Send data arrangement

The following example shows an arrangement of the send data to be sent to the target device when storing it in the send area.

Ex.

When transmitting "ABCDEFG123"

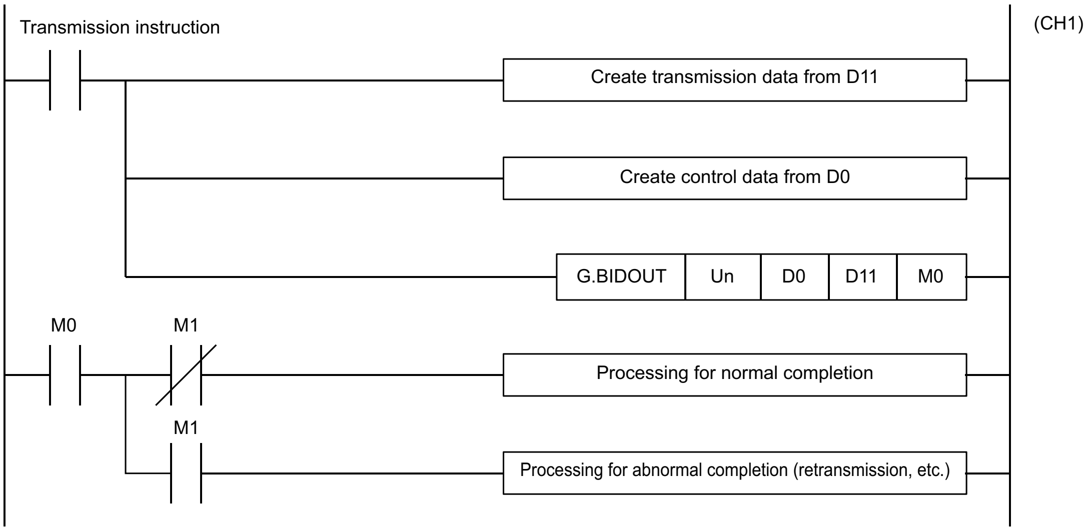


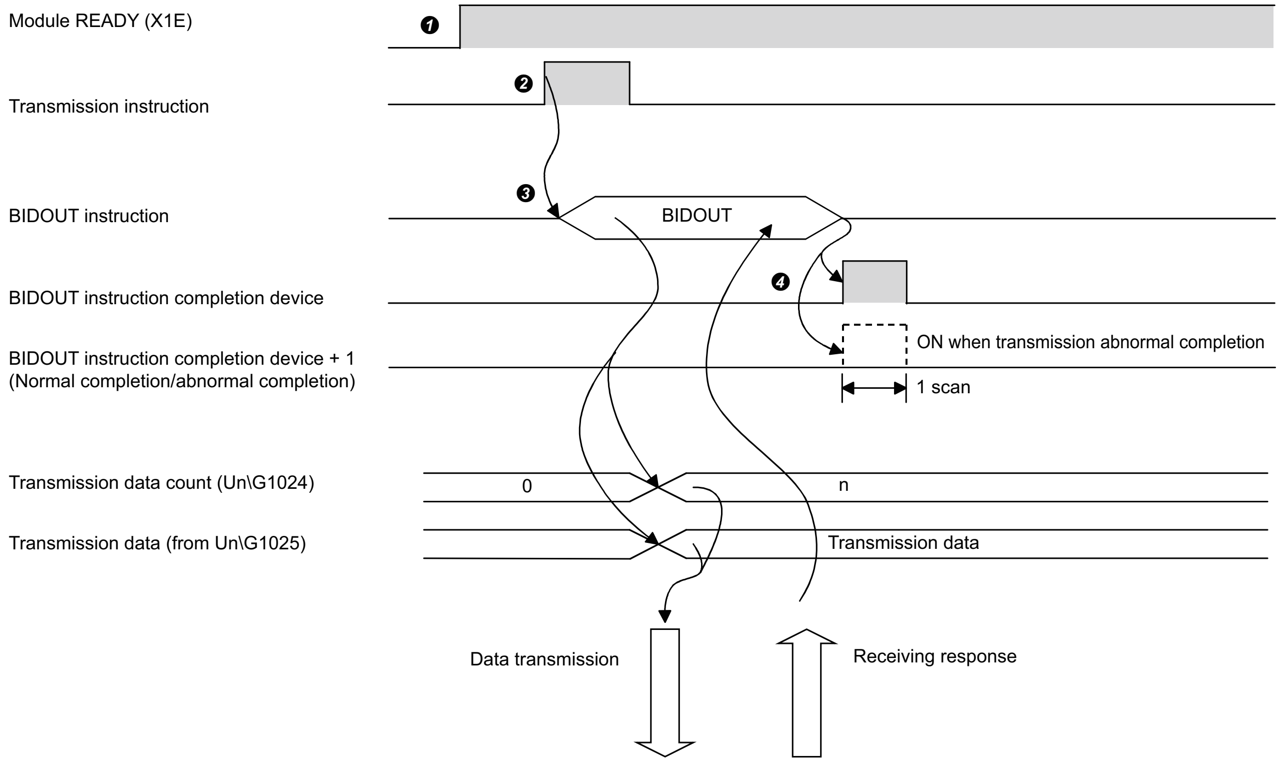
Program for data transmission

Program example for data transmission

The following shows the program for data transmission and program example.

For details on the BIDOUT instruction, refer to the dedicated instructions. (☞ Page 580 Dedicated Instructions)


Transmission instruction
(CH1)
Create transmission data from D11
Create control data from D0
G.BIDOUT
Un
D0
D11
M0
M0
M1
Processing for normal completion
M1
Processing for abnormal completion (retransmission, etc.)


Module READY (X1E)
❶
Transmission instruction
❷
BIDOUT instruction
❸
BIDOUT
BIDOUT instruction completion device
❹
BIDOUT instruction completion device + 1
(Normal completion/abnormal completion)
ON when transmission abnormal completion
1 scan
Transmission data count (Un\G1024)
0
n
Transmission data (from Un\G1025)
Transmission data
Data transmission
Receiving response

❶ Start the programmable controller on the host station.

The values specified with Engineering tool are stored in C24.

❷ Inputs the user data transmission command signal.

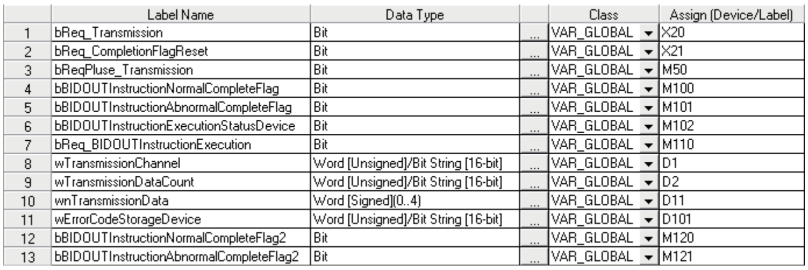
❸ Execute the BIDOUT instruction after storing the send data and the control data for the BIDOUT instruction in the device. Data is sent when the BIDOUT instruction is executed.

❹ A response for the data transmission (for normal completion: ACK message; for abnormal completion: NAK message) is received.

❺ When the response is received, C24 transmission processing completes and the device that has completed the BIDOUT instruction turns ON.

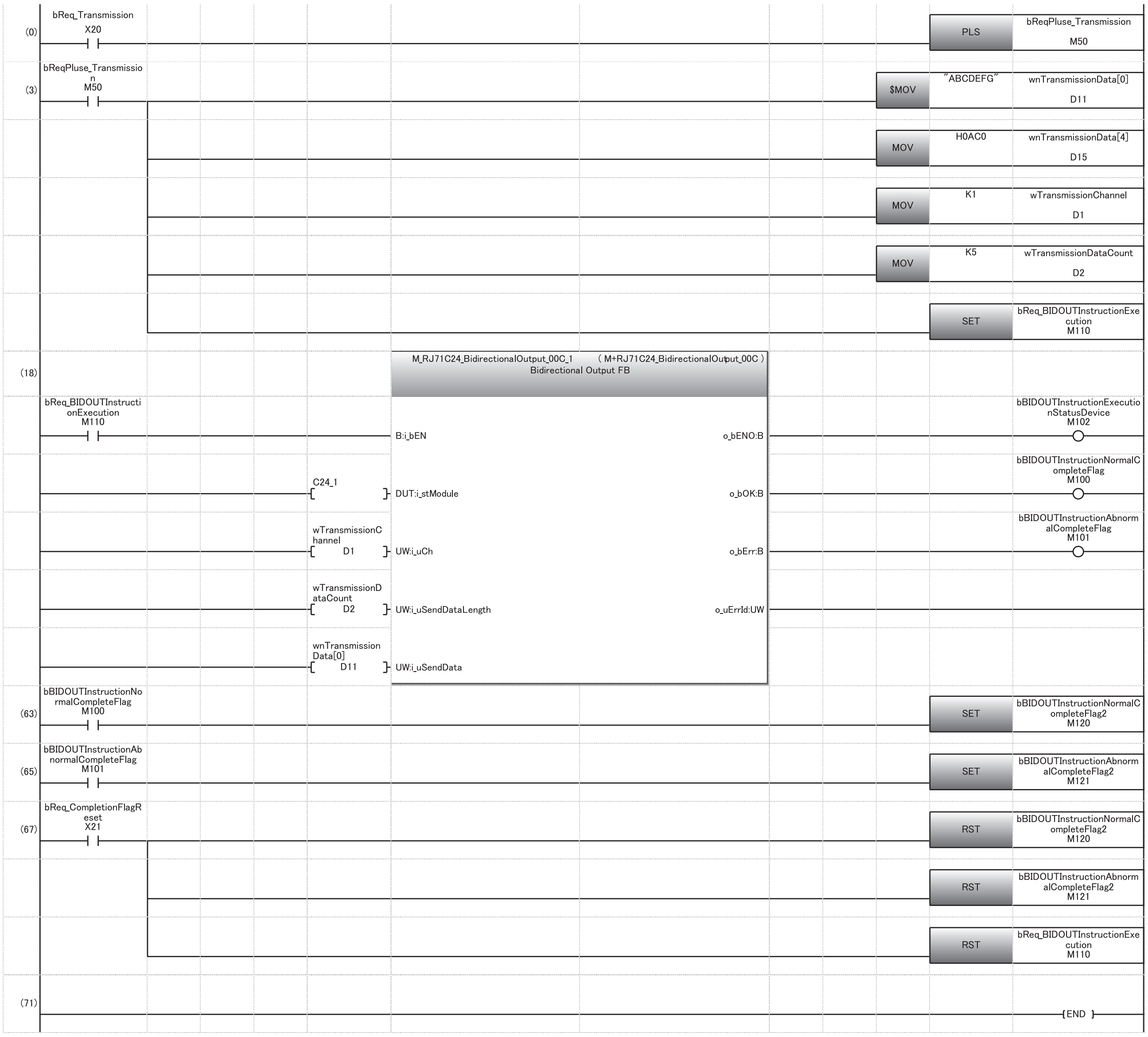
When the BIDOUT instruction ends abnormally due to the reception of an NAK message, the complete device +1 (abnormal completion signal) turns ON and the error code is stored in the control data completion status (S1+1).

■Program example

<table>
<tr><th>Category</th><th>Label name/FB name</th><th>Description</th><th>Device</th></tr>
<tr><td>Module label</td><td>C24_1</td><td>Target module</td><td>—</td></tr>
<tr><td>Label to be defined</td><td colspan="3">Define the global label as shown below.

<table>
<tr><th></th><th>Label Name</th><th>Data Type</th><th></th><th>Class</th><th>Assign (Device/Label)</th></tr>
<tr><td>1</td><td>bReq_Transmission</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>X20</td></tr>
<tr><td>2</td><td>bReq_CompletionFlagReset</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>X21</td></tr>
<tr><td>3</td><td>bReqPluse_Transmission</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M50</td></tr>
<tr><td>4</td><td>bBIDOUTInstructionNormalCompleteFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M100</td></tr>
<tr><td>5</td><td>bBIDOUTInstructionAbnormalCompleteFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M101</td></tr>
<tr><td>6</td><td>bBIDOUTInstructionExecutionStatusDevice</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M102</td></tr>
<tr><td>7</td><td>bReq_BIDOUTInstructionExecution</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M110</td></tr>
<tr><td>8</td><td>wTransmissionChannel</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D1</td></tr>
<tr><td>9</td><td>wTransmissionDataCount</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D2</td></tr>
<tr><td>10</td><td>wnTransmissionData</td><td>Word [Signed](0..4)</td><td>...</td><td>VAR_GLOBAL</td><td>D11</td></tr>
<tr><td>11</td><td>wErrorCodeStorageDevice</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D101</td></tr>
<tr><td>12</td><td>bBIDOUTInstructionNormalCompleteFlag2</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M120</td></tr>
<tr><td>13</td><td>bBIDOUTInstructionAbnormalCompleteFlag2</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M121</td></tr>
</table></td></tr>
<tr><td>FB</td><td>M+RJ71C24_BidirectionalOutput</td><td>Sends data for specified data points.</td><td>—</td></tr>
</table>

For information on the module FB to be used, refer to the following manual.

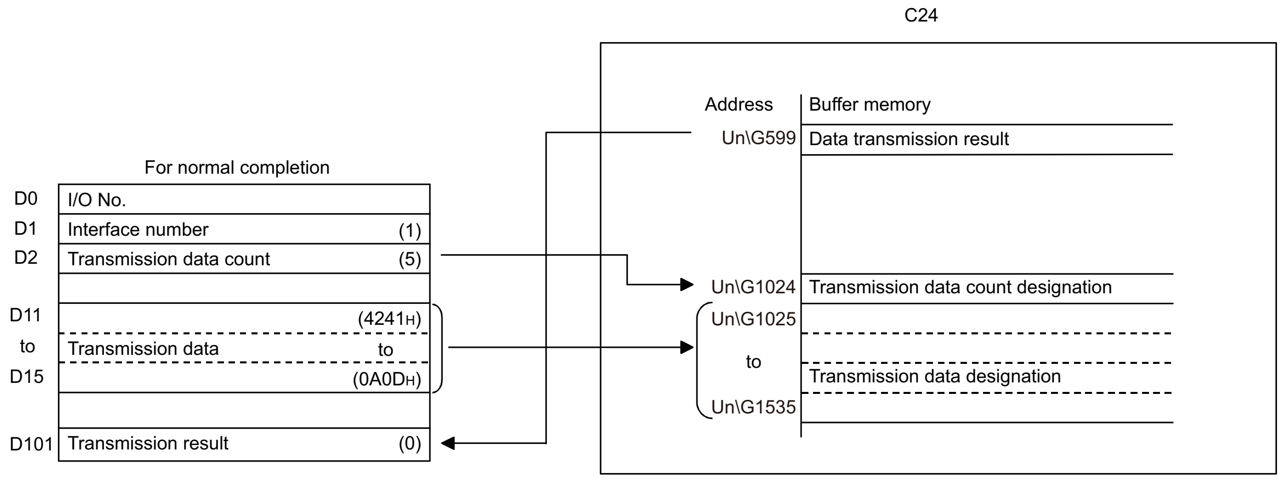
📖MELSEC iQ-R Serial Communication Module Function Block Reference

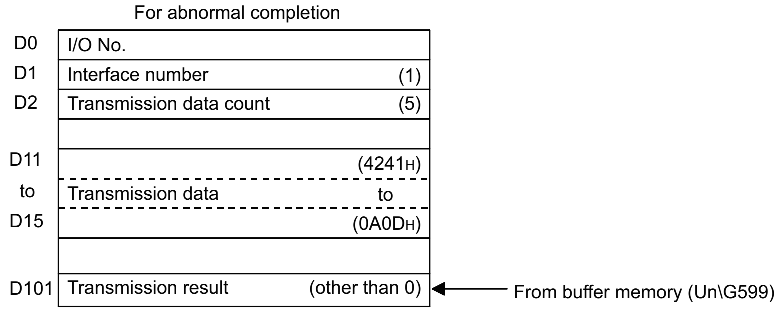


(0) Convert the transmission command into pulse.

(3) Store the data to be transmitted.
Designate the transmission channel.
Designate the send data quantity.
Set the instruction execution flag.

(67) Reset the completion flag by an external command.





Point

The SPBUSY instruction is used to read the execution status when using a dedicated instruction. (☞ Page 580 Dedicated Instructions)

Restriction

More than one BIDOUT instruction cannot be executed simultaneously on the same interface.
Execute the next BIDOUT instruction only after the execution of the first BIDOUT instruction is completed.

Transmission error detection and check methods

The following explains how to detect and confirm errors that may occur when transmitting data to target devices.

The following items are considered as the primary causes of errors that may occur during data transmission.

Cause of transmission error	Reference
A transmission error occurred due to noise.	MELSEC iQ-R Serial Communication Module User's Manual(Startup)
A timeout occurred for the no-reception monitoring time (timer 0).	Page 260 No-reception Monitoring Time (timer 0) Setting
A timeout occurred for the response monitoring time (timer 1).	Page 263 Response Monitoring Time (timer 1) Setting
A timeout occurred for the send monitoring time (timer 2).	Page 265 Send Monitoring Time (timer 2) Setting
Data that could not be converted with the ASCII-BIN conversion was included.	Page 391 ASCII-BIN Conversion for Data Communicated with Bidirectional Protocol
The send data quantity exceeding the size that can be stored in the send area is designated.	Page 100 Send area and send data arrangement
Simultaneous transmissions occurred.	Page 107 Processing when Simultaneous Transmission is Performed during Full-Duplex Communications

Confirmation using a program

■Detecting a transmission error

The following device and input signal turn ON.

- BIDOUT instruction completion device + 1
- 'CH□ Error occurrence' (XE/XF)

■Checking a transmission error

The transmission error code can be checked using the BIDOUT instruction control data ((S1) + 1).

Or, it can be checked by reading the 'data transmission result' (Un\G599/615).

For details on how to check the error code contents and corrective actions, refer to the following section.

☞ Page 519 Error Code List

'CH□ Error occurrence' (XE/XF) turns OFF when the data transmission/reception processing is performed normally.

Confirmation using the module and Engineering tool

■Confirmation using the indicator LED

When C24 detects an error, including a reception error, the ERR LED turns ON.

■Confirmation using Engineering tool

Monitor the 'data transmission result' (Un\G599/615) to check the error code.

4.3 Processing when Simultaneous Transmission is Performed during Full-Duplex Communications

This section explains the processing when simultaneous transmission is performed during full-duplex communications.

Processing when simultaneous transmissions occur

The following explains the processing by C24 when the target device and C24 transmit data at the same time during data communications using the bidirectional protocol.

Note that transmission from a target device and from C24 are not performed simultaneously for data communication by half-duplex communications. (☞ Page 275 DATA COMMUNICATIONS USING HALF-DUPLEX COMMUNICATIONS)

The processing on the C24 side differs depending on the settings with Engineering tool when transmission from a target device and C24 is performed at the same time.

Set "Send data enable/disable" and "Receive data enable/disable" to "Enable" or 'Disable" with Engineering tool.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

The setting value for "Send data enable/disable" and "Receive data enable/disable" is stored in the following area on the buffer memory.

For processing of transmitted/received data on the C24 side for each setting of the 'simultaneous transmitted data valid/invalid designation' (Un\G155/315), refer to the "Processing of transmitted/received data when simultaneous transmission occur".

Setting contents with Engineering tool	Value stored to buffer memory (Un\G155/315)	C24 processing contents	
		Processing related to message transmission	Processing related to massage reception
Send data enable/disable: Enable Receive data enable/disable: Enable	0000H	Waits to receive the response message(❶-2) while checking time-out after data transmission (❶-1) is complete. Notifies normal completion or abnormal completion according to whether or not the response message is received to the CPU module via the buffer memory.	Transmits a response message (❷-2) after data reception (❷-1) is complete. Notifies the received data and reception result to the CPU module via the buffer memory.
Send data enable/disable: Disable Receive data enable/disable: Enable	0100H	Notifies a simultaneous transmission error to the CPU module via the buffer memory after data transmission(❶-1). Does not wait for a response massage (❶-2) in reply to data transmission (❶-1).	Transmits a response message (❷-2) after data reception (❷-1) is completed. Notifies the received data and reception result to the CPU module via the buffer memory.
Send data enable/disable: Enable Receive data enable/disable: Disable	0001H	Waits to receive the response message (❶-2) while checking time-out after data transmission (❶-1) is complete. Notifies normal completion or abnormal completion according to whether or not the response message is received to the CPU module via the buffer memory.	Ignores data reception (❷-1) and discards the received data. Does not transmit a response message (❷-2). Does not inform the CPU module that data was received.
Send data enable/disable: Disable Receive data enable/disable: Disable	0101H	Notifies a simultaneous transmission error to the CPU module via the buffer memory after data transmission(❶-1). Does not wait for a response massage (❶-2) in reply to data transmission (❶-1).	Ignores data reception (❷-1) and discards the received data. Does not transmit a response message (❷-2). Does not inform the CPU module that data was received.

(●-m) is the number showing the correspondence with the messages in the figures below.

☞ Page 108 Processing of transmitted/received data when simultaneous transmissions occur

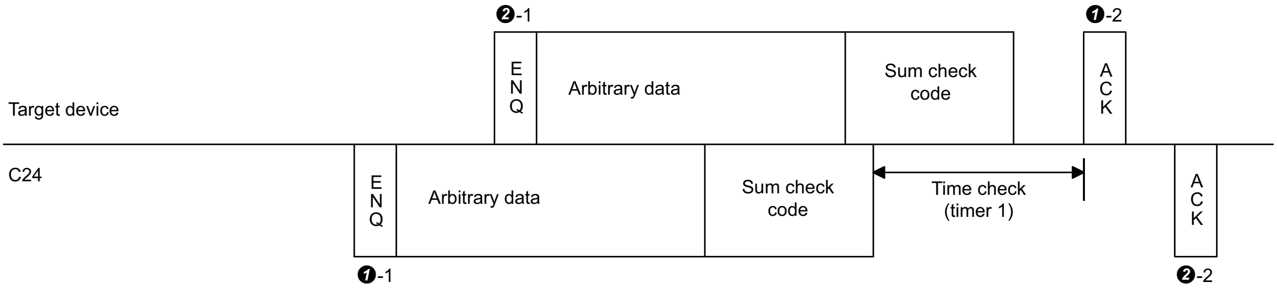
Processing of transmitted/received data when simultaneous transmissions occur

The following shows the C24 communication data processing based on the settings of 'Send data' and 'Receive data' with Engineering tool by using the examples.

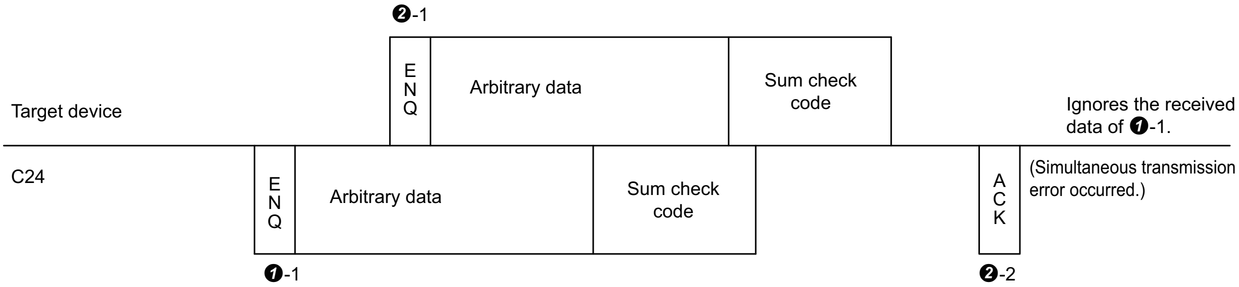
For details on the time check timer 1 (response monitoring time) shown in the figures, refer to the following section.

☞ Page 263 Response Monitoring Time (timer 1) Setting

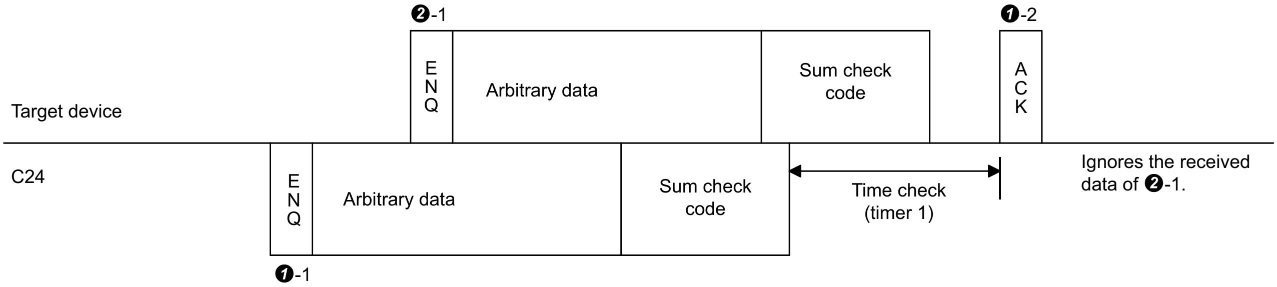
Send data: Enable, Receive data: Enable



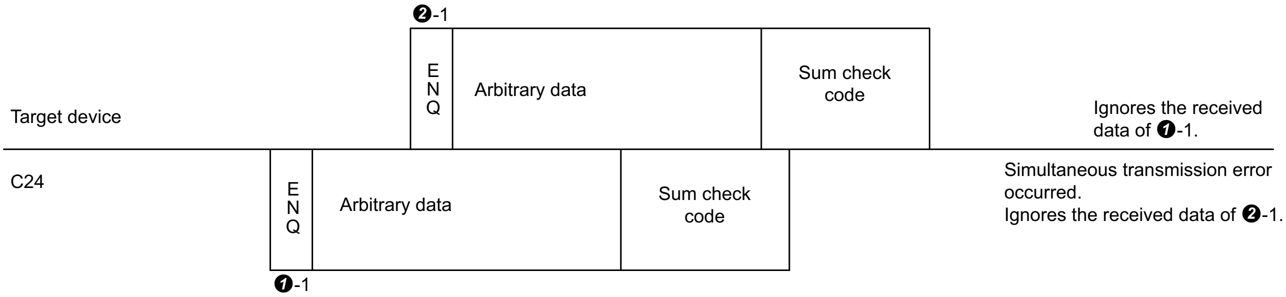
Send data: Disable, Receive data: Disable



Send data: Enable, Receive data: Disable



Send data: Disable, Receive data: Disable



❷-1
Target device
E N Q
Arbitrary data
Sum check code
Ignores the received data of ❶-1.
C24
E N Q
Arbitrary data
Sum check code
Simultaneous transmission error occurred.
Ignores the received data of ❷-1.
❶-1

Point

When the transmission control is performed with "Send data" set to "Enable" and "Receive data" set to "Enable", C24 performs message transmission and message reception processing as described below. (☞ Page 271 Control Contents of DC Code Control)

During message transmission processing, time check by timer 1 (response monitoring time) is performed.

● Message transmission (❶-1 in the figure)

- When C24 receives a transmission terminate request (receiving DC3/DR(DSR) signal OFF) from the target device during message transmission, C24 terminates data transmission.
- When the C24 receives a reception available signal (receiving DC1/DR(DSR) signal ON), C24 restarts data transmission.

● Message reception

- If C24 cannot send a response message to the target device in response to message reception because the target device issued a transmission terminate request (receiving DC3/DR(DSR) signal OFF), C24 transmits the response message after it's in the data transmission available state (receiving DC1/DR(DSR) signal ON).

4.4 Considerations for Data Communication

The following describes the considerations for data communications using the bidirectional protocol.

Factor of C24 transmission sequence initialization

The initial status means the status where transmission has stopped and received data has been discarded.

C24 transmission sequence initialization factors are as follows.

- When the power is turned ON, the CPU module is reset, or the mode is switched.
- When a response message (ACK, NAK, etc.) was received in reply to data transmission.
- When a response message (ACK, NAK, etc.) was transmitted in reply to data reception.

Data transmission procedure

As a data transmission procedure, data transmission from the target device or C24 should be performed after a response message for the immediately preceding data transmission has been received.

Units of data length (word/byte) in message to be transmitted/received

Specify the same units of data length (words/bytes) of the messages to be transmitted/received between a target device and the CPU module.

The word/byte units designation can be set with Engineering tool for the CPU module.

Make the length of the data portion in the message to be transmitted and received as long as the size of the send data specification area and the receive data storage area, or less.

NAK code response

■Response from C24 to target device

After the completion of error detection message reception, it transmits the NAK code to the target device.

■Response from target device to C24

Transmit the error code (0022H to 005FH) immediately after the NAK response.

Point

- Perform error processing according to the error code received immediately after the NAK message at the device that received NAK as the response message after data transmission.
  For more details on error codes that are transmitted from C24, refer to the following section.
  (☞ Page 519 Error Code List)
- If C24 receives an NAK response while transmitting data to a target device, it completes data transmission, then reads the NAK to complete the processing abnormally.
- If C24 detects an error while receiving data, it ignores the received data equivalent to the data length.
  If the data length is abnormal, C24 ignores all the data up to the head data for reception message (ENQ, etc.) received thereafter.

Target device time-out check

When checking time-out up to reception of the response message at the target device during data transmission from the target device to C24, set the timeout time to the time shown below, or longer.

- (CPU module maximum scan time×2) + 100 ms

Framing error on the target device side

While nothing is being transmitted from C24 to the target device though the RS-422/485 interface, a framing error may be generated in the target device.

Make the target device skip the data up to transmission of the head data (ENQ, NAK, etc.) of the message from C24.

Before communicating data through the RS-422/485 interface, check the interface specifications on the C24 side.

📖MELSEC iQ-R Serial Communication Module User's Manual(Startup)

Data bit settings

When adding a sum check code to a message, set "Data bit" to "8".

🖱 Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

5 DATA COMMUNICATION USING MODBUS

Data can be sent and received between a master and a slave by using a MODBUS data format.

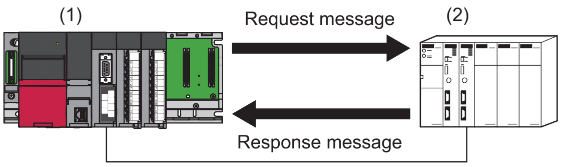
For sending and receiving data in a redundant system, refer to the following section.

☞ Page 624 Using MODBUS in a Redundant System

5.1 MODBUS Master Function

C24 communicates with MODBUS slave devices as a master station by using the following function.

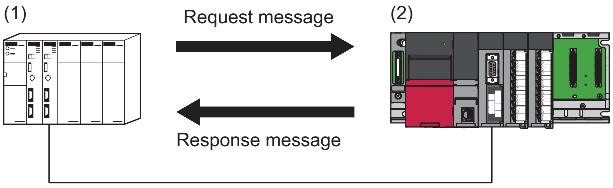
Function	Description	Reference
Communication with predefined protocol	Selects MODBUS from the predefined protocol library and executes the protocol with a dedicated instruction.	Page 30 DATA COMMUNICATION USING PREDEFINED PROTOCOL
Simple CPU communication function	Issues a request message for reading/writing to a MODBUS slave device automatically.	Page 120 Communication with a MODBUS Slave Device



(1) C24
(2) MODBUS slave device

5.2 MODBUS Slave Function

C24 supports the MODBUS slave function and can communicate with MODBUS master devices.



(1) MODBUS master device
(2) C24

The following functions are available:

Function	Description	Reference
Automatic response function	Automatically performs processing corresponding to the function code in a request message received from a MODBUS master device, and automatically sends a response message.	Page 113 Automatic response function
MODBUS device assignment function	Converts access to MODBUS devices into access to devices of a programmable controller CPU.	Page 114 MODBUS device assignment function

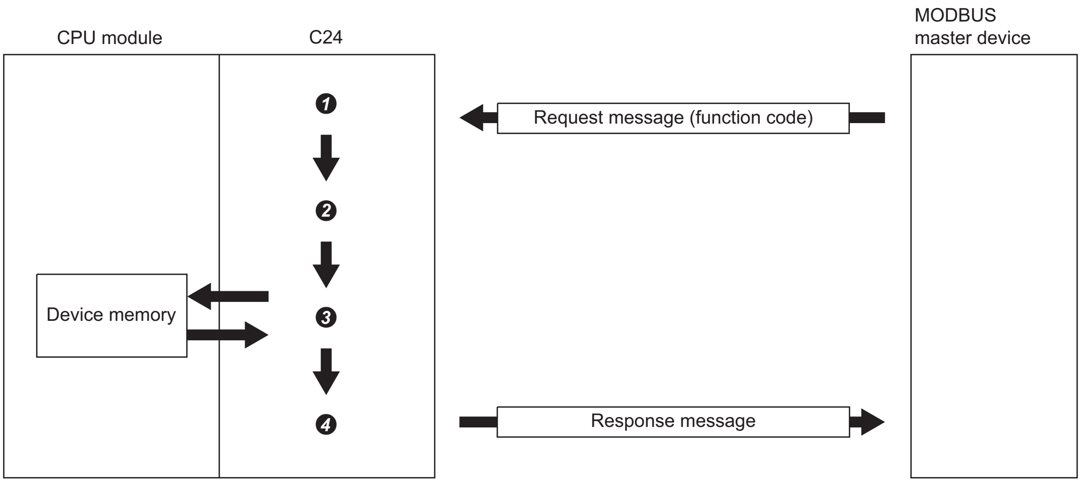
Automatic response function

The automatic response function enables C24s to automatically perform corresponding processing according to the function code in a request message received from a MODBUS master device, and automatically send a response message to it.

No programs are required for sending response messages on the C24 side.

This function uses the MODBUS device assignment function. For details, refer to the following section.

☞ Page 114 MODBUS device assignment function



➊ A request message is received.
➋ The function code is determined.
➌ The corresponding processing is performed.
➍ A response message is sent.

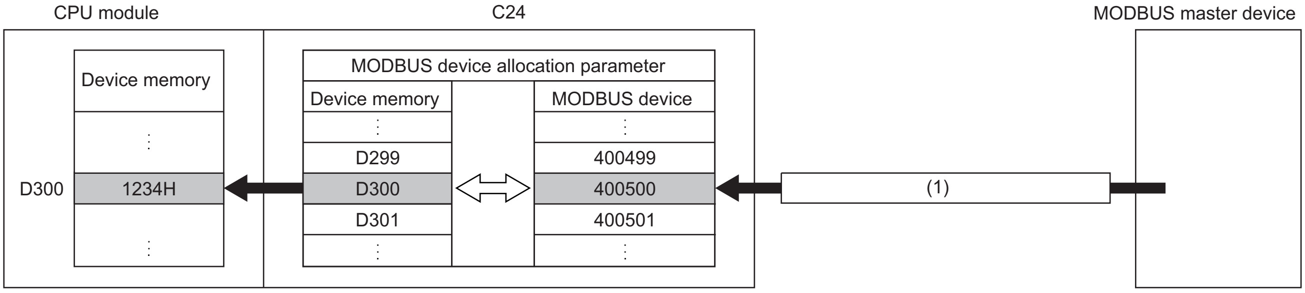
MODBUS device assignment function

The MODBUS device assignment function automatically converts access to MODBUS devices of a slave into access to devices of a CPU module.

MODBUS devices can be associated with device memories of a CPU module with MODBUS device assignment parameters. This enables C24s to convert access to MODBUS devices specified in a request message by MODBUS master devices into access to the corresponding devices of a CPU module and access them.

Associating MODBUS devices with device memories of a programmable controller CPU

When a request message such as 'write coil' is received from a master, access to MODBUS devices is automatically converted into access to devices of a CPU module.



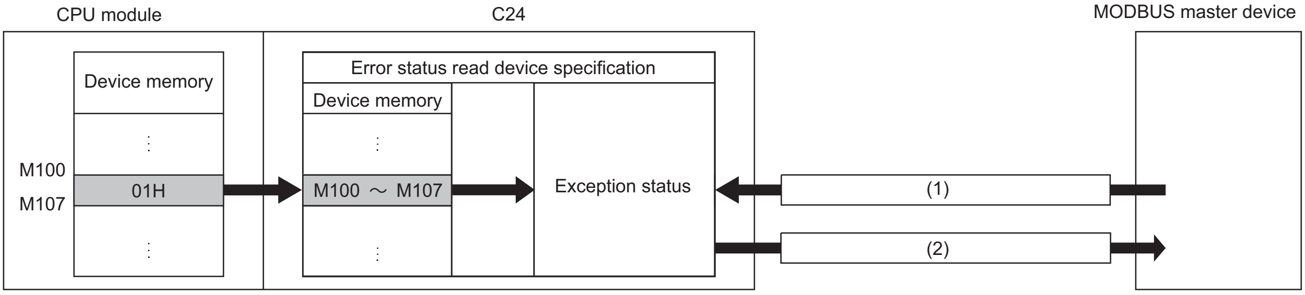
(1) Request message (request to write 1234H to the holding register 400500)

Point

If specifying a device not supported by a CPU module or a device number out of range, an exception code (04) is returned in a response message and 4031H (device specification error) is stored in the detailed error code of 'error log' (Un\G31998 to 32255) in 'MODBUS operating status.'

Specifying the error status read device

Specify data to be read as an error status by a slave (C24) when the read exception status (FC: 07) is received from a MODBUS master device.



(1) Request message (read exception status (FC: 07))
(2) Response message (exception status 01H)

Values of devices specified in 'error status read device' (Un\G28682 to 28685) are stored in a response message.

Specifying an access target

Specify an access target when a C24 is connected to a CC-Link IE Field Network remote head module.

A CC-Link IE Field Network remote head module or CC-Link IE Field Network master station can be selected as an access target.

When selecting a master station, the current master station is accessed.

Precautions

An access target can be specified in a C24 the firmware version of which is '13' or later when the C24 is connected to a CC-Link IE Field Network remote head module. For a C24 the firmware version of which is '12' or earlier, only a CC-Link IE Field Network remote head module can be specified as an access target.

Specifying a CPU response monitoring timer

Specify the timer so that a C24 will monitor the processing of an access target CPU.

If an error occurs in an access target CPU module and a response message cannot be sent, a response message (error completion) can be sent after the elapse of a specified time.

This can release the state in which a master waits for a response message.

Setting method

Set MODBUS device assignment in "MODBUS slave setting" of the parameter.

For details, refer to the following section.

☞ Page 465 MODBUS slave settings

5.3 Interlink Function

For using the interlink function in a MODBUS slave, refer to the following chapter.

☞ Page 429 INDEPENDENT/LINKED OPERATION OF INTERFACE

5.4 MODBUS Devices

The following table shows the MODBUS devices supported by a C24.

MODBUS device type	Unit	Read/write	Number of access points	MODBUS device No.
Coil	Bit	Read/write	65536 points	1 to 65536
Input	Bit	Read	65536 points	100001 to 165536
Input register	Word	Read	65536 points	300001 to 365536
Holding register	Word	Read/write	65536 points	400001 to 465536
Extended file register	Word	Read/write[*1]	10027008 points[*2,*3]	File number: 0 to 1002[*2] Device number: 600000 to 609999

*1 For the CPU module, read-only memories cannot be set in the file register.

*2 The maximum values of the number of access points and the file number for the extended file register depend on the size of the file register (ZR) set in the CPU module. (☞ Page 469 MODBUS extended file register assignment)

*3 The number of access points for the extended file register is the maximum size that can be set in the CPU module (when the R120CPU + NZ2MC-16MBS are mounted).

5.5 Frame Specifications

Frame specifications comply with the specifications of MODBUS protocol.

Frames in RTU mode and ASCII mode can be used.

A frame mode can be selected in "Communication protocol setting" in the module parameter.

For details on the frames, refer to the following manual.

📖MELSEC iQ-R MODBUS and MODBUS/TCP Reference Manual

5.6 Function List

The following table shows the MODBUS standard functions that can be executed from a MODBUS master device to a C24.

The numbers in the following table show the following:

- (1) For broadcast
- (2) For accessing modules other than CC-Link IE Field Network remote head modules
- (3) For accessing CC-Link IE Field Network remote head modules

Executable: ○, Executable (with restrictions): △, Not executable: —

Function code (FC) (DEC (HEX))	Sub code (SC) (DEC (HEX))	Sub-function code (DEC (HEX))	Function name	(1)	(2)	(3)
01 (01H)	—	—	Read coils	—	○	△[*1]
02 (02H)	—	—	Read discrete inputs	—	○	△[*1]
03 (03H)	—	—	Read holding registers	—	○	△[*1]
04 (04H)	—	—	Read input registers	—	○	△[*1]
05 (05H)	—	—	Write single coil	○	○	△[*1]
06 (06H)	—	—	Write single register	○	○	△[*1]
07 (07H)	—	—	Read exception status	—	○	△[*1]
08 (08H)	—	00 (00H)	Return query data	—	○	○
		01 (01H)	Restart communications option	—	○	○
		02 (02H)	Return diagnostic register	—	○	○
		03 (03H)	Change ASCII input delimiter	—	○	○
		04 (04H)	Force listen only mode	—	○	○
		10 (0AH)	Clear counters and diagnostic register	—	○	○
		11 (0BH)	Return bus message count	—	○	○
		12 (0CH)	Return bus communication error count	—	○	○
		13 (0DH)	Return bus exception error count	—	○	○
		14 (0EH)	Return slave message count	—	○	○
		15 (0FH)	Return slave no response count	—	○	○
		16 (10H)	Return slave NAK count	—	○	○
		17 (11H)	Return slave busy count	—	○	○
		18 (12H)	Return bus character overrun count	—	○	○
		19 (13H)	Return IOP overrun error count	—	○	○
		20 (14H)	Clear overrun counter and flag	—	○	○
11 (0BH)	—	—	Get communications event counter	—	○	○
12 (0CH)	—	—	Get communications event log	—	○	○
15 (0FH)	—	—	Write multiple coils	○	○	△[*1]
16 (10H)	—	—	Write multiple registers	○	○	△[*1]
17 (11H)	—	—	Report slave ID	—	○	○
20 (14H)	06 (06H)	—	Read file record	—	○	○
21 (15H)	06 (06H)	—	Write file record	—	○	○
22 (16H)	—	—	Mask write register	○	○	△[*1]
23 (17H)	—	—	Read/Write multiple registers	—	○	△[*1]

[*1] Accessing MODBUS devices assigned to devices not supported by CC-Link IE Field Network remote head modules results in error completion. (Exception code: 04H)

MODBUS standard function details

For details on the functions, refer to the following manual.

📖MELSEC iQ-R MODBUS and MODBUS/TCP Reference Manual

■Buffer memories used for functions

The following table shows the buffer memories used for each function.

Application	Buffer memory	Address Dec (Hex)	
		CH1	CH2
Storage location of exception code and error code	Error log	31998 to 32255 (7CFEH to 7DFFH)	
Storage location of error information of the read exception status (FC: 07H)	Error status read device	28682 to 28685 (700AH to 700DH)	
• Data cleared with the restart communications option (FC: 08H (01H)) • Diagnostic register for the return diagnostic register (FC: 08H (02H)) • Diagnostic register for the clear counters and diagnostic register (FC: 08H (0AH))	CH1/2 side exception code storage area in the buffer memory	28674 (7002H)	28676 (7004H)
	CH1/2 side detailed LED status storage area in the buffer memory	513 (0201H)	514 (0202H)
Storage location of the change ASCII input delimiter (FC: 08H (03H))	2nd byte of end code	32524 (7F0CH)	32588 (7F4CH)
Checking the force listen only mode (FC: 08H (04H))	Communications mode	32525 (7F0DH)	32589 (7F4DH)
Value returned with the return bus message count (FC: 08H (0BH))	Bus message count	32512 (7F00H)	32576 (7F40H)
Value returned with the return bus communication error count (FC: 08H (0CH))	Bus communication error count	32513 (7F01H)	32577 (7F41H)
Value returned with the return bus exception error count (FC: 08H (0DH))	Exception error count	32522 (7F0AH)	32586 (7F4AH)
Value returned with the return slave message count (FC: 08H (0EH))	Slave message count	32518 (7F06H)	32582 (7F46H)
Value returned with the return slave no response count (FC: 08H (0FH))	Slave no-response count	32519 (7F07H)	32583 (7F47H)
Value returned with the return slave NAK count (FC: 08H (10H))	Slave NAK count (0000H is always returned.)	32520 (7F08H)	32584 (7F48H)
Value returned with the return slave busy count (FC: 08H (11H))	Slave busy count (0000H is always returned.)	32521 (7F09H)	32585 (7F49H)
• Value returned with the return bus character overrun count (FC: 08H (12H)) • Value returned with the return IOP overrun error count (FC: 08H (13H))	Character overrun count	32514 (7F02H)	32578 (7F42H)
Value returned with the get communications event counter (FC: 0BH)	Communications event count	32523 (7F0BH)	32587 (7F4BH)
Value returned with the get communications event log (FC: 0DH)	Communications event log 1 to 64	32544 to 32575 (7F20H to 7F3FH)	32608 to 32639 (7F60H to 7F7FH)

6 DATA COMMUNICATION IN SIMPLE CPU COMMUNICATION

Data communications between specified devices can be performed at specified timings by configuring simple parameter settings for a C24 in an engineering tool. Data is sent to/received from a specified communication destination on a 1:1 or 1:n (multi-drop connection) basis.

Restriction

When using the simple CPU communication function, check the firmware versions of a C24. (☞ Page 686 Added and Changed Functions)

Communication pattern

Either of the following can be selected.

Item	Description
Read	To read device data in a specified communication destination (transfer source) to a specified device of the host station (transfer destination).
Write	To write specified device data in the host station (transfer source) to a device of a specified communication destination (transfer destination).

Communication setting

Either of the following can be selected.

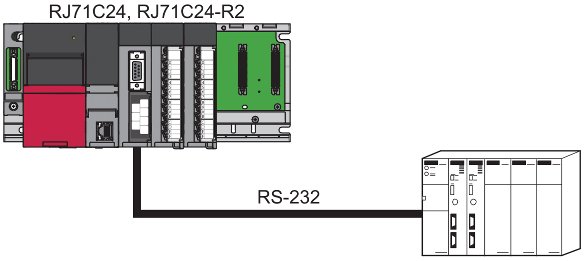
Item	Description
Fixed Interval	To send and receive device data at a specified execution interval.
On Request	To send and receive device data only on request.

Communication destination

C24 can communicate with the following devices.

- MELSEC-A series CPU module (☞ Page 119 Communication with a MELSEC-A Series CPU)
- MODBUS slave device (☞ Page 120 Communication with a MODBUS Slave Device)

6.1 Communication with a MELSEC-A Series CPU



C24 can communicate data with the following MELSEC-A series CPU modules.

Type	Model
An	A2CCPU
AnN	A1NCPU, A1NCPUP21, A1NCPUP21-S3, A1NCPUR21 A2NCPU, A2NCPU-S1, A2NCPUP21, A2NCPUP21-S1, A2NCPUP21-S3, A2NCPUP21-S4, A2NCPUR21, A2NCPUR21-S1 A3NCPU, A3NCPUP21, A3NCPUP21-S3, A3NCPUR21
AnA	A2ACPU, A2ACPU-S1, A2ACPUP21, A2ACPUP21-S1, A2ACPUP21-S3, A2ACPUP21-S4, A2ACPUR21, A2ACPUR21-S1 A3ACPU, A3ACPUP21, A3ACPUP21-S3, A3ACPUR21
A0J2H	A0J2HCPU, A0J2HCPU-DC24V, A0J2HCPUP21, A0J2HCPUR21
AnS	A1SCPUC24-R2
AnSH	A1SHCPU, A1SJHCPU, A2SHCPU
A2C	A2CCPUC24, A2CCPUC24-PRF, A2CCPUP21, A2CCPUR21, A2CJCPU-S3
AnU	A2UCPU, A2UCPU-S1, A3UCPU, A4UCPU
A2US	A2USCPU, A2USHCPU-S1

Wiring

For this data communication, use an RS-232 interface of a C24 to connect to a communication destination.

Connect an RS-232 interface of a C24 and an RS-422 connector of a MELSEC-A series CPU with an RS-232/RS-422 conversion cable.

For connection, use a gender changer for a D-sub 9 pin between the interface and the cable.

Precautions

Only a connector of a MELSEC-A series CPU can be connected.

■Number of connectable modules

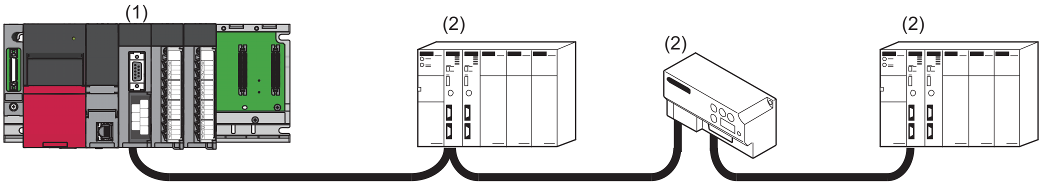
Channel	RJ71C24	RJ71C24-R2	RJ71C24-R4
CH1	1	1	—
CH2	—	1	—

When using an RJ71C24-R2, both CH1 and CH2 can be connected.

Considerations for performing simple CPU communication

A bit device is read and written in one bit units and a word device in one word units. This may cause data inconsistency.

6.2 Communication with a MODBUS Slave Device



(1) C24
(2) MODBUS slave device

C24 can communicate data with the following MODBUS slave devices.

- MODBUS slave (RTU)-compatible device
- MODBUS slave (ASCII)-compatible device

Wiring

Connect a C24 and a MODBUS slave device.

■Number of connectable modules

Channel	RJ71C24	RJ71C24-R2	RJ71C24-R4
CH1	1	1	32
CH2	32	1	32

Considerations for performing simple CPU communication

- Data inconsistency may occur according to the frame specifications of MODBUS protocol.
- For data communication using RS-485 (2-wire type), set "Echo back enable/prohibit specification" to "Echo back prohibit." (☞ Page 436 RS-422/485 INTERFACE ECHO BACK ENABLE/PROHIBIT SPECIFICATION)

6.3 Data Communication Procedure

Data communication with a target device can be performed by the following procedure.

1. Set a channel for each communication destination in "Basic Settings" in "Module Parameter."

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter] ⇨ [Basic Settings]

Item			MELSEC-A/AnS (CPU COM)	MODBUS (RTU)	MODBUS (ASCII)
Various control specification	Communication protocol setting		Simple PLC communication		
	Communication speed setting		9600 bps	Select according to the communication destination.	
	Transmission setting	Operation setting	Independent (default)		
		Data bit	8 bits	8 bits	7 bits (default)
		Parity bit	Yes	Select according to the communication destination.	
		Odd/even parity	Odd (default)	Select according to the communication destination.	
		Stop bit	1 (default)	• Parity bit is set to "None": 2 • Parity bit is set to "Yes": 1 (default) or 2	
	Transmission control setting	Transmission control	DC code control	DTR/DSR control (default)	

Use the default values for the other settings.

2. Set "Simple PLC Communication Setting" in "Module Extended Parameter."

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Extended Parameter] ⇨ [Simple PLC Communication Setting]

For details of the setting, refer to the following section.

☞ Page 478 Simple CPU communication setting

3. Write the settings to the CPU module with an engineering tool after the parameter setting is completed.

4. Reset the CPU module or turn the power OFF and ON, then simple CPU communication starts.

6.4 Devices that can be Specified

A bit device and word device can be set at the same time for each setting number.

A bit device can be specified in units of 16 points and a word device in units of 1 point. Specify '0' or a multiple of 16 for the start device number of a bit device.

Communication destination	Settable device	
	Bit	Word
Host station	X, Y, M, L, B, SB, SM	D, W, SW, SD, R, ZR, RD, G
MELSEC-A/AnS (CPU COM)	X, Y, M, B	D, W
MODBUS slave device • MODBUS (RTU) • MODBUS (ASCII)	Coil, Input	Input Register, Holding Register

Point

- The number of points per setting is up to 1024 words (bit: up to 8192 points, word: up to 512 points).
- Set a device of a module within the range specified in the parameter of a CPU module.

Device range

■Host station

The following table lists the maximum numbers of points when an NZ2MC-16MBS is attached to an R120CPU. The number of points varies depending on the type of a CPU module to be used, whether to use an extended SRAM cassette, and its type.

Type	Settable device		
	Symbol	Range	Remarks
Bit device	X	0H to 2FFFH	—
	Y	0H to 2FFFH	—
	M	0 to 161882111	Local devices cannot be specified.
	L	0 to 32767	—
	B	0H to 9A61FFFH	—
	SB	0H to 9A61FFFH	—
	SM	0 to 4095	—
Word device	D	0 to 10117631	Local devices cannot be specified.
	W	0H to 9A61FFH	—
	SW	0H to 9A61FFFH	—
	SD	0 to 4095	—
	R	0 to 32767	Data is read and written according to the setting for a file register of a CPU module.
	ZR	0 to 10027007	Data is read and written according to the setting for a file register of a CPU module.
	RD	0 to 1048575	—
	G	53248 to 57343	• Buffer memory of a C24 • A G device is expressed as Un\G. The search for the device requires the search string to be prefixed by Un\G. When searching for a G device if a C24 is not mounted, U0\G is prefixed for the search. (An error for unmounted module is not detected.)

■Communicating with a MELSEC-A/AnS (CPU COM)

The following table lists the maximum numbers of points of an A4UCPU. The number of points varies depending on the type of a CPU module to be used.

Type	Settable device		
	Symbol	Range	Remarks
Bit device	X	0H to 1FFFH	—
	Y	0H to 1FFFH	—
	M	0 to 8191, 9000 to 9255,	• For specifying 9000 to 9255 when "Write" is selected for the communication pattern, refer to the manual of a CPU module used. • Specify a value obtained by adding a multiple of 16 to 9000 for 9000 to 9255.
	B	0H to 1FFFH	—
Word device	D	0 to 8191, 9000 to 9255	For specifying 9000 to 9255 when "Write" is selected for the communication pattern, refer to the manual of a CPU module used.
	W	0H to 1FFFH	—

■Communicating with a MODBUS slave device

Devices that can be used differ depending on a MODBUS slave device. Refer to the manual of each device.

Set the number of points per setting for "Simple PLC Communication setting" in "Parameter" to be up to 1024 words (bit: up to 8192 points, word: up to 512 points).

Type		Range	Remarks
Bit	Coil	0 to 65535	—
	Input	0 to 65535	Can be set only when "Communication Pattern" is "Read."
Word	Input Register	0 to 65535	Can be set only when "Communication Pattern" is "Read."
	Holding Register	0 to 65535	—

6.5 Operations When Performing Simple CPU Communication

This section explains the operations when performing simple CPU communication.

Operations when selecting "Fixed Interval" for the communication setting

Starting communication

Communication starts after a latency time elapses when turning the power OFF and ON or resetting a CPU module.
After resetting the CPU module, preparation processing is performed on the system side such as parameter setting check (station numbers and the channel of the host station). If an error occurs in the processing, data is not sent and received.

■Latency time

During a latency time, '1' (preparing) is stored in 'Communication status' for each setting number.
When the latency time elapses, 'Preparation completion for each setting number' (Un\G32768 to Un\G32799) turns from OFF to ON.

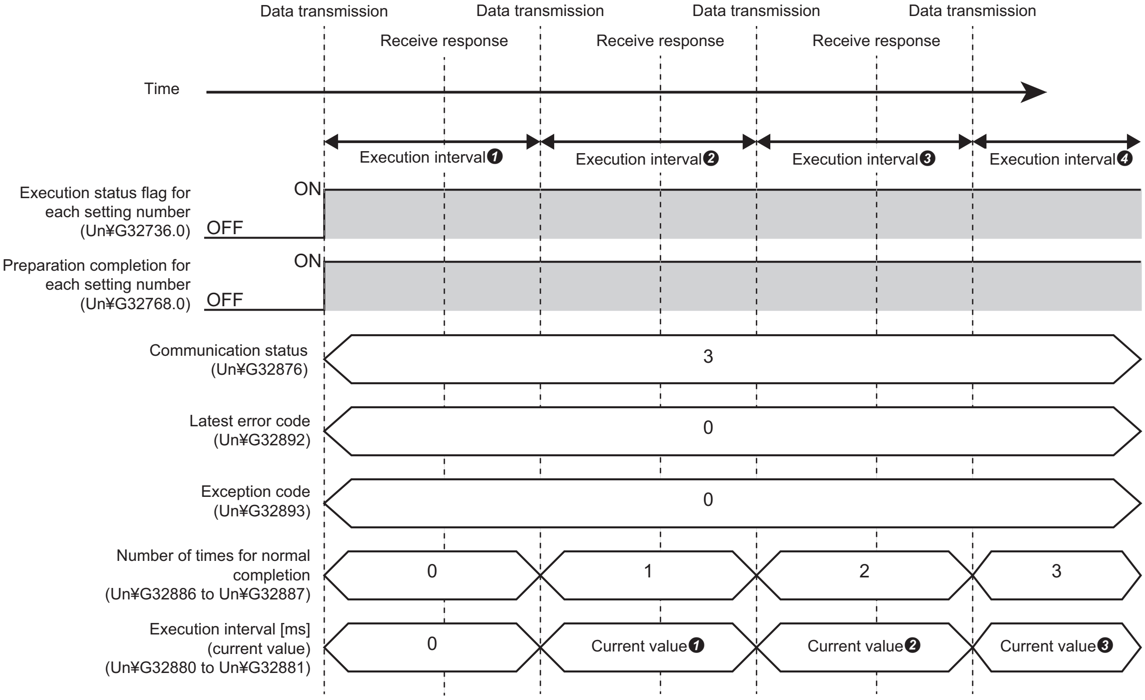
■If an error occurs in preparation processing

'10' (communications impossible) is stored in 'Communication status' for each setting number, and an error is stored in 'Latest error code' for each setting number.

Normal operation

Ex.

The following shows a normal operation for setting No.1.


Data transmission
Data transmission
Data transmission
Data transmission
Receive response
Receive response
Receive response
Time
Execution interval❶
Execution interval❷
Execution interval❸
Execution interval❹
Execution status flag for each setting number (Un¥G32736.0)
ON
OFF
Preparation completion for each setting number (Un¥G32768.0)
ON
OFF
Communication status (Un¥G32876)
3
Latest error code (Un¥G32892)
0
Exception code (Un¥G32893)
0
Number of times for normal completion (Un¥G32886 to Un¥G32887)
0
1
2
3
Execution interval [ms] (current value) (Un¥G32880 to Un¥G32881)
0
Current value❶
Current value❷
Current value❸

Point

- When sending data to a different communication destination during data communications on the same channel, send data after the data communications being executed are completed.

■Storing an execution interval [ms] (current value)

An execution interval is stored in 'Execution interval [ms] (current value)' for each setting number when an execution interval set in the parameter elapses after the first response is received from a communication destination.

Operation when an error occurs

If there is no response from a communication destination or an error occurs (error response) after sending data to the communication destination, a retry is performed (data is resent) when a communication time-out period elapses.

A retry is performed for a time obtained by adding '1' to the number of retries set in the parameter.

During a retry, '5' (retry being executed) is stored in 'Communication status' for each setting number.

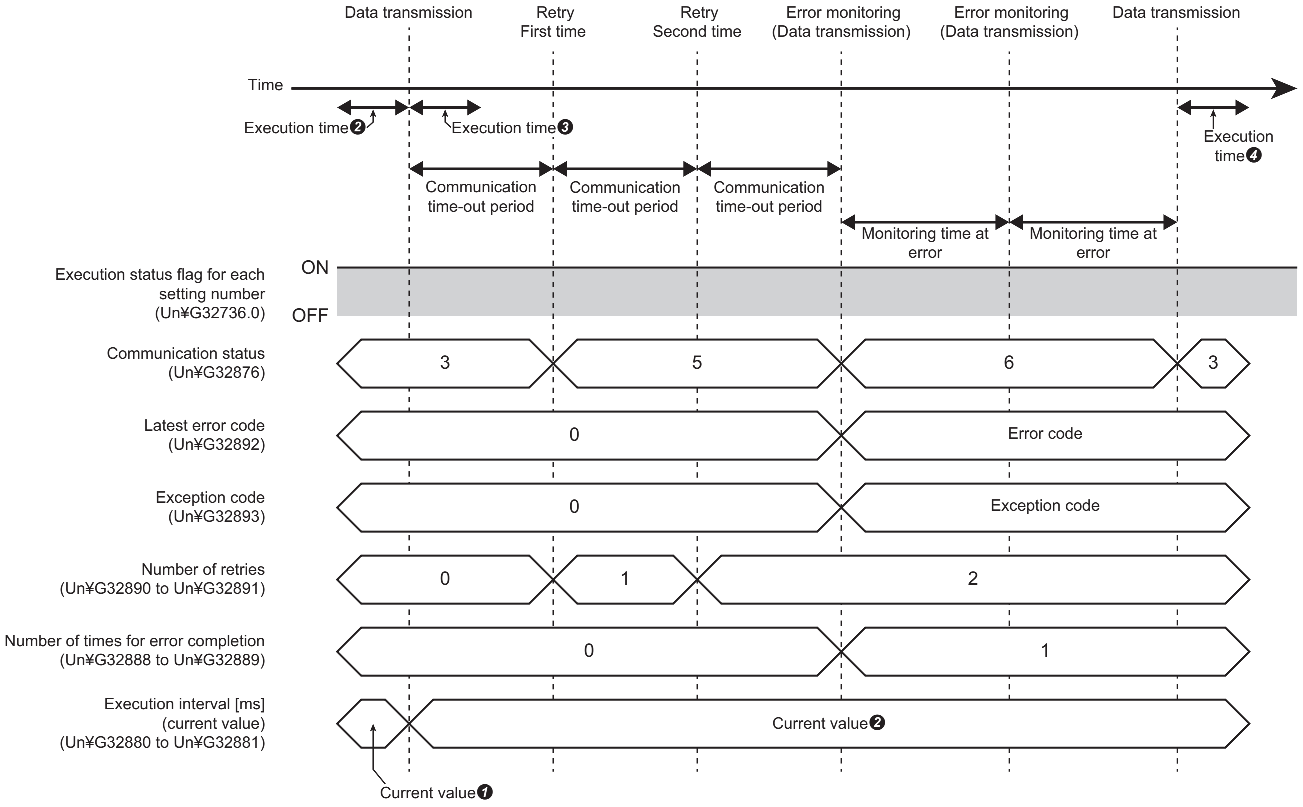
Point

- A value in 'Execution interval [ms] (current value)' for each setting number is not updated during a retry.
- '3' (communicating) remains in 'Communication status' for another setting number for which the same communication destination is set.

If an error response is returned even after a retry, it is determined to be in an abnormal state and an error is stored in 'Latest error code' for each setting number.

Ex.

The following shows an operation when an error occurs in setting No.1 (two retries).



Data transmission
Retry First time
Retry Second time
Error monitoring (Data transmission)
Error monitoring (Data transmission)
Data transmission
Time
Execution time❷
Execution time❸
Execution time❹
Communication time-out period
Communication time-out period
Communication time-out period
Monitoring time at error
Monitoring time at error
Execution status flag for each setting number (Un¥G32736.0)
ON
OFF
Communication status (Un¥G32876)
3
5
6
3
Latest error code (Un¥G32892)
0
Error code
Exception code (Un¥G32893)
0
Exception code
Number of retries (Un¥G32890 to Un¥G32891)
0
1
2
Number of times for error completion (Un¥G32888 to Un¥G32889)
0
1
Execution interval [ms] (current value) (Un¥G32880 to Un¥G32881)
Current value❷
Current value❶

■Operation in an abnormal state

'6' (monitoring at error) is stored in 'Communication status' for each setting number in an abnormal state, and communication is performed at an error monitoring time (low-speed cycle) set in the parameter.

If a response is received from a communication destination during error monitoring, the status returns to normal. (Communication at an execution interval specified in the parameter)

After returning to the normal state, '3' (communicating) is stored in 'Communication status' for each setting number.

Point

- A value in 'Execution interval [ms] (current value)' for each setting number is not updated in an abnormal state.
- If no response can be received from a communication destination, an error is stored in 'Latest error code' for each setting number.
- When an error response is received from a communication destination, an error is stored in 'Latest error code' for each setting number and the code of an abnormal response is stored in 'Exception code' for each setting number. (An exception code refers to a code returned from a target device (error code and end code).)
- If there is no response or an error response (an error occurs) from a communication destination, a communication for the same setting number is repeated until it is established normally. To continue communications for other setting numbers, turn ON 'Request to stop communication for each setting number' (Un\G32672 to Un\G32703) for the setting number for which an error occurs and stop the communication.

Request to stop and restart communication

When selecting "Fixed Interval" for the communication setting, communication can be stopped and restarted.

■Stopping communication

To stop communication, turn 'Request to stop communication for each setting number' (Un\G32672 to Un\G32703) from OFF to ON when '3' (communicating), '5' (retry being executed), or '6' (monitoring at error) is stored in 'Communication status' for each setting number.

The communication stops at a communication timing of the next execution interval.

If there is no response from a communication destination while a stop request is accepted, communication stops after a communication time-out period elapses. During a retry, communication stops after the communication time-out period elapses regardless of the number of retries.

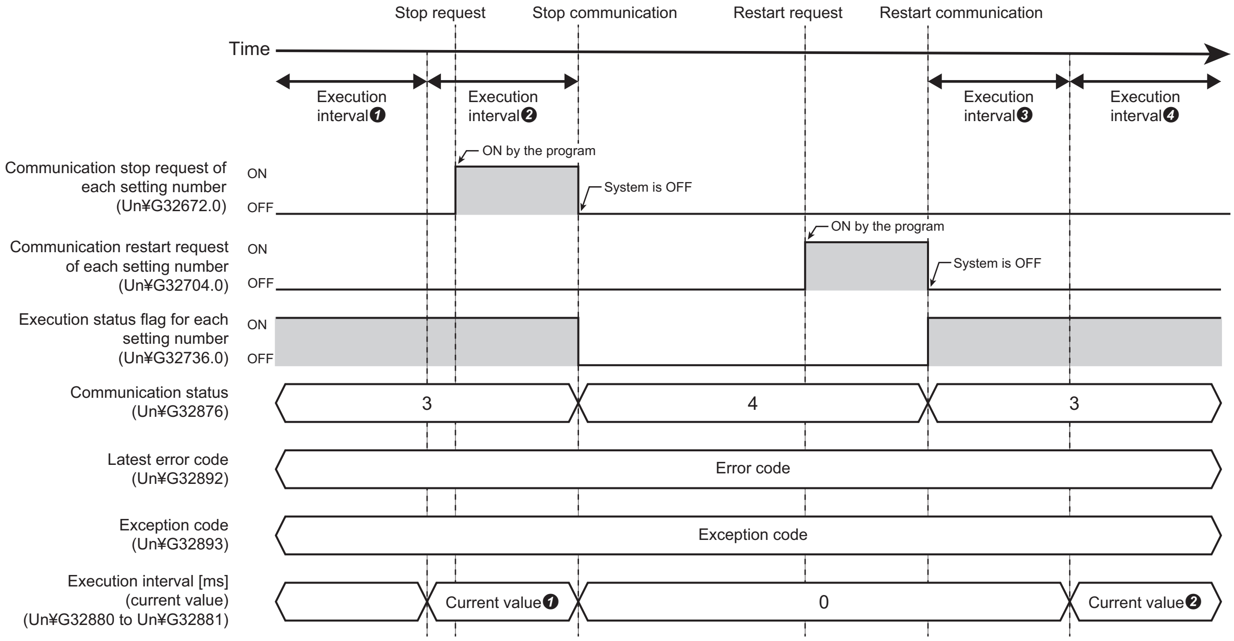
■Restarting communication

To restart communication, turn 'Request to restart communication for each setting number' (Un\G32704 to Un\G32735) from OFF to ON when '4' (communication stop) is stored in 'Communication status' for each setting number.

After the communication restarts, 'Request to restart communication for each setting number' (Un\G32704 to Un\G32735) turns from ON to OFF.

Ex.

The following shows an operation when stopping and restarting communication for setting No.1.


Stop request
Stop communication
Restart request
Restart communication
Time
Execution interval❶
Execution interval❷
Execution interval❸
Execution interval❹
Communication stop request of each setting number (Un¥G32672.0)
ON
OFF
ON by the program
System is OFF
Communication restart request of each setting number (Un¥G32704.0)
ON
OFF
ON by the program
System is OFF
Execution status flag for each setting number (Un¥G32736.0)
ON
OFF
Communication status (Un¥G32876)
3
4
3
Latest error code (Un¥G32892)
Error code
Exception code (Un¥G32893)
Exception code
Execution interval [ms] (current value) (Un¥G32880 to Un¥G32881)
Current value❶
0
Current value❷

Point

Values in 'Latest error code' and 'Exception code' are not cleared even if a request to stop or restart communication is sent. To clear them, turn the buffer memory 'Error clear request' from OFF to ON in a program and others.

■Operation when restarting communication

An operation after restarting communication differs depending on a value in 'Communication status' before stopping the communication.

Communication status before stopping communication	Operation after restarting communication
• Communicating • Retry being executed	A fixed cycle communication is performed at an execution interval set in the parameter. The communication restarts at the next execution interval after turning 'Request to restart communication for each setting number' (Un\G32704 to Un\G32735) from OFF to ON. When restarting the communication, each buffer memory will be in the following states: • 'Execution status flag for each setting number' (Un\G32736 to Un\G32767): ON • 'Communication status' for each setting number: 3 (communicating) • 'Execution interval [ms] (current value)' for each setting number: An execution interval is stored.
Monitoring at error	A fixed cycle communication is performed at an error monitoring time. When restarting the communication, each buffer memory will be in the following states: • 'Execution status flag for each setting number' (Un\G32736 to Un\G32767): ON • 'Communication status' for each setting number: 6 (monitoring at error)

Precautions

- If a value other than '3' (communicating), '5' (retry being executed), or '6' (monitoring at error) is stored in 'Communication status' for each setting number, a request is ignored even when turning 'Request to stop communication for each setting number' (Un\G32672 to Un\G32703) from OFF to ON. In this case, turn it from ON to OFF.
- If a value other than '4' (communication stop) is stored in 'Communication status' for each setting number, a request is ignored even when turning 'Request to restart communication for each setting number' (Un\G32704 to Un\G32735) from OFF to ON. In this case, turn it from ON to OFF.
- If the communication setting is not set or "On Request" is selected, a request is ignored even when turning 'Request to restart communication for each setting number' (Un\G32704 to Un\G32735) or 'Request to stop communication for each setting number' (Un\G32672 to Un\G32703) from OFF to ON. Therefore, they are not turned from ON to OFF by the system and remain ON.

Operations when selecting "On Request" for the communication setting

Starting communication

Communication starts by turning 'Request to start communication at request for each setting number' (Un\G32640 to Un\G32671) from OFF to ON when a latency time elapses after turning the power OFF and ON or resetting a CPU module.

Point

After resetting the CPU module, preparation processing is performed on the system side such as parameter setting check (station numbers and the channel of the host station). If an error occurs in the processing, data is not sent and received.

■Latency time

During a latency time, '1' (preparing) is stored in 'Communication status' for each setting number.

When the latency time elapses, 'Preparation completion for each setting number' (Un\G32768 to Un\G32799) turns from OFF to ON.

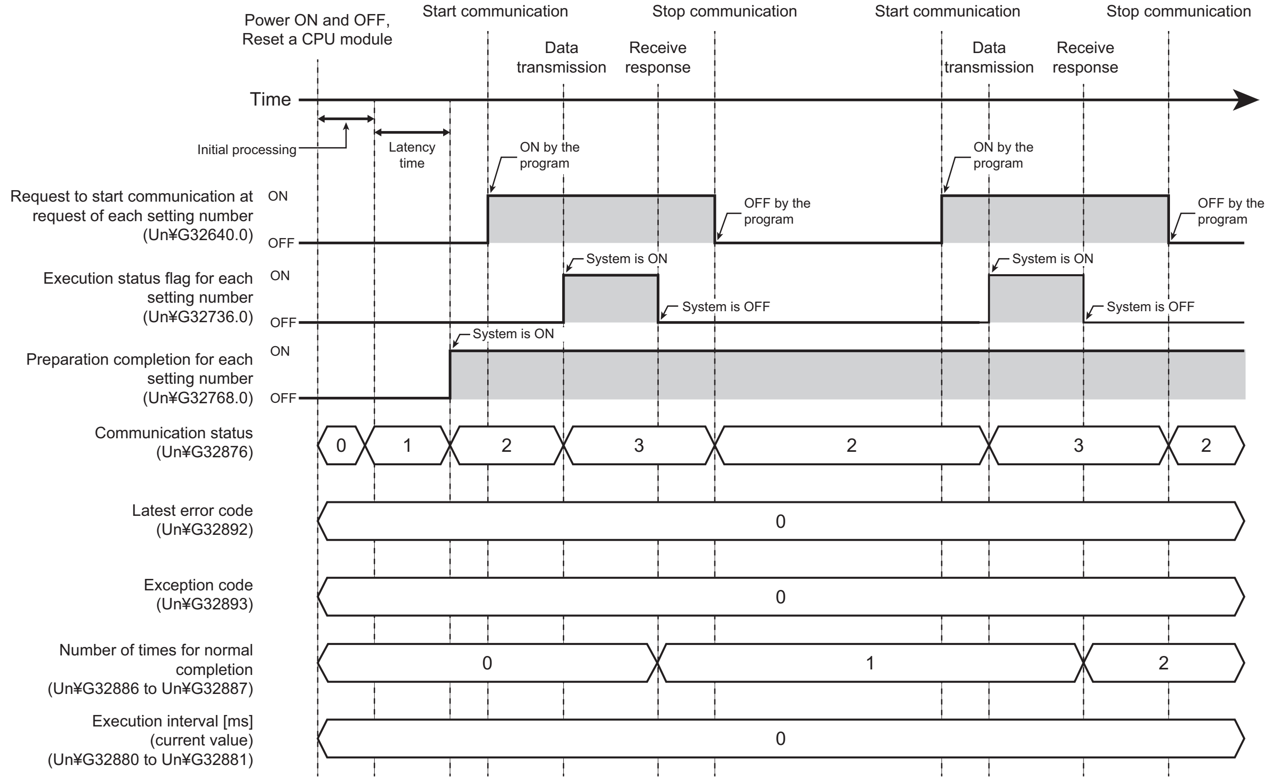
■If an error occurs in preparation processing

'10' (communications impossible) is stored in 'Communication status' for each setting number, and an error is stored in 'Latest error code' for each setting number.

Normal operation

Ex.

The following shows a normal operation for setting No.1.



Power ON and OFF, Reset a CPU module
Start communication
Stop communication
Start communication
Stop communication
Data transmission
Receive response
Data transmission
Receive response
Time
Initial processing
Latency time
ON by the program
ON by the program
OFF by the program
OFF by the program
Request to start communication at request of each setting number (Un¥G32640.0)
ON
OFF
Execution status flag for each setting number (Un¥G32736.0)
ON
OFF
System is ON
System is OFF
System is ON
System is OFF
System is ON
Preparation completion for each setting number (Un¥G32768.0)
ON
OFF
Communication status (Un¥G32876)
0
1
2
3
2
3
2
Latest error code (Un¥G32892)
0
Exception code (Un¥G32893)
0
Number of times for normal completion (Un¥G32886 to Un¥G32887)
0
1
2
Execution interval [ms] (current value) (Un¥G32880 to Un¥G32881)
0

During a latency time, a request is ignored even when turning 'Request to start communication at request for each setting number' (Un\G32640 to Un\G32671) from OFF to ON.

When turning it from ON to OFF after data is sent or received, '2' (waiting for the request) is stored in 'Communication status' for each setting number and a single communication is completed.

Point

- When 'Request to start communication at request for each setting number' (Un\G32640 to Un\G32671) turns from OFF to ON during preparation processing or a latency time on the system side, communication starts when 'Preparation completion for each setting number' turns ON and '2' (waiting for the request) is stored in 'Communication status' for each setting number.
- After it is detected that 'Request to start communication at request for each setting number' (Un\G32640 to Un\G32671) has turned from OFF to ON, a request is ignored before 'Execution status flag for each setting number' (Un\G32736 to Un\G32767) turns OFF (communication stop) even when turning 'Request to start communication at request for each setting number' (Un\G32640 to Un\G32671) from ON to OFF and ON again in a program and others. (No error occurs.)

Operation when an error occurs

If there is no response from a communication destination or an error occurs (error response) after sending data to the communication destination, a retry is performed (data is resent) when a communication time-out period elapses.

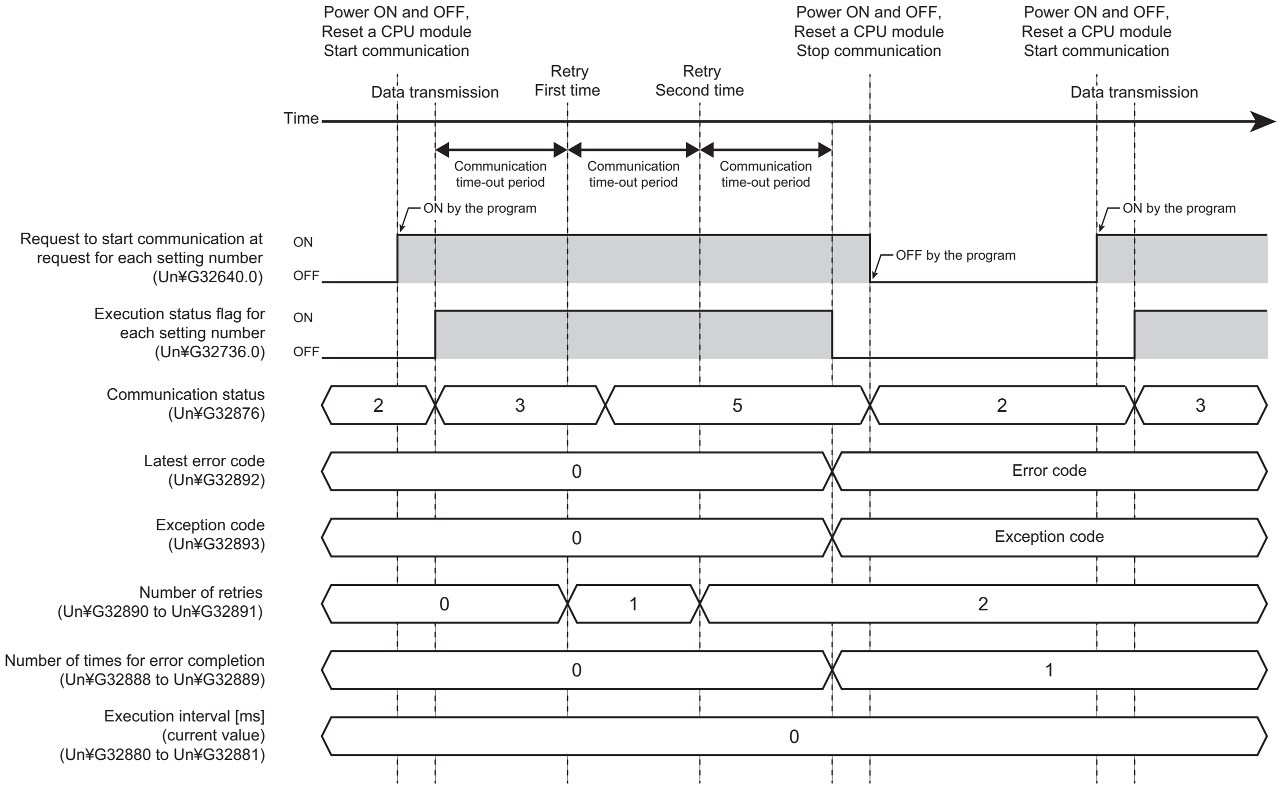
A retry is performed for a time obtained by adding '1' to the number of retries set in the parameter.

During a retry, '5' (retry being executed) is stored in 'Communication status' for each setting number.

If an error response is returned even after a retry, it is determined to be in an abnormal state and an error is stored in 'Latest error code' for each setting number.

Ex.

The following shows an operation when an error occurs in setting No.1 (two retries).


Power ON and OFF,
Reset a CPU module
Start communication
Power ON and OFF,
Reset a CPU module
Stop communication
Power ON and OFF,
Reset a CPU module
Start communication
Data transmission
Retry
First time
Retry
Second time
Data transmission
Time
Communication time-out period
Communication time-out period
Communication time-out period
ON by the program
ON by the program
OFF by the program
Request to start communication at request for each setting number (Un¥G32640.0)
ON
OFF
Execution status flag for each setting number (Un¥G32736.0)
ON
OFF
Communication status (Un¥G32876)
2
3
5
2
3
Latest error code (Un¥G32892)
0
Error code
Exception code (Un¥G32893)
0
Exception code
Number of retries (Un¥G32890 to Un¥G32891)
0
1
2
Number of times for error completion (Un¥G32888 to Un¥G32889)
0
1
Execution interval [ms] (current value) (Un¥G32880 to Un¥G32881)
0

■Operation in an abnormal state

In an abnormal state, 'Execution status flag for each setting number' (Un\G32736 to Un\G32767) turns OFF (communication stop). Therefore, when turning 'Request to start communication at request for each setting number' (Un\G32640 to Un\G32671) from ON to OFF in a program and others, '2' (waiting for the request) is stored in 'Communication status' for each setting number.

Point

- In an abnormal state, '2' (waiting for the request) remains in 'Communication status' for another setting number for which the same communication destination is set.
- If no response can be received from a communication destination, an error is stored in 'Latest error code' for each setting number.
- When an error response is received from a communication destination, an error is stored in 'Latest error code' for each setting number and the code of an abnormal response is stored in 'Exception code' for each setting number. (An exception code refers to a code returned from a target device (error code and end code).)

Broadcast operation

When "Station No." is set to '0' in the "Communication Destination Setting" screen (in the "Simple PLC Communication Setting" screen of "Module Extended Parameter"), broadcast to MODBUS slave devices can be performed. (☞ Page 481 Host station channel number, station number)

When broadcasting to MODBUS slave devices, the communication time-out period set in the parameter operates as the broadcast delay.

After sending data, a normal completion is established by timeout for the broadcast delay (communication time-out period), and the processing is repeated.

Point

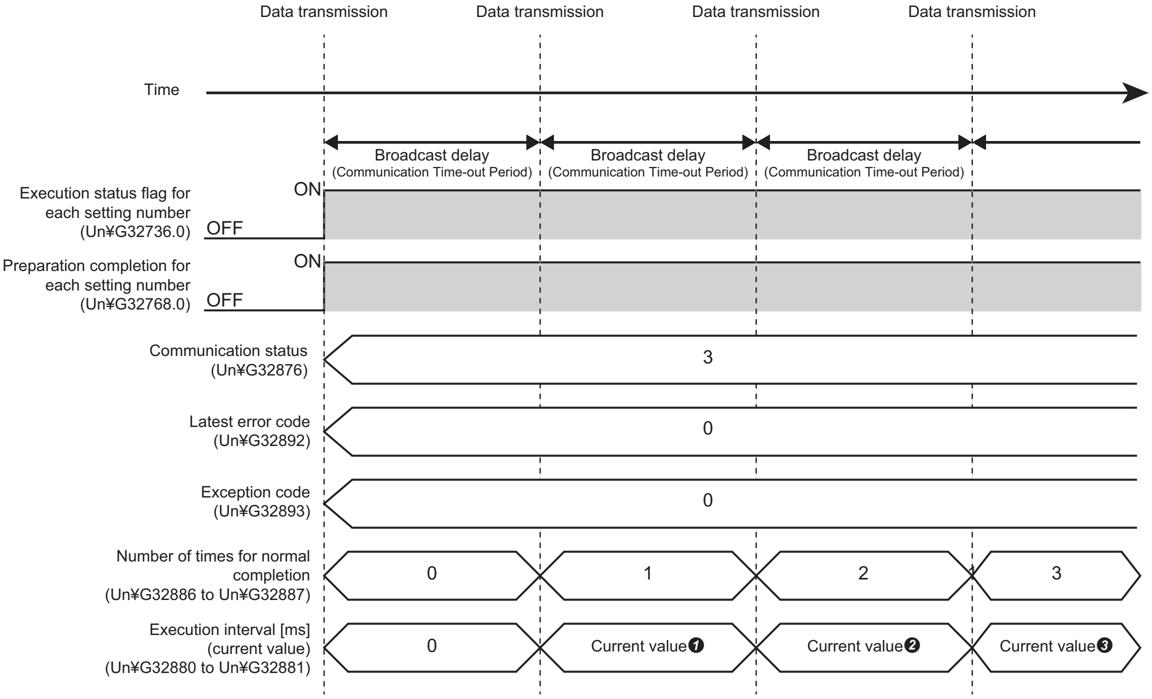
Broadcast is sent to all slave stations; therefore, consider each processing time of all slave devices and set the sufficient time for communication time-out period (broadcast delay).

If there is a slave device the communication time-out period of which is set insufficiently, an error may occur for another request to the slave device.

Normal operation

Ex.

The following shows a normal operation for setting No.1.


Data transmission
Data transmission
Data transmission
Data transmission
Time
Broadcast delay
(Communication Time-out Period)
Broadcast delay
(Communication Time-out Period)
Broadcast delay
(Communication Time-out Period)
Execution status flag for each setting number (Un¥G32736.0)
ON
OFF
Preparation completion for each setting number (Un¥G32768.0)
ON
OFF
Communication status (Un¥G32876)
3
Latest error code (Un¥G32892)
0
Exception code (Un¥G32893)
0
Number of times for normal completion (Un¥G32886 to Un¥G32887)
0
1
2
3
Execution interval [ms] (current value) (Un¥G32880 to Un¥G32881)
0
Current value❶
Current value❷
Current value❸

■Storing an execution interval [ms] (current value)

An execution interval is stored in 'Execution interval [ms] (current value)' for each setting number after normal completion.

6.6 Checking Simple CPU Communication Status

The status of simple CPU communication can be checked in the buffer memory.

Point

Values in the buffer memory are all cleared when switching a CPU module from STOP to RUN after changing module extended parameters and writing them to the CPU module.

Item	Address	Description
Request to start communication at request for each setting number	Un\G32640 to Un\G32671	For 512 settings (1 bit per setting) • ON: Requested • OFF: Not requested
Request to stop communication for each setting number	Un\G32672 to Un\G32703	For 512 settings (1 bit per setting) • ON: Requested • OFF: Not requested
Request to restart communication for each setting number	Un\G32704 to Un\G32735	For 512 settings (1 bit per setting) • ON: Requested • OFF: Not requested
Execution status flag for each setting number	Un\G32736 to Un\G32767	For 512 settings (1 bit per setting) • ON: Communicating • OFF: Communication stop
Preparation completion for each setting number	Un\G32768 to Un\G32799	For 512 settings (1 bit per setting) • ON: Ready • OFF: Not ready

Item			Address	Description
Simple CPU communication diagnostic information	Setting status of each setting number		Un\G32832 to Un\G32863	For 512 settings (1 bit per setting) • ON: Set • OFF: Not set
	Number of settings		Un\G32864	• 0: No diagnostic information • 1 to 512: Number of diagnostic information settings
	Error clear request		Un\G32865	• 0: Not requested • 1: Requested
	Setting No.1 diagnostic information[*1]	Setting No.	Un\G32872	1 to 512: Each setting No.
		Communication pattern	Un\G32873	• 0: Read • 1: Write
		Communication setting	Un\G32874	• 0: On request • 1: Fixed interval
		Communication destination	Un\G32875	10H: MELSEC-A/AnS (CPU COM) 11H: MODBUS (RTU) 12H: MODBUS (ASCII)
		Communication status	Un\G32876	• 0: Not set • 1: Preparing • 2: Waiting for the request[*2] • 3: Communicating • 4: Communication stop[*3] • 5: Retry being executed • 6: Monitoring at error[*3] • 10: Communications impossible
		Station number	Un\G32877	FFFH: Not specified Communicating with a MODBUS (RTU) or a MODBUS (ASCII) • 0: Broadcast • 1 to 247: Station number
		Host station CH number	Un\G32878	• 0: CH1 • 1: CH2
		Target PLC number	Un\G32879	• 0: Not specified • 1 to n: CPU No.1 to CPU No.n[*4]
		Execution interval [ms] (current value)	Un\G32880 to Un\G32881	0 to 4294967295
		Execution interval [ms] (maximum value)	Un\G32882 to Un\G32883	
		Execution interval [ms] (minimum value)	Un\G32884 to Un\G32885	
		Number of times for normal completion	Un\G32886 to Un\G32887	0 to 4294967295[*5]
		Number of times for abnormal completion	Un\G32888 to Un\G32889	
		Number of retries	Un\G32890 to Un\G32891	
		Latest error code[*6]	Un\G32892	• 0H: No error codes • 1H to 7FFFH: Error code
		Exception code[*6]	Un\G32893	0H to FFFFH: Response code of a communication destination
		System area	Un\G32894 to Un\G32895	—
	Setting No.2 to No.512 diagnostic information[*1]		Un\G32896 to Un\G45159	Same as setting No.1 diagnostic information

[*1] Enabled when the corresponding setting number in 'Setting status of each setting number' (Un\G32832 to Un\G32863) is ON.

[*2] Stored only when selecting "On Request" for the communication setting.

[*3] Stored only when selecting "Fixed Interval" for the communication setting.

[*4] A selectable CPU number differs depending on the communication destination.

[*5] When the maximum value is exceeded, counting continues from 1.

[*6] When 'Error clear request' (Un\G32865) turns ON, values stored in 'Latest error code' (Un\G32892) and 'Exception code' (Un\G32893) are cleared. Then, 'Error clear request' (Un\G32865) turns OFF.

Exception codes

The following table lists the exception codes returned from a communication destination when communication fails.

Communicating with a MELSEC-A/AnS (CPU COM)

Exception code	Item	Description	Corrective action
1H	Busy	Processing failed because the CPU module was busy.	• Restart data communications. • Increase the execution interval. • If an error occurs in a MELSEC-A series CPU, take the corrective actions.
4H	Cancellation of processing	New processing was requested during processing.	• Increase the communication time-out period. • If an error occurs in a MELSEC-A series CPU, take the corrective actions.
5H	Sum check error	The received sum value and the calculated one did not match.	• Review the cable connection and wiring. • Take measures against noise.
6H	Busy	Processing failed because the CPU module was busy.	• Restart data communications. • Increase the execution interval. • If an error occurs in a MELSEC-A series CPU, take the corrective actions.
1BH	Busy	Processing failed because the CPU module was busy.	• Restart data communications. • Increase the execution interval. • If an error occurs in a MELSEC-A series CPU, take the corrective actions.

Communicating with a MODBUS slave device

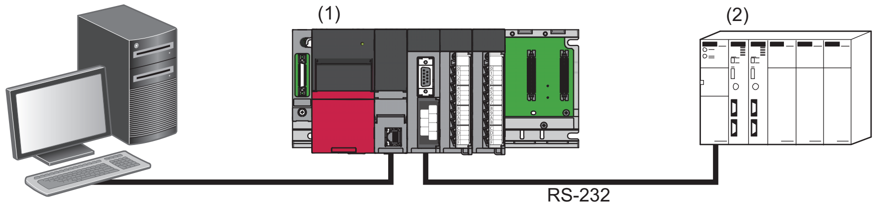
For exception codes from a MODBUS slave device, refer to the following section.

☞ Page 543 MODBUS Exception Code List

6.7 Communication Example

This section shows an example of transferring B0 to B1FFF and W0 to W1FF of the host station to M0 to M8191 and D0 to D511 of a communication destination respectively by using the simple CPU communication function.

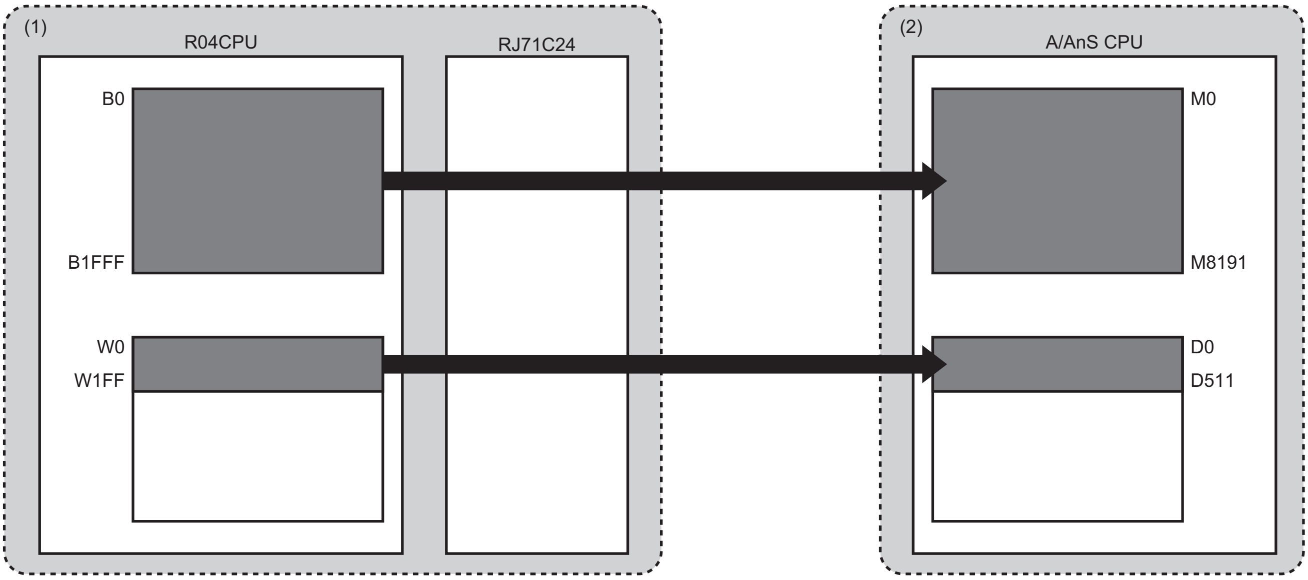
System configuration



(1) Host station (transfer source)
(2) Communication destination (transfer destination)

Device assignment

The following shows assignment of devices sent and received in simple CPU communication.



(1) Host station (transfer source)
(2) Communication destination (transfer destination)

Parameter setting

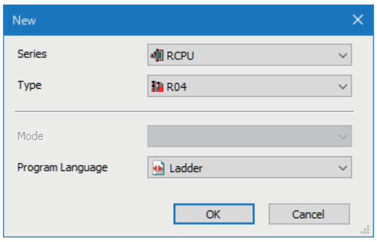
The following shows the procedure for connecting an engineering tool to a CPU module and setting parameters.

■Setting for the host station (transfer source)

Operating procedure

1. Set the CPU module as follows:

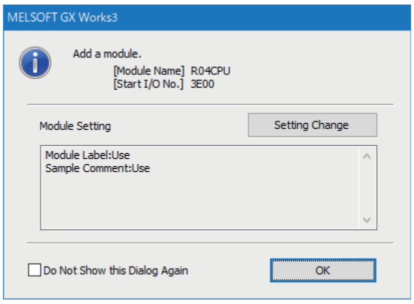
[Project] ⇨ [New]


New

Series	RCPU
Type	R04
Mode	
Program Language	Ladder

2. Click the [Setting Change] button to set to use module labels.

3. Click the [OK] button to add module labels of the CPU module.


MELSOFT GX Works3

Add a module.
[Module Name] R04CPU
[Start I/O No.] 3E00

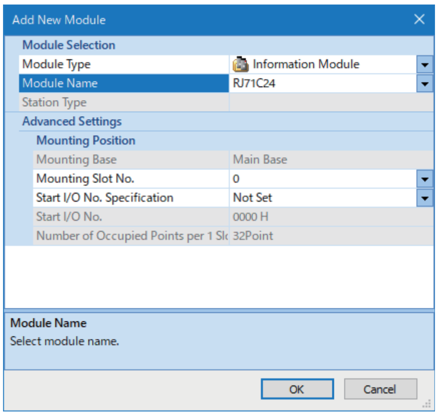
Module Setting

Module Label:Use
Sample Comment:Use

□Do Not Show this Dialog Again

4. Set the RJ71C24 as follows:

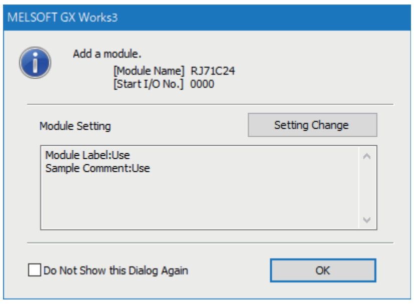
[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ right-click ⇨ [Add New Module]


Add New Module

Module Selection	
Module Type	Information Module
Module Name	RJ71C24
Station Type	
Advanced Settings	
Mounting Position	
Mounting Base	Main Base
Mounting Slot No.	0
Start I/O No. Specification	Not Set
Start I/O No.	0000 H
Number of Occupied Points per 1 Sl	32Point

Module Name
Select module name.

5. Click the [OK] button to add module labels of the RJ71C24.


MELSOFT GX Works3

Add a module.
[Module Name] RJ71C24
[Start I/O No.] 0000

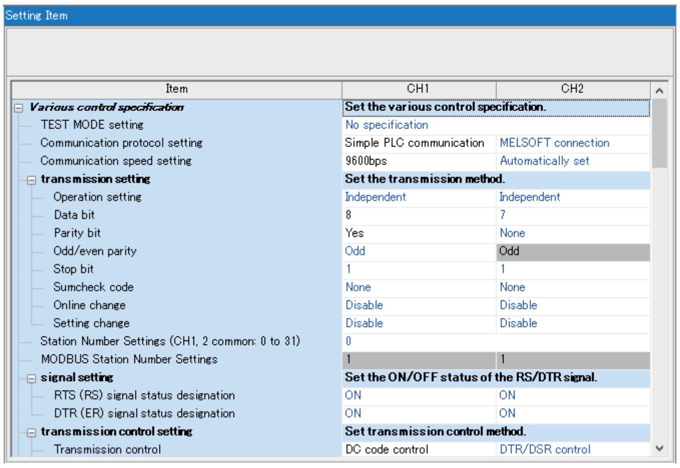
Module Setting

Module Label:Use
Sample Comment:Use

□Do Not Show this Dialog Again

6. Set the items in "Basic Settings" as follows:

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter] ⇨ [Basic Settings] ⇨ [Various control specification] ⇨ [Communication protocol setting]

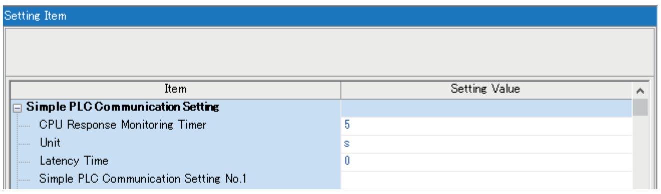


Setting Item

Item	CH1	CH2
⊟ *Various control specification*	Set the various control specification.	
TEST MODE setting	No specification	
Communication protocol setting	Simple PLC communication	MELSOFT connection
Communication speed setting	9600bps	Automatically set
⊟ transmission setting	Set the transmission method.	
Operation setting	Independent	Independent
Data bit	8	7
Parity bit	Yes	None
Odd/even parity	Odd	Odd
Stop bit	1	1
Sumcheck code	None	None
Online change	Disable	Disable
Setting change	Disable	Disable
Station Number Settings (CH1, 2 common: 0 to 31)	0	
MODBUS Station Number Settings	1	1
⊟ signal setting	Set the ON/OFF status of the RS/DTR signal.	
RTS (RS) signal status designation	ON	ON
DTR (ER) signal status designation	ON	ON
⊟ transmission control setting	Set transmission control method.	
Transmission control	DC code control	DTR/DSR control

7. Set the items in "Module Extended Parameter" as follows:

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Extended Parameter] ⇨ [Simple PLC Communication Setting]



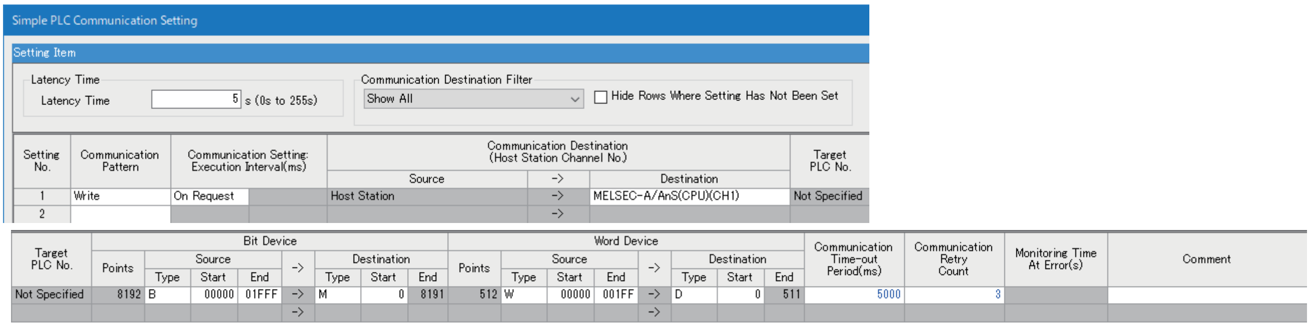
Setting Item

Item	Setting Value
⊟ Simple PLC Communication Setting	
CPU Response Monitoring Timer	5
Unit	s
Latency Time	0
Simple PLC Communication Setting No.1	

A setting screen appears by double-clicking any one of the following in the "Simple PLC Communication Setting" screen.

- "Latency Time"
- "Simple PLC Communication Setting No.1" to "Simple PLC Communication Setting No.512"

8. Set the items in "Simple PLC Communication Setting" as follows:



Simple PLC Communication Setting

Setting Item

Latency Time

Latency Time: 5 s (0s to 255s)

Communication Destination Filter: Show All

☐ Hide Rows Where Setting Has Not Been Set

Setting No.	Communication Pattern	Communication Setting: Execution Interval(ms)	Communication Destination (Host Station Channel No.)			Target PLC No.
			Source	->	Destination	
1	Write	On Request	Host Station	->	MELSEC-A/AnS(CPU)(CH1)	Not Specified
2				->		

Target PLC No.	Bit Device								Word Device								Communication Time-out Period(ms)	Communication Retry Count	Monitoring Time At Error(s)	Comment
	Points	Source			->	Destination			Points	Source			->	Destination						
		Type	Start	End		Type	Start	End		Type	Start	End		Type	Start	End				
Not Specified	8192	B	00000	01FFF	->	M	0	8191	512	W	00000	001FF	->	D	0	511	5000	3		
					->								->							

9. Write the set parameters to the CPU module, and reset the CPU module or turn the power OFF and ON.

[Online] ⇨ [Write to PLC]

Point

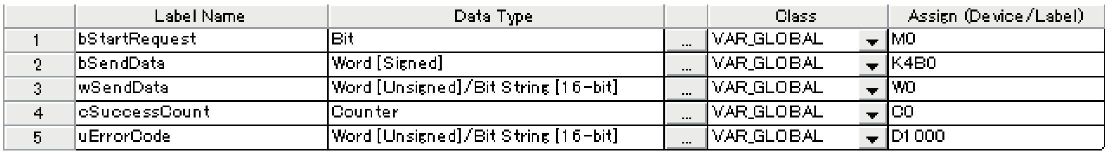
The default settings are used for parameters other than the above in a program example. For the parameters, refer to the following chapter. (☞ Page 455 Module Parameters)

Program examples

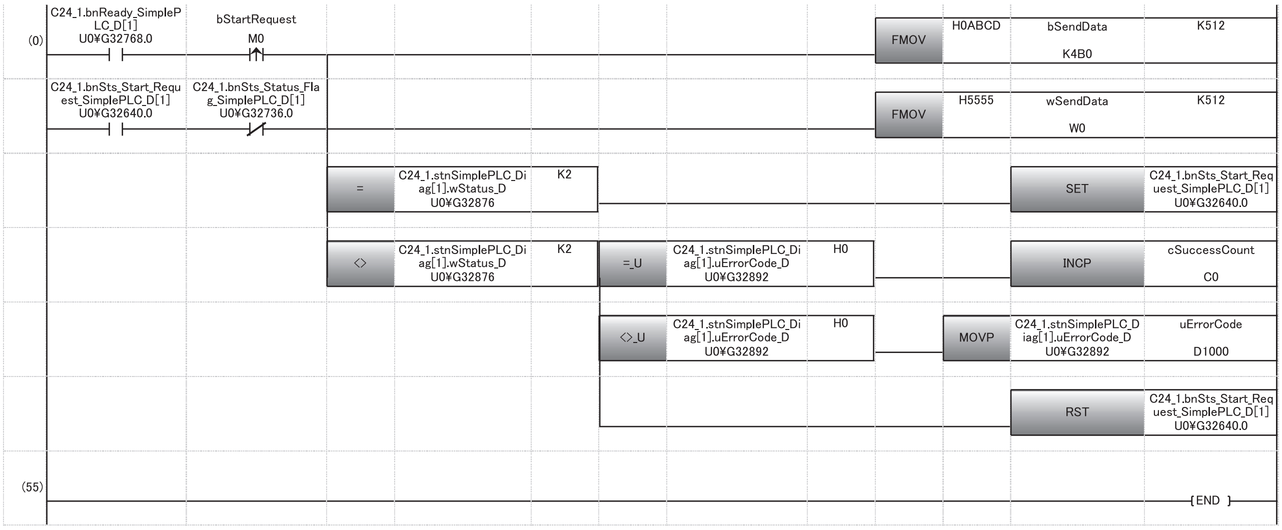
The following shows examples of programs written to a transfer source CPU module.

■Communication on request

The following shows an example of storing ABCDH and 5555H in a bit device and word device respectively and transferring them, when selecting "On Request" for the communication setting.

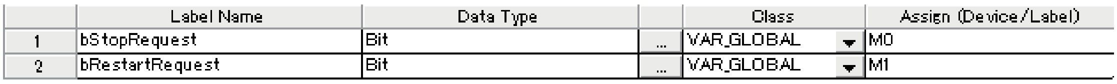
Category	Label name	Description	Device
Module label	C24_1.bnSts_Start_Request_SimplePLC_D[1]	Request to start communication at request for simple CPU communication setting No.1	Un\G32640.0
	C24_1.bnSts_Status_Flag_SimplePLC_D[1]	Execution status flag for simple CPU communication setting No.1	Un\G32736.0
	C24_1.bnReady_SimplePLC_D[1]	Preparation completion flag for simple CPU communication setting No.1	Un\G32768.0
	C24_1.stnSimplePLC_Diag[1].wStatus_D	Communication status storage location of simple CPU communication setting No.1	Un\G32876
	C24_1.stnSimplePLC_Diag[1].uErrorCode_D	Error code storage location of simple CPU communication setting No.1	Un\G32892
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bStartRequest	Bit	...	VAR_GLOBAL	M0
2	bSendData	Word [Signed]	...	VAR_GLOBAL	K4B0
3	wSendData	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W0
4	cSuccessCount	Counter	...	VAR_GLOBAL	C0
5	uErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1000


(0)
C24_1.bnReady_SimplePLC_D[1]
U0¥G32768.0
bStartRequest
M0
FMOV
H0ABCD
bSendData
K4B0
K512
C24_1.bnSts_Start_Request_SimplePLC_D[1]
U0¥G32640.0
C24_1.bnSts_Status_Flag_SimplePLC_D[1]
U0¥G32736.0
FMOV
H5555
wSendData
W0
K512
=
C24_1.stnSimplePLC_Diag[1].wStatus_D
U0¥G32876
K2
SET
C24_1.bnSts_Start_Request_SimplePLC_D[1]
U0¥G32640.0
<>
C24_1.stnSimplePLC_Diag[1].wStatus_D
U0¥G32876
K2
=_U
C24_1.stnSimplePLC_Diag[1].uErrorCode_D
U0¥G32892
H0
INCP
cSuccessCount
C0
<>_U
C24_1.stnSimplePLC_Diag[1].uErrorCode_D
U0¥G32892
H0
MOVP
C24_1.stnSimplePLC_Diag[1].uErrorCode_D
U0¥G32892
uErrorCode
D1000
RST
C24_1.bnSts_Start_Request_SimplePLC_D[1]
U0¥G32640.0
(55)
END

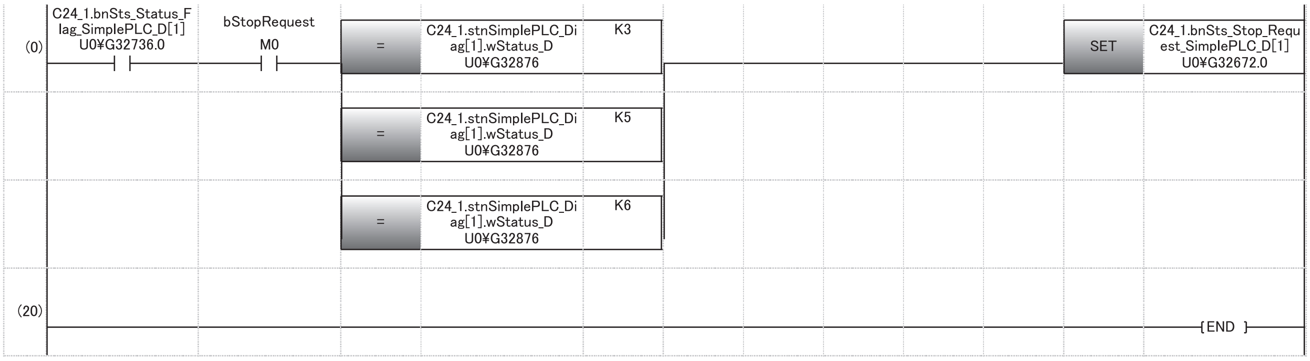
■Stopping and restarting communication

The following shows an example of stopping and restarting communication, when selecting "Fixed Interval" for the communication setting.

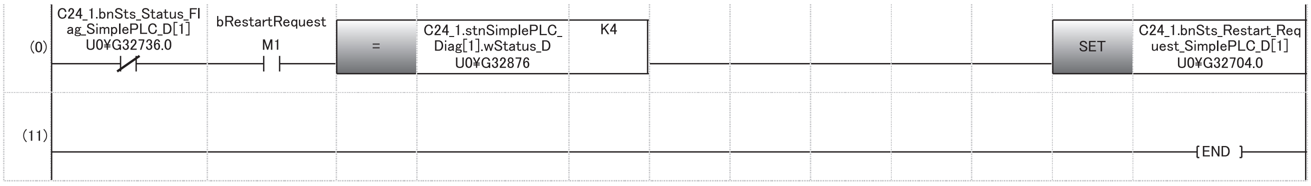
Category	Label name	Description	Device
Module label	C24_1.bnSts_Stop_Request_SimplePLC_D[1]	Request to stop communication for simple CPU communication setting No.1	Un\G32672.0
	C24_1.bnSts_Restart_Request_SimplePLC_D[1]	Request to restart communication for simple CPU communication setting No.1	Un\G32704.0
	C24_1.bnSts_Status_Flag_SimplePLC_D[1]	Execution status flag for simple CPU communication setting No.1	Un\G32736.0
	C24_1.stnSimplePLC_Diag [1].wStatus_D	Communication status storage location of simple CPU communication setting No.1	Un\G32876
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bStopRequest	Bit	...	VAR_GLOBAL	M0
2	bRestartRequest	Bit	...	VAR_GLOBAL	M1

- To stop communication

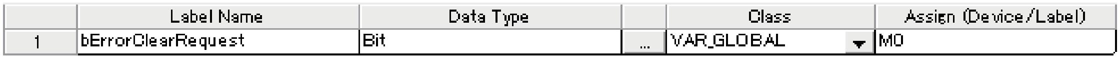

(0)
C24_1.bnSts_Status_Flag_SimplePLC_D[1]
U0¥G32736.0
bStopRequest
M0
=
C24_1.stnSimplePLC_Diag[1].wStatus_D
U0¥G32876
K3
=
C24_1.stnSimplePLC_Diag[1].wStatus_D
U0¥G32876
K5
=
C24_1.stnSimplePLC_Diag[1].wStatus_D
U0¥G32876
K6
SET
C24_1.bnSts_Stop_Request_SimplePLC_D[1]
U0¥G32672.0
(20)
{END }

- To restart communication

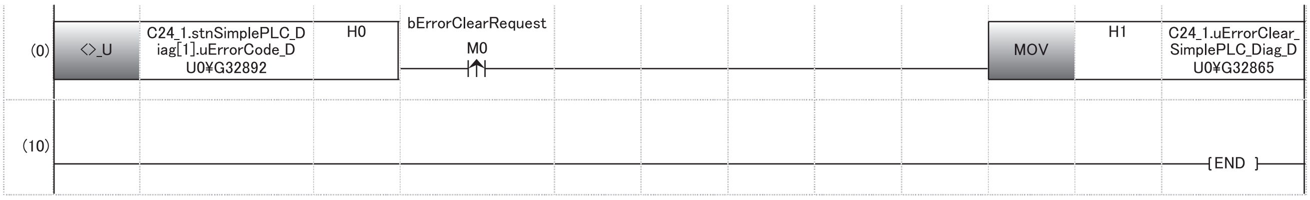

(0)
C24_1.bnSts_Status_Flag_SimplePLC_D[1]
U0¥G32736.0
bRestartRequest
M1
=
C24_1.stnSimplePLC_Diag[1].wStatus_D
U0¥G32876
K4
SET
C24_1.bnSts_Restart_Request_SimplePLC_D[1]
U0¥G32704.0
(11)
{END }

■Clearing errors

The latest error code in the buffer memory can be cleared.

Category	Label name	Description	Device
Module label	C24_1.wErrorClear_SimplePLC_Diag_D	Error clear request for simple CPU communication setting	Un\G32865
	C24_1.stnSimplePLC_Diag[1].uErrorCode_D	Error code storage location of simple CPU communication setting No.1	Un\G32892
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bErrorClearRequest	Bit	...	VAR_GLOBAL	M0


<>_U
C24_1.stnSimplePLC_Diag[1].uErrorCode_D
U0¥G32892
H0
bErrorClearRequest
M0
MOV
H1
C24_1.uErrorClear_SimplePLC_Diag_D
U0¥G32865
(0)
(10)
{END }

6.8 Considerations

This section shows the considerations for using this data communication.

Setting an execution interval

The execution interval of the simple CPU communication function and the time to complete communication vary depending on the load rate of the line, scan time of a communication destination, and system configuration; therefore, a set interval and time may be extended.

■Causes of a long execution interval of the simple CPU communication function

- The scan time of the control CPU module of a C24 with the simple CPU communication function set is long.
- The load on the line is high.
- There are many settings in "Simple PLC Communication Setting."
- A response from a communication destination is slow.
- Multiple C24s with the simple CPU communication function set are mounted on a base unit (main base unit and extension base unit).

Set a sufficient execution interval to perform communication at the set execution interval. Check an execution interval in the buffer memory by actually performing communication if necessary.

Devices of a communication destination

Check the types and ranges of read/write devices of a communication destination.

When selecting "Write" for "Communication Pattern," control data of a communication destination may be rewritten, which may cause a malfunction.

Security

If a remote password is set for a communication destination, simple CPU communication cannot be performed.

Remove the remote password.

Using devices of the host station

When using devices of the host station, be sure to assign devices within the setting range.

Otherwise, data in another device is read or written.

Corrective action if an error occurs during communication

If an error occurs during communication, data may not be written normally.

Check the error and take a corrective action, then perform the communication again.

Data inconsistency

Data in a setting is divided to send and receive depending on the communication destination.

This may cause data inconsistency.

Clearing an error

A value in 'Latest error code' for each setting number is not cleared even after the communication status returns to normal.

To clear an error, turn 'Error clear request' (Un\G32865) from OFF to ON.

Writing data to the buffer memory

Do not write data to 'RS and DTR signal status designation' when performing simple CPU communication.
Otherwise, communication may fail and an error such as a timeout may occur.

Reconnection

When disconnecting and reconnecting the cable, note the following:

- If "Fixed Interval" is selected for the communication setting, connect the cable when '4' (communication stop) or '6' (monitoring at error) is stored in 'Communication status' for each setting number.
- If "On Request" is selected for the communication setting, connect the cable when '2' (waiting for the request) is stored in 'Communication status' for each setting number.

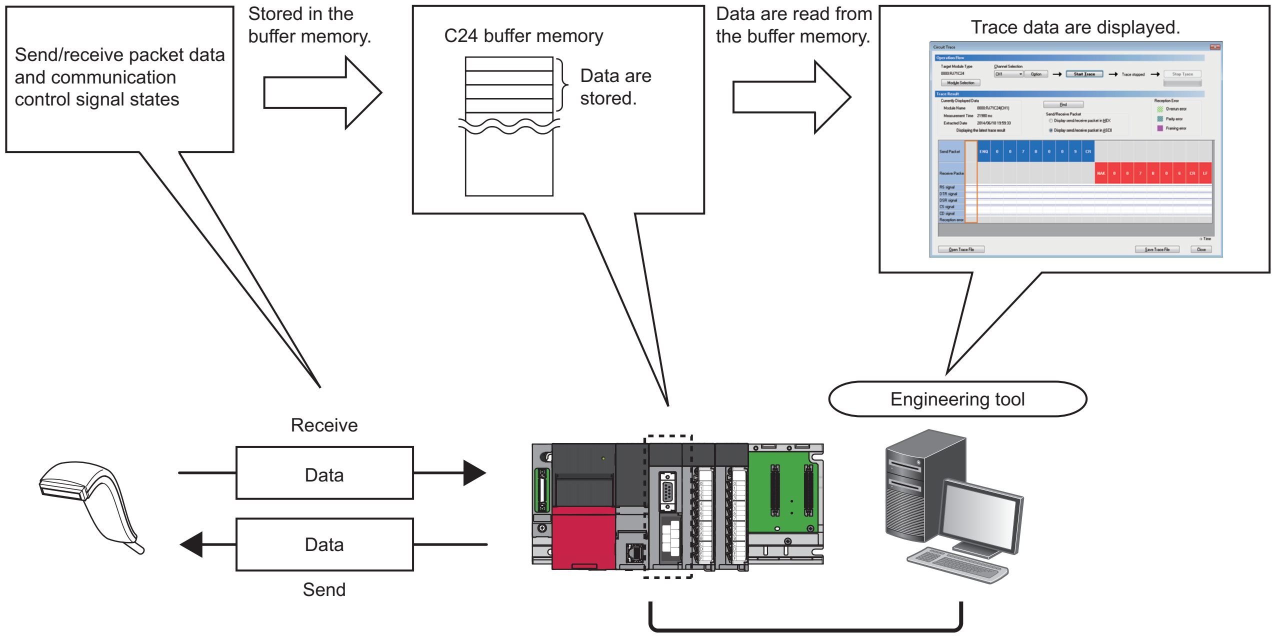
7 DEBUG SUPPORT FUNCTION

The debug support function is designed to support the debugging of communication processing between C24 and target device. The following functions are available to ease system startup work.

- Circuit trace
- Protocol execution log storage function (for predefined protocol only)

7.1 Circuit Trace

The communication data and communication control signal between C24 and target device are traced.


Send/receive packet data and communication control signal states
Stored in the buffer memory.
C24 buffer memory
Data are stored.
Data are read from the buffer memory.
Trace data are displayed.
Engineering tool
Receive
Data
Data
Send

Procedure for the circuit trace

The following shows the procedure for the circuit trace.

Operating procedure

1. Start circuit trace from Engineering tool.
2. Perform communications with the target device.
3. Stop circuit trace in Engineering tool.
4. Circuit trace results are displayed.
5. Confirm the send/receive packet data and communication control signal.

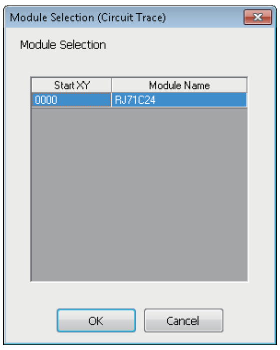
Performing circuit trace

The trace data is stored in the monitor buffer to trace the communication data and communication control signal status.

Operating procedure

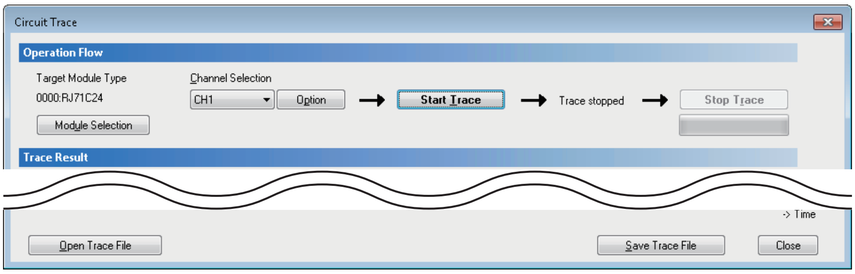
1. Select [Tool] ⇨ [Circuit Trace] in Engineering tool.

2. Click the [Module Selection] button to select the module to be traced in the "Module Selection (Circuit Trace)" screen.



Start XY	Module Name
0000	RJ71C24

3. Select the channel to be traced from "Channel Selection", and click the [Start Trace] button.



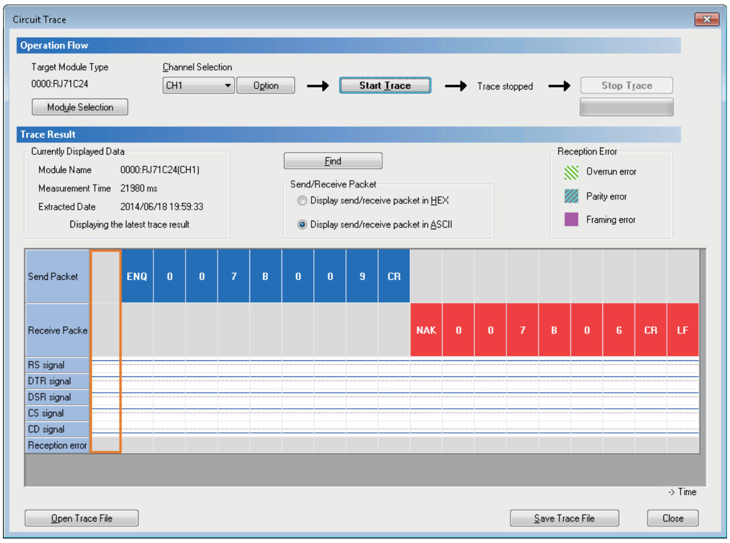
Target Module Type: 0000:RJ71C24

Channel Selection: CH1

4. When the monitor buffer is full, or the [Stop Trace] button is clicked, the trace data are displayed.

5. Confirm the send/receive packets and communication control signals from the displayed trace result.

Window



Target Module Type: 0000:RJ71C24

Channel Selection: CH1

Currently Displayed Data

Module Name 0000:RJ71C24(CH1)

Measurement Time 21980 ms

Extracted Date 2014/06/18 19:59:33

Displaying the latest trace result

Send/Receive Packet

○ Display send/receive packet in HEX

● Display send/receive packet in ASCII

Send Packet		ENQ	0	0	7	B	0	0	9	CR								
Receive Packe											NAK	0	0	7	B	0	6	CR	LF
RS signal																			
DTR signal																			
DSR signal																			
CS signal																			
CD signal																			
Reception error																			

Displayed items

Item		Display/setting content
Trace Result	Currently Displayed Data	Displays the model name, measurement time, and extraction date/time of the module on which the circuit trace is executed.
	[Find] button	Searches the trace data.
	Send/Receive Packet	Select the display format of the send/receive packets. The hexadecimal or ASCII code can be selected.
	Communication control signals	The RS(RTS), ER(DTR), DR(DSR), CS(CTS), and CD(DCD) signal status and reception error are displayed as described below. ■RS, ER, DSR, CS, and CD signals All signals are displayed with blue lines —. When signal is ON: _|‾ When signal is OFF: ‾|_ When the obtained data does not have signal information, the signal is displayed as an OFF status. ■Reception Error Three different errors of overrun error, parity error, and framing error are displayed. Overrun error: ▨ (Green) Parity error : ▩ (Light Blue) Framing error: ■ (Purple)
[Open Trace File] button		Reads and displays the trace data saved in a personal computer.
[Save Trace File] button		Saves the trace data obtained by the circuit trace to a personal computer.
[Close] button		Closes the Circuit Trace screen.

Circuit trace option setting

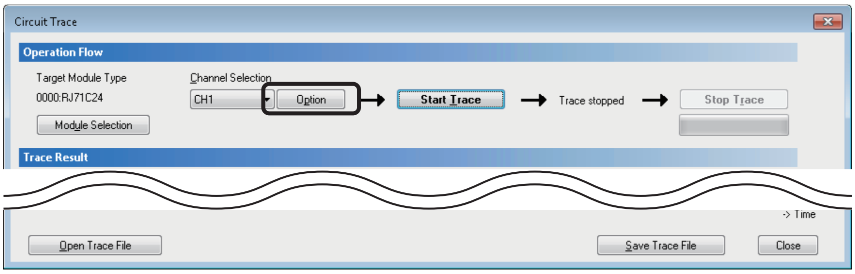
In the circuit trace option setting, the following two items can be set.

- Setting a start address and size of buffer area of C24 in which circuit trace data is stored (hereinafter, 'monitor buffer area')
- Setting whether the circuit trace is stopped or continued when the timer 0 error occurs

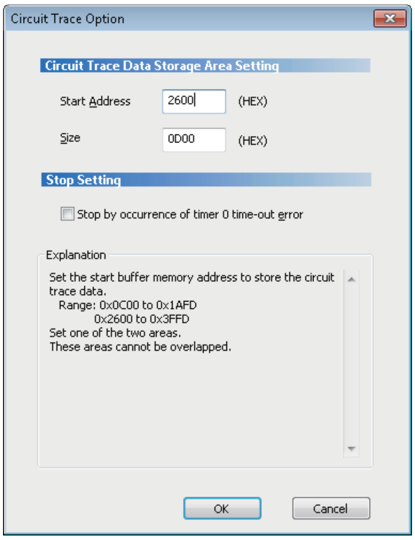
Operating procedure

1. Select [Tool] ⇨ [Circuit Trace] in Engineering tool.

2. Click the [Options] button.



3. On the "Circuit Trace Option" screen, set "Start Address", "Size", and "Stop Setting" of the buffer memory of C24, and click the [OK] button.



Start Address 2600 (HEX)

Size 0D00 (HEX)

□ Stop by occurrence of timer 0 time-out error

Set the start buffer memory address to store the circuit trace data.
Range: 0x0C00 to 0x1AFD
0x2600 to 0x3FFD
Set one of the two areas.
These areas cannot be overlapped.

Displayed items

Item		Display/Setting content
Circuit Trade Data Storage Area Setting	Start Address	Set the start address of the C24 buffer memory. Input the setting in hexadecimal. ■Input range CH1/CH2: 2600H to 3FFDH (C00H to 1AFDH for the user setting area)
	Size	Set the size of the C24 buffer memory. Input the setting in hexadecimal. ■Input range CH1/CH2: 3 to 1A00 words (3 to F00 words for the user setting area) Set the maximum address[*1] for the trace data storage area to be in the range between 2602H to 3FFFH (C02H to 1AFFH for the user setting area). Note that the range of this value is checked at start of the circuit trace when using the user setting area.
Stop Setting	Stop by occurrence of timer 0 timeout error	Set whether or not the circuit trace is stopped when the timer 0 timeout error occurs.

[*1] The maximum address for the trace data storage area is calculated by the following formula.
Maximum address for the trace data storage area = "Start Address" + "Size" – 1

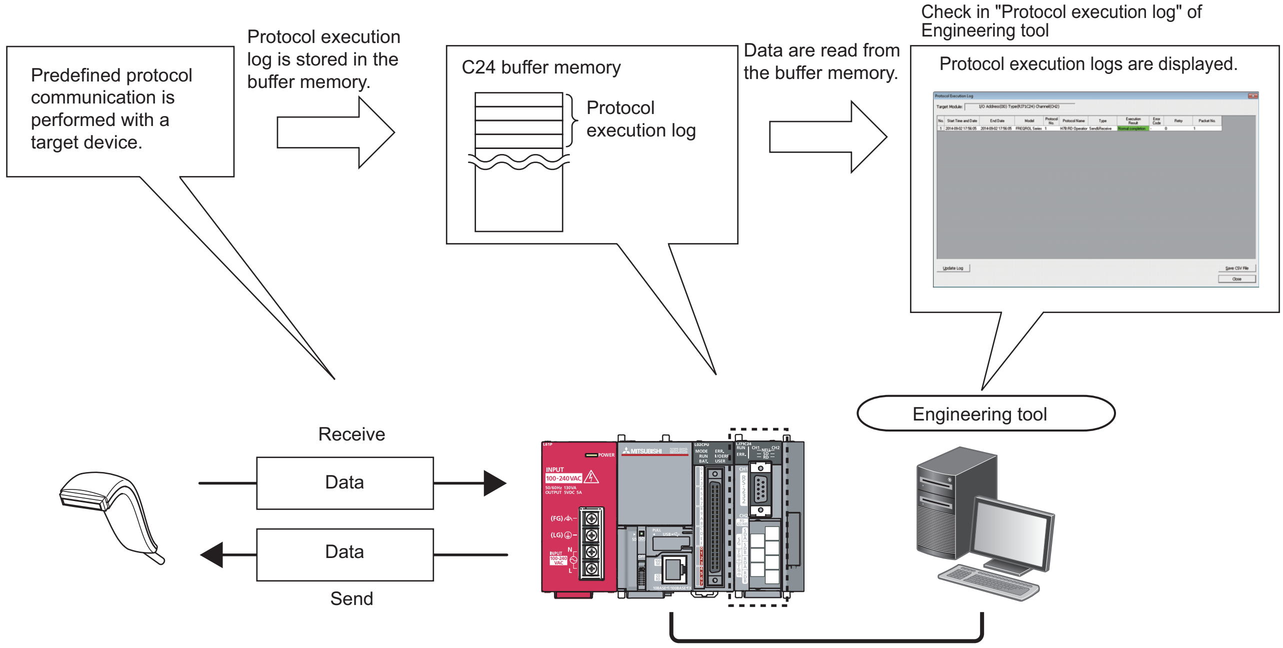
7.2 Protocol Execution Log Storage Function (for Predefined Protocol only)

This function checks the detailed predefined protocol execution status and results for each channel. Up to 32 protocol execution logs can be checked.

If the number of the stored logs exceeds 32, the oldest log will be overwritten.

The protocol execution log can be checked in the following method.

- Checking with Engineering tool
- Checking with the buffer memory


Predefined protocol communication is performed with a target device.
Protocol execution log is stored in the buffer memory.
C24 buffer memory
Protocol execution log
Data are read from the buffer memory.
Check in "Protocol execution log" of Engineering tool
Protocol execution logs are displayed.
Protocol Execution Log
Update Log
Save CSV File
Close
Engineering tool
Receive
Data
Data
Send

Target Module: I/O Address(00) Type(RJ71C24) Channel(CH2)

No.	Start Time and Date	End Date	Model	Protocol No.	Protocol Name	Type	Execution Result	Error Code	Retry	Packet No.
1	2014-09-02 17:56:05	2014-09-02 17:56:05	FREQROL Series	1	H7B:RD Operation	Send&Receive	Normal completion	-	0	1

Precautions

Do not perform the above two methods at the same time.

The latest execution logs are not stored in the buffer memory while they are being updated with the predefined protocol support function of Engineering tool.

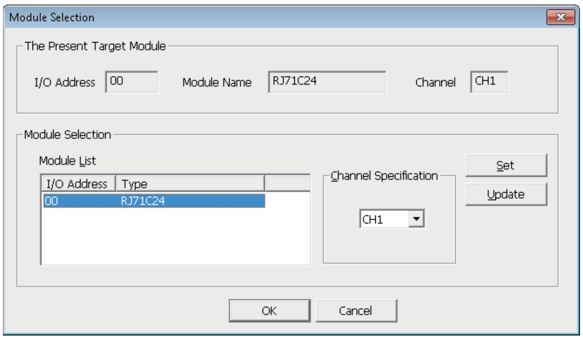
Checking with Engineering tool

Check the protocol execution logs and protocol execution results in the "Protocol Execution Log" screen of Engineering tool.

Operating procedure

1. Select a module to be debugged.

- Display the "Protocol Setting" screen. (☞ Page 31 Data Communication Procedure)
- Select "Protocol Setting" ⇨ [Debugging Support Function] ⇨ [Module Selection].
- Select the I/O address and channel of the module to be debugged, and click the [Set] button.
- As the [OK] button is clicked, the module information is set.

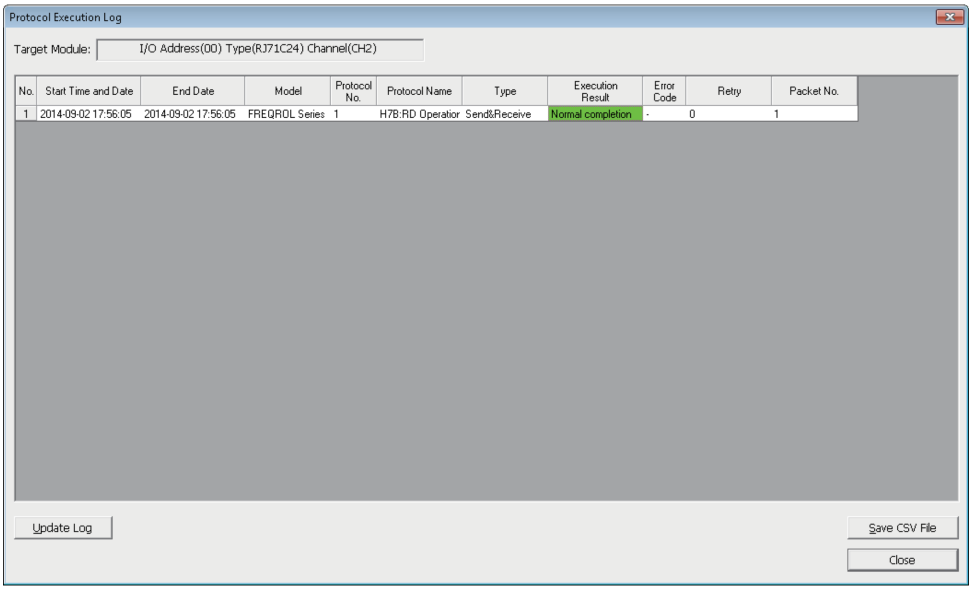


I/O Address	Type
00	RJ71C24

2. Execute the CPRTCL instruction.

3. Select "Protocol Setting" ⇨ [Debugging Support Function] ⇨ [Protocol Execution Log].

Window



Target Module: I/O Address(00) Type(RJ71C24) Channel(CH2)

No.	Start Time and Date	End Date	Model	Protocol No.	Protocol Name	Type	Execution Result	Error Code	Retry	Packet No.
1	2014-09-02 17:56:05	2014-09-02 17:56:05	FREQROL Series	1	H7B:RD Operatior	Send&Receive	Normal completion	-	0	1

Point

The logs displayed in the "Protocol Execution Log" screen can be selected from the following two items, according to the log registration condition.

- Displaying only protocol that completed abnormally.
- Displaying all protocol execution status and execution logs.

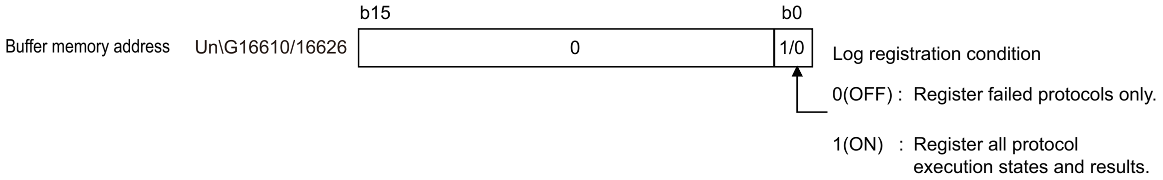
The log registration condition can be set in "Execution log options" in the "Various control specification" screen of Engineering tool.

Checking with the buffer memory

Check the protocol execution logs in the execution log storage area.

Operating procedure

1. The log registration condition can be set by the 'Execution log options' in the buffer memory (Un\G16610/16626) .


b15
b0
Buffer memory address
Un\G16610/16626
0
1/0
Log registration condition
0(OFF) : Register failed protocols only.
1(ON) : Register all protocol execution states and results.

2. Execute the CPRTCL instruction.

3. Check the protocol execution logs stored in the execution log storage area (Un\G16643 to 18177/18435 to 19969).

For details, refer to the following section.

☞ Page 550 Buffer memory list

PART 2 ADDITIONAL FUNCTIONS

This part explains the additional functions of C24.

8 USING THE PROGRAMMABLE CONTROLLER CPU MONITORING FUNCTION

This chapter explains the programmable controller CPU monitoring function with which the C24 monitors the programmable controller CPU based on the monitoring information reregistered by the user.

8.1 Overview

This section explains an overview of the programmable controller CPU monitoring function.

Transmitting data without a program

- The programmable controller CPU monitoring function enables C24 to monitor the CPU module on the host station at time intervals set by the user by reregistering data to be used for the programmable controller CPU monitoring function. Data transmission to the target device communicating with the MC or nonprocedural protocol is possible without a program.
- The following monitoring information selected by the user can be sent to the target device as the CPU module monitoring results.

Transmission method	Monitoring result		Without the modem function	Combined use of the modem function (modem communication)
Data transmission	Devices on the host station CPU module (information of the device to be monitored)	Numeric value stored in a word device	○	○
		ON/OFF status of a bit device		
	Status of module of the host station CPU module			

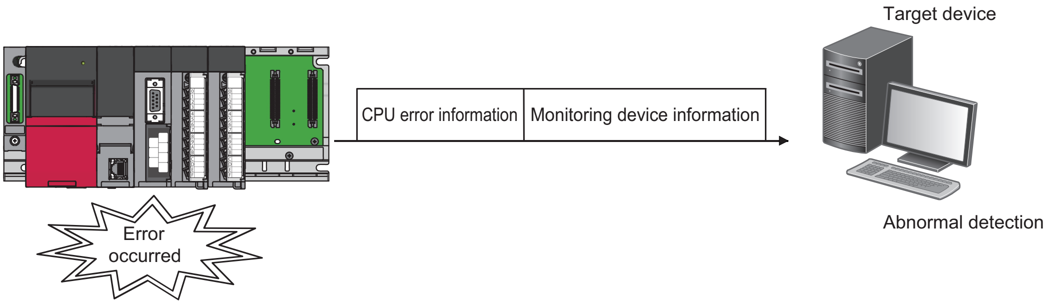
- There are two timings for transmitting the programmable controller CPU monitoring results to the target device; fixed cycle send and condition match send.
  The fixed cycle send transmits the programmable controller CPU monitoring results each time when CPU module is monitored.
  The condition match send transmits when the information read from the CPU module satisfies the defined conditions and when an error is detected in the CPU module.

Simplifying device monitoring procedure

When device monitoring is performed by communication using the MC protocol, the target device must repeatedly perform monitor request transmission and monitor data reception processing after it executes monitor registration.
By designating the fixed cycle send for the programmable controller CPU monitoring function, the device data can be monitored without performing the monitor request reception processing.

Notifying programmable controller CPU error

In the condition match send, error information can be transmitted to the target device whenever CPU module error occurs without any programming.



8.2 Programmable Controller CPU Monitoring Function

This section explains the programmable controller CPU monitoring function.

Data registration for using the programmable controller CPU monitoring function

The following explains the data registration by the user to use the programmable controller CPU monitoring function.

Registration procedure for the Programmable Controller CPU monitoring function

Programmable controller CPU monitor registration for C24 that is required to use the programmable controller CPU monitoring function is described in the following sections. The registration can be performed using one of the following methods:

- Registration with an Engineering tool
- Registration using the programmable controller CPU monitoring registration command (0630) for communication with the MC protocol (📖MELSEC Communication Protocol Reference Manual)
- Registration using the programmable controller CPU "CSET" instruction (📖 MELSEC iQ-R Programming Manual (Module Dedicated Instructions))

Combination with modem function

Before transmitting the CPU module monitoring result when used in conjunction with the modem function, register the data for connection of the modem function.

🖱 Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ Module Name ⇨ [Module Parameter] ⇨ "Application Settings" ⇨ "Modem function setting"

Starting programmable controller CPU monitoring function

By registering the data for using the programmable controller CPU monitoring function, C24 begins monitoring the programmable controller CPU.

CPU module monitoring information

The following explains the monitoring target information used to perform the programmable controller CPU monitoring function.

Target of programmable controller CPU monitoring function

The following information can be registered as the target of the programmable controller CPU monitoring function.

- Device monitoring for the host station CPU module
  Monitoring of the numeric values stored in the word device
  Monitoring of bit devices ON/OFF status
- Monitoring status of the host station CPU module

Registering word devices and bit devices

For monitoring word and bit devices, up to 960 device points (up to 15360 bits for only bit devices) in total, which is equivalent to 10 blocks in total when any continuous device range comprised one block, can be registered.

Since monitoring of the CPU module status on the host station will be also registered as one block, up to 11 blocks can be registered.

- 11 ≥ (Number of registered word device blocks + Number of registered bit device blocks) + CPU status monitoring (1 block)
- 960 points ≥ (Total number of all word device block points (1 point = 1 word) + Total number of all bit device block points (1 point = 16 bits))

Point

To register the device memory to monitor, designate the word device blocks for the registered word blocks, then designate the bit device blocks for the registered bit blocks.

Example for registering word devices and bit devices

The start device of each block is the monitoring target for device monitoring of the blocks for which the word and bit devices are registered.

Ex.

For a block in which 10 points of word devices from D100 to D109 are registered

Monitoring target: Numeric value to be stored in D100

Data transmission: Numeric values to be stored in D100 to D109

Ex.

For a block in which two points of bit devices from M100 to M131 are registered

Monitoring target: ON/OFF status of M100

Data transmission: ON/OFF status of M100 to M131

List of device code

The word and bit devices that can be designated as the monitoring targets and the device codes that are used to register the monitoring devices are listed in the table below.

Register the devices using the device ranges existing in the CPU module.

Category	Device		Device type		Device code		Device range (Default)
			Bit	Word	ASCII	Binary	
Internal system	Special relay		○	—	SM	91H	0 to 4096
	Special register		—	○	SD	A9H	
Internal user	Input		○	—	X*	9CH	0 to 2FFFH
	Output		○	—	Y*	9DH	
	Internal relay		○	—	M*	90H	0 to 12287
	Latch relay		○	—	L*	92H	0 to 8191
	Annunciator		○	—	F*	93H	0 to 2047
	Edge relay		○	—	V*	94H	
	Link relay		○	—	B*	A0H	0 to 1FFFH
	Data register		—	○	D*	A8H	0 to 18431
	Link register		—	○	W*	B4H	0 to 1FFH
	Timer	Contact	○	—	TS	C1H	0 to 1023
		Coil	○	—	TC	C0H	
		Current value	—	○	TN	C2H	
	Retentive timer	Contact	○	—	SS	C7H	0
		Coil	○	—	SC	C6H	
		Current value	—	○	SN	C8H	
	Counter	Contact	○	—	CS	C4H	0 to 511
		Coil	○	—	CC	C3H	
		Current value	—	○	CN	C5H	
	Link special relay		○	—	SB	A1H	0 to 7FFH
	Link special register		—	○	SW	B5H	
	Direct input		○	—	DX	A2H	0 to 2FFFH
	Direct output		○	—	DY	A3H	
	Index register		—	○	Z*	CCH	0 to 20
Register	File register		—	○	R*	AFH	0 to 32767
			—	○	ZR	B0H	—

Point

- Designating a non-existent device code will result in an error.
- When the device range in the parameter setting has been changed, the new device range can be set as the programmable controller CPU monitoring target.

Monitoring timing of CPU module

The following explains the timing for programmable controller CPU monitoring when the programmable controller CPU monitoring function is performed.

To monitor the CPU module, C24 reads the monitoring information (device information and CPU module status information) from the CPU module at the set interval.

Cycle time

1 to 65535 (unit: 100 ms/s/min) can be registered.

Use the following formula as a reference when registering the cycle time.

■When transmitting device data or the programmable controller CPU status

- Cycle time designation > K + SST + PT + DTT

■When transmitting data in conjunction with the modem function

- Cycle time designation > K + SST + PT + DTT + DTTm + MCDT + CDWT

  When modem initialization has not been performed, the modem initialization time will be added. (Performing modem initialization in advance is recommended.)

The items in the formula above is as follows:

- K: 60 ms constant (C24 internal processing time)
- SST (Sequence scan time): Time required for one scan
- PT (Processing time): Processing time for the "Multiple block batch read word unit command 0406"

  For 1 point : 10.2 ms

  For 480 points: 21.1 ms

  For 960 points: 32.6 ms
- DTT (data transmission time) = 1 ÷ Transmission rate × Number of bit for one byte during transmission × Number of byte for send data

  Number of bit for one byte portion during transmission = 1 + Data bit count + Parity bit + Stop bit count (parity bit: Yes = 1, None = 0)
- DTTm (Data transmission delay time by the modem): Depends on the modem specification, line specification, and line status.
- MCDT (modem connection and disconnection time): Depends on the modem specifications, line specifications, and line status.
- CDWT (Line disconnection wait time): Wait time for disconnecting line
- Modem initialization time: Depends on the modem specifications.

Point

Since C24 reads the monitoring information (device data, programmable controller CPU status) at the time of the next END process of the CPU module after the cycle time elapses, make the cycle time as long as possible.

The following phenomenon may be occurred if the cycle time is short.

- The number of scan of which CPU module scan time gets longer will be increased.
- The processing time for other data communication functions will be increased due to the increase of the C24 programmable controller CPU monitoring function processing time.
- The load on the target device will be increased.

Timings of transmitting monitoring results to the target device

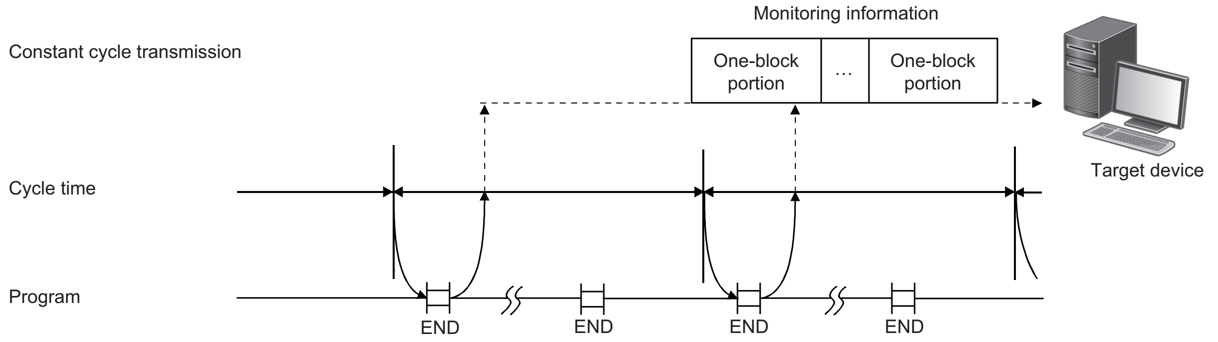
The following explains the timings for the transmission of CPU module monitoring results.

There are two transmission methods for transmitting the monitoring results of the CPU module on the host station to the target device; fixed cycle send and condition match send. One of these methods must be selected by the user during programmable controller CPU monitoring registration.

Fixed cycle send

The monitoring results are transmitted each time the monitoring information is read from the CPU module.

(Timing to transmit data)



Condition match send

■Transmission of monitoring results

For device monitoring, the monitoring conditions registered by the user (conditions for sending monitoring results), the monitoring condition values and the monitoring information read from the CPU module are compared. The monitoring results are transmitted when there is a block where the monitoring conditions match.

For programmable controller CPU status monitoring, the monitoring results are transmitted only once when an error is detected for the first time from the status information read from the CPU module. (This corresponds to the edge triggered transmission described below.)

■Edge triggered/level triggered transmission

Two transmission methods of the monitoring results are available for the condition match send for device monitoring: edge triggered transmission and level triggered transmission.

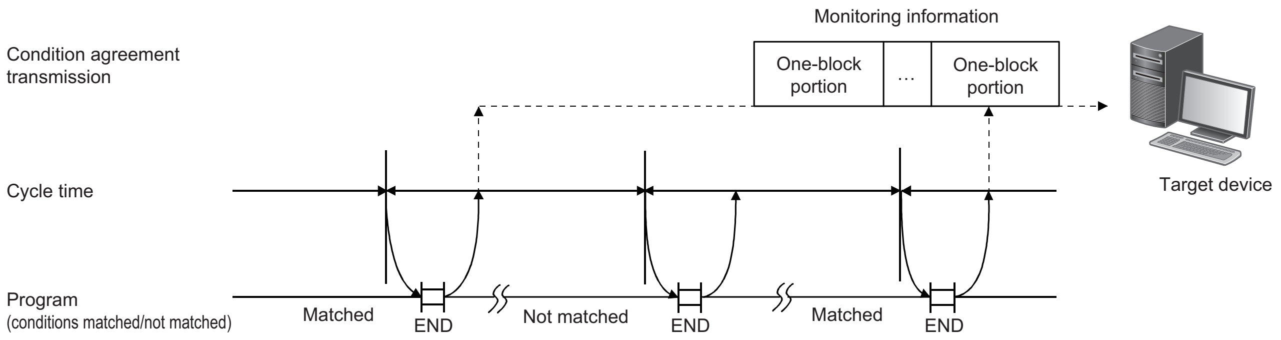
- Edge triggered transmission

  The monitoring conditions registered by the user (conditions for transmitting monitoring results), the monitoring condition values and the monitoring information read from the CPU module are compared. Then, the monitoring results are transmitted only once when a satisfaction of the monitoring conditions is detected for the first time.

  After that, when the monitoring information read from the CPU module does not match the monitoring conditions and then it matches the monitoring conditions once again, the monitoring results are transmitted.
- Level triggered transmission

  The monitoring conditions registered by the user (conditions for transmitting monitoring results), the monitoring condition values and the monitoring information read from the CPU module are compared. While the monitoring conditions are satisfied, the monitoring results are transmitted at each cycle time.

(Timing to transmit data)



■Monitoring condition

In the condition match send for device monitoring, the start device for each block is the monitoring target for condition monitoring of each block device.

For the condition match send, the monitoring conditions that can be designated for the device to be registered by the user and the registration values when designating the monitoring condition are listed in the table below.

Register the monitoring conditions for the start device of each block using the following table.

Monitoring condition		Registration value		Valid designated device	
		For edge triggered transmissions	For level triggered transmissions	Bit	Word
Device value or status = device monitoring condition value or status		0001H	0101H	○	○
Device value or status ≠ device monitoring condition value or status		0002H	0102H		
Unsigned	Monitoring device ≤ monitoring condition value	0003H	0103H	×	
	Monitoring device < monitoring condition value	0004H	0104H		
	Monitoring device ≥ monitoring condition value	0005H	0105H		
	Monitoring device > monitoring condition value	0006H	0106H		
Signed	Monitoring device ≤ monitoring condition value	0007H	0107H		
	Monitoring device < monitoring condition value	0008H	0108H		
	Monitoring device ≥ monitoring condition value	0009H	0109H		
	Monitoring device > monitoring condition value	000AH	010AH		

■Monitoring condition value, status

In device monitoring, register the monitoring condition value or status for C24 to judge that the numeric value/status of the monitoring device for condition match send (start device of each block) satisfy the condition with numeric values listed below.

Type of monitoring device	Monitoring condition value or status	Registration value
Bit device	OFF	0000H
	ON	0001H
Word device	Numerical value	0000H to FFFFH

Ex.

Condition satisfies when M0 = ON

·Registration value for the monitoring condition: 0001H

·Registration value for the monitoring condition value or status: 0001H

Ex.

Condition satisfies when D0 > 100 (signed)

·Registration value for the monitoring condition: 000AH

·Registration value for the monitoring condition value or status: 100 (64H)

Point

In status monitoring for the CPU module, the monitoring conditions and condition values for condition match send are not registered.

Instead, it is registered whether or not the status monitoring of CPU module will be performed.

Monitoring results in the condition match send are transmitted only once when an error is detected by the status information read from the CPU module for the first time.

Transmission methods of monitoring results and data transmitted to the target device

This section explains the transmission method of the CPU module monitoring results and the data to be transmitted to the target device.

Data transmission to target device communicating with MC protocol

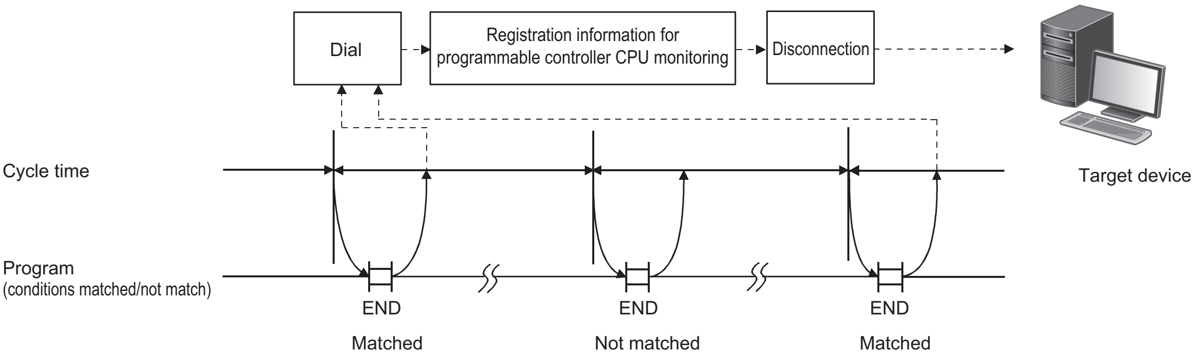
Data is transmitted in the same format as that for a message transmitted with the on-demand function with the on-demand data portion replaced with the device information and programmable controller CPU status information, as explained in "When sending the monitoring results as data during fixed cycle send" or "When sending the monitoring results as data during condition match send" below. (📖MELSEC Communication Protocol Reference Manual)

■When the interface that uses the modem function is set to the MC protocol

The connection processing and disconnection processing to the modem are performed when the programmable controller CPU monitoring results are transmitted.

Ex.

Data transmission by MC protocol with modem function (condition match send)


Dial
Registration information for programmable controller CPU monitoring
Disconnection
Target device
Cycle time
Program
(conditions matched/not match)
END
END
END
Matched
Not matched
Matched

■When the transmission of on-demand data using user frames is designated

Data is transmitted in the transmission format for on-demand data using a user frame with the on-demand data portion replaced with the device information and programmable controller CPU status information, as explained in "When sending the monitoring results as data during fixed cycle send" and "When sending the monitoring results as data during condition match send" below.

See the following explanation for data reception for the target device side.

- Device information, programmable controller CPU status information arrangement (📖 MELSEC Communication Protocol Reference Manual)
- Arrangement of data in the user frame portion to be transmitted (☞ Page 299 ON-DEMAND DATA COMMUNICATIONS USING USER FRAMES)

■When transmitting the monitoring results as data during fixed cycle send

Data equivalent to the entire blocks of the monitoring target device information and programmable controller CPU status information is transmitted in batch.

■When transmitting the monitoring results as data during condition match send

The device information for a block which the monitoring conditions are matched and the programmable controller CPU status information at an occurrence of an error are transmitted with the head data (header) and last data (footer) for the on-demand function added for each block.

Transmission is performed in the order of the programmable controller CPU status information, the device information registered in the word block, and then the device information registered in the bit block.

Point

When data is communicated using the MC protocol format 1 to 4, all of the device monitoring start device number is converted to hexadecimal ASCII data and transmitted. (The same conversion is performed during either fixed cycle send or condition match send.)

Data transmission to target device communicating with nonprocedural protocol

The device information and CPU information are transmitted by the word/byte unit designation.

■When ASCII-BIN conversion for communication data is designated

Data is converted to ASCII code data and transmitted. (The transmission example is shown in the "Device information and programmable controller CPU status information".)

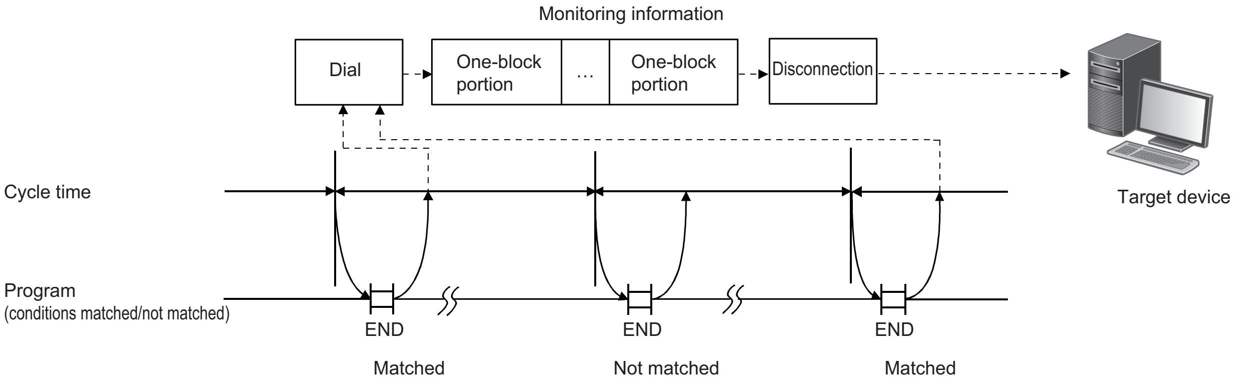
- When the word/byte units designation is set to word
  The device information and CPU information is transmitted by 1-word in the order of (H) and (L).
- When the word/byte units designation is set to byte
  The device information and CPU information is transmitted by 1-word in the order of (L) and (H).

■When the interface that uses the modem function is set to the nonprocedural protocol

The connection processing and disconnection processing to the modem are performed when the programmable controller CPU monitoring results are transmitted.

Ex.

Data transmission by nonprocedural protocol with modem function (condition match send)



■When transmitting monitoring results as data during fixed cycle send

The device information of one or more user frame numbers and programmable controller CPU status information that have been currently designated for the fixed cycle send in C24 is transmitted in batch.

■When transmitting the monitoring results as data during condition match send

The device information of one or more user frame numbers and programmable controller CPU status information that have been currently designated for the condition match send of the block where the monitoring conditions satisfied in C24 is transmitted in batch.

When the monitoring conditions of two or more blocks match, the device information and programmable controller CPU status information are transmitted for each block.

Point

When the ASCII-BIN conversion for the communication data using the nonprocedural protocol is performed, all of the device monitoring start device number is converted to hexadecimal ASCII data and transmitted. (The same conversion is performed during either fixed cycle send or condition match send.)

■The user frame numbers that can be designated for data transmission of programmable controller CPU monitoring results

- 1H to 3E7H (Default registration frame numbers)
- 3E8H to 4AFH (Frame numbers registered by the user in the flash ROM)
- 8001H to 801FH (Frame numbers registered by the user in the buffer memory)
- B001H to B01FH (Dedicated frame numbers for this function) (☞ Page 161 Transmission instruction of device information and programmable controller CPU status information when transmitting the monitoring results)

For details on how to designate user frame numbers, refer to the following section.

☞ Page 308 DATA COMMUNICATIONS USING USER FRAMES

■Transmission instruction of device information and programmable controller CPU status information when transmitting the monitoring results

Use the following dedicated user frame numbers.

Frame No.	Information to be transmitted		Valid function	
			Fixed cycle send	Condition match send
B001H	Device information for the block registered in number n	Number 1	○	○
B002H		Number 2		
B003H		Number 3		
B004H		Number 4		
B005H		Number 5		
B006H		Number 6		
B007H		Number 7		
B008H		Number 8		
B009H		Number 9		
B00AH		Number 10		
B061H	Programmable controller CPU status information (CPU error monitoring data)		○	○
B080H	Number of blocks to be sent		○	○
B081H	Monitoring result information for all blocks			
B082H	Monitoring result information for blocks satisfying the conditions		×	○

■Device information and programmable controller CPU status information

It is transmitted in the order of data shown below.

☞ Page 162 Data arrangement of device information and programmable controller CPU status information

For ASCII-BIN conversion, set "ASCII-BIN conversion designation" to "ON".

When user frame is designated with the number, of which bit 14 for the numeric value indicating the user frame number is ON, ASCII-BIN conversion of the corresponding send data is not performed. It will be transmitted as binary data. (☞ Page 380 ASCII-BIN Conversion for Data Communicated with Nonprocedural Protocol)

Data arrangement of device information and programmable controller CPU status information

When user frame numbers B001H to B00AH are designated (example for one block)

■When word device data (W100 to W103, (4 points)) is transmitted

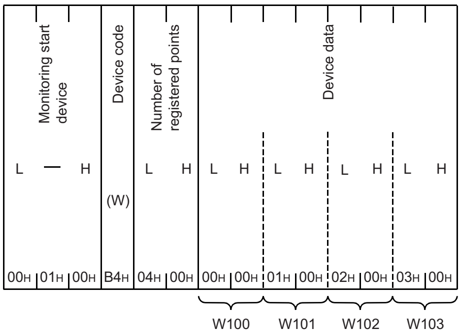
When the word/byte units designation is set to word unit, the device data will be transmitted in the order of (H)→(L).

The number of registered points is the number of points in word units.

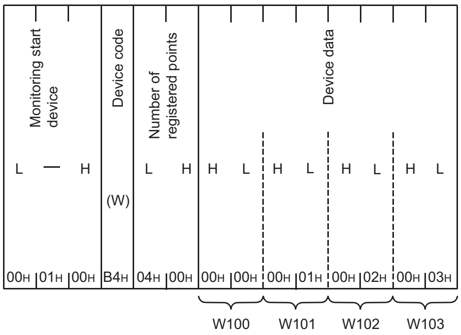
- (When ASCII-BIN conversion is not performed)

  The total number of bytes for the device data portion is the number of device points × 2.

When the word/byte units designation is byte


Monitoring start device
Device code
Number of registered points
Device data
L — H
(W)
L H
L H | L H | L H | L H
00H | 01H | 00H | B4H | 04H | 00H | 00H | 00H | 01H | 00H | 02H | 00H | 03H | 00H
W100
W101
W102
W103

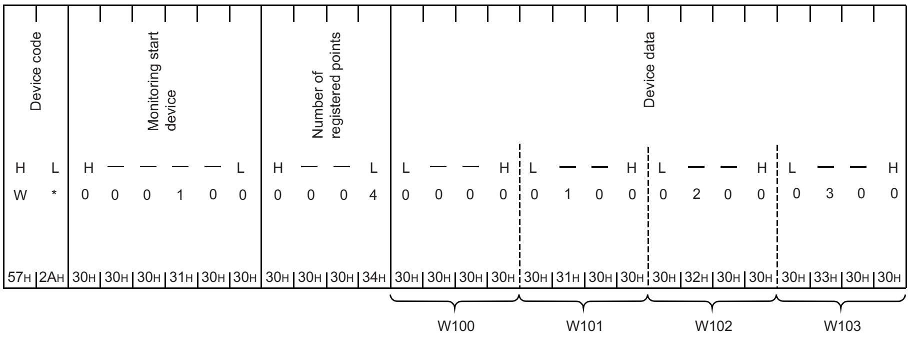
When the word/byte units designation is word


Monitoring start device
Device code
Number of registered points
Device data
L — H
(W)
L H
H L | H L | H L | H L
00H | 01H | 00H | B4H | 04H | 00H | 00H | 00H | 00H | 01H | 00H | 02H | 00H | 03H
W100
W101
W102
W103

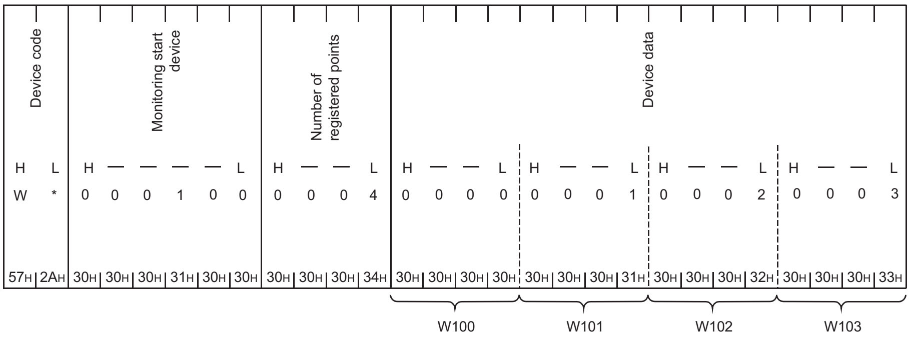
- (When ASCII-BIN conversion is performed)

  The total number of bytes for the device data portion is the number of device points × 4.

When the word/byte units designation is byte


Device code
Monitoring start device
Number of registered points
Device data
H L
W *
H — — — — L
0 0 0 1 0 0
H — — L
0 0 0 4
L — — H | L — — H | L — — H | L — — H
0 0 0 0 | 0 1 0 0 | 0 2 0 0 | 0 3 0 0
57H | 2AH | 30H | 30H | 30H | 31H | 30H | 30H | 30H | 30H | 30H | 34H | 30H | 30H | 30H | 30H | 30H | 31H | 30H | 30H | 30H | 32H | 30H | 30H | 30H | 33H | 30H | 30H
W100
W101
W102
W103

When the word/byte units designation is word


Device code
Monitoring start device
Number of registered points
Device data
H L
W *
H — — — — L
0 0 0 1 0 0
H — — L
0 0 0 4
H — — L | H — — L | H — — L | H — — L
0 0 0 0 | 0 0 0 1 | 0 0 0 2 | 0 0 0 3
57H | 2AH | 30H | 30H | 30H | 31H | 30H | 30H | 30H | 30H | 30H | 34H | 30H | 30H | 30H | 30H | 30H | 30H | 30H | 31H | 30H | 30H | 30H | 32H | 30H | 30H | 30H | 33H
W100
W101
W102
W103

■When data for bit device (M16 to M175, (10 point)) is transmitted

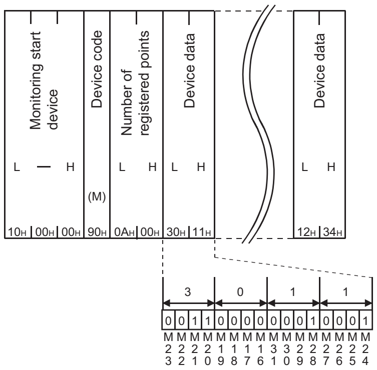
When the word/device units designation is set to word unit, the device data will be sent in the order of (H)→(L).

The number of registered points is the number of points in word units.

- (When ASCII-BIN conversion is not performed)

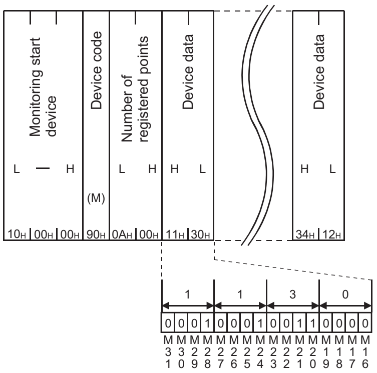
  The total number of bytes for the device data portion is the number of device points × 2.

When the word/byte units designation is byte



Monitoring start device
Device code
Number of registered points
Device data
Device data
L — H
(M)
L H
L H
L H
10H 00H 00H 90H 0AH 00H 30H 11H
12H 34H
3 0 1 1
0 0 1 1 0 0 0 0 0 0 0 1 0 0 0 1
M23 M22 M21 M20 M19 M18 M17 M16 M31 M30 M29 M28 M27 M26 M25 M24

When the word/byte units designation is word

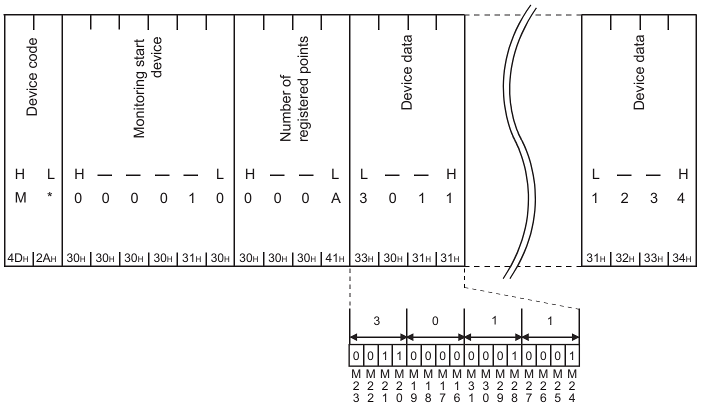


Monitoring start device
Device code
Number of registered points
Device data
Device data
L — H
(M)
L H
H L
H L
10H 00H 00H 90H 0AH 00H 11H 30H
34H 12H
1 1 3 0
0 0 0 1 0 0 0 1 0 0 1 1 0 0 0 0
M31 M30 M29 M28 M27 M26 M25 M24 M23 M22 M21 M20 M19 M18 M17 M16

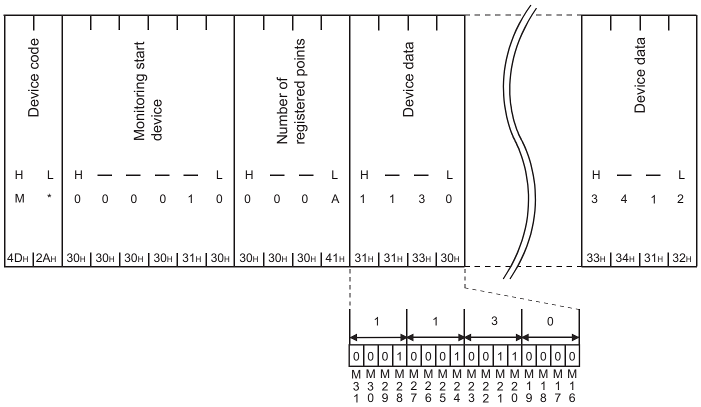
- (When ASCII-BIN conversion is performed)

  The total number of bytes for the device data portion is the number of device points × 4.

When the word/byte units designation is byte



When the word/byte units designation is word



When user frame No. B061H is designated

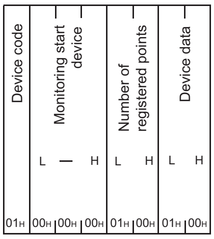
Programmable controller CPU status information (for one block) is transmitted as the following data:

Item		Communicating with ASCII code	Communicating with binary code	Remarks
Device code		'01'	01H	All fixed value
Number of registered points		'0001'	0001H	
Monitoring start device		'000000'	000000H	
Device data	During normal operation	'0000'	0000H	
	Module warning occurring	'0001'	0001H	
	Module error/module system error occurring	'0002'	0002H	

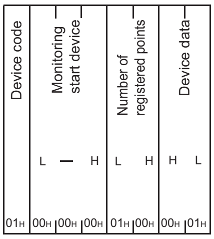
- (When ASCII-BIN conversion is not performed)

  The total number of bytes for the device data portion is the number of device points × 2.

When the word/byte units designation is byte



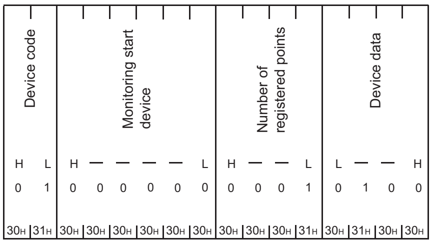
When the word/byte units designation is word



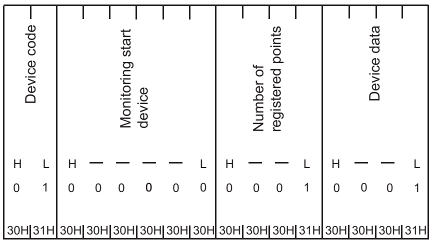
- (When ASCII-BIN conversion is performed)

  The total number of bytes for the device data portion is the number of device points × 4.

When the word/byte units designation is byte



When the word/byte units designation is word



When user frame No. B080H is designated

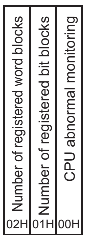
The transmission block is transmitted as follows.

Ex.

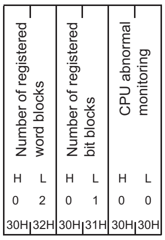
Number of registered word blocks: 2 (D0 to D3 (4 points), W100 to W107 (8 points))

Number of registered bit blocks : 1 (M0 to M31 (2 points))

(When ASCII-BIN conversion is not performed)


Number of registered word blocks
Number of registered bit blocks
CPU abnormal monitoring
02H
01H
00H

(When ASCII-BIN conversion is performed)


Number of registered word blocks
Number of registered bit blocks
CPU abnormal monitoring
H L
H L
H L
0 2
0 1
0 0
30H 32H
30H 31H
30H 30H

When user frame No. B081H is designated

The information on monitoring results for all blocks are sent as shown below.

Results are transmitted in the order of the device information registered in the word block, the device information registered in the bit block, and the programmable controller CPU status information.

Ex.

Number of registered word blocks: 1 (W100 to W103 (4 points))

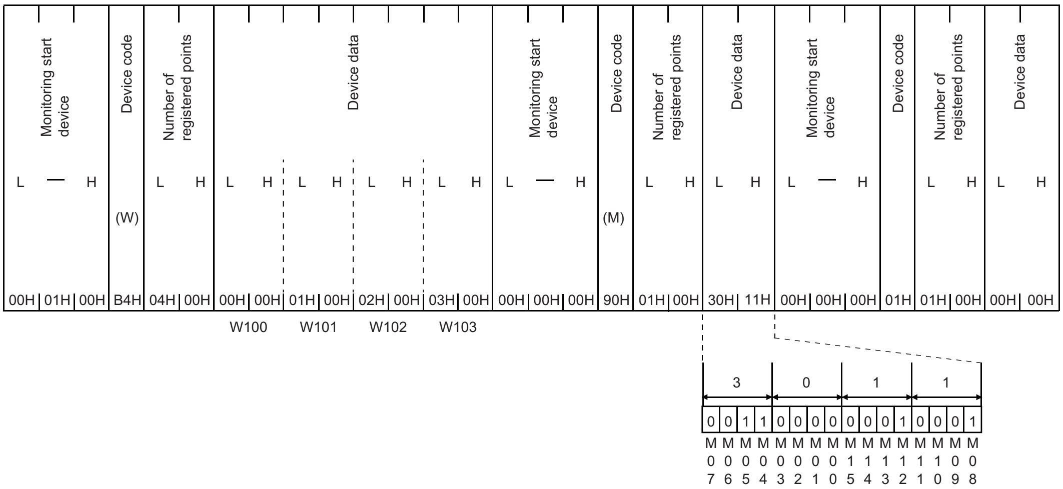
Number of registered bit blocks : 1 (M0 to M15 (1 point))

Perform CPU status monitoring : 1 (1 point)

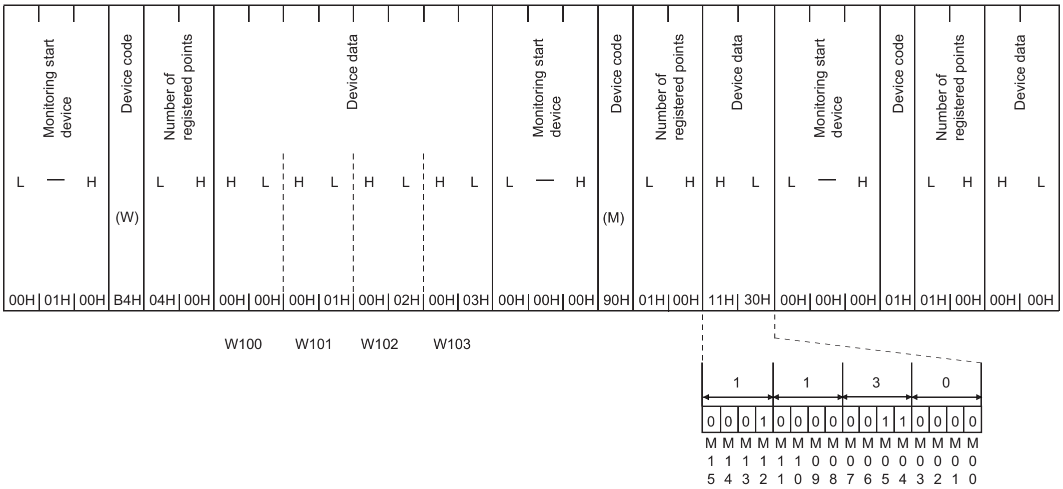
- (When ASCII-BIN conversion is not performed)

  The total number of bytes for the device data portion is the number of device points × 2.

When the word/byte units designation is byte


Monitoring start device
Device code
Number of registered points
Device data
Monitoring start device
Device code
Number of registered points
Device data
Monitoring start device
Device code
Number of registered points
Device data
L — H
(W)
L H
L H
L H
L H
L H
L — H
(M)
L H
L H
L — H
L H
L H
00H 01H 00H B4H 04H 00H 00H 00H 01H 00H 02H 00H 03H 00H 00H 00H 00H 90H 01H 00H 30H 11H 00H 00H 00H 01H 01H 00H 00H 00H
W100
W101
W102
W103
3 0 1 1
0 0 1 1 0 0 0 0 0 0 0 1 0 0 0 1
M7 M6 M5 M4 M3 M2 M1 M0 M15 M14 M13 M12 M11 M10 M9 M8

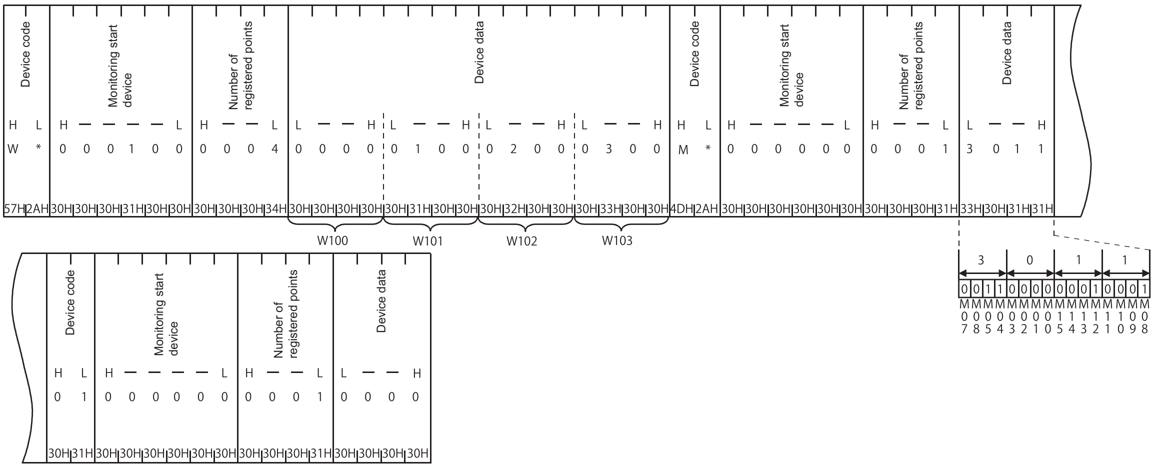
When the word/byte units designation is word


Monitoring start device
Device code
Number of registered points
Device data
Monitoring start device
Device code
Number of registered points
Device data
Monitoring start device
Device code
Number of registered points
Device data
L — H
(W)
L H
H L
H L
H L
H L
L — H
(M)
L H
H L
L — H
L H
H L
00H 01H 00H B4H 04H 00H 00H 00H 00H 01H 00H 02H 00H 03H 00H 00H 00H 90H 01H 00H 11H 30H 00H 00H 00H 01H 01H 00H 00H 00H
W100
W101
W102
W103
1 1 3 0
0 0 0 1 0 0 0 0 0 0 1 1 0 0 0 0
M15 M14 M13 M12 M11 M10 M9 M8 M7 M6 M5 M4 M3 M2 M1 M0

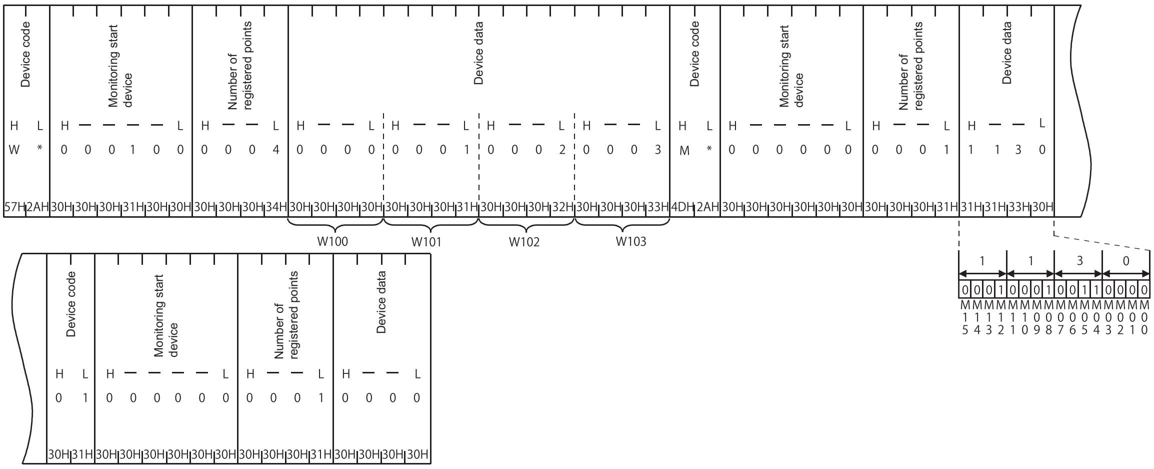
- (When ASCII-BIN conversion is performed)

  The total number of bytes for the device data portion is the number of device points × 4.

When the word/byte units designation is byte


Device code
Monitoring start device
Number of registered points
Device data
Device code
Monitoring start device
Number of registered points
Device data
H L H — — — — L H — — L L — — H L — — H L — — H L — — H H L H — — — — L H — — L L — — H
W * 0 0 0 1 0 0 0 0 0 4 0 0 0 0 0 1 0 0 0 2 0 0 0 3 0 0 M * 0 0 0 0 0 0 0 0 0 1 3 0 1 1
57H 2AH 30H 30H 30H 31H 30H 30H 30H 30H 30H 34H 30H 30H 30H 30H 30H 31H 30H 30H 30H 32H 30H 30H 30H 33H 30H 30H 4DH 2AH 30H 30H 30H 30H 30H 30H 30H 30H 30H 31H 33H 30H 31H 31H
W100
W101
W102
W103
3 0 1 1
0 0 1 1 0 0 0 0 0 0 0 1 0 0 0 1
M7 M6 M5 M4 M3 M2 M1 M0 M15 M14 M13 M12 M11 M10 M9 M8
Device code
Monitoring start device
Number of registered points
Device data
H L H — — — — L H — — L L — — H
0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0
30H 31H 30H 30H 30H 30H 30H 30H 30H 30H 30H 31H 30H 30H 30H 30H

When the word/byte units designation is word


Device code
Monitoring start device
Number of registered points
Device data
Device code
Monitoring start device
Number of registered points
Device data
H L H — — — — L H — — L H — — L H — — L H — — L H — — L H L H — — — — L H — — L H — — L
W * 0 0 0 1 0 0 0 0 0 4 0 0 0 0 0 0 0 1 0 0 0 2 0 0 0 3 M * 0 0 0 0 0 0 0 0 0 1 1 1 3 0
57H 2AH 30H 30H 30H 31H 30H 30H 30H 30H 30H 34H 30H 30H 30H 30H 30H 30H 30H 31H 30H 30H 30H 32H 30H 30H 30H 33H 4DH 2AH 30H 30H 30H 30H 30H 30H 30H 30H 30H 31H 31H 31H 33H 30H
W100
W101
W102
W103
1 1 3 0
0 0 0 1 0 0 0 1 0 0 1 1 0 0 0 0
M15 M14 M13 M12 M11 M10 M9 M8 M7 M6 M5 M4 M3 M2 M1 M0
Device code
Monitoring start device
Number of registered points
Device data
H L H — — — — L H — — L H — — L
0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0
30H 31H 30H 30H 30H 30H 30H 30H 30H 30H 30H 31H 30H 30H 30H 30H

When user frame No. B082H is designated

Information on the monitoring results for the condition matched blocks are transmitted for each block.

Results are transmitted in the order of the programmable controller CPU status information, the device information registered in the word block and then the device information registered in the bit block.

Ex.

Number of registered word blocks: 2 (D0 to D3 (4 points), W100 to W103 (4 points))

Number of registered bit blocks : 1 (M0 to M15 (1 point))

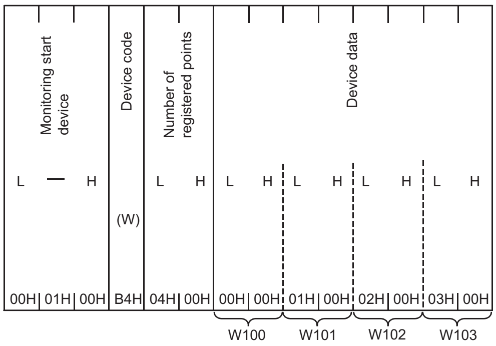
When the condition satisfied monitoring device is W100 = 0 and M0≠ON

- (When ASCII-BIN conversion is not performed)

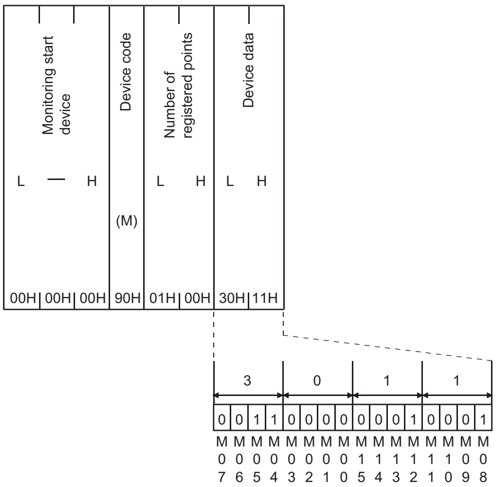
  The total number of bytes for the device data portion is the number of device points × 2.

When the word/byte units designation is byte

For W100 to W103 block data



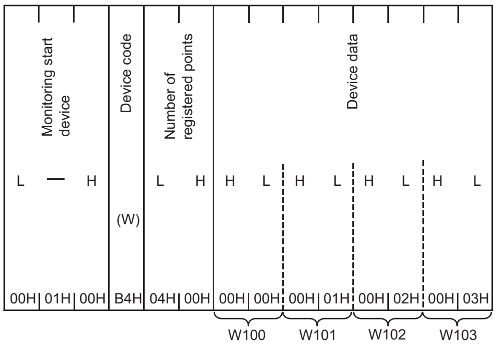
For M0 to M15 block data



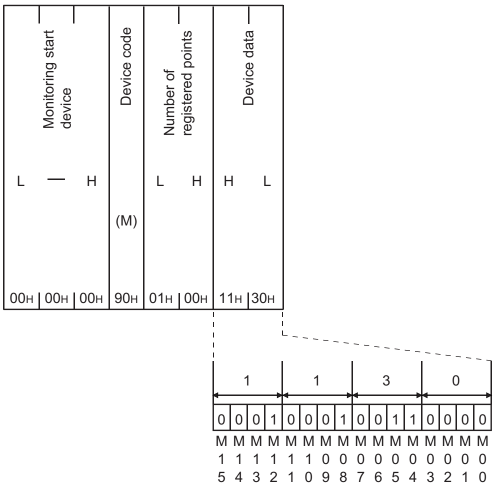
Monitoring start device
Device code
Number of registered points
Device data
L — H
(M)
L H
L H
00H 00H 00H 90H 01H 00H 30H 11H
3
0
1
1
0 0 1 1 0 0 0 0 0 0 0 1 0 0 0 1
M7 M6 M5 M4 M3 M2 M1 M0 M15 M14 M13 M12 M11 M10 M9 M8

When the word/byte units designation is word

For W100 to W103 block data



For M0 to M15 block data

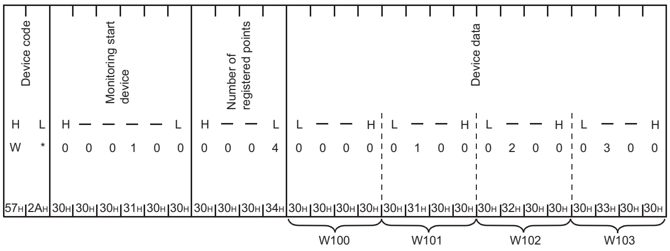


Monitoring start device
Device code
Number of registered points
Device data
L — H
(M)
L H
H L
00H 00H 00H 90H 01H 00H 11H 30H
1
1
3
0
0 0 0 1 0 0 0 1 0 0 1 1 0 0 0 0
M15 M14 M13 M12 M11 M10 M9 M8 M7 M6 M5 M4 M3 M2 M1 M0

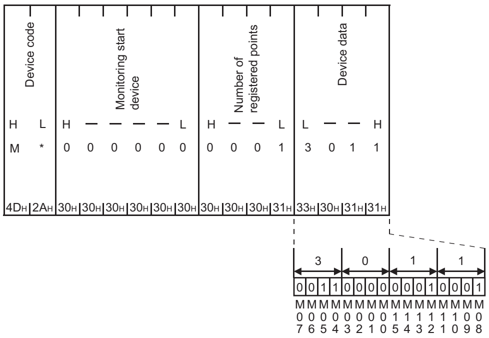
- (When ASCII-BIN conversion is performed)
  The total number of bytes for the device data portion is the number of device points × 4.

When the word/byte units designation is byte

For W100 to W103 block data

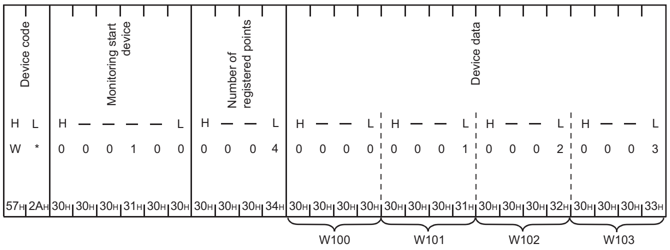

Device code
Monitoring start device
Number of registered points
Device data
H L H — — — — — L H — — L L — — H L — — H L — — H L — — H
W * 0 0 0 1 0 0 0 0 0 4 0 0 0 0 0 1 0 0 0 2 0 0 0 3 0 0
57H 2AH 30H 30H 30H 31H 30H 30H 30H 30H 30H 34H 30H 30H 30H 30H 30H 31H 30H 30H 30H 32H 30H 30H 30H 33H 30H 30H
W100
W101
W102
W103

For M0 to M15 block data

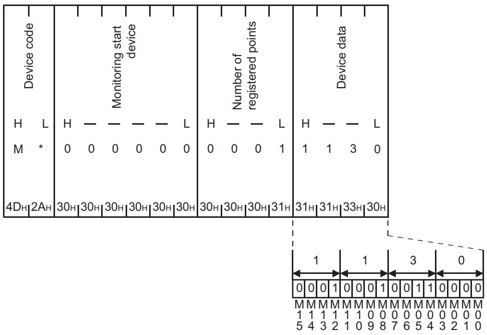

Device code
Monitoring start device
Number of registered points
Device data
H L H — — — — — L H — — L L — — H
M * 0 0 0 0 0 0 0 0 0 1 3 0 1 1
4DH 2AH 30H 30H 30H 30H 30H 30H 30H 30H 30H 31H 33H 30H 31H 31H
3 0 1 1
0 0 1 1 0 0 0 0 0 0 0 1 0 0 0 1
M7 M6 M5 M4 M3 M2 M1 M0 M15 M14 M13 M12 M11 M10 M9 M8

When the word/byte units designation is word

For W100 to W103 block data


Device code
Monitoring start device
Number of registered points
Device data
H L H — — — — — L H — — L H — — L H — — L H — — L H — — L
W * 0 0 0 1 0 0 0 0 0 4 0 0 0 0 0 0 0 1 0 0 0 2 0 0 0 3
57H 2AH 30H 30H 30H 31H 30H 30H 30H 30H 30H 34H 30H 30H 30H 30H 30H 30H 30H 31H 30H 30H 30H 32H 30H 30H 30H 33H
W100
W101
W102
W103

For M0 to M15 block data


Device code
Monitoring start device
Number of registered points
Device data
H L H — — — — — L H — — L H — — L
M * 0 0 0 0 0 0 0 0 0 1 1 1 3 0
4DH 2AH 30H 30H 30H 30H 30H 30H 30H 30H 30H 31H 31H 31H 33H 30H
1 1 3 0
0 0 0 1 0 0 0 1 0 0 1 1 0 0 0 0
M15 M14 M13 M12 M11 M10 M9 M8 M7 M6 M5 M4 M3 M2 M1 M0

Execution order for using the programmable controller CPU monitoring function

This section explains the execution order to use the programmable controller CPU monitoring function.

- When transmitting the monitoring results with the modem function, configure the following settings in order to use the modem function.

Setting item	Reference
Initial setting using the Engineering tool	Page 206 Initial settings of Serial communication module
Registration of data number for initialization and data number for connection	Page 209 Register/read/delete data for initialization Page 213 Register/read/delete data for connection
Initialization of C24 modem/TA	Page 217 Initialization of modem/TA (terminal adapter)

- Register programmable controller CPU monitoring for C24 using one of the methods shown below.
  Registration with Engineering tool
  Registration using the programmable controller CPU monitoring registration command (0630) for communication with the MC protocol (📖 MELSEC Communication Protocol Reference Manual)
  Registration by the dedicated instruction for the CPU module (CSET) (📖MELSEC iQ-R Programming Manual (Module Dedicated Instructions))
- By registering programmable controller CPU monitoring, C24 monitors the CPU module on the host station regardless of the RUN/STOP status and sends the monitoring information to the target device.
- When reregistering programmable controller CPU monitoring in order to change the registration data for the programmable controller CPU monitoring, reregister after canceling the programmable controller CPU monitoring.
  When registering with communication using the MC protocol (📖 MELSEC Communication Protocol Reference Manual)
  When registering by "CSET" instruction for the CPU module (📖MELSEC iQ-R Programming Manual (Module Dedicated Instructions))
  To cancel by using Engineering tool, set the CPU module in the STOP status first to configure the settings again, and then restart the CPU module.

8.3 Settings for Using the Programmable Controller CPU Monitoring Function

This section describes system settings required for fixed cycle send and condition match send.

Parameter setting items for the programmable controller CPU monitoring function

The following explains parameter setting items for the programmable controller CPU monitoring function.

Configure the parameter settings.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ Module Name ⇨ [Module Parameter] ⇨ "Application Settings" ⇨ "Programmable Controller CPU monitoring function setting"

Point

- Register send user frames by Engineering tool, after checking the specifications and setting methods explained in the following sections. (☞ Page 282 CONTENTS AND REGISTRATION OF USER FRAMES FOR DATA COMMUNICATION, Page 308 DATA COMMUNICATIONS USING USER FRAMES)
- Register connection data for the modem function by Engineering tool, after checking the specifications in the following section. (☞ Page 214 Method to register/read/delete data for connection)

Setting items and requirement for communication using the MC protocol

○: Setting required, ×: Setting not required

Setting item			Fixed cycle send	Condition match send	Reference
Programmable controller CPU monitoring function designation			○	○	Page 171 Programmable controller CPU monitoring function designation
Cycle time units designation			○	○	Page 171 Cycle time units designation
Cycle time designation (Programmable controller CPU monitoring interval time)			○	○	Page 171 Cycle time designation (Programmable controller CPU monitoring interval time)
Send pointer designation			×	×	Page 173 Setting items for transmitting monitoring results
Output quantity designation					
Block monitoring device	Registered word block quantity designation		○	○	Page 172 Registered word block quantity designation, Registered bit block quantity designation
	Registered bit block quantity designation				
	CPU error monitoring designation				Page 172 CPU error monitoring designation
	n-th block monitoring device[*1]	Monitoring device designation			Page 172 Monitoring device designation, start device number designation, number of read points designation
		Start device No. specification			
		Number of read points specification			
		Monitoring condition value designation (judgment condition designation)	×		Page 172 Monitoring condition specification (Judgment condition specification)
		Monitoring condition specification			Page 172 Monitoring condition value designation
		Send pointer designation		×	Page 173 Setting items for transmitting monitoring results
		Output quantity designation			
	CPU error monitoring designation	Send pointer designation			Page 173 Setting items for transmitting monitoring results
		Output quantity designation			

[*1] n=1 to 10

Setting items and requirement for communication using the nonprocedural protocol

○: Setting required, ×: Setting not required

Setting item			Fixed cycle send	Condition match send	Reference
Programmable controller CPU monitoring function designation			○	○	Page 171 Programmable controller CPU monitoring function designation
Cycle time units designation			○	○	Page 171 Cycle time units designation
Cycle time designation (programmable controller CPU monitoring interval time)			○	○	Page 171 Cycle time designation (Programmable controller CPU monitoring interval time)
Send pointer designation			○	×	Page 173 Setting items for transmitting monitoring results
Output quantity designation					
Block monitoring device	Registered word block quantity designation		○	○	Page 172 Registered word block quantity designation, Registered bit block quantity designation
	Registered bit block quantity designation				
	CPU error monitoring designation				Page 172 CPU error monitoring designation
	n-th block monitoring device[*1]	Monitoring device designation			Page 172 Monitoring device designation, start device number designation, number of read points designation
		Start device No. specification			
		Number of read points specification			
		Monitoring condition value designation (judgment condition designation)	×		Page 172 Monitoring condition specification (Judgment condition specification)
		Monitoring condition specification			Page 172 Monitoring condition value designation
		Send pointer designation			Page 173 Setting items for transmitting monitoring results
		Output quantity designation			
	CPU error monitoring designation	Send pointer designation			Page 173 Setting items for transmitting monitoring results
		Output quantity designation			

[*1] n=1 to 10

Contents of setting items

The following shows the data setting items with Engineering tool and the setting contents to use the programmable controller CPU monitoring function.

■Programmable controller CPU monitoring function designation

Designate the timing (fixed cycle send or condition match send) when transmitting information on the programmable controller CPU monitoring results (device information/CPU status information) to the target device.

- The conditions for condition match send are designated using data items in "Monitoring condition specification (Judgment condition specification)" and "Monitoring condition value designation".

■Cycle time units designation

- Designate the unit for cycle time for reading information from the CPU module using the programmable controller CPU monitoring function.
- The time designated by "Cycle time units designation" and "Cycle time designation (programmable controller CPU monitoring interval time)" is also used as the transmission time interval for fixed cycle send.

■Cycle time designation (Programmable controller CPU monitoring interval time)

Designate the time for one cycle when reading information from the CPU module in order to perform programmable controller CPU monitoring function.

- The time designated for "Cycle time designation (Programmable controller CPU monitoring interval time)" is also used as the transmission time interval for fixed cycle send.

■Registered word block quantity designation, Registered bit block quantity designation

Designate the number of word device blocks (number of registered word blocks) and the number of bit device blocks (number of registered bit blocks) registered in C24 as the target when performing device data monitoring or transmission.

■CPU error monitoring designation

Designate whether or not C24 monitors abnormality of the CPU module on the host station (status monitoring) in the programmable controller CPU monitoring.

■Monitoring device designation, start device number designation, number of read points designation

When performing device data monitoring or transmission, designate the device range for each block for the number of blocks designated by "Registered word block quantity designation" and "Registered bit block quantity designation".

The target of device data monitoring for condition match send is the start device for each block.

Word device designated block: Head word device (for one word)

Bit device designated block : Head bit device (for one bit)

- The monitored device is the item that indicates the target device of the corresponding block and designated with the codes below. (☞ Page 154 List of device code)
- The start device is the data that designates the head of the target device range for the corresponding block.
- The number of read points (number of registered point) is the item that indicates the target device range for the corresponding block which designate points from the start device number.
  The bit device designates points in word units (1 point = 16 bits).
- The methods for designating these data are the same as the designation methods used when reading from or writing to the device memory, which are described in the following manual.
  (📖 MELSEC Communication Protocol Reference Manual)

Point

When the user performs the programmable controller CPU monitoring registration, the device No. will designated in either a decimal or hexadecimal depending on devices. The number of read points (number of registered points) are designated in hexadecimal.

However, for data communication using either MC protocol (format 1 to 4) or nonprocedural protocol, when ASCII-BIN conversion of the communication data has been designated, the start device number for all devices to be transmitted to target devices as the monitoring results will be converted to hexadecimal ASCII data and transmitted.

■Monitoring condition specification (Judgment condition specification)

When the condition match send is designated by the "Programmable controller CPU monitoring function designation", designate the conditions for transmitting information for the "Monitoring condition value designation".

■Monitoring condition value designation

When the condition match send is designated by the "Programmable controller CPU monitoring function designation", this data designates the status/numeric value for "Monitoring condition specification (Judgment condition specification)".

- When the monitoring device is a word device: Designate the monitoring condition value with a numeric value
- When the monitoring device is a bit device : Designate the monitoring condition with a numeric value (1/0) corresponding to ON/OFF.

Setting items for transmitting monitoring results

The following shows an example of setting items and data transmission when sending the monitored results of the programmable controller CPU monitoring function to the target device using the nonprocedural protocol.

Ex.

This example shows when the D0 to D3 device information and user frame data are transmitted by the edge triggered transmission method with a condition of D0 = 0.

The setting is configured in the parameters of Engineering tool.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ Module Name ⇨ [Module Parameter] ⇨ "Application Settings" ⇨ "Programmable Controller CPU monitoring function setting"

Setting item			Setting content	Remarks
Programmable controller CPU monitoring function designation			Condition match send	Settings other than those listed at left are not required
Cycle time units designation			min	
Cycle time designation (Programmable controller CPU monitoring interval time)			3	
Block monitoring device	Registered word block quantity designation		1	
	Registered bit block quantity designation		0	
	CPU error monitoring designation		Do not monitor	
	1st block monitoring device	Monitoring device designation	D	
		Start device number designation	0	
		Number of read points specification	4	
		Monitoring condition specification (Judgment condition specification)	=(Edge)	
		Monitoring condition value designation	0	
		Send pointer designation	49	
		Output quantity designation	3	

- Setting the send user frames

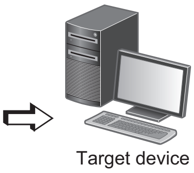
Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ Module Name ⇨ [Module Parameter] ⇨ "Application Settings" ⇨ "User frame specification"

Transmission frame No. designation 46th	User frame number	
Transmission frame No. designation 47th	User frame number	
Transmission frame No. designation 48th	User frame number	
Transmission frame No. designation 49th	User frame number (02H)	← Set the user frame data (STX)
Transmission frame No. designation 50th	User frame number (B001H)	← Set the first block (from D0 to D3)
Transmission frame No. designation 51th	User frame number (03H)	← Set the user frame data (ETX)
Transmission frame No. designation 52th	User frame number	
Transmission frame No. designation 53th	User frame number	
Transmission frame No. designation 54th	User frame number	
Transmission frame No. designation 55th	User frame number	
Transmission frame No. designation 56th	User frame number	
Transmission frame No. designation 57th	User frame number	

Data sent when the condition D0 = 0 is satisfied

ETX	Device data (D3)	Device data (D2)	Device data (D1)	Device data (D0)	STX

⇨



How to register and cancel the programmable controller CPU monitoring function

The following shows the method to register/cancel the programmable controller CPU monitoring function from the CPU module.

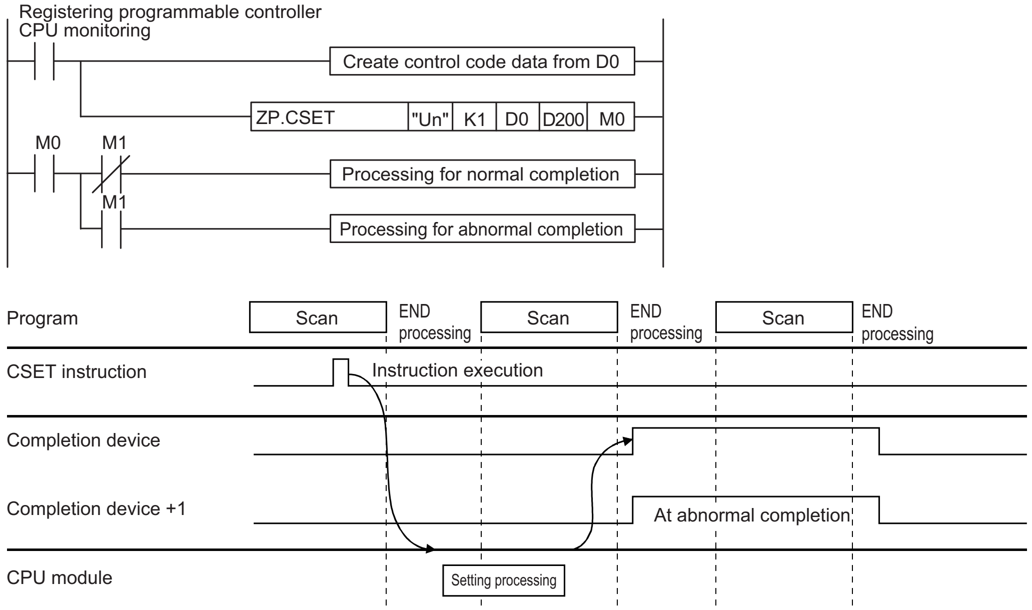
Point

- For details on the method for registering or canceling the programmable controller CPU monitoring with a command of MC protocol, refer to the following manual.
  (📖 MELSEC Communication Protocol Reference Manual)

When registering or canceling from the CPU module

For details on the CSET command, refer to the following section.

☞ Page 580 Dedicated Instructions

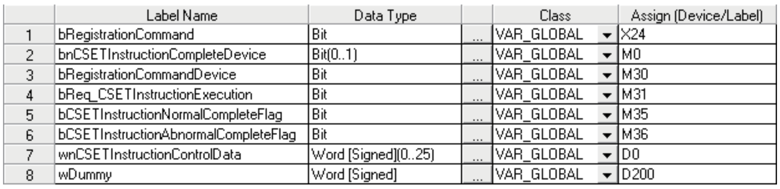


- Store the data for programmable controller CPU monitoring registration in the device that designates the control data for the CSET instruction.
- Execute the CSET instruction.
  At the end of the scan in which the CSET instruction was completed, the completion device (M0) designated by (D2) turns ON and then turns OFF at the next END processing.
- When an error occurs, (D2) + 1 turns ON and the error code is stored in the completion status (S2) + 1.

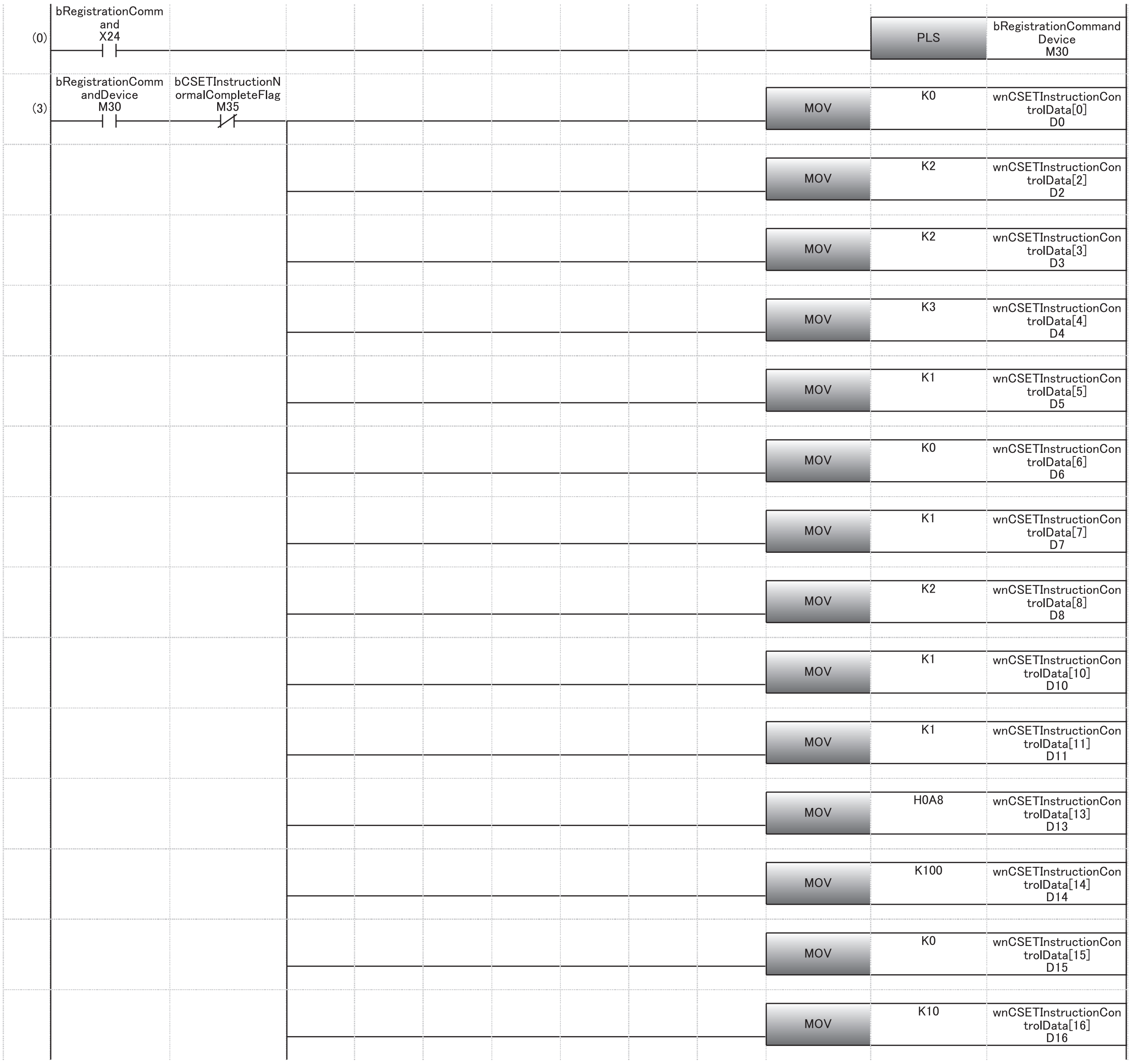
■Program example for performing programmable controller CPU monitoring registration

This example shows a program that registers programmable controller CPU monitoring for the CH1 side interface.

This registration is for transmitting the contents of M0 to M15 and D100 to D109 to the target device using the fixed cycle send (cycle time is 3 min).

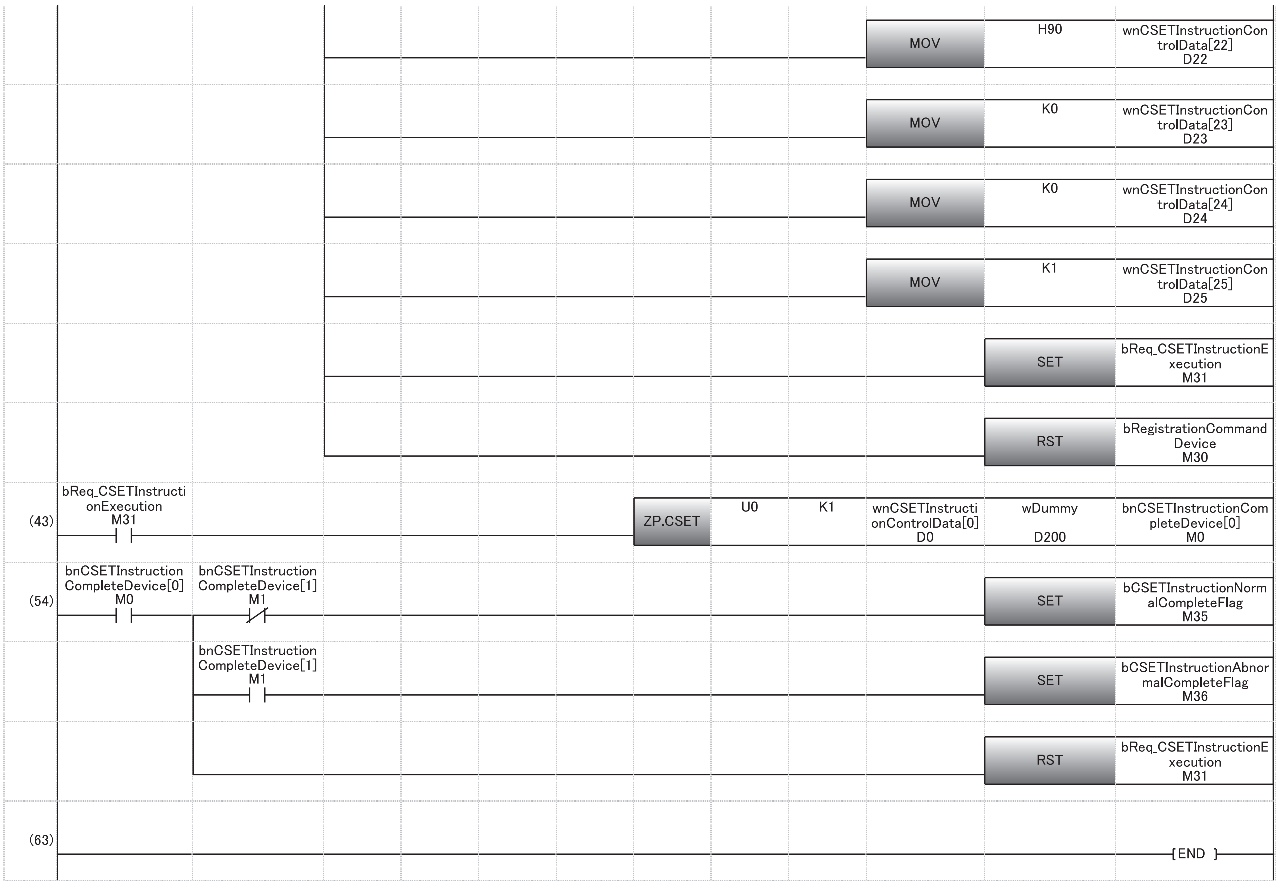
Category	Setting content
Label to be defined	Define the global label as shown below. 

	Label Name	Data Type		Class	Assign (Device/Label)
1	bRegistrationCommand	Bit	...	VAR_GLOBAL	X24
2	bnCSETInstructionCompleteDevice	Bit(0..1)	...	VAR_GLOBAL	M0
3	bRegistrationCommandDevice	Bit	...	VAR_GLOBAL	M30
4	bReq_CSETInstructionExecution	Bit	...	VAR_GLOBAL	M31
5	bCSETInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M35
6	bCSETInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M36
7	wnCSETInstructionControlData	Word [Signed](0..25)	...	VAR_GLOBAL	D0
8	wDummy	Word [Signed]	...	VAR_GLOBAL	D200



```
(0) bRegistrationCommand X24 ─┤ ├─ [PLS bRegistrationCommandDevice M30]

(3) bRegistrationCommandDevice M30 ─┤ ├─ bCSETInstructionNormalCompleteFlag M35 ─┤/├─
        [MOV K0    wnCSETInstructionControlData[0]  D0 ]
        [MOV K2    wnCSETInstructionControlData[2]  D2 ]
        [MOV K2    wnCSETInstructionControlData[3]  D3 ]
        [MOV K3    wnCSETInstructionControlData[4]  D4 ]
        [MOV K1    wnCSETInstructionControlData[5]  D5 ]
        [MOV K0    wnCSETInstructionControlData[6]  D6 ]
        [MOV K1    wnCSETInstructionControlData[7]  D7 ]
        [MOV K2    wnCSETInstructionControlData[8]  D8 ]
        [MOV K1    wnCSETInstructionControlData[10] D10]
        [MOV K1    wnCSETInstructionControlData[11] D11]
        [MOV H0A8  wnCSETInstructionControlData[13] D13]
        [MOV K100  wnCSETInstructionControlData[14] D14]
        [MOV K0    wnCSETInstructionControlData[15] D15]
        [MOV K10   wnCSETInstructionControlData[16] D16]
```


MOV
H90
wnCSETInstructionControlData[22]
D22
MOV
K0
wnCSETInstructionControlData[23]
D23
MOV
K0
wnCSETInstructionControlData[24]
D24
MOV
K1
wnCSETInstructionControlData[25]
D25
SET
bReq_CSETInstructionExecution
M31
RST
bRegistrationCommandDevice
M30
(43)
bReq_CSETInstructionExecution
M31
ZP.CSET
U0
K1
wnCSETInstructionControlData[0]
D0
wDummy
D200
bnCSETInstructionCompleteDevice[0]
M0
(54)
bnCSETInstructionCompleteDevice[0]
M0
bnCSETInstructionCompleteDevice[1]
M1
SET
bCSETInstructionNormalCompleteFlag
M35
bnCSETInstructionCompleteDevice[1]
M1
SET
bCSETInstructionAbnormalCompleteFlag
M36
RST
bReq_CSETInstructionExecution
M31
(63)
END

(0) Convert the registration command into pulse.

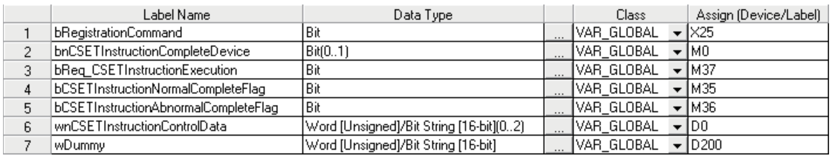
(3) Set the execution type.
Set the execution type (Programmable controller monitoring registration).
Set the unit of the cycle time to 'Minute'.
Set the cycle time to three minutes.
Set the monitoring function to fixed cycle send.
Set the transmission measure to data transmission.
Set the output start pointer.
Set the number of the user frame to be transmitted.
Set the number of registered word blocks designation to '1'.
Set the number of registered bit blocks designation to '1'.
Register the device of D100 to D109 as the first block.
Register the device of M0 to M15 as the second block.

(43) Execute the programmable controller CPU monitoring registration.

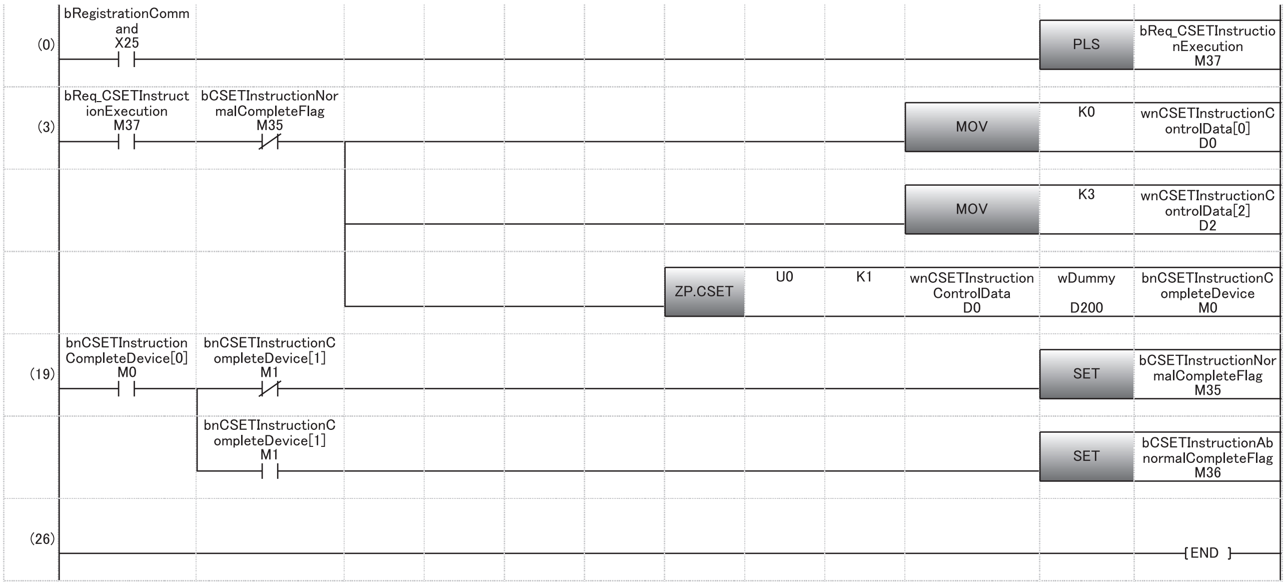
(54) Processing for normal completion or abnormal completion

■Program example for performing programmable controller CPU monitoring cancellation

This example shows a program that cancels programmable controller CPU monitoring for the CH1 side interface

Category	Setting content
Label to be defined	Define the global label as shown below. 

	Label Name	Data Type		Class	Assign (Device/Label)
1	bRegistrationCommand	Bit	...	VAR_GLOBAL	X25
2	bnCSETInstructionCompleteDevice	Bit(0..1)	...	VAR_GLOBAL	M0
3	bReq_CSETInstructionExecution	Bit	...	VAR_GLOBAL	M37
4	bCSETInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M35
5	bCSETInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M36
6	wnCSETInstructionControlData	Word [Unsigned]/Bit String [16-bit](0..2)	...	VAR_GLOBAL	D0
7	wDummy	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D200


(0)
bRegistrationCommand
X25
PLS
bReq_CSETInstructionExecution
M37
(3)
bReq_CSETInstructionExecution
M37
bCSETInstructionNormalCompleteFlag
M35
MOV
K0
wnCSETInstructionControlData[0]
D0
MOV
K3
wnCSETInstructionControlData[2]
D2
ZP.CSET
U0
K1
wnCSETInstructionControlData
D0
wDummy
D200
bnCSETInstructionCompleteDevice
M0
(19)
bnCSETInstructionCompleteDevice[0]
M0
bnCSETInstructionCompleteDevice[1]
M1
SET
bCSETInstructionNormalCompleteFlag
M35
bnCSETInstructionCompleteDevice[1]
M1
SET
bCSETInstructionAbnormalCompleteFlag
M36
(26)
END

(0) Set the command pulse.

(3) Execute the programmable controller CPU monitoring registration.

(19) Processing for normal completion or abnormal completion

8.4 Considerations for Using the Programmable Controller CPU Monitoring Function

Influence on the cycle time

The cycle time will be affected by the following factors.

Keep these in mind when setting the cycle time.

- When the programmable controller CPU is accessed by a module other than C24.
- When a data communication function other than the programmable controller CPU monitoring function is used.
- When transmission interruption occurs by DTR/DSR control.

Mixing of fixed cycle send and condition match send

Both of the fixed cycle send and the condition match send cannot be designated together for the same interface.

Target CPU module

Only the CPU module on the host station can be the target for the programmable controller CPU monitoring function.

Registration during operation

A new programmable controller CPU monitoring registration cannot be performed while the programmable controller CPU monitoring function is in operation.

- Perform the new programmable controller CPU monitoring registration after canceling the programmable controller CPU monitoring.
- If the programmable controller CPU monitoring registration is performed without canceling the monitoring, an error will occurs.

  Before performing the programmable controller CPU monitoring registration with Engineering tool, set the CPU module in the STOP state, and then restart the CPU module.

Error during operation

While the programmable controller CPU monitoring function is in operation, even if an error occurs in transmission of the CPU module monitoring results or reading of data from the CPU module, the programmable controller CPU monitoring function operation will not stop.

System configuration

The programmable controller CPU monitoring function can be used when the system configuration is in a 1:1 basis.

Operation when information cannot be transmitted to target device

The following describes how C24 operates when the information of the programmable controller CPU monitoring results cannot be sent to the target device due to line disconnection or other reasons.

Even if an error occurs while the programmable controller CPU monitoring function is in operation, the ERR LED does not turn ON. (This is the same as when using the on-demand function of the MC protocol.)

■When the setting for the send monitoring time designation (timer 2) is 'Unlimited wait' (0H)

- Reading of monitoring data from the CPU module stops until the transmission of the monitoring data is completed.
- When transmission resumes, reading of monitoring data from the CPU module resumes and monitoring data and information are transmitted.

■When the setting for the send monitoring time designation (timer 2) is other than 'Unlimited wait' (0H)

- A transmission timeout error occurs. Reading of monitoring information from the CPU module and transmission of the information resumes.
- The error code is stored in the 'programmable controller CPU monitoring function execution result (current)' (Un\G8709/8965).

When CPU module error occurs

When device data for the programmable controller CPU cannot be read because of a CPU module error (hardware failure, etc.), the error code is stored in the programmable controller CPU monitoring function error code storage area and C24 performs the monitoring processing based on previously read data.

Modem connection error

When the monitoring information is transmitted with the modem function, a modem connection error will occur if a modem connection request is issued for the following reason.

- Connection request by the 'connection request' (Y11)

If possible, provide a dedicated C24 for using the programmable controller CPU monitoring function.

When using both the programmable controller CPU monitoring function and data communication function with a single C24 and one of the above modem connection errors occurs, re-execute a connection request in consideration of the transmission timing set by the user for the programmable controller CPU monitoring function.

9 COMMUNICATIONS BY THE MODEM FUNCTION

This chapter explains the overview and how to use the modem function, which can be used for data communication with remote target devices.

For the transmission specification on C24 side, connectable modems/TA's (terminal adapter) for the usage of the modem function, refer to the following manual.

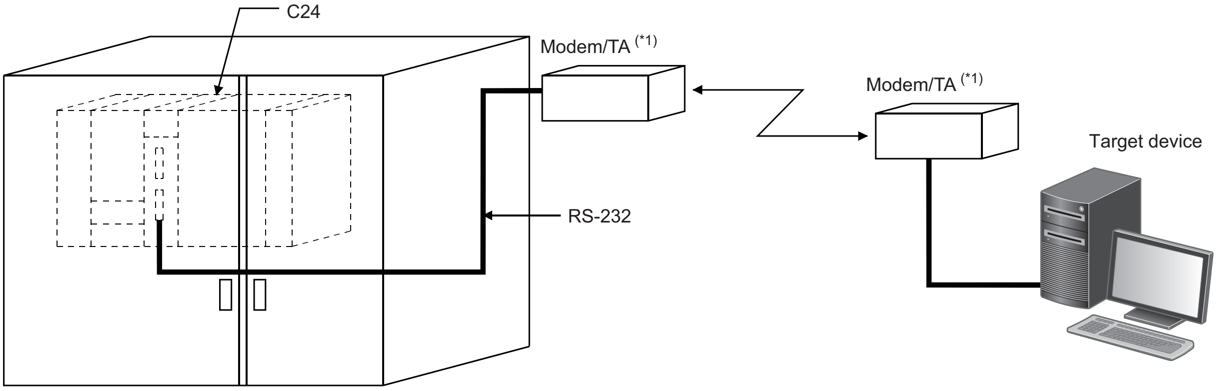
📖MELSEC iQ-R Serial Communication Module User's Manual(Startup)

9.1 Overview

The modem function easily performs data transmission/reception to remote devices via public lines/office telephone systems/digital lines (ISDN) by connecting a modem or TA (terminal adapter) to RS-232 interface of C24.

Initialization of the modem or TA, line connection (dialing), and modem disconnection are performed by CPU module.

Once the line is connected, data communication with the target device via public line/office telephone system/digital line can be performed.



*1 TA indicates the abbreviation for the terminal adapter.

Point

The modem function does not support dedicated lines.

When transmitting/receiving data via a dedicated line, prepare a modem having an automatic connection function and supporting dedicated lines.

This chapter can be skipped because data can be transmitted/received without using the modem function when transmitting/receiving data via a modem having an automatic connection function and supporting dedicated lines.

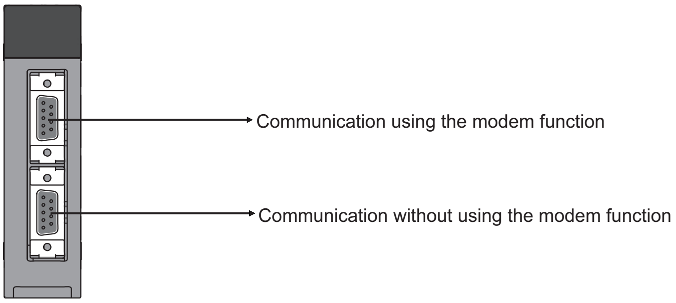
Automatic connection function: Indicates a function for connecting to the modem of the target device automatically. (The function name varies depending on the manufacturer of the modem.)

Features

The following explains the features of the modem function.

Interface that can use the modem function

- The modem function can be used with an RS-232 interface of C24.
- For the RJ71C24-R2, the modem function can be used for only one of the two RS-232 interfaces. With the other interface of C24, direct data communication with a target device can be performed using MC protocol, nonprocedural protocol, or bidirectional protocol (independent operation).


Communication using the modem function
Communication without using the modem function

Initialization/connection/disconnection of modem or TA

- The following setting value for line connection can be used by registering in the module extended parameter or flash ROM of C24 in multiple sets.
  Data for Modem/TA initialization (AT command)
  User setting: 30 sets (78 bytes/set), default value: 13 sets
  Data for connection
  User setting: 30 sets (80 bytes/set) (display message to telephone number of the connection destination, line type, etc.)
- By registering the above data to C24 in advance, the modem/TA (terminal adapter) initialization, line connection (dialing), and line disconnection can be performed easily.
- When the no-communication interval time (1 min to 120 min) is set, C24 disconnects the line when a no-communication condition continues for the set period of time following the line connection.

Communication between remote device and CPU module

- Data communication can be performed via full-duplex communication.
- From the target device to the CPU module, communication using the MC protocol, nonprocedural protocol, or bidirectional protocol can be performed.
- From the CPU module to the target device, communication using the MC protocol (transmission by the on-demand function only), the nonprocedural protocol, or bidirectional protocol can be performed.

Remote password check

If the remote password check has been set for C24, the C24 performs a remote password check when accessing to the programmable controller from a target device using the C24 modem function.

The following is an overview of the CPU module remote password function.

☞ Page 187 Applicability of remote password function

■Remote password function

The remote password function allows/prohibits access to the CPU module from a target device via the following modules.

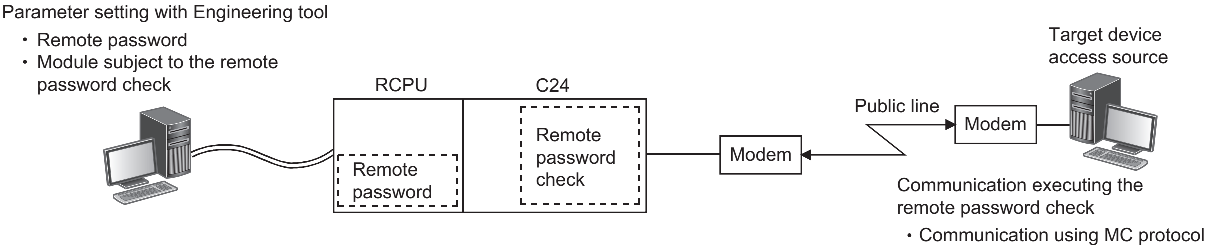
- C24
- Ethernet module
- Built-in Ethernet port CPU

In the case of the Ethernet module, the remote password function can be used for data communications connections with a target device. For details, refer to following manual.

📖MELSEC iQ-R Ethernet User's Manual (Application)

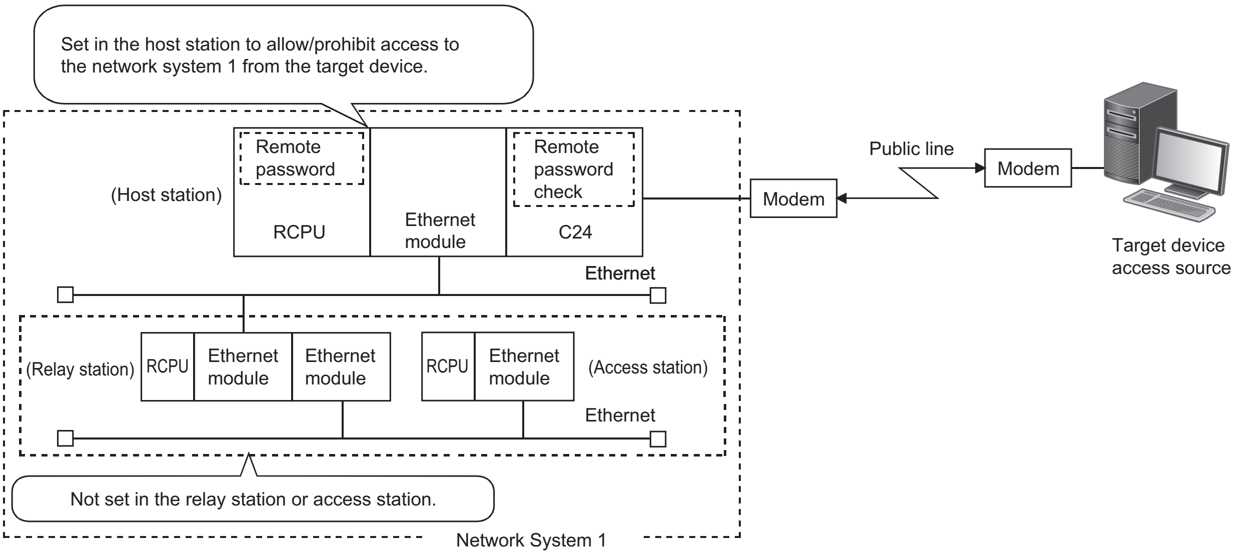
■Station to which the remote password and remote password check are set

- When a programmable controller system with one CPU module station


Parameter setting with Engineering tool
・Remote password
・Module subject to the remote password check
RCPU
C24
Remote password
Remote password check
Modem
Public line
Modem
Target device access source
Communication executing the remote password check
・Communication using MC protocol

- When a programmable controller system consisted of multiple CPU module stations

  Set to the CPU module station (the CPU module on the host station in the figure below) which is the entrance of the programmable controller system as viewed from the target device.



When set to a station other than the CPU module which is the entrance of the programmable controller system (relay station or access station in the above figure), access to other stations beyond the set station is prohibited.

Function list

The modem function overviews are as follows:

Function	Overview	
Modem/TA initialization	Initializes the modem/TA using the user-designated initialization data (AT command). (Auto initialization of the modem/TA is possible.)	
Line connection (dialing)	Dials the target telephone number according to the user-designated data for connection and enables data communication after establishing the line connection. When the modem/TA is not initialized, performs the initialization.	
Data communication	Performs communication with a target device using the MC protocol, nonprocedural protocol, or bidirectional protocol.	Communication method: full-duplex communication Synchronization method: Asynchronous communication (asynchronous)
	Performs communication with the target C24-installed station by modem/TA connection using nonprocedural protocol or bidirectional protocol. (Station-to-station communication)	
Modem disconnection	Disconnects forcefully the line to the connected target device.	
Flash ROM reading, writing (registration), and deletion	Reads, writes (registers) and deletes the initialization data (AT command) and data for connection from/to the flash ROM in C24 according to the request from CPU module.	
Remote password check	Allows C24 to execute the check of the remote password set in the CPU module when communication from a target device to C24 is performed using the MC protocol.	

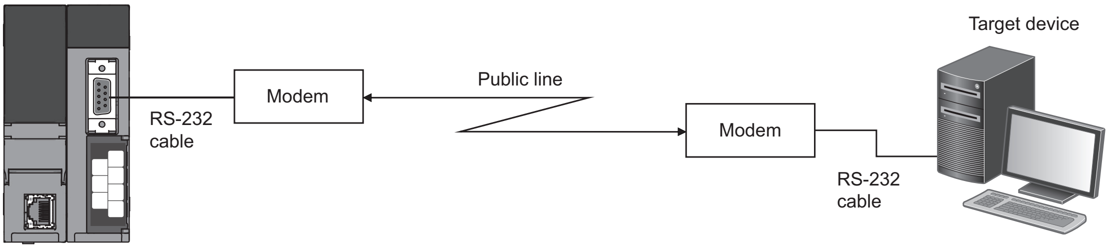
9.2 System Configuration

This section explains the system configurations when the modem function is used to perform data communication with a target device via public lines.

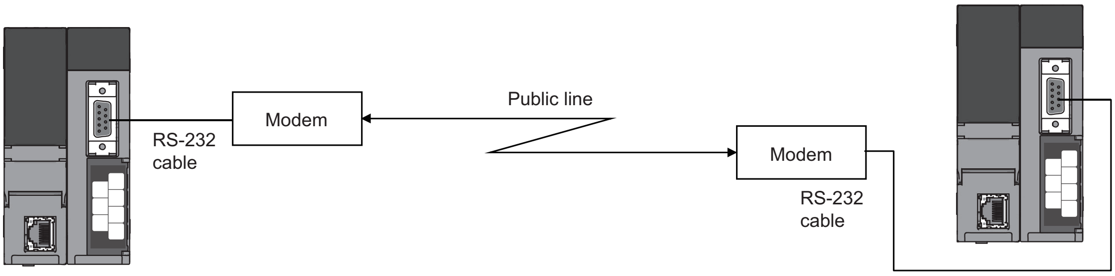
System configuration when performing data communication with target device

The following shows the system configuration examples used for data communications between a target device and CPU module using the MC protocol/nonprocedural protocol/bidirectional protocol of C24.

Connection example to a target device



Connection example to C24



- The public lines indicated above are compatible with the office telephone system as well.
- For connection to a target device or C24, the digital line (ISDN) can be replaced by the public line.
  When connecting via a digital line, a TA (terminal adapter) and a DSU (digital service module) are used instead of a modem.

Considerations for system configurations

The following shows considerations for the system configurations to perform data communication with a target device via public line, an office telephone system or digital line (ISDN) using the modem function.

Usable C24 interface

- The modem function can be used for the RS-232 interface only.
- For the RJ71C24-R2, the modem function can be used for only either one of the two RS-232 interfaces.
- Interactive data communication via two interface of C24 (interlink operation) is not available.

Connectable modem/TA

Only the modems/TA shown below can be used for the C24 RS-232 interface to use the modem function.
📖MELSEC iQ-R Serial Communication Module User's Manual(Startup)

Number of connectable modems/TAs

Only one modem/TA can be connected to the C24 RS-232 interface to use the modem function.

Modem/TA connection cables

- Perform the wiring connection as designated on the modem/TA side for connection between C24 and modem/TA.
- For the RS-232 interface of C24, a D-sub 9-pin connector (female) is used.
  For the connection cable of C24 side, refer to the following manual.
  (📖MELSEC iQ-R Serial Communication Module User's Manual(Startup))

Modem/TA installation

- Install the modem/TA according to the modem/TA manual.
  Installing it in an area where a lot of noises are generated may cause malfunctions.
- In order to prevent the effects of noise and power surges, do not connect near or tie the cable together with a main circuit line, high-voltage line or load line other than for the programmable controller with the modem/TA connection cable.

Connectable lines

- The connections with the following lines, which are equivalent to NTT communication protocol are available.
  Perform connection tests beforehand and confirm that connection is possible.
  Public line or office telephone system of analog two-line method
  Digital line (ISDN)
- It is not possible to connect to call-waiting lines, in order to avoid data errors or automatic line disconnection due to the call-waiting interrupt tone.
- Avoid connections with party-line telephones to avoid interrupted calls during communication.
- If an alert sound is sent at fixed intervals from the communication machine to prevent long-term calls, data corruption may occur.
  Transmitting a response for normal/abnormal data reception between devices, and performing transmission retry processing when any abnormality is detected is recommended.
- Refer to the manuals for a modem/TA for connection from a modem to public line/office telephone system, or for connection from a TA to a digital line.

Communication method

Communication via the modem function is performed using full-duplex communication.

Connections to devices designed for half-duplex communication is not available.

Data communication to target devices

- Data communication with target devices is performed using the public line or electric wave transmitted from the electric wave transmission base.
  In some conditions, normal data communication may not be carried out due to the system installation environment, electric-wave transmission status, error in the target device, or other reasons.
  Perform a connection test beforehand, and confirm that connection is possible.

9.3 Specifications of Modem Function

This section explains the applicability to the remote password function, input/output signals and buffer memory related to the modem function.

Applicability of remote password function

The following explains C24 data communication for the remote password function of the CPU module.

For an overview of the C24 check function for the CPU module remote password, refer to the following section.

☞ Page 182 Remote password check

Precautions

The remote password function cannot prevent improper access completely.

Incorporate the user's own safeguards when it is necessary to protect the security of the programmable controller system from improper access from an external device.

Mitsubishi Electric Corporation cannot be held responsible for any system problems that may occur from unauthorized access.

■Countermeasures for preventing unauthorized access

- Install a firewall.
- Install a personal computer as a relay station and control the relay station of the transmission/receive data using application programs.
- Install an external device whose access authority can be controlled.

For the external devices whose access authority can be controlled, contact the network provider or distributor of devices.

Data communication when remote password is set

This section explains how to use and set the CPU module remote password function, and data communication between the target device and the CPU module when a remote password has been set.

■Permitting/prohibiting access from target device to programmable controller

❶Access permission processing (unlock processing)

- To access to the specified CPU module, the target device performs the remote password unlock processing to the C24[*1] on the directly connected station (host station) after line connection for the modem function.
- If the unlock processing has not been performed, accessing to the specified station is prohibited by the remote password check performed by C24[*1], which received the communication request.
(☞ Page 190 Remote password check processing performed by C24)
- All received data before the unlock processing is performed will be processed as an error.

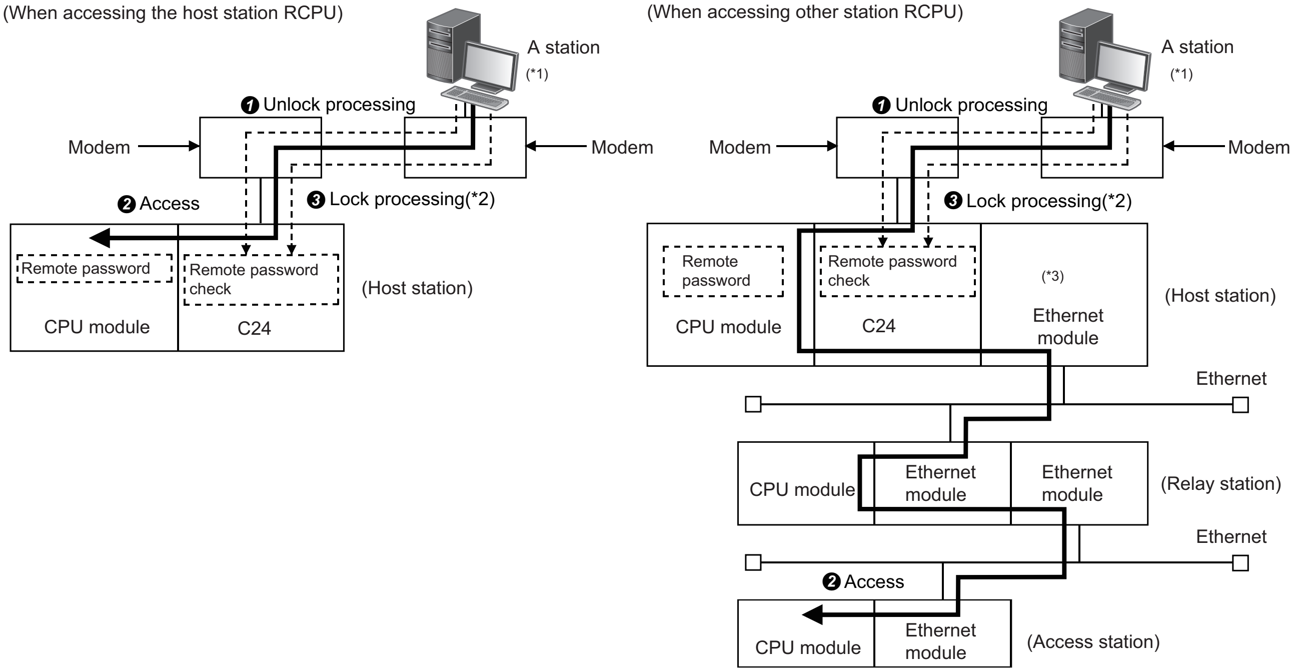
[*1] It indicates the C24 on the CPU module station to which a remote password is set.

❷ Access processing

- Normal completion of the remote password unlock processing allows the access to the specified station.
- Perform communication using the MC protocol.

❸ Access prohibition processing (lock processing)

- To complete the access to the specified station, perform the line disconnection processing for the modem function in order to disable further access.
- When the modem disconnection is completed, the remote password lock processing is performed automatically.



[*1] Unlock and lock processing of the remote password for the host station is possible.
Remote password unlock and lock processing for the relay station and access station cannot be performed.

[*2] Lock processing is performed when a line for the modem function is disconnected.

[*3] For an Ethernet module which transmits a communication request to other Ethernet, the unlock and lock processing is not necessary even it is set as the subject of remote password check.

Restriction

The remote password unlock and lock processing can be performed to the C24 on the host station directly connected to the target device.

The remote password unlock and lock processing cannot be performed for the Ethernet module of the other stations (relay station and access station).

Point

- The remote password unlock processing from a target device is performed using dedicated commands for MC protocol communication.
- For corrective actions for abnormal completion of the remote password unlock processing, refer to the following section.
  (☞ Page 198 Considerations for using the modem function)

Remote password check processing performed by C24

■Communication in which a remote password check is performed

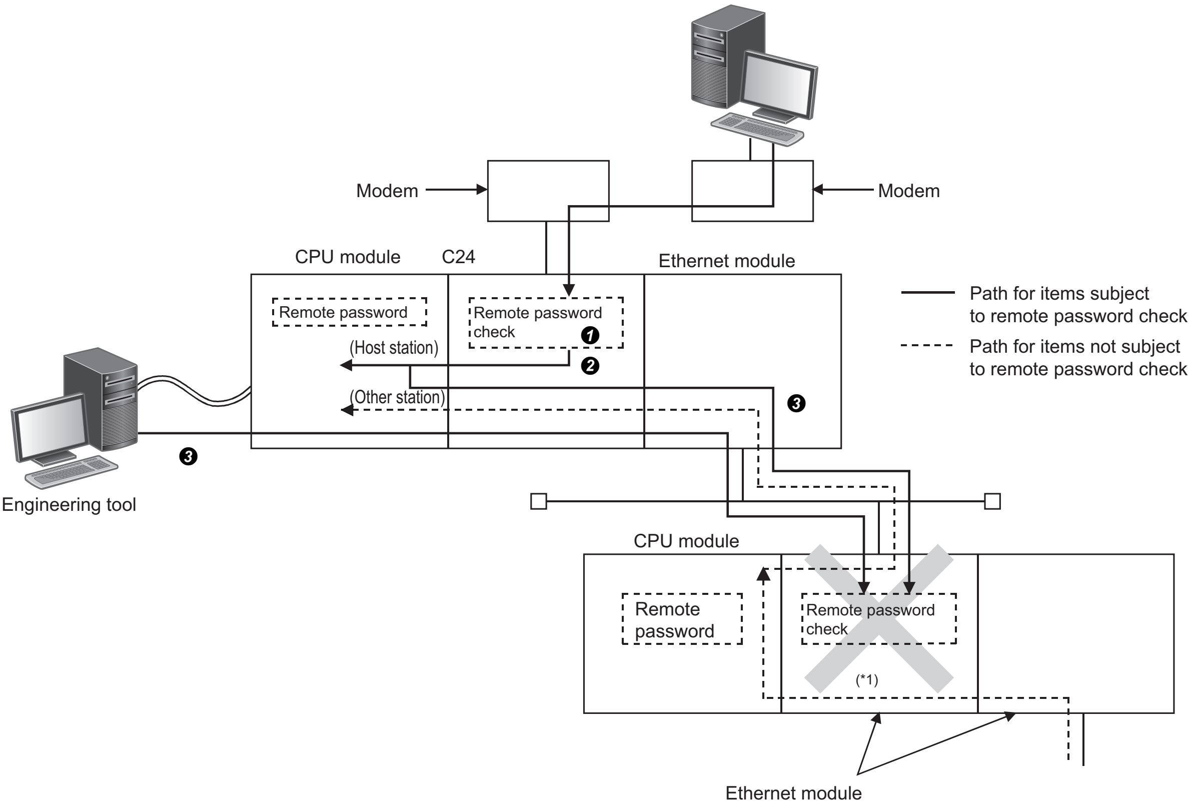
❶ When the following parameters are set for the C24 mounted on the host station CPU module, the C24 performs a remote password check for communication requests listed below.

- When a remote password is set in the CPU module.
- When C24 that is communicating data with the target device has been set as a module subject to the remote password check.

❷ C24 performs a remote password check with respect to a communication request to the host station/other station received from the target device.

❸ C24 does not perform a remote password check for the following communication requests.

- Transmission request from the CPU module on the host station (such as transmission using nonprocedural protocol)
- Communication request from the target device (including Engineering tool connected to the CPU module on the host station) transmitted to other station upon request from the CPU module.



[*1] In the above figure, a communication request from the target device cannot be received since the remote password check setting has been configured.
If the remote password check setting has not been configured, a communication request can be received and data communication from the target device is possible.

■Selecting modules to which remote password check is performed

C24 to which the remote password check is performed can be selected by the user, and it will be set by the parameters of the CPU module.

(This is set on the "Remote Password Setting" screen of Engineering tool.)

■Stations that can be accessed when the remote password check is performed

- The remote password unlock processing to the C24 on the directly connected station (host station) after line connection for the modem function enables the target device to access to the CPU module on the host station.
- When accessing the programmable controller on other stations via the Ethernet module on a relay station or access station, the access is allowed/prohibited by the settings below.
  To prohibit access to other station from a target device using the IP packet transfer function of the Ethernet module, select the checkbox for "MELSOFT Transmission Port (UDP/IP)" on the "Remote Password Detail Setting" screen for the relay station or access station.
  For the stations that can be accessed when accessing to the CPU module on the other stations via Ethernet module, refer to the following manual.
  (Refer to it by substituting the station connected to a target device with C24.)
  (📖 MELSEC iQ-R Ethernet User's Manual (Application))

Data communication procedure

The following shows the procedure when a target device performs data communication via C24 to which the remote password check is performed.

1. Initialization of the modem on the C24 side and target device side is performed at each device sides.
2. The line is connected from the target device.
3. The target device performs the remote password unlock (release) processing for the CPU module on the station where C24 is mounted using dedicated commands for MC protocol communication.
   (The unlock processing cannot be performed for the CPU module of other station.)
   For what to do when the remote password unlock processing is completed abnormally, refer to the following section.
   (☞ Page 182 Remote password check)
4. Data communication is performed from the target device using MC protocol.
5. When data communication using MC protocol is completed, a line for the modem is disconnected from the target device.
   When line disconnection is completed, the remote password lock processing is performed automatically.

■Remarks

- For more details on the unlock processing command to the remote password, refer to the following manual.
  (📖MELSEC Communication Protocol Reference Manual)

How to set the remote password

For the remote password setting method, refer to the following manual.

📖GX Works3 Operating Manual

List of input/output signals for the modem function

The input/output signals to/from the CPU module for the modem function are shown below.

For information on the input/output signals not related to the modem function, refer to the following section.

☞ Page 545 Input/Output Signal List

Ex.

The start I/O number of C24 is '0030'. ('n' indicates start I/O number of C24.)

Xn0 to X(n+1)F → X30 to X4F, Yn0 to Y(n+1)F → Y30 to Y4F

Function and description of input/output signal

I/O signal	Signal name	Function/description	Description
X10	Modem initialization completion	Indicates the normal completion of the initialization for the modem/TA connected to the host station side by C24 according to the initialization data designated.	Page 217 Initialization of modem/TA (terminal adapter)
X11	Dialing	Indicates that C24 is dialing (performing connection processing) the target device side according to the data for connection designated.	Page 222 Line connection
X12	Connection	• Indicates normal completion of the line connection processing from or to the target side. • When this signal is ON, data communication with the target device is possible.	
X13	Initialization/connection abnormal completion	• Indicates abnormal completion of the modem/TA initialization or line connection processing (dialing) to the target device. • Check the cause of the abnormal completion in the modem-error code storage area (address: 221H) and take corrective actions.	Page 217 Initialization of modem/TA (terminal adapter) Page 222 Line connection
X14	Modem disconnection completion	Indicates that the line for data communication with the target device has been disconnected.	Page 234 Line disconnection
Y10	Modem initialization request (standby request)	• Indicates the initialization request to the modem connected to the C24 on the host station. • Turn ON the initialization request signal after designating the data for initialization to the buffer memory when it is not set with Engineering tool.	Page 217 Initialization of modem/TA (terminal adapter)
Y11	Connection request	• Indicates the connection request (dialing) to enable data communication with the target device. • Turn ON the connection request signal after designating the data for connection to the buffer memory when it is not set with Engineering tool. • If the modem/TA connected to the host station is not initialized, the C24 side modem is initialized as well prior to dialing, according to the designated data for initialization.	Page 222 Line connection
Y12	Modem disconnection request	Indicates a line disconnection request to the target device upon completion of data communication.	Page 234 Line disconnection

Point

The descriptions hereafter show an example in which I/O numbers of the C24 are assigned to X/Y00 to X/Y1F.

Buffer memory

The buffer memory to be used for the modem function is shown below.

For the buffer memory that are not related to the modem function, refer to the following section.

☞ Page 550 Buffer memory list

Point

Read/write the setting values from/to the buffer memory using Engineering tool.

This section provides supplementary explanations on setting values used to perform settings and monitoring with an Engineering tool.

Precautions

Do not write data in the system area of the buffer memory.

If data is written in the system area, the programmable controller system may operate abnormally.

The system area exists in the user area partially. Therefore, take caution when reading from/writing to the buffer memory.

Details of the buffer memory (for modern function)

■Modem connection CH specification (Un\G46)

Designate the interface on the C24 side to which a modem/TA is connected.

■Connection retry count designation (Un\G48)

- Designate the number of retries for the connection request when the connection could not be made to the target device by the connection request.
- Using the default value for the connection re-try count is recommended.

■Connection retry interval designation (Un\G49)

- Designate the interval time of the retry processing for the connection request when the connection could not be made to the target device by the connection request.
- The default value is recommended to use for the connection retry interval.

■Initialization/connection timeout designation (Un\G50)

- Designate the following wait time:
  Wait time until the modem/TA initialization is complete
  Wait time per wait when the connection could not be made to the destination by the connection request
- The default value is recommended to use for the initialization/connection retry timeout.

■Number of initialization retries designation (Un\G51)

Designate the number of retries when the initialization by the initialization request to the modem on the C24 side has failed.

■Data No. for initialization designation (Un\G52)

- Designate the registration number of data for initialization which is transmitted by the initialization request to the modem on the C24 side.
  The registration number is the number registered in C24.
- For details on the designation by program, refer to the following section.
  (☞ Page 217 Initialization of modem/TA (terminal adapter))

■Data No. for connection designation (Un\G53)

- Designate the registration number of the data for connection used by C24 for the connection processing to the target device in order to perform data communication.
  The registration number is the number registered in C24.
- For details on the designation by program, refer to the following section.
  (☞ Page 222 Line connection)

■No-communication interval time designation (Un\G55)

- Designate the wait time until the line is closed when the data communication has ceased with the target device after the line connection.
- C24 automatically performs the line disconnection processing when no data communication is performed with the target device for a designated time.
  The 'Connection' (X12) and the 'Modem initialization completion' (X10) are turned OFF and the 'Modem disconnection completion' (X14) is turned ON when the line disconnection processing is performed.

■RS/CS control enable/disable designation (Un\G56)

- Designate whether or not the control to notify the data reception capability on the host station to the target side with the RS/CS signal during data communication between C24 and modem/TA.
- This setting is for the interface designated for the modem connection channel indicated in the 'Modem connection CH specification' (Un\G46).
  The control of the other interface that does not use the modem function is performed by the settings in the 'DTR/DSR (ER/DR), DC control designation' (Un\G147/307).

Point

The overview of the RS/CS control is shown below.

● When transmitting data

- C24 detects the modem/TA data reception capability by ON/OFF of the CS (CTS) signal.
- When the CS (CTS) signal is ON, data transmission from C24 starts or continues.
  When the CS (CTS) signal is OFF, data transmission from C24 is interrupted.

● When receiving data

- The C24 side reception capability is notified to the modem/TA by ON/OFF of the RS (RTS) signal.
- When the RS (RTS) signal is ON, C24 can receive data. Start/continue data transmission from the modem/TA to C24.
  When the RS (RTS) signal is OFF, C24 cannot receive data. Cancel data transmission from the modem/TA to C24.
- C24 turns ON/OFF the RS (RTS) signal in the following conditions:
  ON → OFF control of the RS (RTS) signal
  C24 turns OFF the RS (RTS) signal when the free OS area for receive data storage becomes 64 bytes (default) or less.
  OFF → ON control of the RS (RTS) signal
  C24 turns ON the RS (RTS) signal when the free OS area for receive data storage becomes 263 bytes (default) or more.

■Modem function error code (Un\G545)

- Error code for an error occurred while the modem function is used or when abnormal signal ('Initialization/connection abnormal completion' (X13)) turns ON is stored.
- For information on error codes, refer to the following section.
  (☞ Page 519 Error Code List)

■Modem function sequence status (Un\G546)

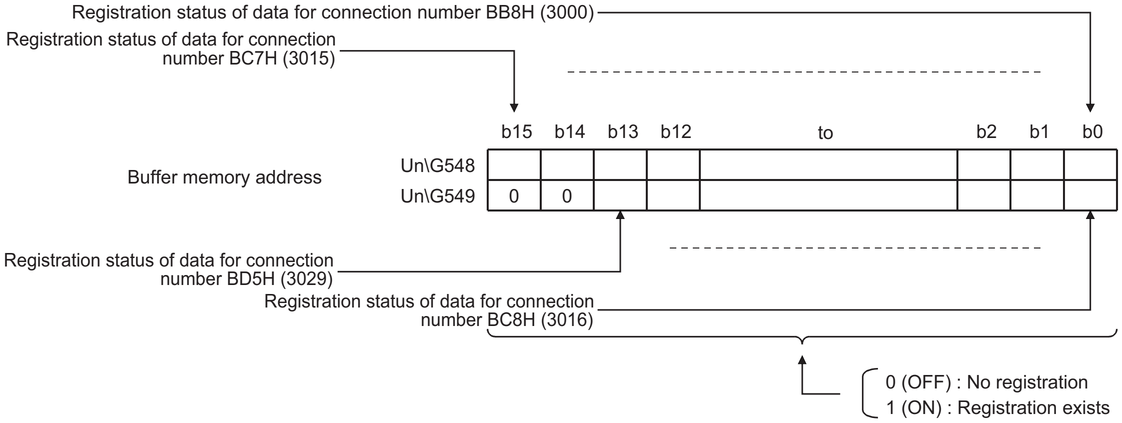
- The current status while the modem function is used is stored as a number.
- For information on the storage value of the modem function sequence status when using the modem function, refer to the following section.
  (☞Page 204 Startup procedure of the modem function)

■Number of data registration for connection (Un\G547)

- The number of registered data for connection, which has been registered in the flash ROM, used by C24 for connection processing with the target device to perform data communication is stored.
  The number of registration is the number of data for connection registered in the flash ROM by the user.
- For details on registration of data for connection, refer to the following section.
  (☞Page 213 Register/read/delete data for connection)

■Data registration status for connection (for confirmation of registration No.) (Un\G548 to 549)

- The registration status of data for connection, which has been registered in flash ROM, used by C24 for the connection processing with the target device to perform data communication is stored.
- The registration status of each data for connection with registration numbers, No.BB8H to BD5H (3000 to 3029), is indicated in the corresponding bit in the range shown in the figure below.


Registration status of data for connection number BB8H (3000)
Registration status of data for connection number BC7H (3015)
b15
b14
b13
b12
to
b2
b1
b0
Buffer memory address
Un\G548
Un\G549
0
0
Registration status of data for connection number BD5H (3029)
Registration status of data for connection number BC8H (3016)
0 (OFF) : No registration
1 (ON) : Registration exists

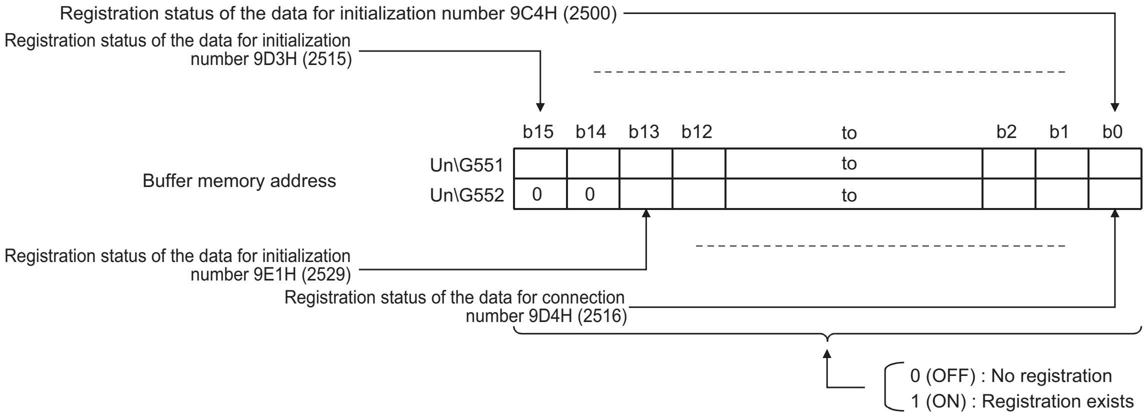
■Number of data registration for initialization (Un\G550)

The number of registered data for initialization, which has been registered in the flash ROM, transmitted to the modem connected to the C24 side for initialization request.

The number of registrations is the number of data for initialization registered in the Flash ROM by the user.

■Data registration status for initialization (for confirmation of registration No.) (Un\G551 to 552)

- The registration status of data for initialization, which has been registered in flash ROM, transmitted to the modem connected to the C24 side for initialization request.
- The registration status of each data for initialization with registration numbers, No.9C4H to 9E1H (2500 to 2529), is indicated in the corresponding bit in the range shown in the figure below.
- For details on registration of data for initialization, refer to the following section.
  (☞Page 209 Register/read/delete data for initialization)


Registration status of the data for initialization number 9C4H (2500)
Registration status of the data for initialization number 9D3H (2515)
b15
b14
b13
b12
to
b2
b1
b0
Buffer memory address
Un\G551
Un\G552
to
to
0
0
Registration status of the data for initialization number 9E1H (2529)
Registration status of the data for connection number 9D4H (2516)
0 (OFF) : No registration
1 (ON) : Registration exists

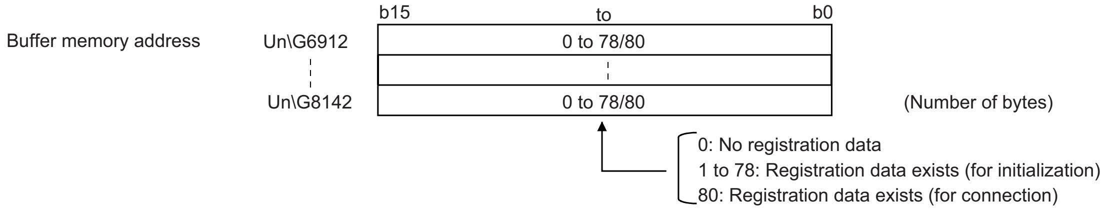
■For user frame registration: Registration data byte count designation area (Un\G6912, 6953···)

- The initialization data or data for connection can be registered in the buffer memory as well as the flash ROM of C24.

Data type	Registration destination		Registration number (Decimal (Hex.))
Initialization data	Flash ROM	Data registered by the operating system	2000 to 2013 (7D0H to 7DDH)
		User-registered data	2500 to 2529 (9C4H to 9E1H)
	Buffer memory	(All registered by user)	-32767 to -32737 (8001H to 801FH)
Data for connection	Flash ROM	(All registered by user)	3000 to 3029 (BB8H to BD5H)
	Buffer memory	(All registered by user)	-32767 to -32737 (8001H to 801FH)

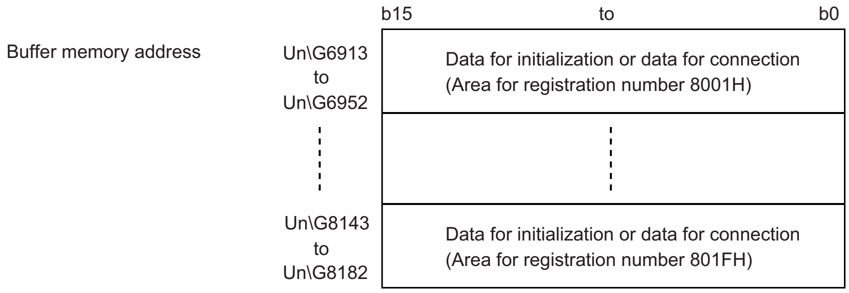
A registration number to register data for initialization or data for connection to the buffer memory is in the range of -32767 to -32737 (8001H to 801FH) respectively, which is determined by the area to be used.

- In this area, the number of bytes for the data for initialization or data for connection (for 1 data) to be registered in the buffer memory is designated.
- For more information on how to register data for initialization or data for connection in the buffer memory, refer to the following sections.
  (☞ Page 209 Register/read/delete data for initialization, ☞ Page 213 Register/read/delete data for connection)



■For user frame registration: User frame specification area (Un\G6913 to 6952, 6954 to 6993 ···)

- Designate the applicable data (for 1 data) for the above registration data byte count when registering the data for initialization or data for connection in the buffer memory.
- For more information on how to register data for initialization or data for connection in the buffer memory, refer to the following sections.
  (☞ Page 209 Register/read/delete data for initialization, ☞ Page 213 Register/read/delete data for connection)


Buffer memory address
b15
to
b0
Un\G6913
to
Un\G6952
Data for initialization or data for connection
(Area for registration number 8001H)
Un\G8143
to
Un\G8182
Data for initialization or data for connection
(Area for registration number 801FH)

■Auto modem initialization designation (Un\G8199)

- Designate whether the modem on the C24 side is initialized automatically or not.
- Registration in this area is performed with Engineering tool.
  The modem is initialized automatically when C24 starts up after the CPU module restarts.

■Modem initialization DR (DSR) signal enable/disable designation (Un\G8200)

Set "DR signal enabled" for this setting.

This setting is used to designate how the DR (DSR) signal is treated only when modem initialization is performed. Following the completion of modem initialization, data is sent according to the status of the DR (DSR) signal.

■Complete signal handling for modem function designation (Un\G8201)

Set "Turn ON/OFF X13, X14" (default value) for this setting.

■Line disconnection wait time (for PLC CPU monitor) (Un\G8206)

- When transmitting data using the programmable controller CPU monitoring function, designate the time between data transmission from C24 and completion of data transmission from the modem on the host station side to the target device side (wait time to disconnect the line).
- Specify the line disconnection wait time in accordance with the specifications of the modem to be used.

Details of buffer memory (for the remote password function)

Each of the areas described below is valid when C24 performs the remote password check.

■Remote password mismatch notification count designation (Un\G8204)

- Designate the count that will be the timing to notify the CPU module that a remote password mismatch occurred during the unlock processing by the user/target device after the modem line has been connected with 0 to FFFFH.
- Confirming the accumulated number of times of occurred remote password mismatch up to the present time after the line connection is not available.

■Remote password mismatch notification accumulated count designation (Un\G8205)

- Designate the accumulated count that will be the timing to notify the CPU module that a remote password mismatch occurred during the unlock processing by the user/target device after C24 has been started up with 0 to FFFFH.
- The accumulated number of times remote password mismatch occurred up to the present time from C24 startup (accumulated count value by C24) can be checked in the 'Accumulated count of unlock process abnormal completion' (Un\G8956).

■Accumulated count of unlock process normal completion (Un\G8955)

The accumulated number of times the remote password unlock process has been completed normally is stored.

■Accumulated count of unlock process abnormal completion (Un\G8956)

The accumulated number of times the remote password unlock process has been completed abnormally is stored.

■Accumulated count of lock process by line disconnection (Un\G8959)

The accumulated number of times C24 has automatically performed the lock processing due to a modem line disconnection is stored.

Point

The supplemental information for each of the accumulated count stored in Un\G8955, 8956, and 8959 is shown below:

- The value can be cleared by writing "0" to the corresponding area of the buffer memory.
- The accumulated count will be stored as 0H → 1H → 2H ··· → FFFFH → 0H → 1H → ···.

Considerations for using the modem function

Considerations for using the modem function to perform data communication with a target device via public line are shown below.

Modem connection and disconnection

When performing data communication with a target device, it must be predetermined which station is to perform the line connection (dialing) and disconnection processing with the target device as well as the timings.

Reception data before connection completion

Until the connection processing to the modem is completed, the received data other than modem commands is ignored (read and discarded) at the interface that uses the modem function.

(Example) C24 will ignore the data even when an MC protocol command message is received.

Transmission control

Delays may occur in transmission controls to notify the data reception capability at the host station to the target device.

The amount of transmitted/received data and the interval must be determined beforehand in order to prevent any conditions in which the target device cannot receive transmitted data.

When transmitting/receiving data in the nonprocedural protocol, the procedure must also be predetermined.

Priority of data communication

The data transmission/reception processing with the target device after line connection has been established is performed in the order of the processing request occurrence.

At the same time, when the line disconnection processing or data transmission/reception (including data transmission processing, reception processing and flash ROM access processing) occurs, the line disconnection processing has the priority.

Data communication time

The data transmission/reception time after line connection has been established with the target device is the total time of the transmission time between C24 and modem/TA, between modem and TA, and between modem/TA and target device.

When communicating via the MC protocol, the transmission time (such as T0 and T3) must include the transmission time between the modem/TA on the C24 side and the target device.

For details, refer to the following manual.

📖MELSEC Communication Protocol Reference Manual

Initial Settings

The data for connection, which is used for the modem function, can be registered as follows depending on how to set.

Set the telephone number and message within the allowable number of registration characters for modem/TA.

■Registering using Engineering tool

- Up to 254 bytes can be set for comments. (Not used for control.)
- Up to 62 bytes can be set for telephone number.

■Registering by the CPU module (program)

- Comments cannot be set.
- Up to 18 bytes can be set for telephone number.

Programmable controller CPU monitoring function

For considerations for transmitting monitored results with the modem function, refer to the following section.

☞ Page 179 Modem connection error

Remote password check

■How to unlock the remote password

- When C24 has been set as a subject to the remote password check with the CPU module parameter, a remote password unlock processing must be performed from the target device after line connection before starting data communication.
- The unlock processing for the CPU module remote password is performed as follows:
  When communicating using MC protocol: Perform the unlock processing from the target device using dedicated commands.

■When the remote password unlock processing is completed abnormally

- Repeat the unlock processing after checking the remote password set in the CPU module.
- Start with the line connect processing again if C24 'Connection' (X12) turns OFF due to the unlock processing abnormal completion.
- Perform the line connect processing again after clearing the accumulated count stored in the following buffer memory when the 'CH□ Error occurrence' (XE/XF) of C24 turns ON and the ERR LED lights up due to the unlock processing abnormal completion.
  (Target buffer memory)
  'Accumulated count of unlock process abnormal completion' (Un\G8956)
  (How to clear)
  Write "0" to the relevant area of the buffer memory.

■When the number of times remote password mismatch occurs is large

- When the number of times notification of a remote password mismatch is received exceeds the 'Remote password mismatch notification count designation' (Un\G8204) , C24 disconnects the line automatically. (The 'Connection' (X12) turns OFF.)
  After confirming the remote password registered in the CPU module and the remote password specified in the target device, restart form the line connection.
- When the count exceeds the 'Remote password mismatch notification accumulated count designation'(Un\G8205), C24 performs the following processing every time the mismatch occurs (The line to the modem is not disconnected.)
  An error code (7FE8H) is stored in the 'MC protocol send error code' (Un\G602/618) on the buffer memory.
  The 'CH□ Error occurrence' (XE/XF) turns ON and the ERR LED turns ON.
- The accumulated number of times remote password mismatch occurred up to the present time (accumulated count value by C24) can be checked in the 'Accumulated count of unlock process abnormal completion' (Un\G8956).
- The accumulated number of times up to the present time should be cleared by writing "0" to the 'Accumulated count of unlock process abnormal completion' on the buffer memory (Un\G8956).

Point

- If the 'CH□ Error occurrence' (XE/XF) for C24 turns ON and the ERR LED turns ON, this may indicate improper access from the target device. Refer to "CPU module side countermeasures against improper access from a target device".
  (☞ Page 200 CPU module side countermeasures against unauthorized access from a target device)
- For details on how to turn OFF the ERR LED of C24, refer to the following manual.
  (☞ Page 492 Turning OFF the ERR LED and initializing the error codes)
- 'CH□ Error occurrence' (XE/XF) turns OFF when the data transmission/reception processing is performed normally.

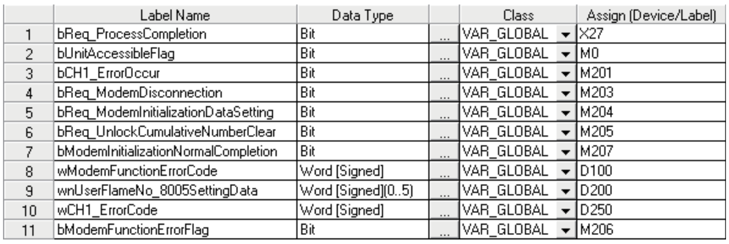
CPU module side countermeasures against unauthorized access from a target device

The following is an example of performing, on the programmable controller CPU side, the line disconnect processing against the target device and prohibiting receive via a modem when the number of "remote password mismatch" detected by the C24 remote password check function exceeds the number set by the user.

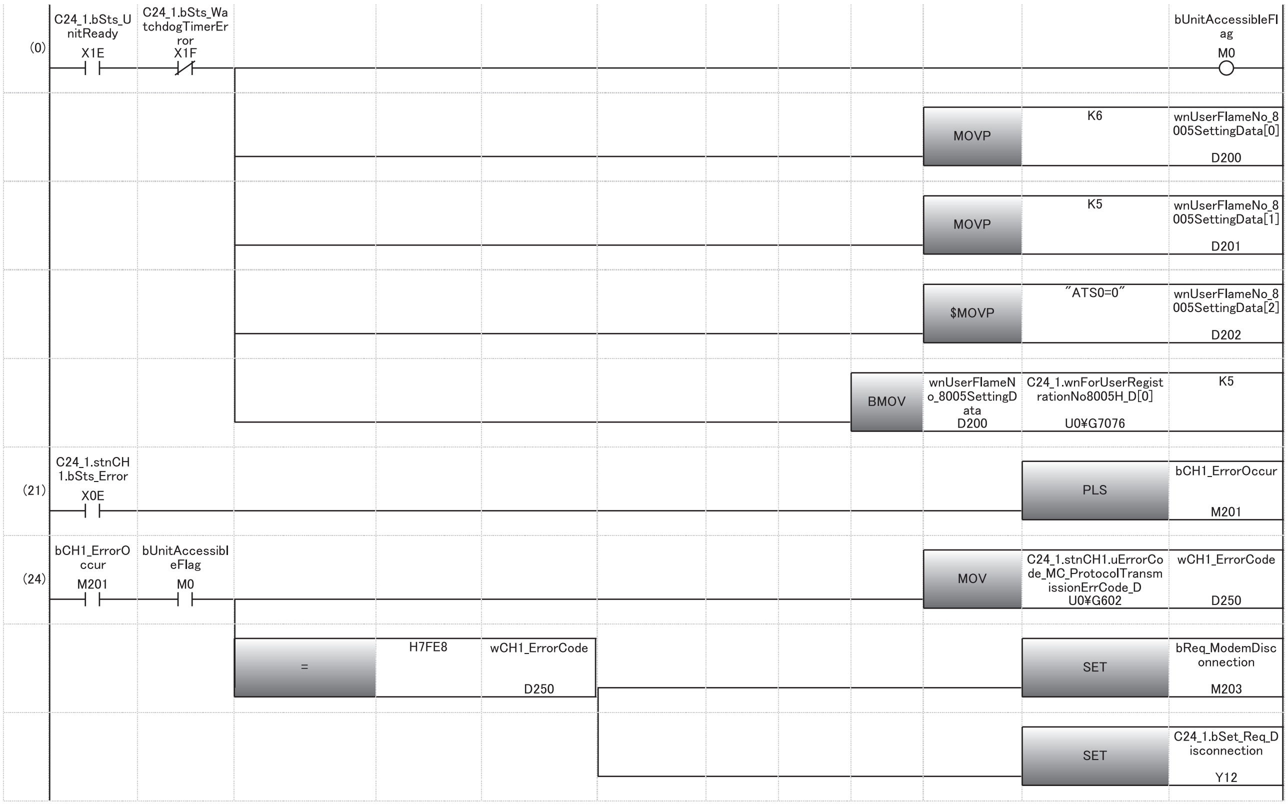
- Determine at the user side the count and accumulated count for notification when a remote password mismatch occurs in the unlock processing from external devices with respect to the remote password set in the CPU module.
  (☞ Page 197 Details of buffer memory (for the remote password function))
- Set the above counts to the items below for the module parameters in order for C24 to operate, and register them in C24.
  Set in "Remote password mismatch notification count."
  Set in "Remote password mismatch notification accumulated count."
- In addition to initialization commands used in normal modem initialization, specify "No automatic receive" and additionally register the modem initialization command for this step to C24. (No automatic receive: This is the setting to prohibit line connection from the target device.)
- Constantly monitor the rise (OFF → ON) of the 'CH□ Error occurrence' (XE/ XF) when connecting the line to the target device using the modem function after the system begins operating.
- Monitor the following buffer memory when the error occurred signal turns ON.
  'MC protocol send error code' (Un\G602)
- Perform the line disconnect processing to the target device when the error code stored in the above buffer memory is 7FE8H.
  (Use the 'Modem disconnection request' (Y12).)
- After the above line disconnection processing is completed, specify the modem initialization command for which "No automatic receive" has been specified and perform modem initialization only. (☞ Page 231 Data communication)
- Describe the above occurrence to the system manager and take necessary measures.

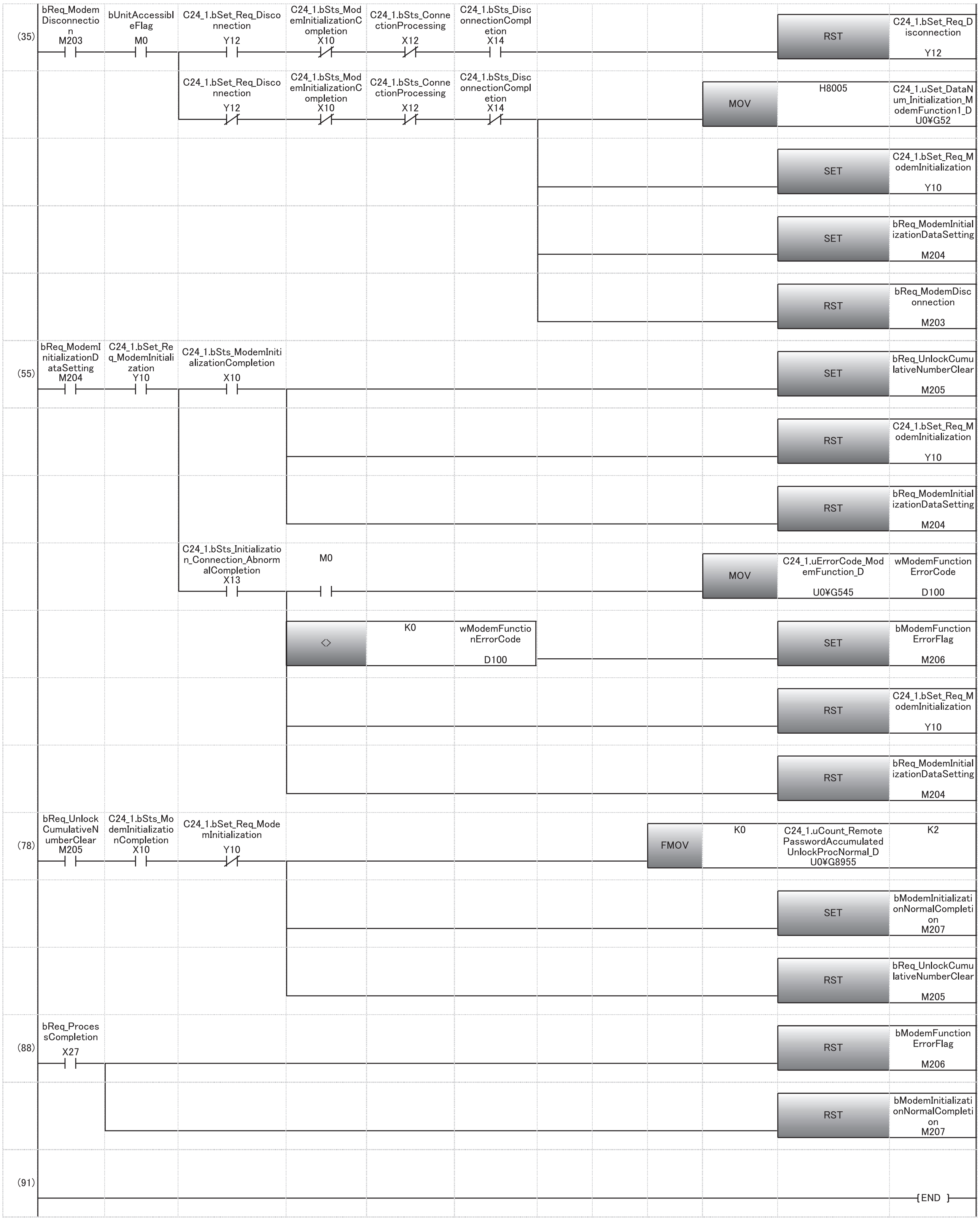
■Program example

When the number of times a remote password mismatch occurred exceeds the notification accumulated count setting value during the remote password unlock processing in communication using MC protocol with the C24 CH1 side interface used.

Category	Label name	Description	Device
Module label	C24_1.stnCH1.bSts_Error	CH1 Error occurrence	X0E
	C24_1.bSts_ModemInitializationCompletion	Modem initialization completion	X10
	C24_1.bSts_ConnectionProcessing	Connection	X12
	C24_1.bSts_Initialization_Connection_AbnormalCompletion	Initialization/connection abnormal completion	X13
	C24_1.bSts_DisconnectionCompletion	Modem disconnection completion	X14
	C24_1.bSts_UnitReady	Module READY	X1E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X1F
	C24_1.bSet_Req_ModemInitialization	Modem initialization request	Y10
	C24_1.bSet_Req_Disconnection	Modem disconnection request	Y12
	C24_1.uSet_DataNum_Initialization_ModemFunction1_D	Data No. for initialization designation	U0\G52
	C24_1.uErrorCode_ModemFunction_D	Modern function error code	U0\G545
	C24_1.stnCH1.uErrorCode_MC_ProtocolTransmissionErrCode_D	Protocol transmission error code	U0\G602
	C24_1.wnForUserRegistrationNo8005H_D[0]	User registration number 8005H	U0\G7076
	C24_1.uCount_RemotePasswordAccumulatedUnlockProcNormal_D	Accumulated count of unlock process normal completion	U0\G8955
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_ProcessCompletion	Bit	...	VAR_GLOBAL	X27
2	bUnitAccessibleFlag	Bit	...	VAR_GLOBAL	M0
3	bCH1_ErrorOccur	Bit	...	VAR_GLOBAL	M201
4	bReq_ModemDisconnection	Bit	...	VAR_GLOBAL	M203
5	bReq_ModemInitializationDataSetting	Bit	...	VAR_GLOBAL	M204
6	bReq_UnlockCumulativeNumberClear	Bit	...	VAR_GLOBAL	M205
7	bModemInitializationNormalCompletion	Bit	...	VAR_GLOBAL	M207
8	wModemFunctionErrorCode	Word [Signed]	...	VAR_GLOBAL	D100
9	wnUserFlameNo_8005SettingData	Word [Signed](0..5)	...	VAR_GLOBAL	D200
10	wCH1_ErrorCode	Word [Signed]	...	VAR_GLOBAL	D250
11	bModemFunctionErrorFlag	Bit	...	VAR_GLOBAL	M206


(0)
C24_1.bSts_UnitReady
X1E
C24_1.bSts_WatchdogTimerError
X1F
bUnitAccessibleFlag
M0
MOVP
K6
wnUserFlameNo_8005SettingData[0]
D200
MOVP
K5
wnUserFlameNo_8005SettingData[1]
D201
$MOVP
"ATS0=0"
wnUserFlameNo_8005SettingData[2]
D202
BMOV
wnUserFlameNo_8005SettingData
D200
C24_1.wnForUserRegistrationNo8005H_D[0]
U0¥G7076
K5
(21)
C24_1.stnCH1.bSts_Error
X0E
PLS
bCH1_ErrorOccur
M201
(24)
bCH1_ErrorOccur
M201
bUnitAccessibleFlag
M0
MOV
C24_1.stnCH1.uErrorCode_MC_ProtocolTransmissionErrCode_D
U0¥G602
wCH1_ErrorCode
D250
=
H7FE8
wCH1_ErrorCode
D250
SET
bReq_ModemDisconnection
M203
SET
C24_1.bSet_Req_Disconnection
Y12


(35)
bReq_ModemDisconnection M203
bUnitAccessibleFlag M0
C24_1.bSet_Req_Disconnection Y12
C24_1.bSts_ModemInitializationCompletion X10
C24_1.bSts_ConnectionProcessing X12
C24_1.bSts_DisconnectionCompletion X14
RST C24_1.bSet_Req_Disconnection Y12
C24_1.bSet_Req_Disconnection Y12
C24_1.bSts_ModemInitializationCompletion X10
C24_1.bSts_ConnectionProcessing X12
C24_1.bSts_DisconnectionCompletion X14
MOV H8005 C24_1.uSet_DataNum_Initialization_ModemFunction1_D U0¥G52
SET C24_1.bSet_Req_ModemInitialization Y10
SET bReq_ModemInitializationDataSetting M204
RST bReq_ModemDisconnection M203
(55)
bReq_ModemInitializationDataSetting M204
C24_1.bSet_Req_ModemInitialization Y10
C24_1.bSts_ModemInitializationCompletion X10
SET bReq_UnlockCumulativeNumberClear M205
RST C24_1.bSet_Req_ModemInitialization Y10
RST bReq_ModemInitializationDataSetting M204
C24_1.bSts_Initialization_Connection_AbnormalCompletion X13
M0
MOV C24_1.uErrorCode_ModemFunction_D U0¥G545 wModemFunctionErrorCode D100
<> K0 wModemFunctionErrorCode D100
SET bModemFunctionErrorFlag M206
RST C24_1.bSet_Req_ModemInitialization Y10
RST bReq_ModemInitializationDataSetting M204
(78)
bReq_UnlockCumulativeNumberClear M205
C24_1.bSts_ModemInitializationCompletion X10
C24_1.bSet_Req_ModemInitialization Y10
FMOV K0 C24_1.uCount_RemotePasswordAccumulatedUnlockProcNormal_D U0¥G8955 K2
SET bModemInitializationNormalCompletion M207
RST bReq_UnlockCumulativeNumberClear M205
(88)
bReq_ProcessCompletion X27
RST bModemFunctionErrorFlag M206
RST bModemInitializationNormalCompletion M207
(91)
END

Number of modules for which the remote password check can be set

Up to eight modules can be registered as the remote password applicable module in the CPU module. [1]

For the settings of the remote password applicable module, use the "Remote Password Setting" screen of Engineering tool.

[1] The remote password applicable modules are as follows:
- C24
- Ethernet module
- Built-in Ethernet port CPU module

9.4 Startup of the Modem Function

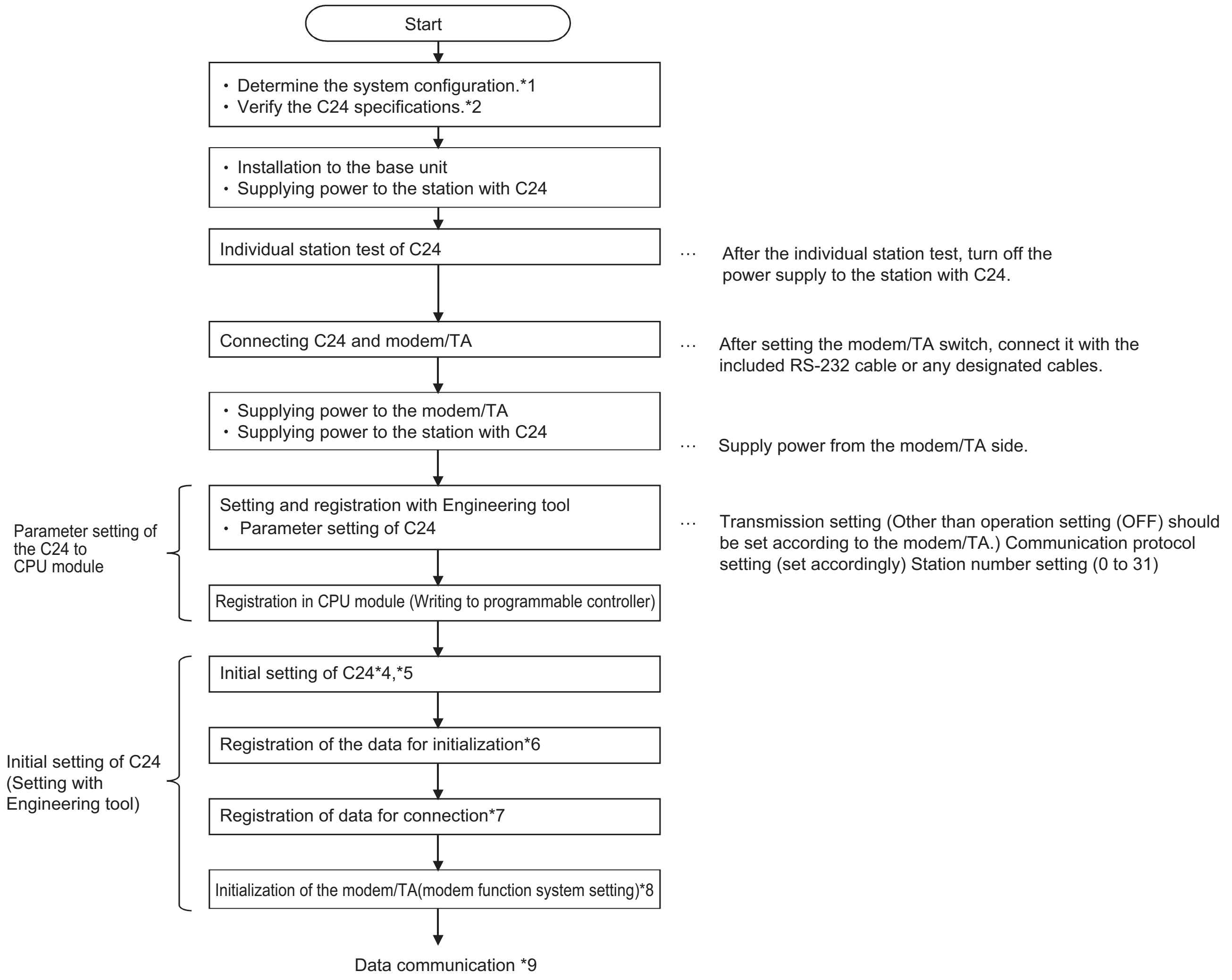
This section explains the startup procedures, processing methods and programming when the modem function of C24 is used.

Startup procedure of the modem function

The following shows the procedure to start the modem function and start data communications.

Procedure up to initialization of C24 modem/TA

Initialize the modem/TA connected to C24 in order to use the modem function. Initialization of the modem/TA enables respective communications.


Start
・ Determine the system configuration.*1
・ Verify the C24 specifications.*2
・ Installation to the base unit
・ Supplying power to the station with C24
Individual station test of C24
Connecting C24 and modem/TA
・ Supplying power to the modem/TA
・ Supplying power to the station with C24
Setting and registration with Engineering tool
・ Parameter setting of C24
Registration in CPU module (Writing to programmable controller)
Parameter setting of the C24 to CPU module
Initial setting of C24*4,*5
Registration of the data for initialization*6
Registration of data for connection*7
Initialization of the modem/TA(modem function system setting)*8
Initial setting of C24 (Setting with Engineering tool)
Data communication *9
...
...
...
...

After the individual station test, turn off the power supply to the station with C24.

After setting the modem/TA switch, connect it with the included RS-232 cable or any designated cables.

Supply power from the modem/TA side.

Transmission setting (Other than operation setting (OFF) should be set according to the modem/TA.) Communication protocol setting (set accordingly) Station number setting (0 to 31)

*1 ☞ Page 184 System Configuration

*2 📖MELSEC iQ-R Serial Communication Module User's Manual(Startup)

*3 ☞ Page 492 Individual station test

*4 ☞ Page 206 Initial settings of Serial communication module

*5 ☞ Page 454 PARAMETER SETTING

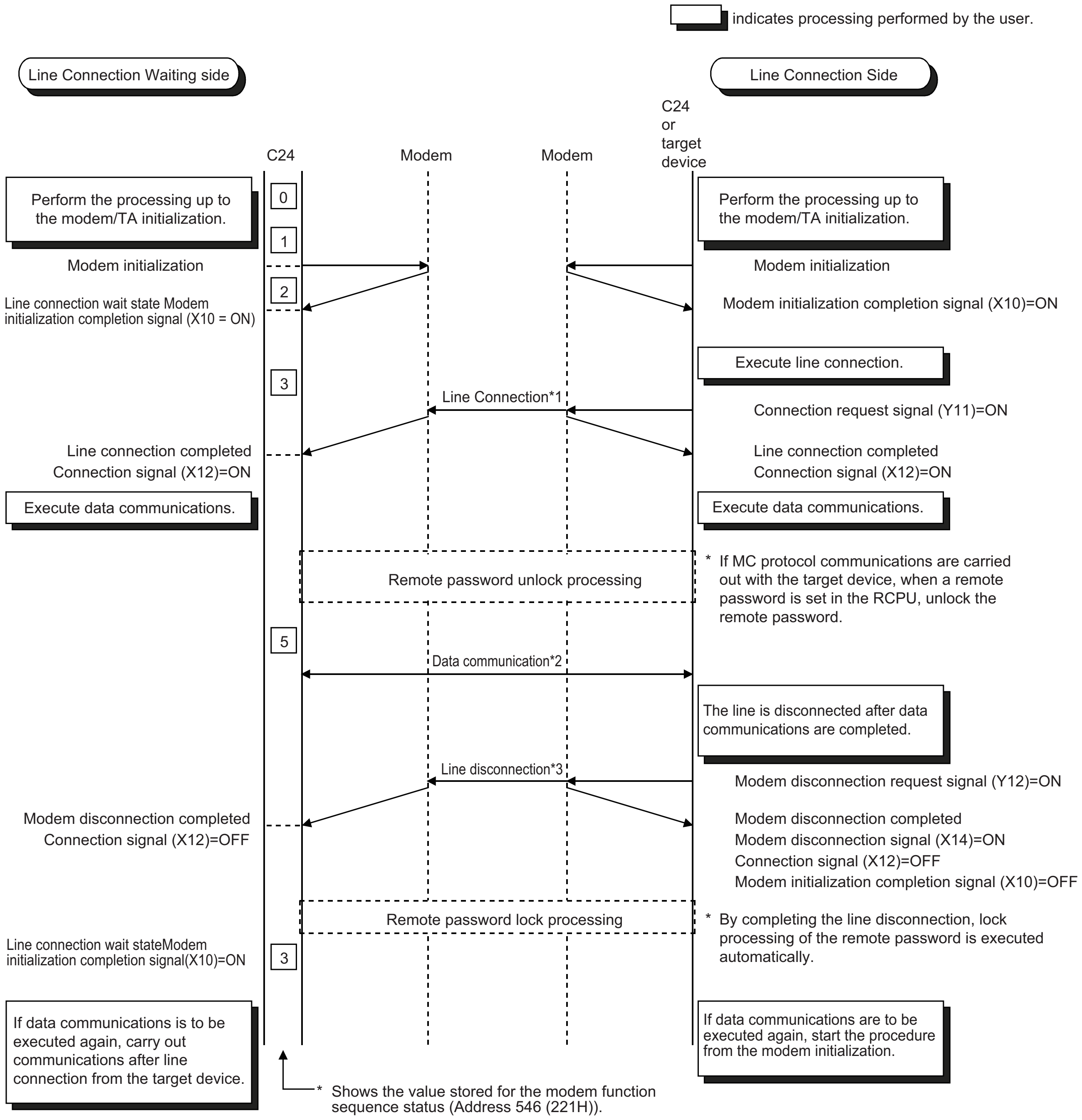
*6 ☞ Page 209 Register/read/delete data for initialization

*7 ☞ Page 213 Register/read/delete data for connection

*8 ☞ Page 217 Initialization of modem/TA (terminal adapter)

*9 ☞ Page 205 Procedure for data communication

Procedure for data communication



indicates processing performed by the user.
Line Connection Waiting side
Line Connection Side
C24
Modem
Modem
C24
or
target
device
Perform the processing up to the modem/TA initialization.
0
1
Perform the processing up to the modem/TA initialization.
Modem initialization
Modem initialization
2
Line connection wait state Modem initialization completion signal (X10 = ON)
Modem initialization completion signal (X10)=ON
Execute line connection.
3
Line Connection*1
Connection request signal (Y11)=ON
Line connection completed
Connection signal (X12)=ON
Line connection completed
Connection signal (X12)=ON
Execute data communications.
Execute data communications.
Remote password unlock processing
* If MC protocol communications are carried out with the target device, when a remote password is set in the RCPU, unlock the remote password.
5
Data communication*2
The line is disconnected after data communications are completed.
Line disconnection*3
Modem disconnection request signal (Y12)=ON
Modem disconnection completed
Connection signal (X12)=OFF
Modem disconnection completed
Modem disconnection signal (X14)=ON
Connection signal (X12)=OFF
Modem initialization completion signal (X10)=OFF
Remote password lock processing
* By completing the line disconnection, lock processing of the remote password is executed automatically.
Line connection wait stateModem initialization completion signal(X10)=ON
3
If data communications is to be executed again, carry out communications after line connection from the target device.
If data communications are to be executed again, start the procedure from the modem initialization.
* Shows the value stored for the modem function sequence status (Address 546 (221H)).

*1 ☞ Page 222 Line connection

*2 ☞ Page 231 Data communication

*3 ☞ Page 234 Line disconnection

Initial settings of Serial communication module

This section explains the initial settings of C24 for data communication with a target device by using the modem function.

Communication protocol setting, Communication speed setting, Transmission setting

Setting item		Data communication
Test mode setting		No specification
Communication protocol setting		• MC protocol (Formats 1 to 5) • Nonprocedural protocol • Bidirectional protocol
Communication speed setting		(Set according to modem/TA on the host station)
Transmission setting	Operation setting	Independent
	Data bit	(Set according to modem/TA on the host station)
	Parity bit	
	Odd/even parity	
	Stop bit	
	Sum check code	(Set according to system specification)
	Online change	
	Setting change	
Station number setting (CH1, 2 in common: 0 to 31)		0 to 31

Initial settings with Engineering tool

■Initial settings for the side of interface that uses the modem function

Perform initial settings on the side of interface that uses the modem function.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ Module Name ⇨ [Module Parameter] ⇨ "Application Settings" ⇨ "Modem function setting"

●: Required item ○: Configurable ×: Setting not required

Setting item		Data communication (MC, Nonprocedural ,Bidirectional)
Modem function specification 1	Modem connection CH specification	●
	Connection re-try count designation	○
	Connection retry interval designation	○
	Initialization/connection timeout designation	○
	Number of initialization retries designation	○
	Data No. for initialization designation[*1]	●
	Data No. for connection designation[*1]	●
	No-communication interval time designation[*2]	○
	RS/CS control enable/disable designation	○
Modem function specification 2	Auto modem initialization designation	○
	Modem initialization DR (DSR) signal enable/disable designation[*3]	○
Modem function specification 3	Line disconnection wait time (for PLC CPU monitor)	○
For remote password function	Remote password mismatch notification count designation	○
	Remote password mismatch notification accumulated count designation	○

[*1] For details on how to register the data No. for initialization, refer to the following section.
☞ Page 209 Register/read/delete data for initialization
For details on how to register the data No. for connection, refer to the following section.
☞ Page 213 Register/read/delete data for connection

[*2] Even if the CPU module on the station with C24 (host station) becomes STOP status under the following circumstances, the line (telephone) with the target devices will be left connected.
To prevent the line from being left connected when the line is not in use, be sure to make the appropriate settings.
· When the CPU module is stopped when the 'Connection' (X12) is in the ON status. (This occurs because the program write after remote stop is enabled.)
· When the CPU module stops abnormally during self-diagnosis, etc.

[*3] The "Modem initialization DR (DSR) signal enable/disable designation" designates whether the status of the DR (DSR) signal output is valid or invalid at the startup of C24 .
·When the DR (DSR) signal = ON is output from the modem, register "valid" for this item.
· When the DR (DSR) signal = ON is not output from the modem, register "invalid" for this item.

Precautions

- When setting the no-communication interval time to 'Infinite wait' (setting value = 0), be sure to perform line disconnection processing after the data has been communicated.
- Leaving the line connected for long periods of time without performing line disconnection after data communication has been completed not only will run up a large telephone bill but also may violate electronic communication business laws.

■Transmissions using the modem function

All communications using the modem function are performed in full-duplex communications. Leave the following initial settings for the interface side that uses the modem function as default.

- CD terminal check : Do not check
- Communication method: Full-duplex communications

■Processing that corresponds to output signals

The processing to the following output signals cannot be aborted.

Output signal	Request processing name
Y10	Initialization request (standby request)
Y11	Connection request
Y12	Modem disconnection request

Using the following initial settings for the modem function as default is recommended. (When any error occurred, it will end due to the time-out.)

- Connection re-try count: 3 times
- Connection retry intervals : 180 s
- Initialization/connection timeout : 60 s

Also configure the settings for a remote password check when the remote password check is executed to the remote password set in the module CPU. (☞ Page 197 Details of buffer memory (for the remote password function))

Register/read/delete data for initialization

The section explains the registration/reading/deletion of the data for initialization such as initialization commands for the modem/TA connected to the C24 side for data communication with the target device using the modem function.

Details of the data for initialization

The following shows the registration destination and type of the data for initialization, registration number, and registration count.

Up to 78 bytes of initialization commands can be registered as one data for initialization. (For 78 characters)

Registration destination of data for initialization		Types of data for initialization that is registered	Registration number of data for initialization	Number of registrations	Remarks
CPU module	Built-in memory	Arbitrary data for initialization registered by the user*1	2500 to 2529 (9C4H to 9E1H)	30	• Register the data for initialization after the debug process has been completed. • By registering data in this area, replacing C24 can be performed easily.
	SD memory card	Arbitrary data for initialization registered by the user*1	2500 to 2529 (9C4H to 9E1H)	30	• Register the data for initialization after the debug process has been completed. • By registering data in this area, replacing C24 can be performed easily.
C24	Flash ROM	Data for initialization that is registered before shipment	2000 to 2013 (7D0H to 7DDH)	13	• ☞ Page 210 Registration contents before shipment
		Arbitrary data for initialization registered by the user*1	2500 to 2529 (9C4H to 9E1H)	30	• Register the data for initialization after the debug process has been completed.
	Buffer memory	Arbitrary data for initialization registered by the user	-32767 to -32737 (8001H to 801FH)	31	• Register the data for initialization before the debug process has been completed.

*1 For the priority of the registered data, refer to the following section.
☞ Page 434 PARAMETER REGISTRATION TO CPU MODULE

Considerations for registration of data for initialization

- Do not include CR/LF (data code : 0DH/0AH) in the data for initialization to be registered to C24.
  The CR/LF is output at the end of the AT command by C24 at the time of the initialization processing (automatically added).
- The registration status of the data for initialization stored in the flash ROM may be checked in the buffer memory (Un\G550 to 552).
  For initial registration, register by designating an unregistered number.
  When designating a registration number that has already been registered, first delete the registration data for the designated number prior to registration.

Registration contents before shipment

- The data for initialization registered in the flash ROM of C24 are listed below:

Registration number		Initialization command	Devices	
Hexadecimal	Decimal		Manufacturer	Model
7D0H	2000	ATQ0V1E1X1\J0\Q2\V2\N3S0=1	Iwa Corporation	PV-AF2881WW, PV-BF288M2
7D1H	2001	ATQ0V1E1X1\Q2\V2\N3S0=1	Micro Research Institute, Inc.	MC288XE, MC288X1
7D2H	2002	ATQ0V1E1X1&K3\N3S0=1	Microcom Technologies, Inc.	DESKPORTE22.8S, DESKPORTE33.6S
7D3H	2003	ATQ0V1E1X1&H1&R2&A3&D2S0=1	OMRON Corporation	ME3314B
7D4H	2004	ATQ0V1E1X1\J0\Q2\N3S0=1	Sun Electronics Int., Inc.	MS336AF
7D5H	2005	ATE1Q0V1&C1&D2&H1&I0&R2&S0S0=1	OMRON Corporation	ME5614B
7D6H	2006	ATE1Q0V1&C1&D2&K3&S0S0=1	Sun Electronics Int., Inc.	MS56KAF
			Micro Research Institute, Inc.	MRV56XL
			Matsushita Electric Industrial Company, Ltd.	VS-2621A
7D7H	2007	ATE1Q0V1&C1&D2&K3&S1S0=1		VC-173
7D8H	2008	ATE1Q0V1&C1&D2&K3&S0S0=1	OMRON Corporation	MT128BⅡ-D
7D9H	2009	ATE1Q0V1&C1&D1\Q2&S0S0=1	Sun Electronics Int., Inc.	TS128JXⅡ
7DAH	2010	ATE1Q0V1&C1&D2\Q3&S0S0=1	Sharp Electronics Corporation.	DN-TA1
7DCH	2012	AT&S0S0=1	General devices • Use the general devices for checking the operation. • If the modem/TA does not work, create an initialization command appropriate for the modem specification.	
7DDH	2013	ATX1&S0S0=1		

- If initialization commands other than listed above are needed, register the data for initialization in the flash ROM or the buffer memory of the C24 and use it.

■Remarks

- Perform the following setting to the modem/TA connected to the C24 side.
  For settings other than listed below, configure the setting in accordance with the modem/TA specifications.
  Refer to the instruction manual for the model to be used for details.

Setting content	Setting command example
—	AT
Display the result code (or, return the result code).	Qn
Set the result code to a word.	Vn
Perform character echo.	En
Dial tone and busy tone detection + X1	Xn
Set register 0 to 2.	Sr=n
The modem and the serial speed are not equal.	\Jn
Control RTS/CTS.	\Qn
Control DSR.	&Sn
Control DTR.	&Dn
Enable extension result code (display MNP class).	\Vn
MNP mode/normal mode auto selection	\N3

Method to register/read/delete data for initialization

The applicable registration area is shown below.

Applicable area	Method	Remarks
Flash ROM in C24	Register/read/delete operations are executed with Engineering tool.	The data for initialization, which is stored in the flash ROM of C24 before shipment, cannot be deleted. Designated with "\\" to a field to register "\" when Engineering tool is used to register the modem initialization data. (Example) To register the data of \Q2: \\Q2
CPU module built-in memory	Register/read/delete data with Engineering tool.	—
SD memory card for the CPU module	Register/read/delete data with Engineering tool.	—
Buffer memory of C24	Designate the applicable area corresponding to the registration No. 8001H to 801FH, and write (register)/read the data for initialization.	• For the buffer memory address for user frames and data for initialization, refer to the following section. (☞ Page 550 Buffer memory list)

■Setting with Engineering tool

Set with the module extended parameters of C24.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ Module Name ⇨ [Module Extended Parameter] ⇨ [Modem initialization data]

■For the buffer memory of the C24

- Designate the applicable area corresponding to the registration No., 8001H to 801FH in respect to the area for user frame registration (Un\G6912 to 8128), and write (register)/read the data for initialization.
  When deleting the data for initialization, write "0" to the registration data byte count designation area of the applicable number.
- The following shows the overview of buffer memory that are used for writing/reading/deleting processing for data initialization and the setting value of each area.
  For more details, refer to the following section. (Refer to it by substituting the user frame to data for initialization.)
  (☞ Page 282 CONTENTS AND REGISTRATION OF USER FRAMES FOR DATA COMMUNICATION)

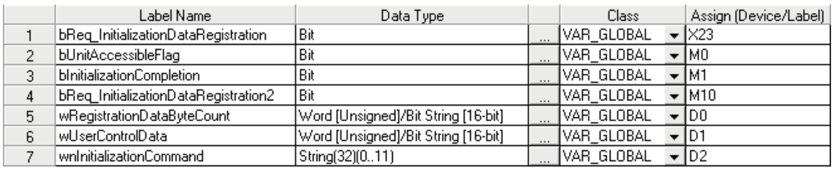
Address		Name		Designated/stored value	Designation required (○)/ not required (×)		
Hexadecimal	Decimal				Write	Read	Delete
1B00H	6912	Registration No. 8001H	Registration data byte count designation	0: When deleting 1 to 78: Number of registered data bytes (Only for the initialization command section)	○	× (Read processing not required)	○
1B01H	6913		User control data	Any data used by the user to control data to be registered (manufacturer code, control number, etc.)			×
1B02H to 1B28H	6914 to 6952		Initialization command	Data code for the initialization command for registration			
1B29H	6953	Registration No. 8002H	Registration data byte count designation	0: When deleting 1 to 78: Number of registered data bytes (Only for the initialization command section)	○	× (Read processing not required)	○
1B2AH	6954		User control data	Any data used by the user to manage the registration data (manufacturer code, control number, etc.)			×
1B2BH to 1B51H	6955 to 6993		Initialization command	Data code for the initialization command for registration			
⋮							
1FCEH	8142	Registration No. 801FH	Registration data byte count designation	0: When deleting 1 to 80: Number of registered data bytes (Only for the initialization command section)	○	× (Read processing not required)	○
1FCFH	8143		User control data	Any data used by the user to manage the registration data (manufacturer code, control number, etc.)			×
1FD0H to 1FF6H	8144 to 8182		Initialization command	Data code for the initialization command for registration			

■Program example

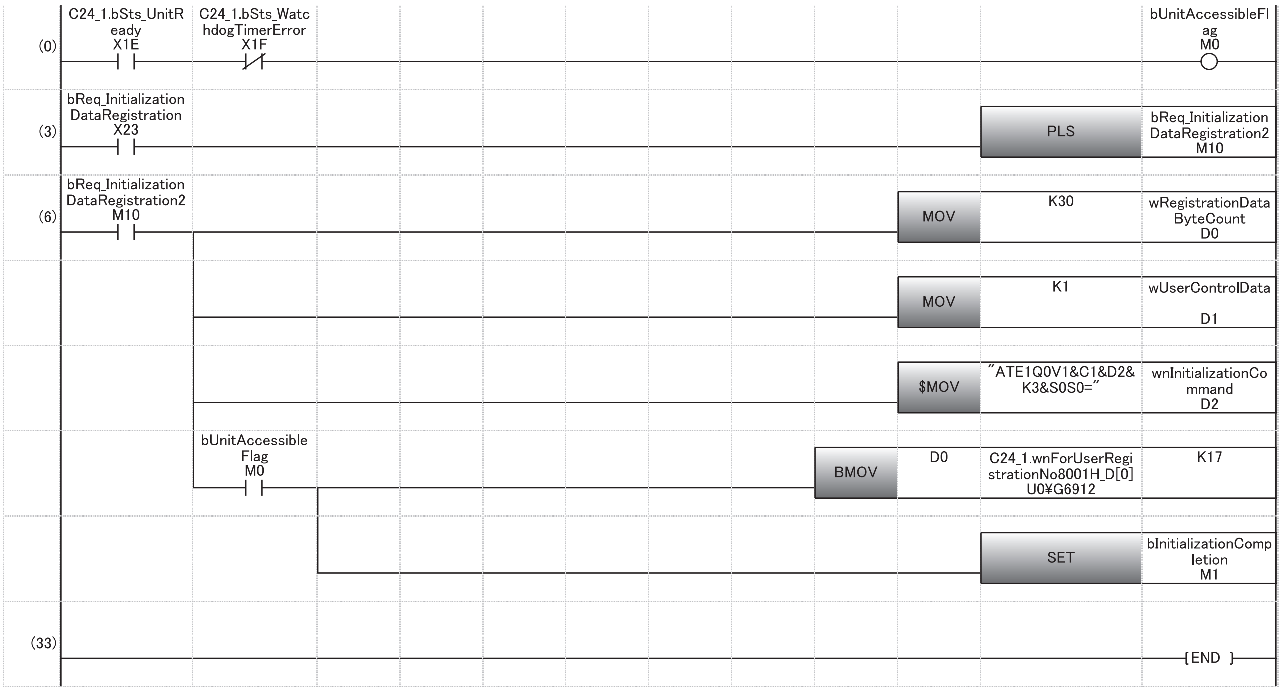
The following shows an example of a program used to write (register) data for initialization.

Writing data for initialization to the area of which registration number 8001H

Category	Label name	Description	Device
Module label	C24_1.bSts_UnitReady	Module READY	X1E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X1F
	C24_1.wnForUserRegistrationNo8001H_D[0]	For user registration number 8001H	U0\G6912
Label to be defined	Define the global label as shown below.		



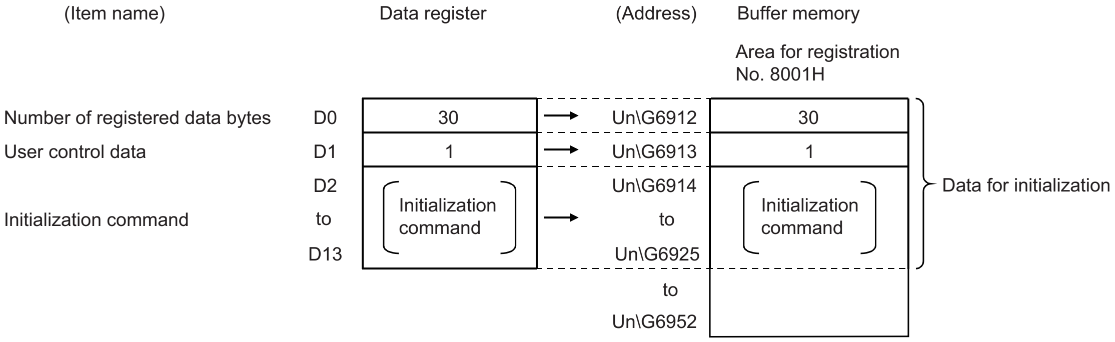
	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_InitializationDataRegistration	Bit	...	VAR_GLOBAL	X23
2	bUnitAccessibleFlag	Bit	...	VAR_GLOBAL	M0
3	bInitializationCompletion	Bit	...	VAR_GLOBAL	M1
4	bReq_InitializationDataRegistration2	Bit	...	VAR_GLOBAL	M10
5	wRegistrationDataByteCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
6	wUserControlData	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
7	wnInitializationCommand	String(32)(0..11)	...	VAR_GLOBAL	D2


(0)
C24_1.bSts_UnitReady X1E
C24_1.bSts_WatchdogTimerError X1F
bUnitAccessibleFlag M0
(3)
bReq_InitializationDataRegistration X23
PLS
bReq_InitializationDataRegistration2 M10
(6)
bReq_InitializationDataRegistration2 M10
MOV
K30
wRegistrationDataByteCount D0
MOV
K1
wUserControlData D1
$MOV
"ATE1Q0V1&C1&D2&K3&S0S0="
wnInitializationCommand D2
bUnitAccessibleFlag M0
BMOV
D0
C24_1.wnForUserRegistrationNo8001H_D[0] U0¥G6912
K17
SET
bInitializationCompletion M1
(33)
END

(0) Turn the READY flag ON.

(3) Designate initialization data registration.

(6) Set the registration data byte count.
Set the user control data (control number).
Set the AT command for initialization to buffer memory.
Set the initialization flag.



Register/read/delete data for connection

This section explains the registration/reading/deletion of data for connection such as the telephone number of the target device that are used for communicating data with target devices using the modem function.

Details of the data for connection

The following shows the registration destination and type of the data for connection, registration number, and registration count.

The maximum number of bytes for data that can be registered as one data for connection is 80 bytes. (Connection data portion: 44 bytes, free: 36 bytes)

Registration destination of data for connection		Types of data for connection that is registered	Registration number of data for connection	Number of registrations	Remarks
CPU module	Built-in memory	Data registered by the user[*1]	3000 to 3029 (BB8H to BD5H)	30	• Register the data for connection after the debug process has been completed. • Registration in this area enables easy replacement of C24.
	SD memory card	Data registered by the user[*1]	3000 to 3029 (BB8H to BD5H)	30	• Register the data for connection after the debug process has been completed. • Registration in this area enables easy replacement of C24.
C24	Flash ROM	Data registered by the user[*1]	3000 to 3029 (BB8H to BD5H)	30	• Register the data for connection after the debug process has been completed.
	Buffer memory	Data registered by the user	-32767 to -32737 (8001H to 801FH)	31	• Register the data for connection after the debug process has been completed.

[*1] For the priority of the registered data, refer to the following section.
☞ Page 434 PARAMETER REGISTRATION TO CPU MODULE

Considerations for registration of data for connection

- The registration status of the data for connection stored in the flash ROM can be checked in the buffer memory (Un\547 to 549).
  For initial registration, register by designating an unregistered number.
  When designating a registration number that has already been registered, first delete the registration data for the designated number prior to registration.

Method to register/read/delete data for connection

■Setting with Engineering tool

Set the data using the C24 module extended parameter.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ Module Name ⇨ [Module Extended Parameter] ⇨ [Modem connection data]

●: Required item ○: Configurable ×: Setting not required

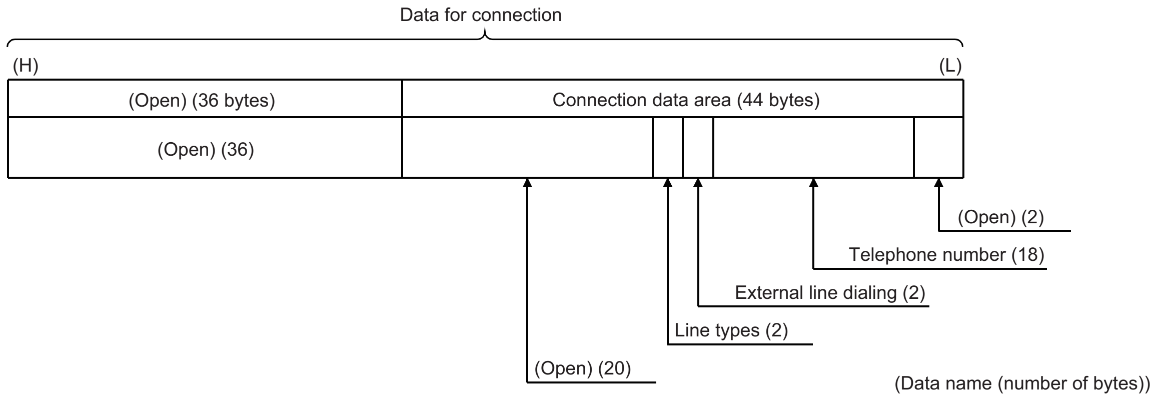
Setting item		Data communication (MC, nonprocedural, bidirectional)
Modem connection data 1 to 30	Telephone number	●
	External line outgoing number	○
	Line type	○
	Wait time for message transmission (unit: s)	×
	Message	×
	Comment	○

■For the buffer memory of C24

- Designate the applicable area corresponding to the registration No., 8001H to 801FH in respect to the area for user frame registration (Un\G6912 to 8128), and write (register)/read the data for connection.
  When deleting the data for connection, write "0" to the registration data byte count designation area of the applicable number.
- The following shows the overview of buffer memory that are used for writing/reading/deleting processing for data connection and the setting value of each area.
  For more details, refer to the following section. (Refer to it by substituting the user frame to data for connection.)
  (☞ Page 282 CONTENTS AND REGISTRATION OF USER FRAMES FOR DATA COMMUNICATION)

Address		Name		Designated/stored value	Designation required (○)/not required (×)		
Hexadecimal	Decimal				Write	Read	Delete
1B00H	6912	Registration No. 8001H	Registration data byte count designation	0: When deleting 80: Number of registered data bytes	○	× (Read processing not required)	○
1B01H to 1B28H	6913 to 6952		Data for connection	Data for connection that is registered			×
1B29H	6953	Registration No. 8002H	Registration data byte count designation	0: When deleting 80: Number of registered data bytes	○	× (Read processing not required)	○
1B2AH to 1B51H	6954 to 6993		Data for connection	Data for connection that is registered			×
⋮							
1FCEH	8142	Registration No. 801FH	Registration data byte count designation	0: When deleting 80: Number of registered data bytes	○	× (Read processing not required)	○
1FCFH to 1FF6H	8143 to 8182		Data for connection	Data for connection that is registered			×

- The following shows the data arrangement of the area for data for connection (area corresponding to registration numbers 8001H to 801FH) in the buffer memory that is used to register, read and delete data for connection, as well as the designated values and stored values.



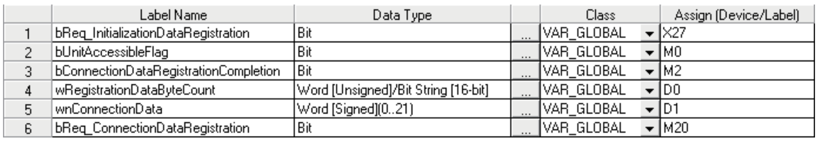
●Connection data portion (44 bytes)

Data name	Designated/stored value and contents	Number of bytes	Data type
(Open)	Designate "0".	2	Binary
Telephone number	• Designate the telephone number of the target device side to establish the line connection to the target device side with which data are communicated. • When the phone number is less than 18 characters, a space (code:20H) must be entered for the remainder.	18	ASCII
External line outgoing number	Designate the external line outgoing number on the C24 side to establish the line connection to the target device with which data is communicated. ■External line outgoing number on the C24 side • 0 to 9 • 10(*) • 11(#) ■No external line outgoing number required on the C24 side • 255	2	Binary
Line type	Designate the line type used to perform data communication with the target device. • 0: Pulse • 1 : Tone • 2: ISDN	2	Binary
(Open)	Designate "0".	20	Binary
(Open)	Designate "0".	36	Binary

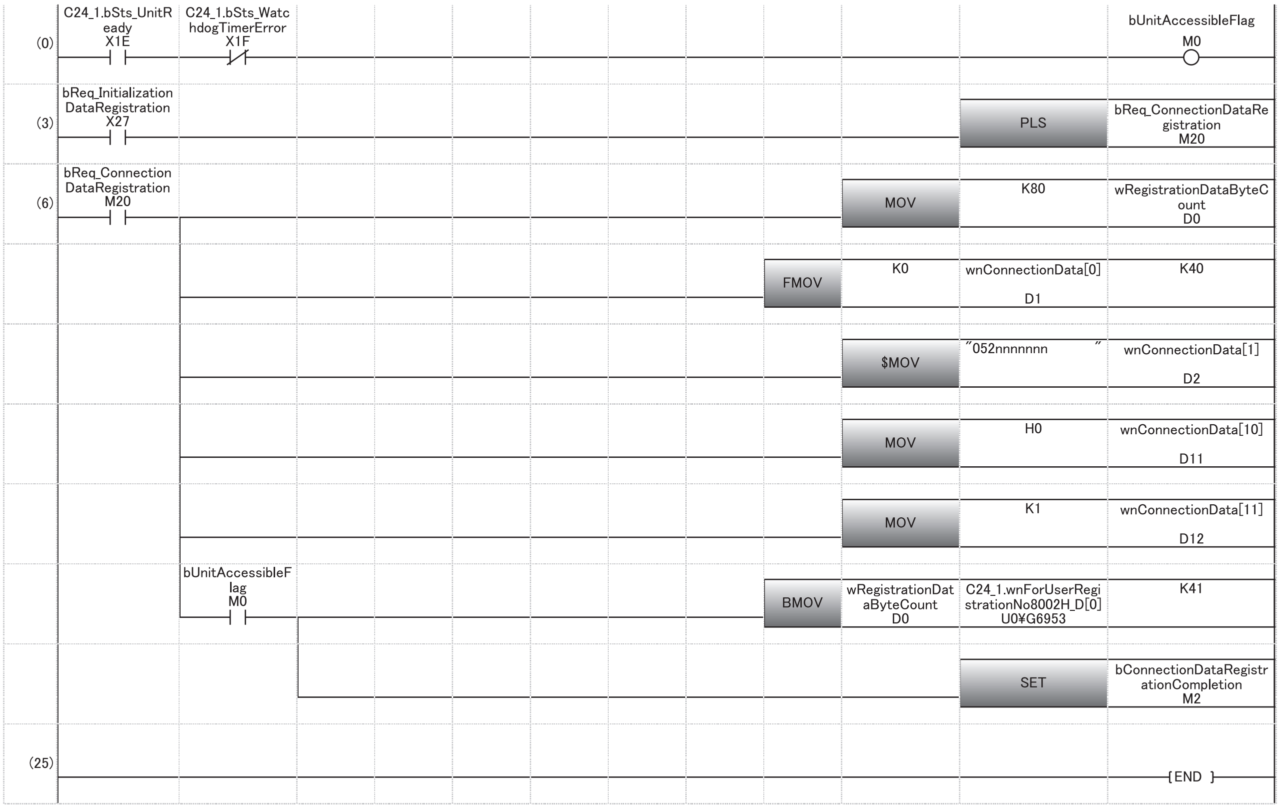
■Program example

An example of a program used for writing data for connection is shown below.

Example of writing data for connection to the registration number 8002H area

Category	Label name	Description	Device
Module label	C24_1.bSts_UnitReady	Module READY	X1E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X1F
	C24_1.wnForUserRegistrationNo8002H_D[0]	For user registration number 8002H	U0\G6953
Label to be defined	Define the global label as shown below. 		

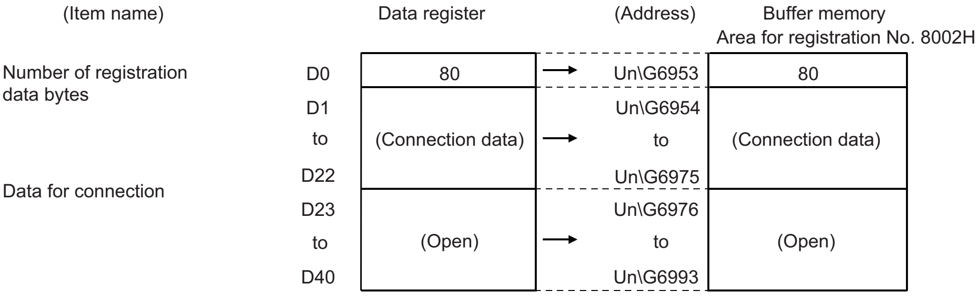
	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_InitializationDataRegistration	Bit	...	VAR_GLOBAL	X27
2	bUnitAccessibleFlag	Bit	...	VAR_GLOBAL	M0
3	bConnectionDataRegistrationCompletion	Bit	...	VAR_GLOBAL	M2
4	wRegistrationDataByteCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
5	wnConnectionData	Word [Signed](0..21)	...	VAR_GLOBAL	D1
6	bReq_ConnectionDataRegistration	Bit	...	VAR_GLOBAL	M20


(0)
C24_1.bSts_UnitReady
X1E
C24_1.bSts_WatchdogTimerError
X1F
bUnitAccessibleFlag
M0
(3)
bReq_InitializationDataRegistration
X27
PLS
bReq_ConnectionDataRegistration
M20
(6)
bReq_ConnectionDataRegistration
M20
MOV
K80
wRegistrationDataByteCount
D0
FMOV
K0
wnConnectionData[0]
D1
K40
$MOV
"052nnnnnnn "
wnConnectionData[1]
D2
MOV
H0
wnConnectionData[10]
D11
MOV
K1
wnConnectionData[11]
D12
bUnitAccessibleFlag
M0
BMOV
wRegistrationDataByteCount
D0
C24_1.wnForUserRegistrationNo8002H_D[0]
U0¥G6953
K41
SET
bConnectionDataRegistrationCompletion
M2
(25)
END

(0) Turn the READY flag ON.

(3) Issue the registration command of data for connection.

(6) Set the number of registered data bytes.
Clear the storage device of data for connection.
Set the connection target telephone number.
Set the external line outgoing number (send 0).
Set the line type to 'Tone'.
Write the data for connection to buffer memory.
Turn the data for connection registration completion flag ON.



(Item name)		Data register		(Address)	Buffer memory Area for registration No. 8002H
Number of registration data bytes	D0	80	→	Un\G6953	80
Data for connection	D1 to D22	(Connection data)	→	Un\G6954 to Un\G6975	(Connection data)
	D23 to D40	(Open)	→	Un\G6976 to Un\G6993	(Open)

Initialization of modem/TA (terminal adapter)

This section explains the initialization of the modem/TA connected to C24, used for communicating data with the target device with the modem function.

Requirements for initialization

Complete the following settings and registration in advance.

- C24 initial settings (☞ Page 206 Initial settings of Serial communication module)
- Registration of data for initialization when initializing with the data for initialization set by the user (☞ Page 209 Register/read/delete data for initialization)

Initialization and connection can be performed at the same time by performing the connection processing by designating data for initialization and connection. (☞ Page 222 Line connection)

Registering data for initialization using Engineering tool

Registration of the number of data for initialization to initialize the modem connected to the C24 side is performed with Engineering tool.

The following explains the number designated by the "Initialization data number" item and the related buffer memory when "0" is designated as the number of data for initialization.

Used buffer memory				The number of data for initialization used and buffer memory designated value	
Name		Address (CH1/CH2)			
		Hexadecimal	Decimal	When number of data used = 1	When number of data used = 2 or more
Data No. for initialization designation		34H	52	7D0H to 801FH :Registration number of data for initialization[*3]	0H
Sending user frame number		B6H/156H	182/342	(Not used)	(During initialization, the registration number of data currently being transmitted is stored.)
CR/LF output designation		B7H/157H	183/343		0 (default value)
Output start pointer designation[*1]		B8H/158H	184/344		1 to 100[*1]
Output quantity designation[*2]		B9H/159H	185/345		1 to 100[*2]
Send frame number designation	1st	BAH/15AH	186/346		7D0H to 801FH :Registration number of data for initialization[*3]
	2nd	BBH/15BH	187/347		
	to	to	to		
	100th	11DH/1BDH	285/445		

*1 'Output start pointer designation' (Un\G184/344)
The head position (nth) in the send frame number designation area to write the registration number of data for initialization to be transmitted is designated.
1 (transmitting from the 1th) to 100 (transmitting from the 100th)

*2 'Output quantity designation' (Un\G185/345)
How many data for initialization is transmitted form the designated position in the output start pointer designation area is designated.
1 (transmitting 1) to 100 (transmitting 100)

*3 The registration number of data for initialization to be used is designated.
·7D0H to 7DDH (2000 to 2013) : Data registered by the operating system
·9C4H to 9E1H (2500 to 2529) : Data registered in the flash ROM by the user
·8001H to 801FH (–32767 to –32737): Data registered in the buffer memory by the user

Considerations for modem/TA initialization

If the DSR signal from the modem/ TA turns OFF when modem/TA initialization is completed (X10 turns ON), C24 executes initialization processing automatically in accordance with the following.

- When the auto modem initialization is specified
  Modem/TA initialization processing is executed in the initialization/connection timeout time interval while the DSR signal is OFF regardless of the ON/OFF status of the 'Modem initialization request' (Y10).
- When the auto modem initialization is not specified
  The modem/TA initialization processing is executed when the DSR signal restarts regardless of the ON/OFF status of the 'Modem initialization request' (Y10).

Automatic modem/TA initialization (Setting with Engineering tool)

By configuring the settings for automatic modem initialization with Engineering tool, modem initialization is executed automatically when C24 starts up.

The 'Modem initialization completion' (X10) turns ON.

■Settings for auto initialization of the modem / TA

Select "Automatically initialize" for the auto modem initialization designation of Engineering tool.

■Considerations for initializing the modem / TA automatically

● If the modem's initialization processing is completed abnormally, C24 executes the following processing.

- C24 does not turn ON the 'Initialization/connection abnormal completion' (X13).
- C24 stores an error code in the 'Modem function error code' (Un\G545).
- C24 retries the modem initialization automatically by using the initialization/connection timeout time stored in the 'Initialization/connection timeout designation' (Un\G50), and repeats it until the initialization is completed normally.
  When modem initialization is not completed normally, perform the following, and restart the station where C24 is installed.
  Check the set initialization data number.
  Check the registered contents of the initialization data corresponding to the set initialization data number. (If any abnormal, correct them and register them again.) Check if the power of the modem is turned ON.

● When the 'Modem disconnection request' (Y12) is used from the C24 side, the 'Modem initialization completion' (X10) turns OFF together with the 'Connection' (X12).

When connecting to the line again, start from the modem initialization.

Point

When the line to C24 is disconnected from the target device side, the C24's 'Modem initialization completion' (X10) does not turn OFF.

When desiring to disable reception by the C24 side's modem, execute line disconnect by the 'Modem disconnection request' (Y12).

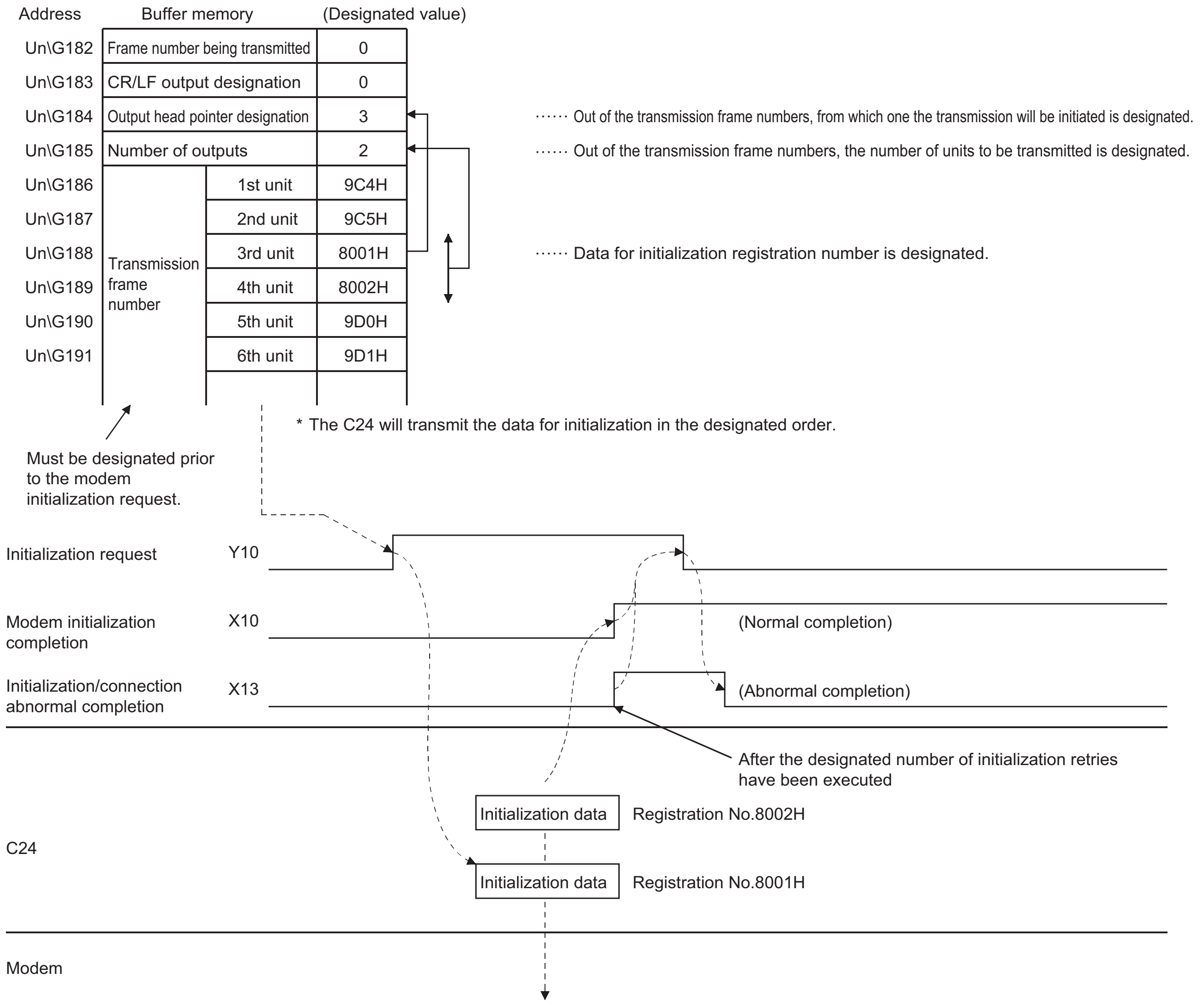
Modem/TA initialization with a program

■I/O signals used for initialization

The 'Modem initialization request' (Y10), 'Modem initialization completion' (X10) and 'Initialization/connection abnormal completion' (X13) are used.

Ex.

When initializing the modem connected to CH1 of C24 using two set of data for initialization (registration numbers 8001H and 8002H) that are registered in the buffer memory



Address	Buffer memory		(Designated value)
Un\G182	Frame number being transmitted		0
Un\G183	CR/LF output designation		0
Un\G184	Output head pointer designation		3
Un\G185	Number of outputs		2
Un\G186	Transmission frame number	1st unit	9C4H
Un\G187		2nd unit	9C5H
Un\G188		3rd unit	8001H
Un\G189		4th unit	8002H
Un\G190		5th unit	9D0H
Un\G191		6th unit	9D1H

······ Out of the transmission frame numbers, from which one the transmission will be initiated is designated.

······ Out of the transmission frame numbers, the number of units to be transmitted is designated.

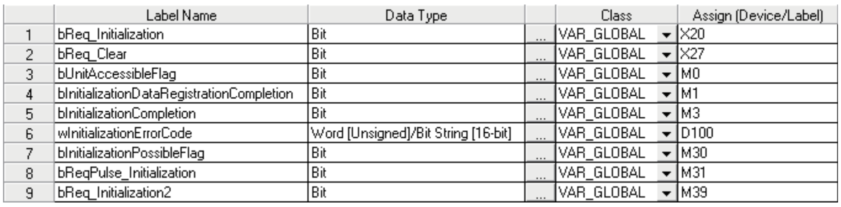
······ Data for initialization registration number is designated.

Must be designated prior to the modem initialization request.

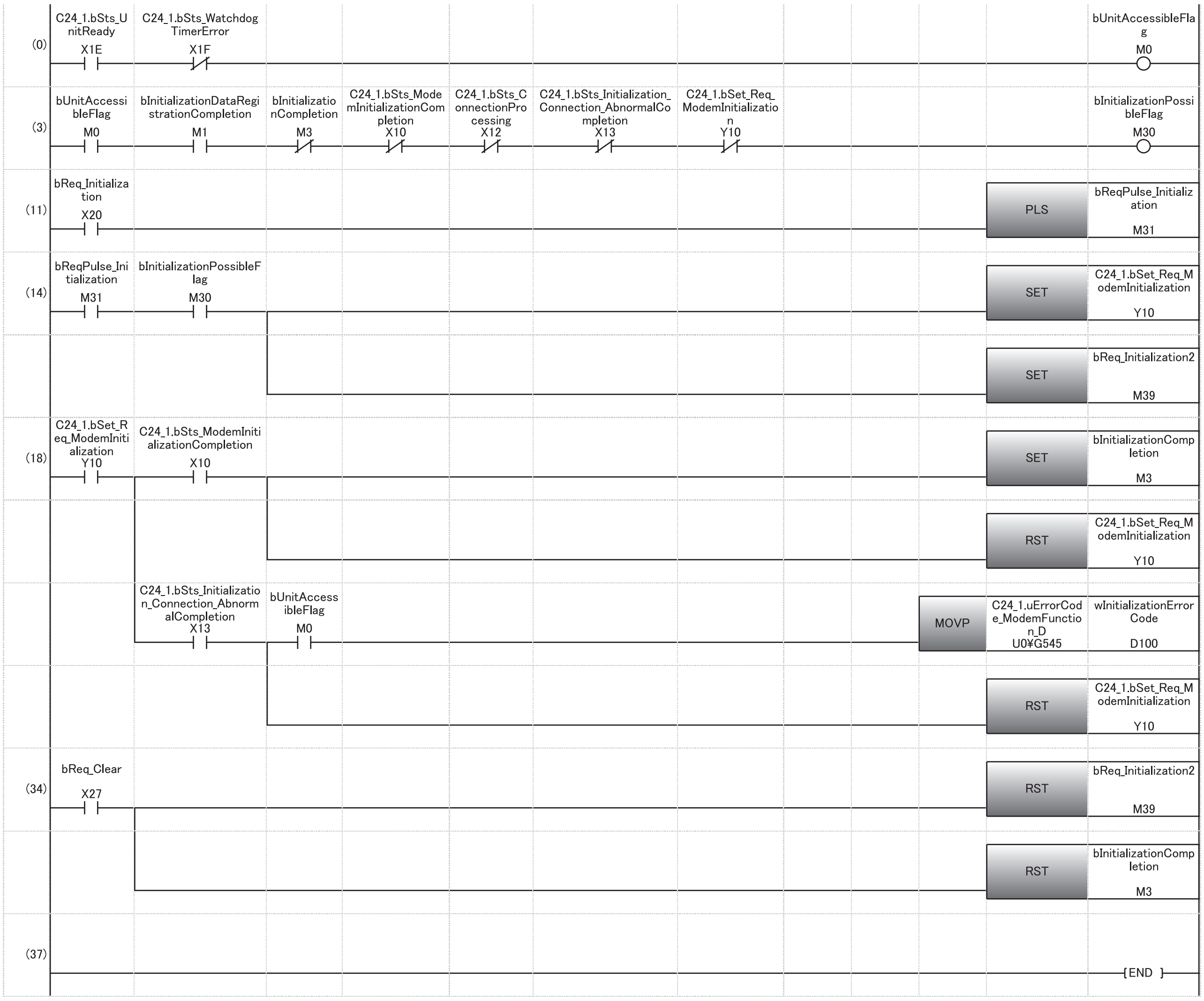
* The C24 will transmit the data for initialization in the designated order.

■Program example

An example of the modem/TA initialization program on the C24 side by the CPU module is shown below.
(When the data for initialization has been registered with Engineering tool or from the CPU module)

Category	Label name	Description	Device
Module label	C24_1.bSts_ModemInitializationCompletion	Modem initialization completion	X10
	C24_1.bSts_ConnectionProcessing	Connection	X12
	C24_1.bSts_Initialization_Connection_AbnormalCompletion	Initialization/connection abnormal completion	X13
	C24_1.bSts_UnitReady	Module READY	X1E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X1F
	C24_1.bSet_Req_ModemInitialization	Modem initialization request	Y10
	C24_1.uErrorCode_ModemFunction_D	Modern function error code	U0\G545
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class		Assign (Device/Label)
1	bReq_Initialization	Bit	...	VAR_GLOBAL	▼	X20
2	bReq_Clear	Bit	...	VAR_GLOBAL	▼	X27
3	bUnitAccessibleFlag	Bit	...	VAR_GLOBAL	▼	M0
4	bInitializationDataRegistrationCompletion	Bit	...	VAR_GLOBAL	▼	M1
5	bInitializationCompletion	Bit	...	VAR_GLOBAL	▼	M3
6	wInitializationErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	▼	D100
7	bInitializationPossibleFlag	Bit	...	VAR_GLOBAL	▼	M30
8	bReqPulse_Initialization	Bit	...	VAR_GLOBAL	▼	M31
9	bReq_Initialization2	Bit	...	VAR_GLOBAL	▼	M39


(0)
C24_1.bSts_UnitReady
X1E
C24_1.bSts_WatchdogTimerError
X1F
bUnitAccessibleFlag
M0
(3)
bUnitAccessibleFlag
M0
bInitializationDataRegistrationCompletion
M1
bInitializationCompletion
M3
C24_1.bSts_ModemInitializationCompletion
X10
C24_1.bSts_ConnectionProcessing
X12
C24_1.bSts_Initialization_Connection_AbnormalCompletion
X13
C24_1.bSet_Req_ModemInitialization
Y10
bInitializationPossibleFlag
M30
(11)
bReq_Initialization
X20
PLS
bReqPulse_Initialization
M31
(14)
bReqPulse_Initialization
M31
bInitializationPossibleFlag
M30
SET
C24_1.bSet_Req_ModemInitialization
Y10
SET
bReq_Initialization2
M39
(18)
C24_1.bSet_Req_ModemInitialization
Y10
C24_1.bSts_ModemInitializationCompletion
X10
SET
bInitializationCompletion
M3
RST
C24_1.bSet_Req_ModemInitialization
Y10
C24_1.bSts_Initialization_Connection_AbnormalCompletion
X13
bUnitAccessibleFlag
M0
MOVP
C24_1.uErrorCode_ModemFunction_D
U0¥G545
wInitializationErrorCode
D100
RST
C24_1.bSet_Req_ModemInitialization
Y10
(34)
bReq_Clear
X27
RST
bReq_Initialization2
M39
RST
bInitializationCompletion
M3
(37)
END

(0) Turn the READY flag ON.

(3) Turn the modem initialization applicable flag ON.

(11) Issue the modem initialization command.

(14) Set the initialization request signal.
Set the initialization request flag.

(18) Set the initialization completion flag.
Read the error code at initialization abnormal completion.

(34) Reset the flag with the clear command.

Line connection

The following explains the connection (dialing) with target devices for data communication with target devices using the modem function.

Requirements for connection

Complete the following settings and registration in advance.

- The initial settings for C24 (☞ Page 206 Initial settings of Serial communication module)
- The registration of the data for initialization (☞ Page 209 Register/read/delete data for initialization)
- The registration of data for connection (☞ Page 213 Register/read/delete data for connection)
- The initialization of the modem/TA connected to C24 side (☞ Page 217 Initialization of modem/TA (terminal adapter))

In addition, both the initialization and line connection can be conducted simultaneously by designating the data for initialization and data for connection to perform connection processing.

For setting of data for initialization to perform initialization and line connection simultaneously, refer to the following sections. It is not explained in this section.

☞ Page 206 Initial settings of Serial communication module

☞ Page 217 Initialization of modem/TA (terminal adapter)

Registering the data for connection using Engineering tool

The number for the data for connection that is used for line connection in order to perform data communication with the target device is registered with Engineering tool.

■When line connecting from the C24 side

Designate the registration number of the data for connection to be used for "Data No. for connection designation".

🖱 Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ Module Name ⇨ [Module Parameter] ⇨ "Application Settings" ⇨ "Modem function setting"

- BB8H to BD5H (3000 to 3029) : Data registered in the flash ROM by the user
- 8001H to 801FH (–32767 to –32737): Data registered in the buffer memory by the user

■When line connecting from the target device

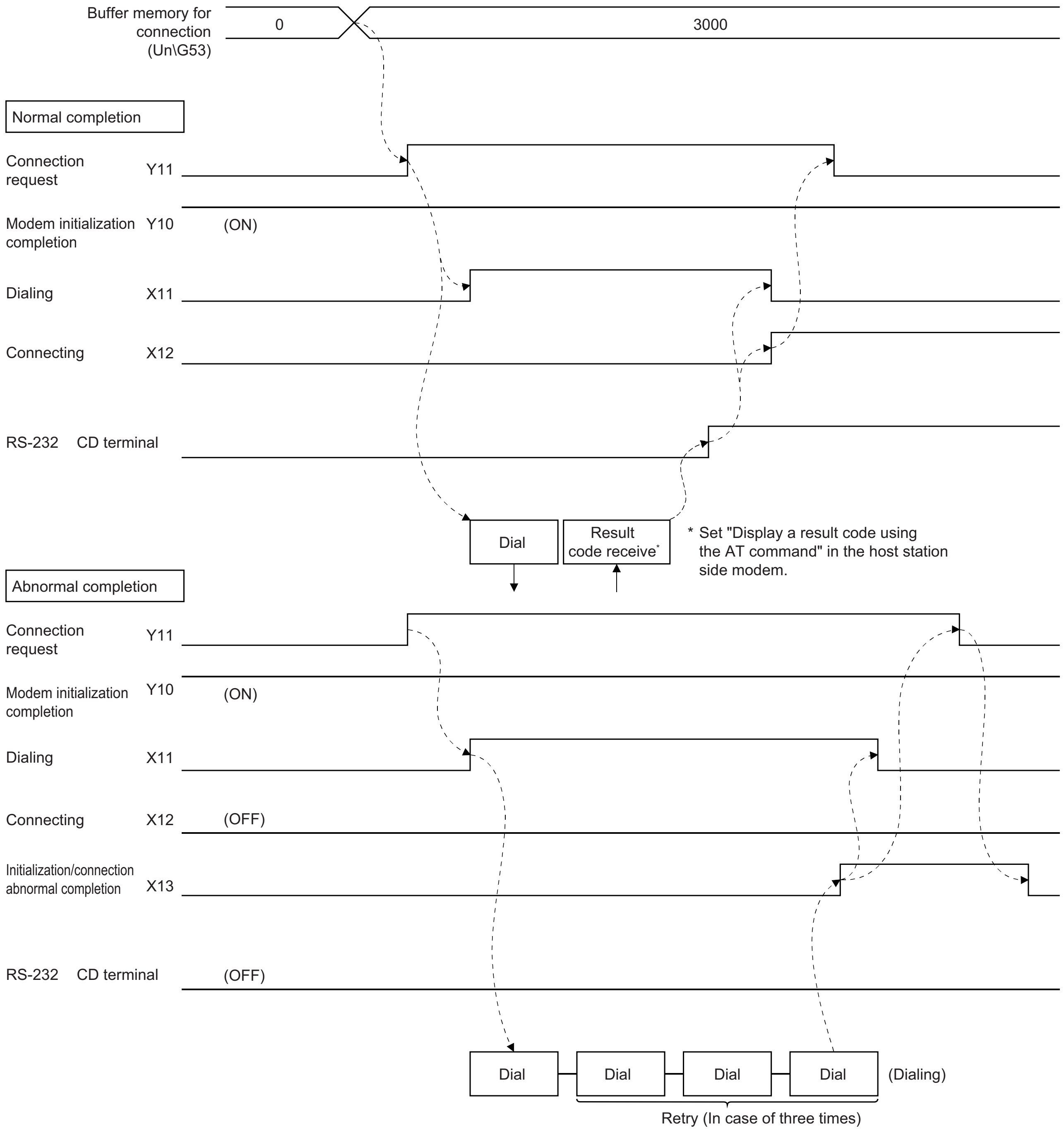
Since the line connection processing is not necessary on the C24 side, registration of data for connection to connect the line and number setting of data for connection are not required.

Input/output signals used for line connection

'Connection request' (Y11), 'Dialing' (X11), 'Connection' (X12) and 'Initialization/connection abnormal completion' (X13) are used.

Ex.

When performing the line connection only from the C24 side after the completion of initialization

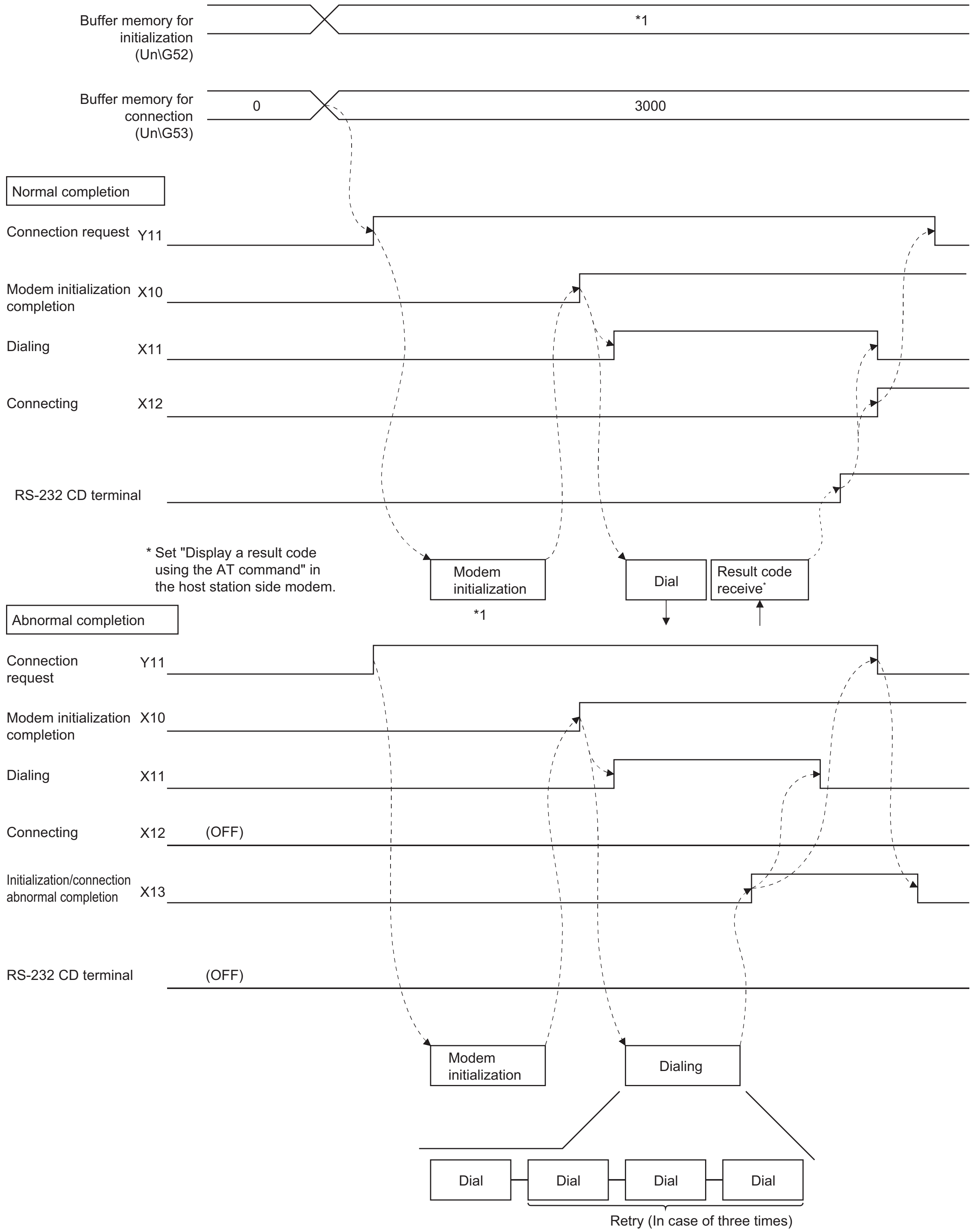


* Set "Display a result code using the AT command" in the host station side modem.

* Connection channel and retry operation are performed using the buffer memory setting.

Ex.

When performing the initialization and the line connection from the C24 side simultaneously

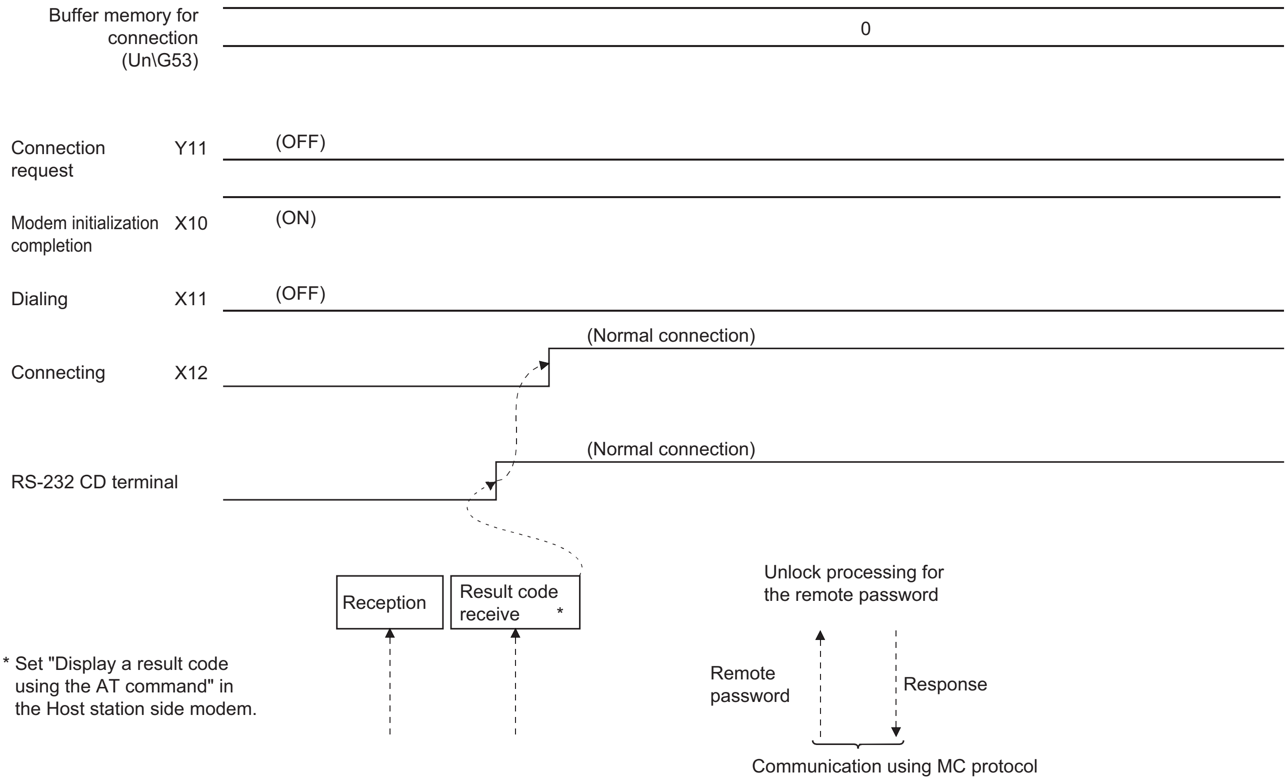

Buffer memory for initialization (Un\G52)
*1
Buffer memory for connection (Un\G53)
0
3000
Normal completion
Connection request Y11
Modem initialization completion X10
Dialing X11
Connecting X12
RS-232 CD terminal
* Set "Display a result code using the AT command" in the host station side modem.
Modem initialization
*1
Dial
Result code receive*
Abnormal completion
Connection request Y11
Modem initialization completion X10
Dialing X11
Connecting X12 (OFF)
Initialization/connection abnormal completion X13
RS-232 CD terminal (OFF)
Modem initialization
Dialing
Dial
Dial
Dial
Dial
Retry (In case of three times)

* Connection channel and retry operation are performed using the buffer memory setting.

*1 ☞Page 217 Initialization of modem/TA (terminal adapter)

Ex.

When performing the line connection from the target device after the completion of initialization



* Set "Display a result code using the AT command" in the Host station side modem.

Point

- The connection channel on the C24 side is set in the module parameter.
- Abnormal processing when the target device initiate the line connection is left entirely to the target device. There is no method on the C24 side to check a line connection error occurrence at the target device.
- When a remote password check is performed for the CPU module, normal completion of the unlock processing enables data communication thereafter.

Considerations during the line connection

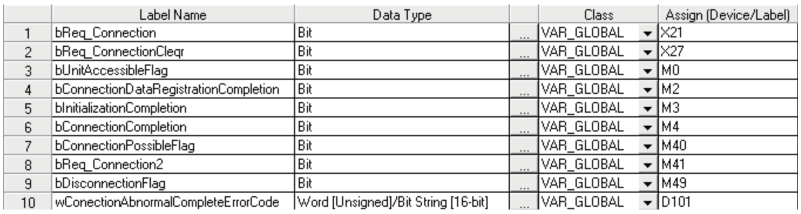
- Prior to data communication with target devices, determine when and which station will perform the line connection (dialing) and line disconnection processing to the target device.
- Use the following method to perform the unlock processing for the CPU module remote password from the target device. During communication using MC protocol, transmit the dedicated command to the C24 side.
  For countermeasures against abnormal completion of the unlock processing, refer to the following section.
  (☞ Page 199 When the remote password unlock processing is completed abnormally)
- When reconnecting the line after disconnection, allow several seconds for the modem before turning ON the 'Connection request' (Y11).
  If it (Y11) is turned ON immediately after line disconnection, the modem may not accept the first connection request, resulting in connection failure, and may require to wait the retry time.

Line connection program

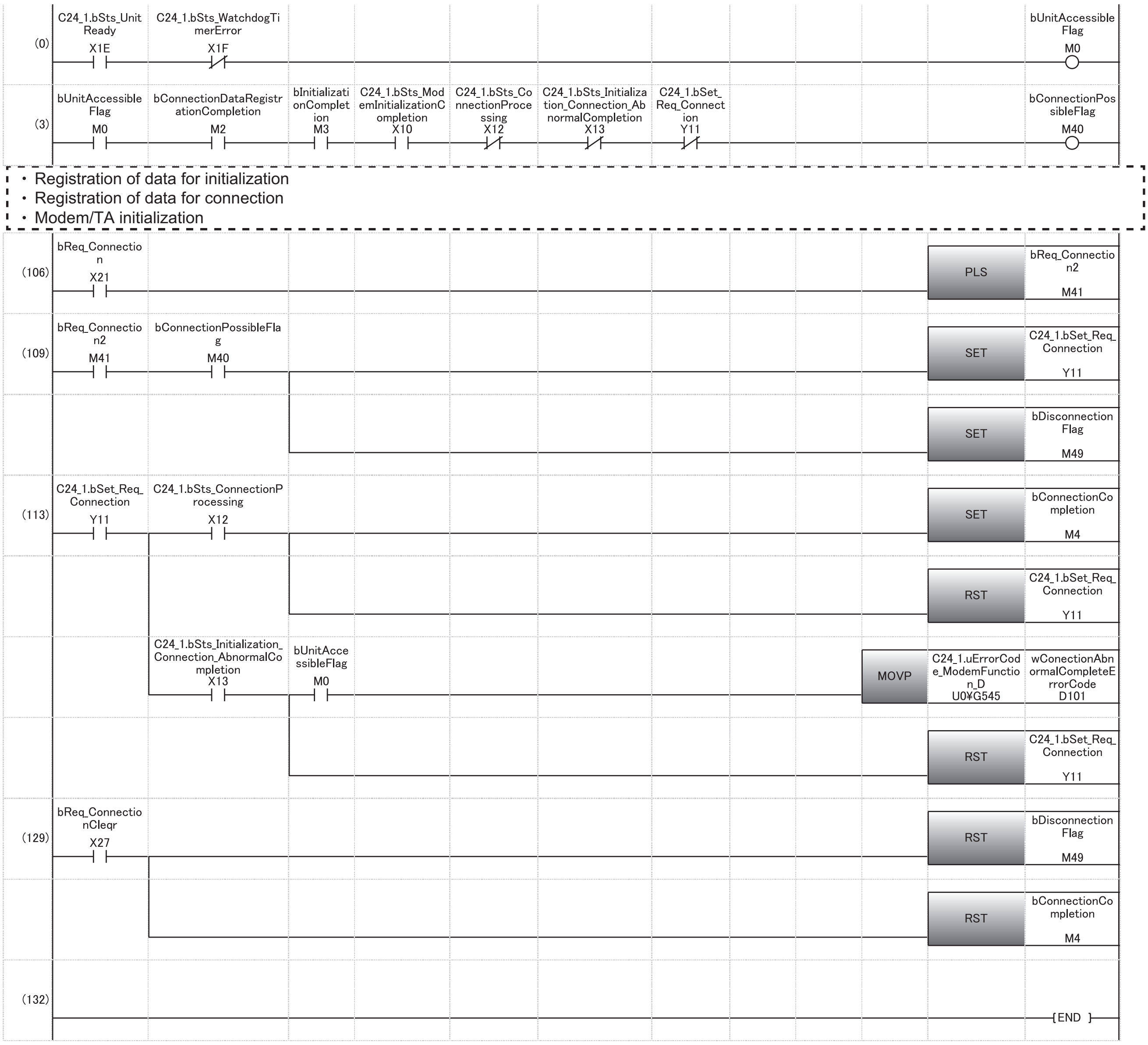
An example of a line connection program is shown below.

■Program example (When establishing the line connection from the C24 side following the completion of initialization)

When the data for connection has been registered with Engineering tool or from the CPU module

Category	Label name	Description	Device
Module label	C24_1.bSts_ModemInitializationCompletion	Modem initialization completion	X10
	C24_1.bSts_ConnectionProcessing	Connection	X12
	C24_1.bSts_Initialization_Connection_AbnormalCompletion	Initialization/connection abnormal completion	X13
	C24_1.bSts_UnitReady	Module READY	X1E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X1F
	C24_1.bSet_Req_ModemInitialization	Modem initialization request	Y10
	C24_1.bSet_Req_Connection	Connection request	Y11
	C24_1.uErrorCode_ModemFunction_D	Modern function error code	U0\G545
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_Connection	Bit	...	VAR_GLOBAL	X21
2	bReq_ConnectionCleqr	Bit	...	VAR_GLOBAL	X27
3	bUnitAccessibleFlag	Bit	...	VAR_GLOBAL	M0
4	bConnectionDataRegistrationCompletion	Bit	...	VAR_GLOBAL	M2
5	bInitializationCompletion	Bit	...	VAR_GLOBAL	M3
6	bConnectionCompletion	Bit	...	VAR_GLOBAL	M4
7	bConnectionPossibleFlag	Bit	...	VAR_GLOBAL	M40
8	bReq_Connection2	Bit	...	VAR_GLOBAL	M41
9	bDisconnectionFlag	Bit	...	VAR_GLOBAL	M49
10	wConectionAbnormalCompleteErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D101


(0)
C24_1.bSts_UnitReady
X1E
C24_1.bSts_WatchdogTimerError
X1F
bUnitAccessibleFlag
M0
(3)
bUnitAccessibleFlag
M0
bConnectionDataRegistrationCompletion
M2
bInitializationCompletion
M3
C24_1.bSts_ModemInitializationCompletion
X10
C24_1.bSts_ConnectionProcessing
X12
C24_1.bSts_Initialization_Connection_AbnormalCompletion
X13
C24_1.bSet_Req_Connection
Y11
bConnectionPossibleFlag
M40
• Registration of data for initialization
• Registration of data for connection
• Modem/TA initialization
(106)
bReq_Connection
X21
PLS
bReq_Connection2
M41
(109)
bReq_Connection2
M41
bConnectionPossibleFlag
M40
SET
C24_1.bSet_Req_Connection
Y11
SET
bDisconnectionFlag
M49
(113)
C24_1.bSet_Req_Connection
Y11
C24_1.bSts_ConnectionProcessing
X12
SET
bConnectionCompletion
M4
RST
C24_1.bSet_Req_Connection
Y11
C24_1.bSts_Initialization_Connection_AbnormalCompletion
X13
bUnitAccessibleFlag
M0
MOVP
C24_1.uErrorCode_ModemFunction_D
U0¥G545
wConectionAbnormalCompleteErrorCode
D101
RST
C24_1.bSet_Req_Connection
Y11
(129)
bReq_ConnectionCleqr
X27
RST
bDisconnectionFlag
M49
RST
bConnectionCompletion
M4
(132)
END

(0) Turn the READY flag ON.

(3) Turn the line connection available flag ON.

(11) to (14) Register data for initialization.
☞ Page 212 Program example

(41) to (44) Register data for connection.
☞ Page 216 Program example

(75) to (90) Modem/TA initialization
☞ Page 220 Program example

(106) Issue the line connection command.

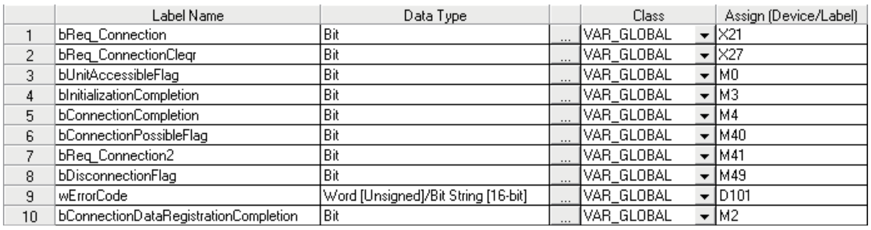
(109) Set the connection request signal.

(113) Set the connection completion flag when the connection signal is turned ON (normal completion).
Read the error code at connection error.

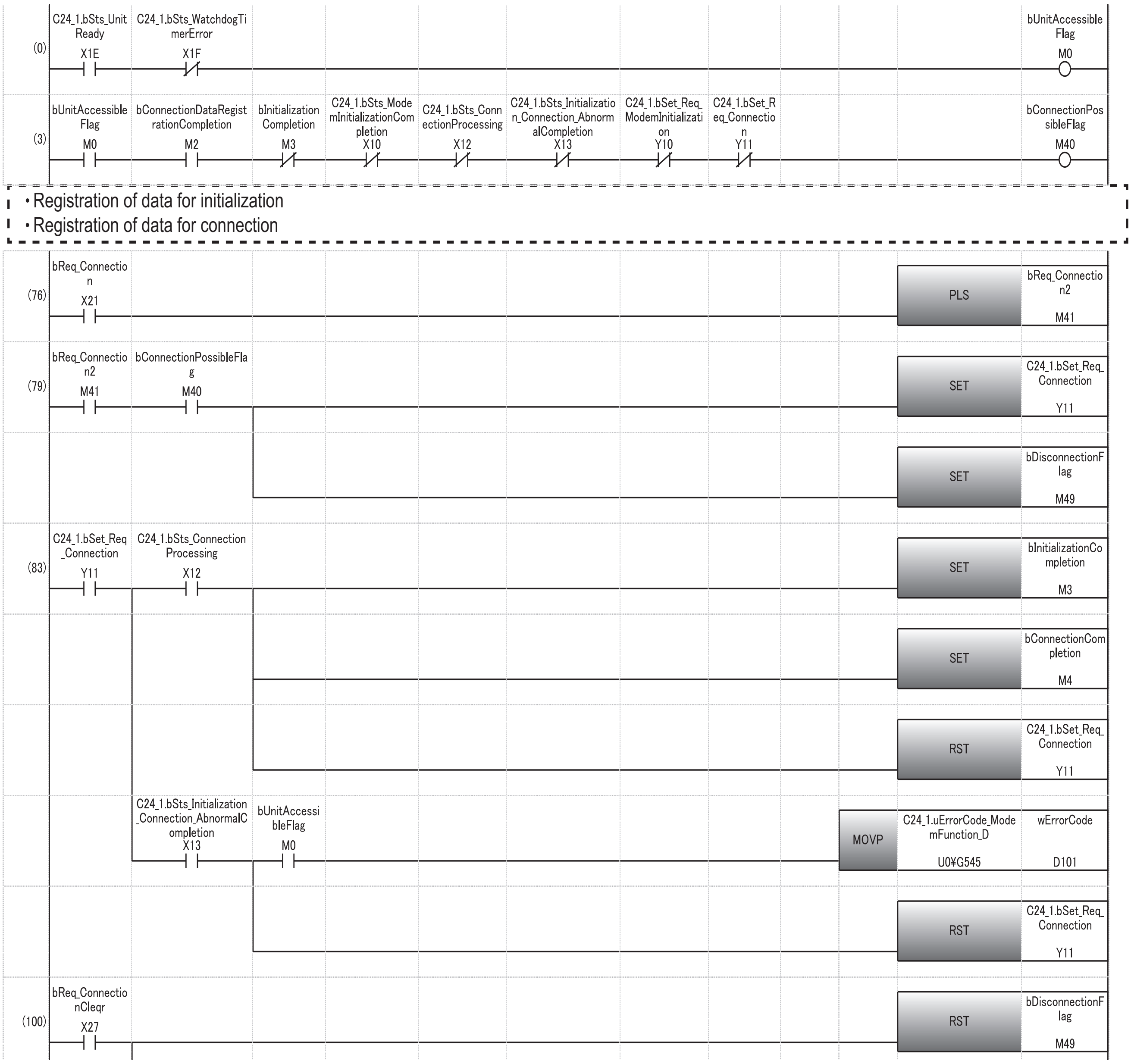
(129) Reset the flag with the clear command.

■Program example (When performing the initialization and the line connection from the C24 side simultaneously)

When the data for initialization and connection have been registered with Engineering tool or from the CPU module

Category	Label name	Description	Device
Module label	C24_1.bSts_ModemInitializationCompletion	Modem initialization completion	X10
	C24_1.bSts_ConnectionProcessing	Connection	X12
	C24_1.bSts_Initialization_Connection_AbnormalCompletion	Initialization/connection abnormal completion	X13
	C24_1.bSts_UnitReady	Module READY	X1E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X1F
	C24_1.bSet_Req_ModemInitialization	Modem initialization request	Y10
	C24_1.bSet_Req_Connection	Connection request	Y11
	C24_1.uErrorCode_ModemFunction_D	Modern function error code	U0\G545
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_Connection	Bit	...	VAR_GLOBAL	X21
2	bReq_ConnectionCleqr	Bit	...	VAR_GLOBAL	X27
3	bUnitAccessibleFlag	Bit	...	VAR_GLOBAL	M0
4	bInitializationCompletion	Bit	...	VAR_GLOBAL	M3
5	bConnectionCompletion	Bit	...	VAR_GLOBAL	M4
6	bConnectionPossibleFlag	Bit	...	VAR_GLOBAL	M40
7	bReq_Connection2	Bit	...	VAR_GLOBAL	M41
8	bDisconnectionFlag	Bit	...	VAR_GLOBAL	M49
9	wErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D101
10	bConnectionDataRegistrationCompletion	Bit	...	VAR_GLOBAL	M2


(0)
C24_1.bSts_UnitReady
X1E
C24_1.bSts_WatchdogTimerError
X1F
bUnitAccessibleFlag
M0
(3)
bUnitAccessibleFlag
M0
bConnectionDataRegistrationCompletion
M2
bInitializationCompletion
M3
C24_1.bSts_ModemInitializationCompletion
X10
C24_1.bSts_ConnectionProcessing
X12
C24_1.bSts_Initialization_Connection_AbnormalCompletion
X13
C24_1.bSet_Req_ModemInitialization
Y10
C24_1.bSet_Req_Connection
Y11
bConnectionPossibleFlag
M40
• Registration of data for initialization
• Registration of data for connection
(76)
bReq_Connection
X21
PLS
bReq_Connection2
M41
(79)
bReq_Connection2
M41
bConnectionPossibleFlag
M40
SET
C24_1.bSet_Req_Connection
Y11
SET
bDisconnectionFlag
M49
(83)
C24_1.bSet_Req_Connection
Y11
C24_1.bSts_ConnectionProcessing
X12
SET
bInitializationCompletion
M3
SET
bConnectionCompletion
M4
RST
C24_1.bSet_Req_Connection
Y11
C24_1.bSts_Initialization_Connection_AbnormalCompletion
X13
bUnitAccessibleFlag
M0
MOVP
C24_1.uErrorCode_ModemFunction_D
U0¥G545
wErrorCode
D101
RST
C24_1.bSet_Req_Connection
Y11
(100)
bReq_ConnectionCleqr
X27
RST
bDisconnectionFlag
M49

(0) Turn the READY flag ON.

(3) Turn the line connection available flag ON.

(12) to (15) Register data for initialization.
☞ Page 212 Program example

(42) to (45) Register data for connection.
☞ Page 216 Program example

(76) Issue the line connection command.

(79) Set the connection request signal.

(83) Set the connection completion flag when the connection signal is turned ON (normal completion).
Read the error code at connection error.

(100) Reset the flag with the clear command.

Point

When the line connection is initiated from the target device, any of the registration, setting or connection processing related to the line connection is not necessary.

Data communication is available when the 'Connection' (X12) turns ON after the completion of the C24 modem/TA initialization.

☞ Page 223 Input/output signals used for line connection

For program example of the modem/TA for initialization, refer to the following section.

☞ Page 219 Modem/TA initialization with a program

Data communication

The following explains the considerations for data communication with target devices using the modem function.

Requirements for data communication

■When communicating data with target devices

Perform the appropriate processing up to line connection or modem/TA initialization, depending on whether or not the line connection is initiated from the C24 side.

After line connection, data communication can be performed using an MC protocol/nonprocedural protocol/bidirectional protocol in full-duplex communication.

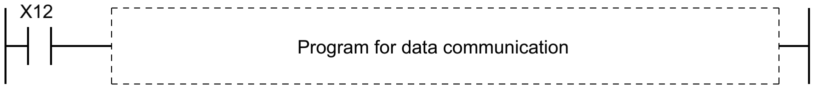
- When line connecting from the C24 side, complete the processing up to line connection beforehand. (☞ Page 222 Line connection)
- When line connecting from the target device, complete the processing up to the initialization of the modem/TA beforehand. (☞ Page 217 Initialization of modem/TA (terminal adapter))

Buffer memory and input/output signals to be used

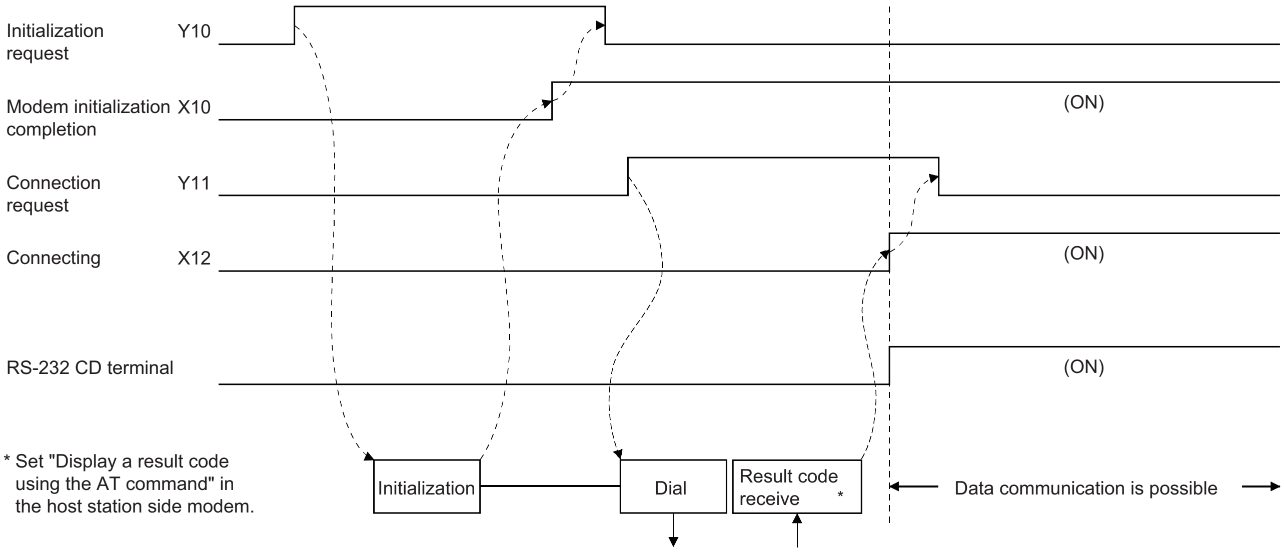
■When communicating data with target devices

Only buffer memory and input/output signals used for data communication (MC protocol/nonprocedural protocol/bidirectional protocol) is used.

Communicate data using the ON of 'Connection' (X12) as the interlock signal.



There is no input/output signal or buffer memory for modem functions used in data communication.

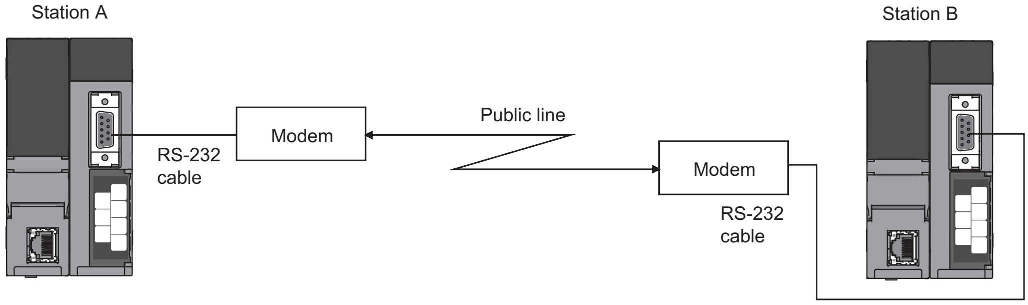


* Set "Display a result code using the AT command" in the host station side modem.

General procedure

This section explains the general procedure for nonprocedural protocol/bidirectional protocol (executed in full-duplex communication) data communication using the modem function between the CPU modules with C24 installed.

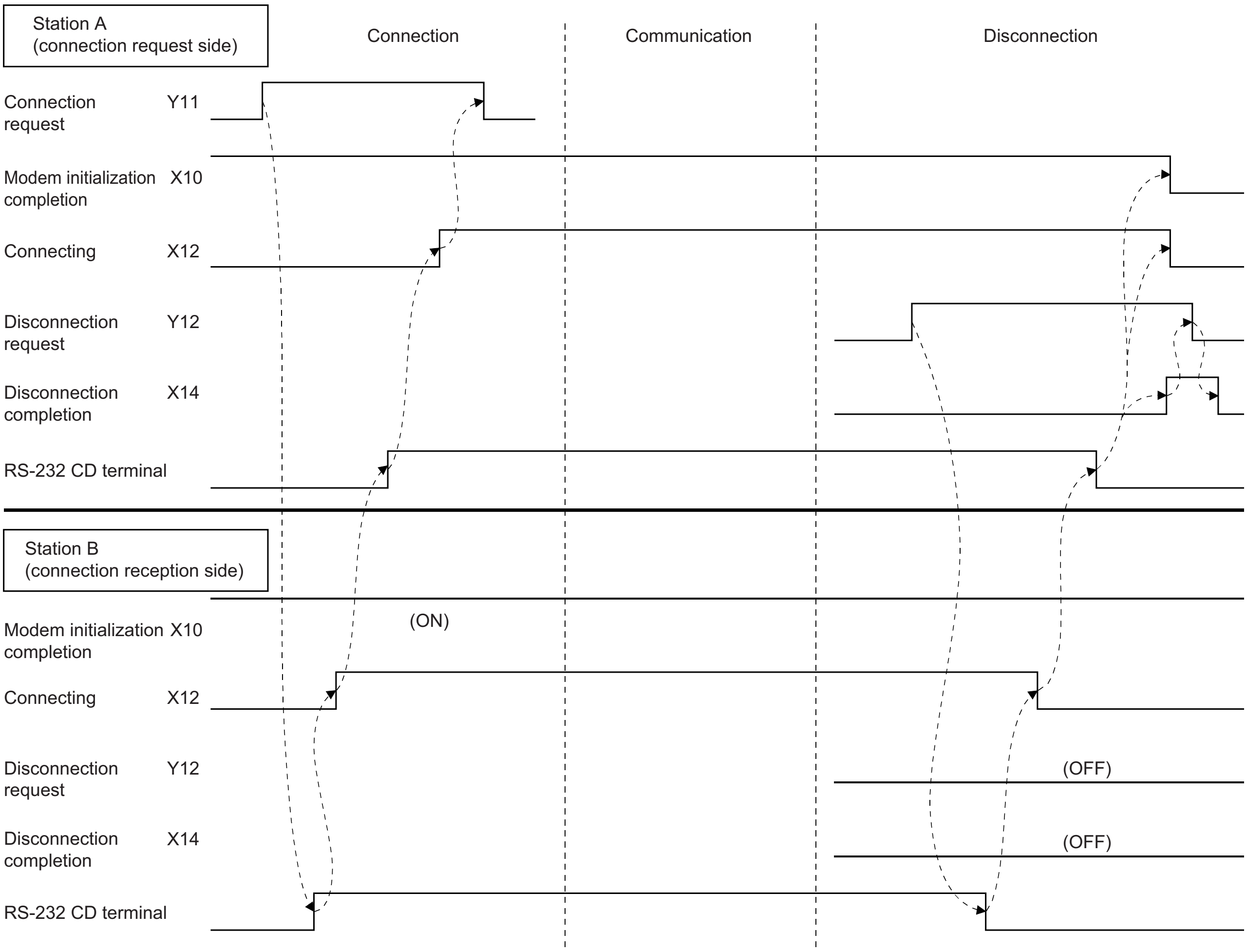
■System configuration



■General procedure

1. Perform initial setting for C24 at both station A and station B.
2. Perform modem/TA initialization on station B.
3. Perform modem/TA initialization and line connection on station A.
4. Communicate data using the nonprocedural protocol/bidirectional protocol.
5. In order to end the communication, disconnect line from station A that initiated the line connection.

■Processing flow


Station A (connection request side)
Connection
Communication
Disconnection
Connection request Y11
Modem initialization completion X10
Connecting X12
Disconnection request Y12
Disconnection completion X14
RS-232 CD terminal
Station B (connection reception side)
(ON)
Modem initialization completion X10
Connecting X12
Disconnection request Y12
(OFF)
Disconnection completion X14
(OFF)
RS-232 CD terminal

* It is possible to disconnect line from Station B, as well.

Considerations for performing data communication

■When communicating data with the target device

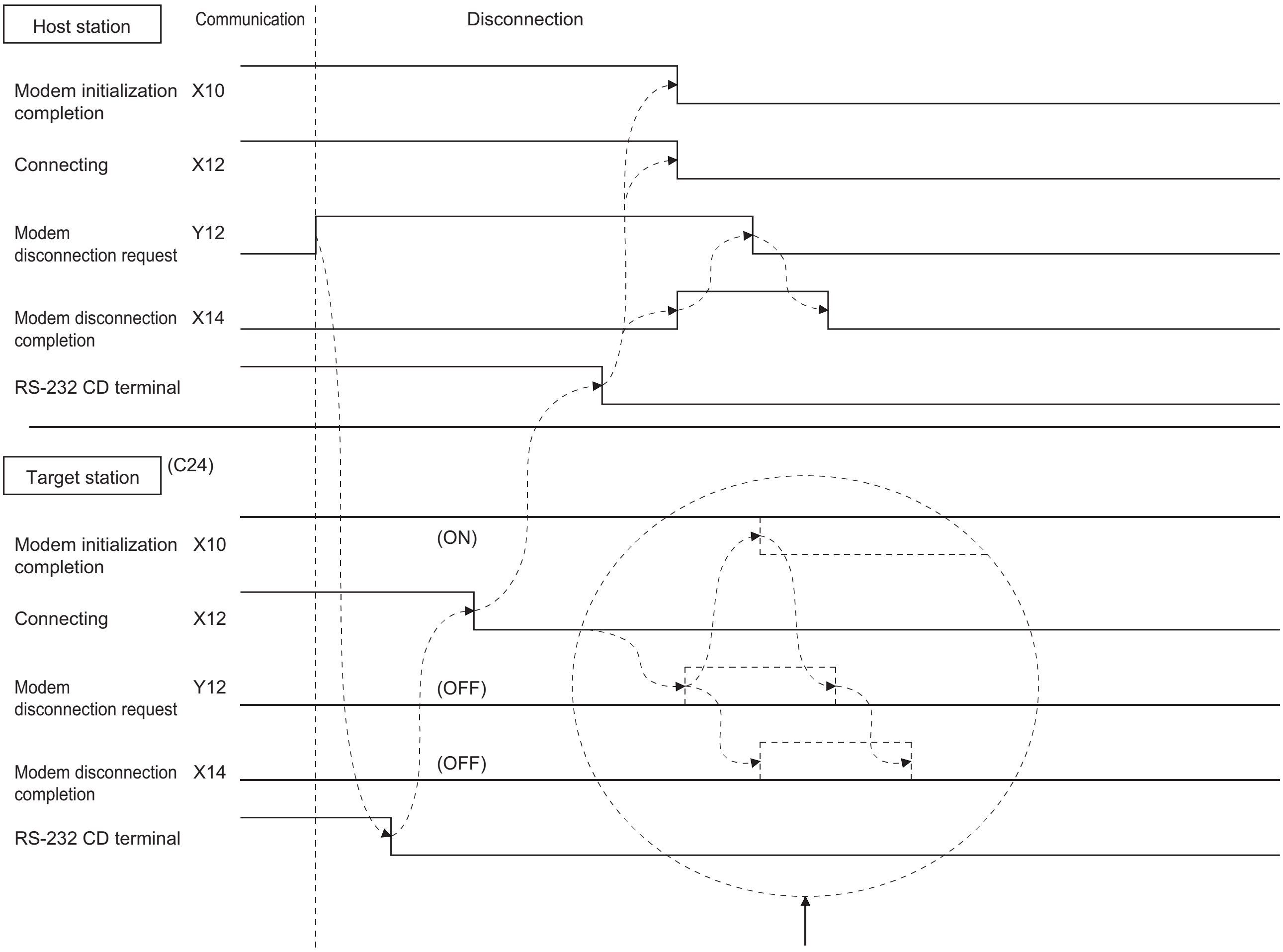
- When setting the no-communication interval time to 'Infinite wait' (setting value = 0) in the initial setting of C24, be sure to perform line disconnection after the completion of data communication.
- Only the nonprocedural protocol/bidirectional protocol data communication can be performed with the CPU module with C24 installed.
- C24 automatically performs line disconnection processing if no data communication is performed during the no-communication interval time.
  The 'Connection' (X12) and the 'Modem initialization completion' (X10) turn OFF and the 'Modem disconnection completion' (X14) turns ON when the line disconnection processing is performed.

Line disconnection

The following explains the line disconnection upon communication completion when communicating data with the target device using the modem functions.

Input/output signals to be used

Use the 'Modem disconnection request' (Y12) and 'Modem disconnection completion' (X14).



Host station
Communication
Disconnection
Modem initialization completion X10
Connecting X12
Modem disconnection request Y12
Modem disconnection completion X14
RS-232 CD terminal
Target station
(C24)
Modem initialization completion X10
(ON)
Connecting X12
Modem disconnection request Y12
(OFF)
Modem disconnection completion X14
(OFF)
RS-232 CD terminal

This is the procedure to take in order to turn off the initialization completion signal (X10).

* In the case of C24, when the line is disconnected from the target device, the initialization completion signal (X10) at the host station is not turned OFF.

Point

- There is no buffer memory for line disconnection processing.
- Line disconnection processing can be conducted from either device as long as the connection is in progress.
- The line disconnection processing disconnects the line connection with the target device as well as the connection with the modem on the C24 side.
- Even when an error occurs during the line disconnection, the disconnection processing will be forced.
- When resuming data communication after line disconnection, start form either one of the following processing depending on the 'Modem initialization completion' (X10).
  'Modem initialization completion' is OFF: Start from the initialization of the modem/TA.
  'Modem initialization completion' is ON: Start from the line connection with the target device.

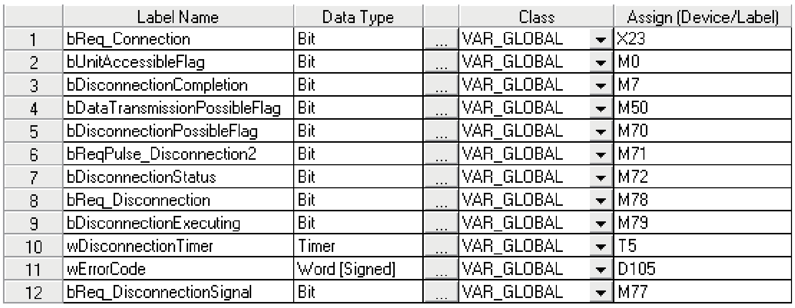
Considerations during the line disconnection

- Prior to data communication with target devices, determine when and which station will perform the line connection (dialing) and line disconnection processing to the target device.
- If the line is disconnected during data transmission, transmission processing will be performed depending on the signal status of the C24 RS-232 interface.
- If the line is disconnected during data reception, data reception will be disabled.
  This may cause an error occurrence such as a reception time out.
- When data communication is not performed for the period equivalent to the no-communication interval time designation, C24 will automatically disconnect the line.
  When the line is disconnected, the 'Connection' (X12) and the 'Modem initialization completion' (X10) turn OFF, and the 'Modem disconnection completion' (X14) turns ON.
  To turn OFF the 'Modem disconnection completion' (X14), turn ON the 'Modem disconnection request' (Y12) for a second after the 'Modem disconnection completion' (X14) turned ON.

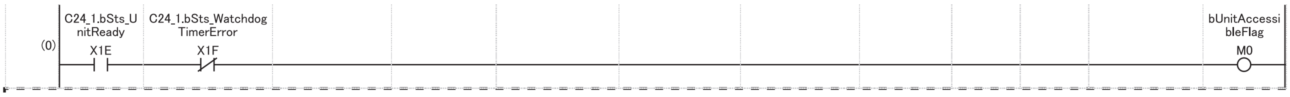
Program for line disconnection

A program example for line disconnection is shown below.

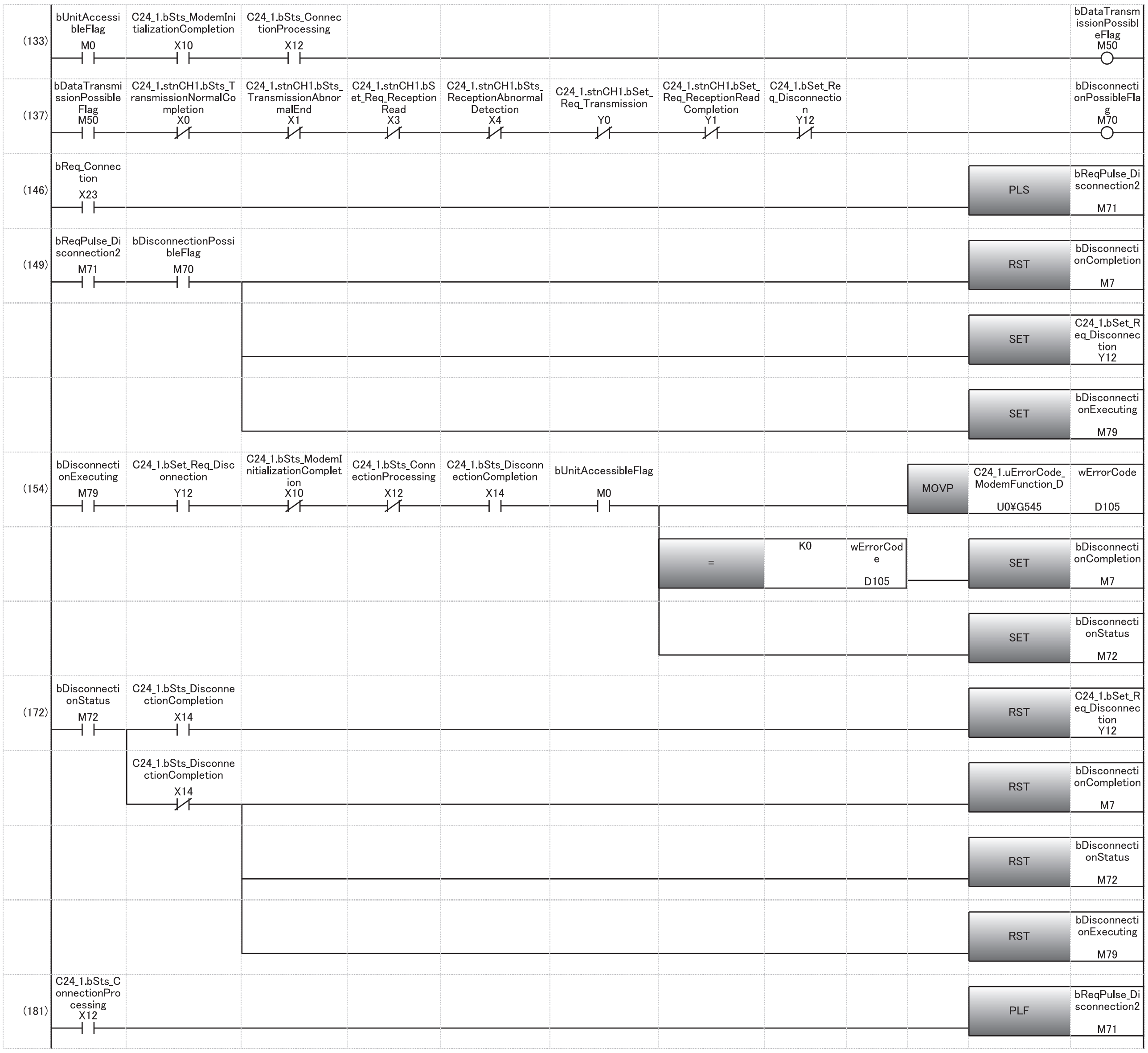
■Program example

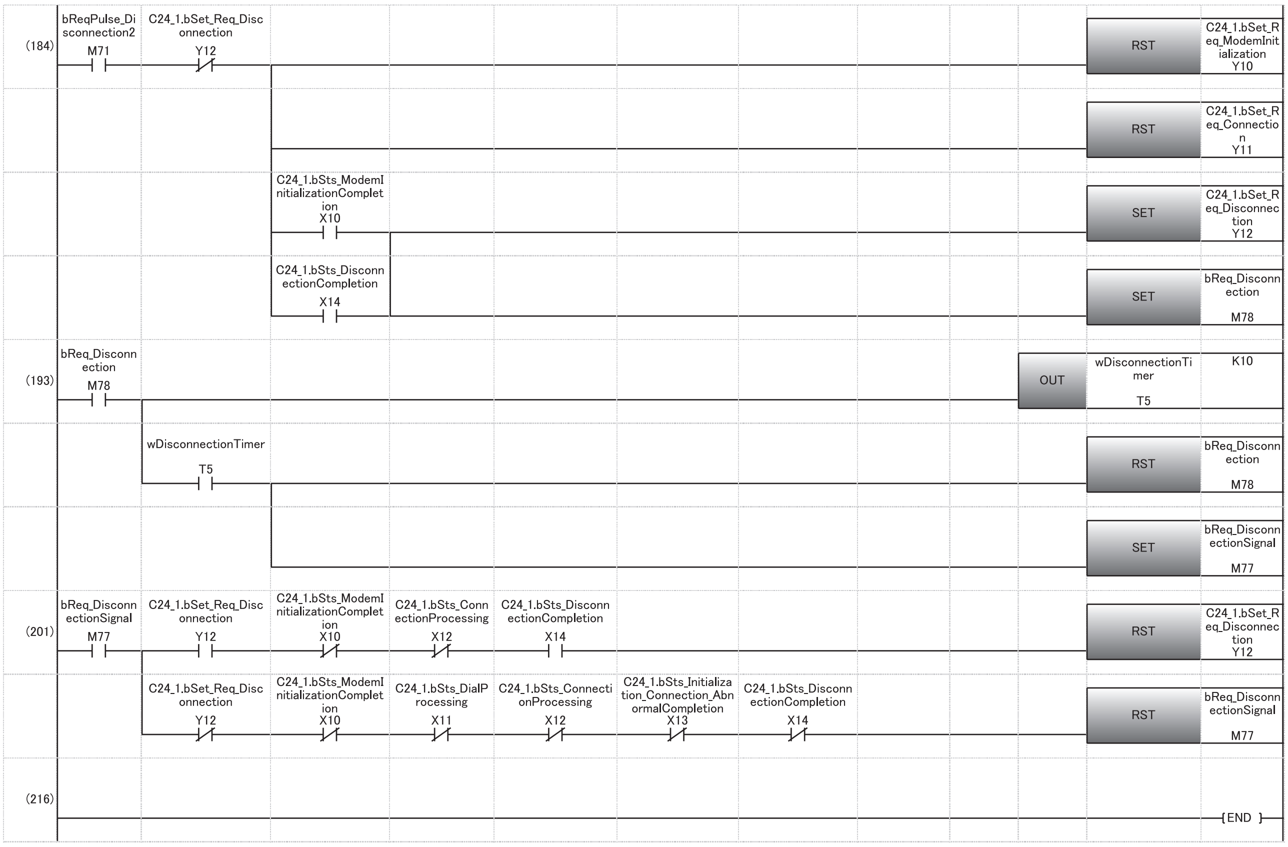
Category	Label name	Description	Device
Module label	C24_1.stnCH1.bSts_TransmissionNormalCompletion	CH1 Transmission normal completion	X0
	C24_1.stnCH1.bSts_TransmissionAbnormalEnd	CH1 Transmission abnormal completion	X1
	C24_1.bSts_ModemInitializationCompletion	Modem initialization completion	X10
	C24_1.bSts_DialProcessing	Dialing	X11
	C24_1.bSts_ConnectionProcessing	Connection	X12
	C24_1.bSts_Initialization_Connection_AbnormalCompletion	Initialization/connection abnormal completion	X13
	C24_1.bSts_DisconnectionCompletion	Modem disconnection completion	X14
	C24_1.bSts_UnitReady	Module READY	X1E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X1F
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection	CH1 Reception abnormal detection	X4
	C24_1.stnCH1.bSet_Req_Transmission	CH1 Transmission request	Y0
	C24_1.stnCH1.bSet_Req_ReceptionReadCompletion	CH1 Reception data read completion	Y1
	C24_1.bSet_Req_ModemInitialization	Modem initialization request	Y10
	C24_1.bSet_Req_Connection	Connection request	Y11
	C24_1.bSet_Req_Disconnection	Modem disconnection request	Y12
	C24_1.uErrorCode_ModemFunction_D	Modern function error code	U0\G545
	C24_1.wnForUserRegistrationNo8001H_D[0]	For user registration number 8001H	U0\G6912
	C24_1.wnForUserRegistrationNo8002H_D[0]	For user registration number 8002H	U0\G6953
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_Connection	Bit	...	VAR_GLOBAL	X23
2	bUnitAccessibleFlag	Bit	...	VAR_GLOBAL	M0
3	bDisconnectionCompletion	Bit	...	VAR_GLOBAL	M7
4	bDataTransmissionPossibleFlag	Bit	...	VAR_GLOBAL	M50
5	bDisconnectionPossibleFlag	Bit	...	VAR_GLOBAL	M70
6	bReqPulse_Disconnection2	Bit	...	VAR_GLOBAL	M71
7	bDisconnectionStatus	Bit	...	VAR_GLOBAL	M72
8	bReq_Disconnection	Bit	...	VAR_GLOBAL	M78
9	bDisconnectionExecuting	Bit	...	VAR_GLOBAL	M79
10	wDisconnectionTimer	Timer	...	VAR_GLOBAL	T5
11	wErrorCode	Word [Signed]	...	VAR_GLOBAL	D105
12	bReq_DisconnectionSignal	Bit	...	VAR_GLOBAL	M77



- Registration of data for initialization
- Registration of data for connection
- Modem/TA initialization
- Line connection


(133)
bUnitAccessibleFlag
M0
C24_1.bSts_ModemInitializationCompletion
X10
C24_1.bSts_ConnectionProcessing
X12
bDataTransmissionPossibleFlag
M50
(137)
bDataTransmissionPossibleFlag
M50
C24_1.stnCH1.bSts_TransmissionNormalCompletion
X0
C24_1.stnCH1.bSts_TransmissionAbnormalEnd
X1
C24_1.stnCH1.bSet_Req_ReceptionRead
X3
C24_1.stnCH1.bSts_ReceptionAbnormalDetection
X4
C24_1.stnCH1.bSet_Req_Transmission
Y0
C24_1.stnCH1.bSet_Req_ReceptionReadCompletion
Y1
C24_1.bSet_Req_Disconnection
Y12
bDisconnectionPossibleFlag
M70
(146)
bReq_Connection
X23
PLS
bReqPulse_Disconnection2
M71
(149)
bReqPulse_Disconnection2
M71
bDisconnectionPossibleFlag
M70
RST
bDisconnectionCompletion
M7
SET
C24_1.bSet_Req_Disconnection
Y12
SET
bDisconnectionExecuting
M79
(154)
bDisconnectionExecuting
M79
C24_1.bSet_Req_Disconnection
Y12
C24_1.bSts_ModemInitializationCompletion
X10
C24_1.bSts_ConnectionProcessing
X12
C24_1.bSts_DisconnectionCompletion
X14
bUnitAccessibleFlag
M0
MOVP
C24_1.uErrorCode_ModemFunction_D
U0¥G545
wErrorCode
D105
=
K0
wErrorCode
D105
SET
bDisconnectionCompletion
M7
SET
bDisconnectionStatus
M72
(172)
bDisconnectionStatus
M72
C24_1.bSts_DisconnectionCompletion
X14
RST
C24_1.bSet_Req_Disconnection
Y12
C24_1.bSts_DisconnectionCompletion
X14
RST
bDisconnectionCompletion
M7
RST
bDisconnectionStatus
M72
RST
bDisconnectionExecuting
M79
(181)
C24_1.bSts_ConnectionProcessing
X12
PLF
bReqPulse_Disconnection2
M71


(184)
bReqPulse_Disconnection2
M71
C24_1.bSet_Req_Disconnection
Y12
RST
C24_1.bSet_Req_ModemInitialization
Y10
RST
C24_1.bSet_Req_Connection
Y11
C24_1.bSts_ModemInitializationCompletion
X10
SET
C24_1.bSet_Req_Disconnection
Y12
C24_1.bSts_DisconnectionCompletion
X14
SET
bReq_Disconnection
M78
(193)
bReq_Disconnection
M78
OUT
wDisconnectionTimer
T5
K10
wDisconnectionTimer
T5
RST
bReq_Disconnection
M78
SET
bReq_DisconnectionSignal
M77
(201)
bReq_DisconnectionSignal
M77
C24_1.bSet_Req_Disconnection
Y12
C24_1.bSts_ModemInitializationCompletion
X10
C24_1.bSts_ConnectionProcessing
X12
C24_1.bSts_DisconnectionCompletion
X14
RST
C24_1.bSet_Req_Disconnection
Y12
C24_1.bSet_Req_Disconnection
Y12
C24_1.bSts_ModemInitializationCompletion
X10
C24_1.bSts_DialProcessing
X11
C24_1.bSts_ConnectionProcessing
X12
C24_1.bSts_Initialization_Connection_AbnormalCompletion
X13
C24_1.bSts_DisconnectionCompletion
X14
RST
bReq_DisconnectionSignal
M77
(216)
END

(0) Turn the READY flag ON.

(11) to (14) Register data for initialization
☞ Page 212 Program example

(41) to (44) Register data for connection
☞ Page 216 Program example

(76) to (91) Modem/TA initialization
☞ Page 220 Program example

(107) to (130) Line connection
☞ Page 226 Program example (When establishing the line connection from the C24 side following the completion of initialization)

(133) Turn the data update available flag ON.

(137) Turn the line disconnection available flag ON.

(146) Issue the line disconnection command.

(149) Set the modem disconnection request signal.

(154) Read the error code when modem disconnection completion signal is turned ON.
Set the modem disconnection completion flag after the normal completion.

(172) Reset the modem disconnection request signal.

(181) Line disconnection processing (line disconnection from the target device)

(184) Reset the request signals.
Set the modem disconnection request signal when either of completion signal or connection signal is turned ON.

(193) To turn the 'Modem disconnection completion' (X14) OFF, wait for one second after the 'Modem disconnection request' (Y12) is turned ON.

(201) Reset the request signal when the modem disconnection completion signal is turned ON.
Reset the request signal when the modem disconnection completion signal is turned OFF.

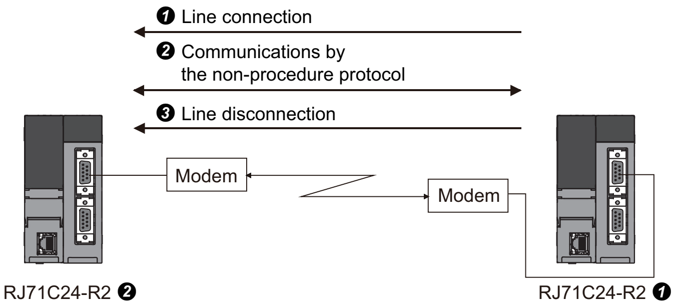
9.5 Sample program

Program for data communication

The following shows sample programs to test the connection between C24s.
Each program contains a minimum set of processing necessary for performing a communication test.
Modify the data for initialization and data for connection in accordance with the system. When adding error-handling procedures, add them separately by referring to the following explanations.

System configuration

The system configuration used in this program is shown below.



Connection request station side (RJ71C24-R2❶) program example

Perform the initialization of the modem connected to the CH1 side interface, line connection, data communication using the nonprocedural protocol, and line disconnection by the command from the user.

■Parameter list

The following shows the parameters of this sample program.

- Module parameter

Category	Setting item			Setting content
Basic settings	Communication protocol setting			Nonprocedural protocol
	Communication speed setting			19200 bps
	Transmission setting	Data bit		8
		Parity bit		None
		Odd/even parity		Odd
		Stop bit		1
		Sum check code		Yes
		Online change		Enable
		Setting change		Enable
	Station number setting (CH1, 2 in common: 0 to 31)			0
Application settings	Modem function setting	Modem function specification 1	Modem connection CH specification	1CH
			Data No. for initialization designation	07D5H
			Data No. for connection designation	0BB8H

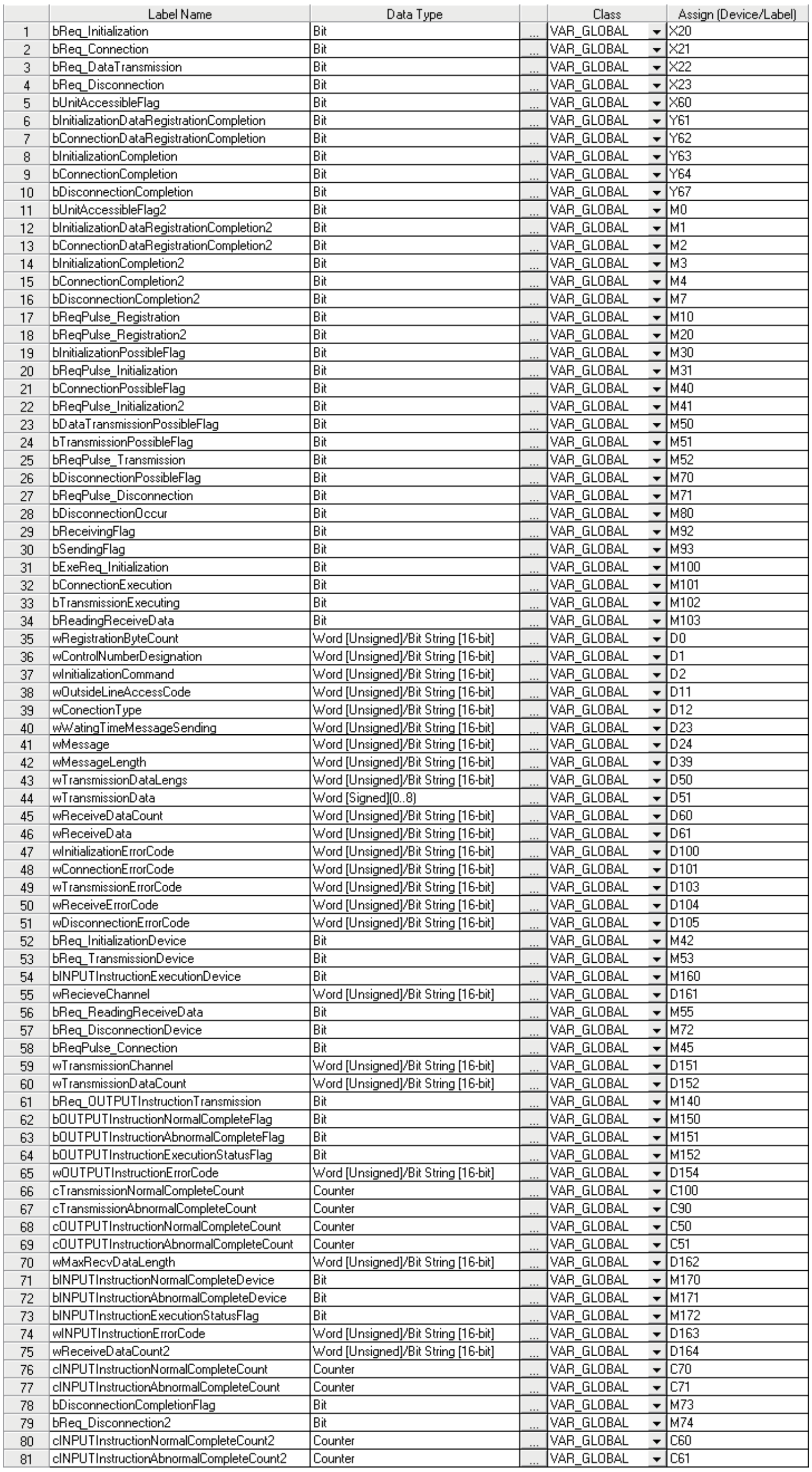
- Module extended parameter

Category	Setting item		Setting contents
Modem connection data	Modem connection data No.3000	Telephone number	0123456789

■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.bSts_TransmissionNormalCompletion	CH1 Transmission normal completion	X0
	C24_1.stnCH1.bSts_TransmissionAbnormalEnd	CH1 Transmission abnormal completion	X1
	C24_1.bSts_ModemInitializationCompletion	Modem initialization completion	X10
	C24_1.bSts_DialProcessing	Dialing	X11
	C24_1.bSts_ConnectionProcessing	Connection	X12
	C24_1.bSts_Initialization_Connection_AbnormalCompletion	Initialization/connection abnormal completion	X13
	C24_1.bSts_DisconnectionCompletion	Modem disconnection completion	X14
	C24_1.bSts_UnitReady	Module READY	X1E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X1F
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection	CH1 Reception abnormal detection	X4
	C24_1.stnCH1.bSet_Req_Transmission	CH1 Transmission request	Y0
	C24_1.stnCH1.bSet_Req_ReceptionReadCompletion	CH1 Reception data read completion	Y1
	C24_1.bSet_Req_ModemInitialization	Modem initialization request	Y10
	C24_1.bSet_Req_Connection	Connection request	Y11
	C24_1.bSet_Req_Disconnection	Modem disconnection request	Y12
	C24_1.uErrorCode_ModemFunction_D	Modem function error code	U0\G545
	C24_1.wnForUserRegistrationNo8001H_D[0]	For user registration number 8001H	U0\G6912
	C24_1.wnForUserRegistrationNo8002H_D[0]	For user registration number 8002H	U0\G6953

Category	Label name/FB name	Description	Device
Label to be defined	Define the global label as shown below.		

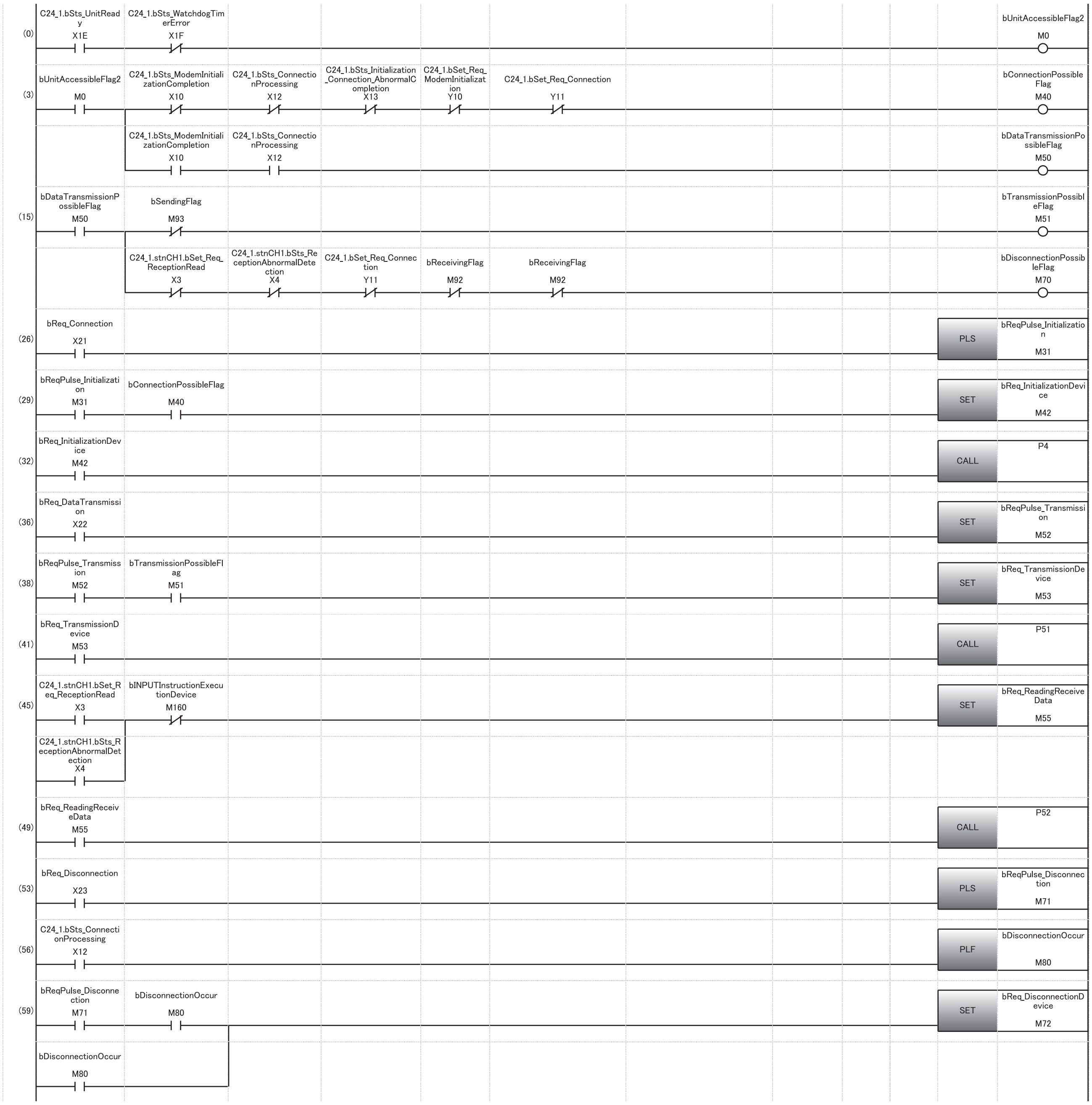


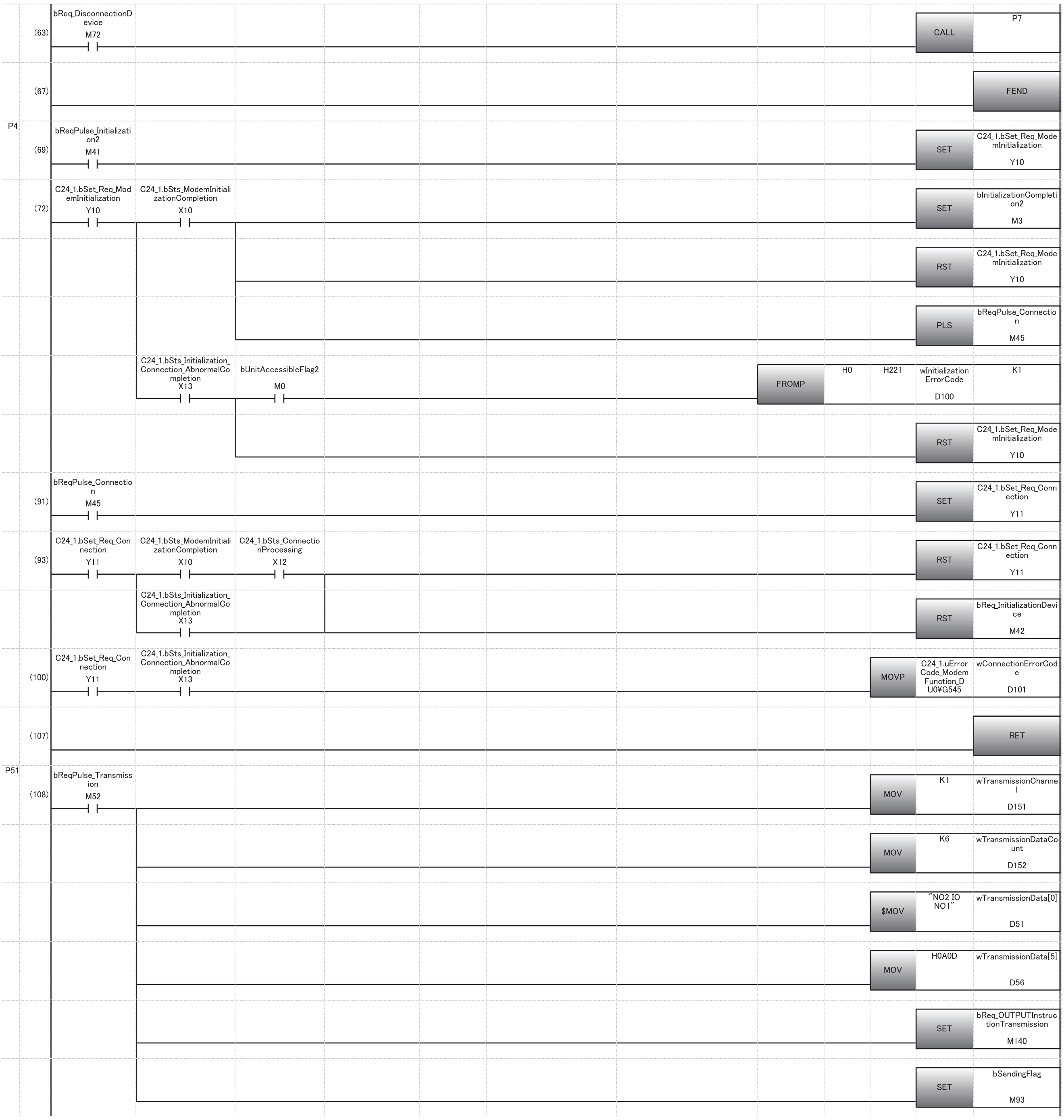
	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_Initialization	Bit	...	VAR_GLOBAL	X20
2	bReq_Connection	Bit	...	VAR_GLOBAL	X21
3	bReq_DataTransmission	Bit	...	VAR_GLOBAL	X22
4	bReq_Disconnection	Bit	...	VAR_GLOBAL	X23
5	bUnitAccessibleFlag	Bit	...	VAR_GLOBAL	X60
6	bInitializationDataRegistrationCompletion	Bit	...	VAR_GLOBAL	Y61
7	bConnectionDataRegistrationCompletion	Bit	...	VAR_GLOBAL	Y62
8	bInitializationCompletion	Bit	...	VAR_GLOBAL	Y63
9	bConnectionCompletion	Bit	...	VAR_GLOBAL	Y64
10	bDisconnectionCompletion	Bit	...	VAR_GLOBAL	Y67
11	bUnitAccessibleFlag2	Bit	...	VAR_GLOBAL	M0
12	bInitializationDataRegistrationCompletion2	Bit	...	VAR_GLOBAL	M1
13	bConnectionDataRegistrationCompletion2	Bit	...	VAR_GLOBAL	M2
14	bInitializationCompletion2	Bit	...	VAR_GLOBAL	M3
15	bConnectionCompletion2	Bit	...	VAR_GLOBAL	M4
16	bDisconnectionCompletion2	Bit	...	VAR_GLOBAL	M7
17	bReqPulse_Registration	Bit	...	VAR_GLOBAL	M10
18	bReqPulse_Registration2	Bit	...	VAR_GLOBAL	M20
19	bInitializationPossibleFlag	Bit	...	VAR_GLOBAL	M30
20	bReqPulse_Initialization	Bit	...	VAR_GLOBAL	M31
21	bConnectionPossibleFlag	Bit	...	VAR_GLOBAL	M40
22	bReqPulse_Initialization2	Bit	...	VAR_GLOBAL	M41
23	bDataTransmissionPossibleFlag	Bit	...	VAR_GLOBAL	M50
24	bTransmissionPossibleFlag	Bit	...	VAR_GLOBAL	M51
25	bReqPulse_Transmission	Bit	...	VAR_GLOBAL	M52
26	bDisconnectionPossibleFlag	Bit	...	VAR_GLOBAL	M70
27	bReqPulse_Disconnection	Bit	...	VAR_GLOBAL	M71
28	bDisconnectionOccur	Bit	...	VAR_GLOBAL	M80
29	bReceivingFlag	Bit	...	VAR_GLOBAL	M92
30	bSendingFlag	Bit	...	VAR_GLOBAL	M93
31	bExeReq_Initialization	Bit	...	VAR_GLOBAL	M100
32	bConnectionExecution	Bit	...	VAR_GLOBAL	M101
33	bTransmissionExecuting	Bit	...	VAR_GLOBAL	M102
34	bReadingReceiveData	Bit	...	VAR_GLOBAL	M103
35	wRegistrationByteCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
36	wControlNumberDesignation	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
37	wInitializationCommand	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2
38	wOutsideLineAccessCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D11
39	wConectionType	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D12
40	wWatingTimeMessageSending	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D23
41	wMessage	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D24
42	wMessageLength	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D39
43	wTransmissionDataLengs	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D50
44	wTransmissionData	Word [Signed](0..8)	...	VAR_GLOBAL	D51
45	wReceiveDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D60
46	wReceiveData	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D61
47	wInitializationErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D100
48	wConnectionErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D101
49	wTransmissionErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D103
50	wReceiveErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D104
51	wDisconnectionErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D105
52	bReq_InitializationDevice	Bit	...	VAR_GLOBAL	M42
53	bReq_TransmissionDevice	Bit	...	VAR_GLOBAL	M53
54	bINPUTInstructionExecutionDevice	Bit	...	VAR_GLOBAL	M160
55	wRecieveChannel	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D161
56	bReq_ReadingReceiveData	Bit	...	VAR_GLOBAL	M55
57	bReq_DisconnectionDevice	Bit	...	VAR_GLOBAL	M72
58	bReqPulse_Connection	Bit	...	VAR_GLOBAL	M45
59	wTransmissionChannel	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D151
60	wTransmissionDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D152
61	bReq_OUTPUTInstructionTransmission	Bit	...	VAR_GLOBAL	M140
62	bOUTPUTInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M150
63	bOUTPUTInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M151
64	bOUTPUTInstructionExecutionStatusFlag	Bit	...	VAR_GLOBAL	M152
65	wOUTPUTInstructionErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D154
66	cTransmissionNormalCompleteCount	Counter	...	VAR_GLOBAL	C100
67	cTransmissionAbnormalCompleteCount	Counter	...	VAR_GLOBAL	C90
68	cOUTPUTInstructionNormalCompleteCount	Counter	...	VAR_GLOBAL	C50
69	cOUTPUTInstructionAbnormalCompleteCount	Counter	...	VAR_GLOBAL	C51
70	wMaxRecvDataLength	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D162
71	bINPUTInstructionNormalCompleteDevice	Bit	...	VAR_GLOBAL	M170
72	bINPUTInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M171
73	bINPUTInstructionExecutionStatusFlag	Bit	...	VAR_GLOBAL	M172
74	wINPUTInstructionErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D163
75	wReceiveDataCount2	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D164
76	cINPUTInstructionNormalCompleteCount	Counter	...	VAR_GLOBAL	C70
77	cINPUTInstructionAbnormalCompleteCount	Counter	...	VAR_GLOBAL	C71
78	bDisconnectionCompletionFlag	Bit	...	VAR_GLOBAL	M73
79	bReq_Disconnection2	Bit	...	VAR_GLOBAL	M74
80	cINPUTInstructionNormalCompleteCount2	Counter	...	VAR_GLOBAL	C60
81	cINPUTInstructionAbnormalCompleteCount2	Counter	...	VAR_GLOBAL	C61

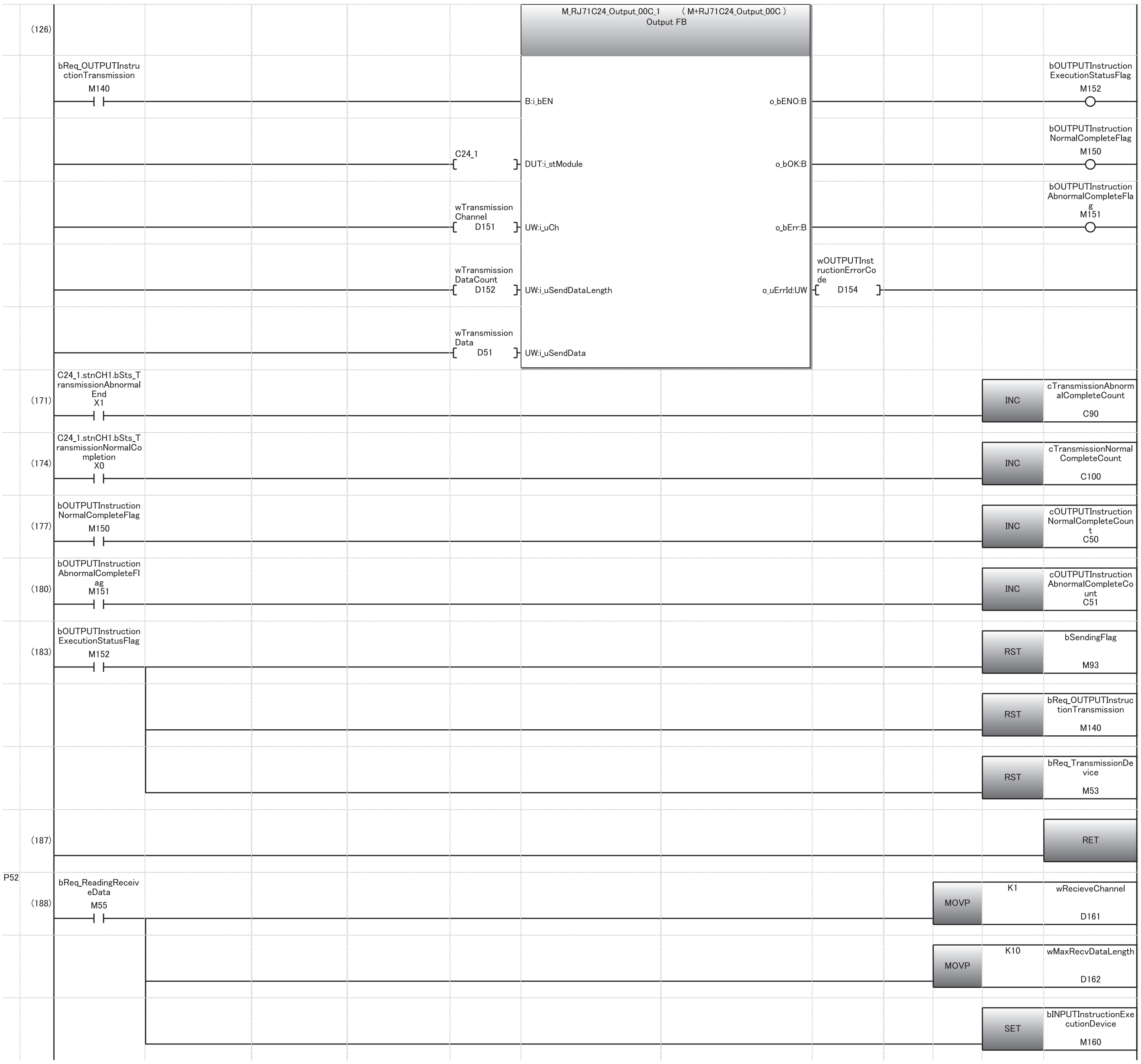
FB	M+RJ71C24_Output	Sends data for specified data points.	—
	M+RJ71C24_Input	Reads the received data.	—

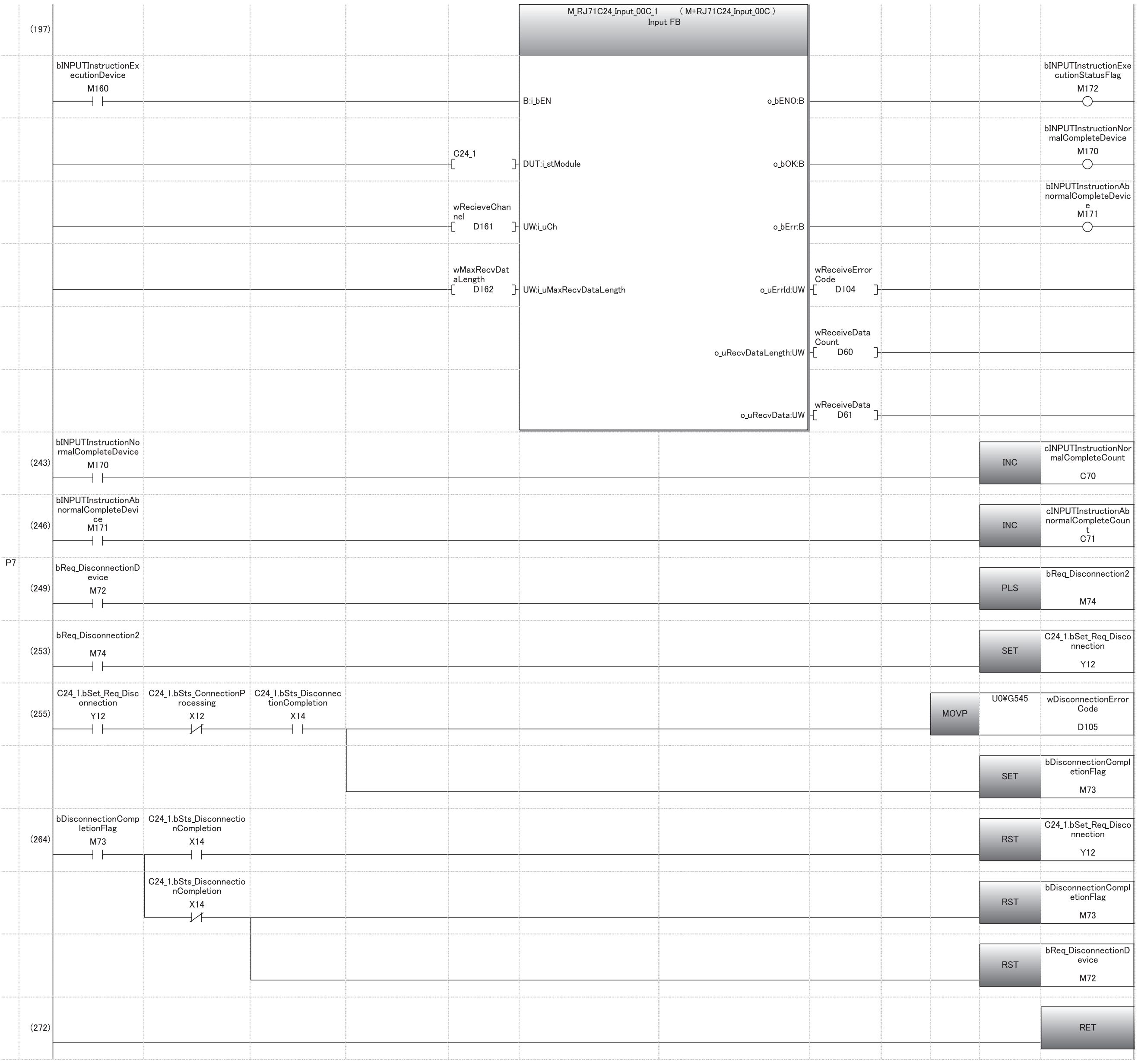
For information on the module FB to be used, refer to the following manual.

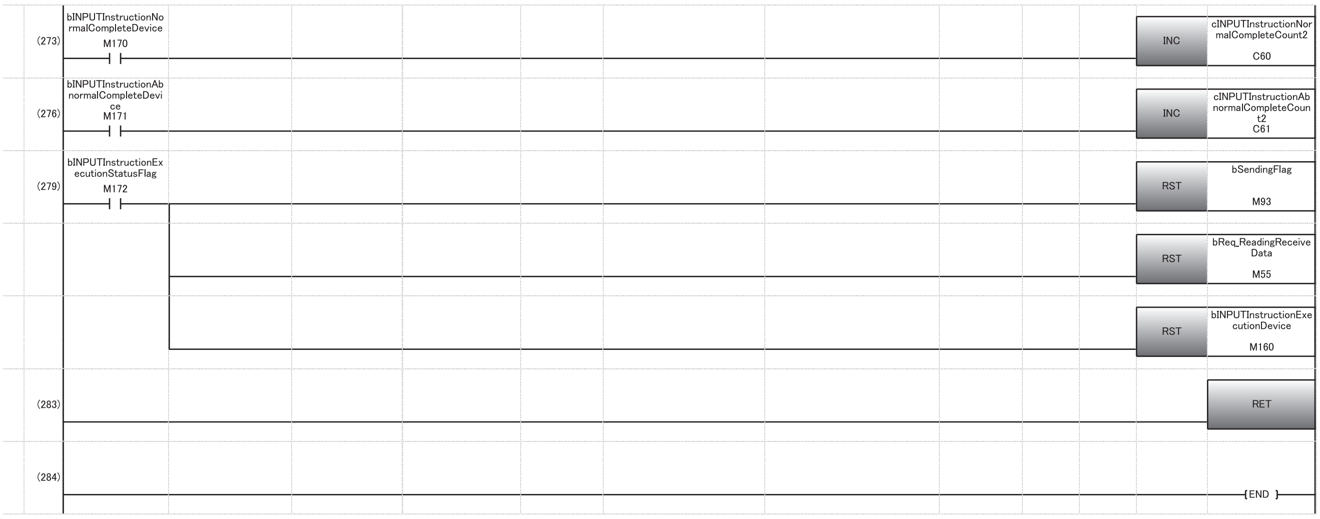
📖MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
C24_1.bSts_UnitReady
X1E
C24_1.bSts_WatchdogTimerError
X1F
bUnitAccessibleFlag2
M0
(3)
bUnitAccessibleFlag2
M0
C24_1.bSts_ModemInitializationCompletion
X10
C24_1.bSts_ConnectionProcessing
X12
C24_1.bSts_Initialization_Connection_AbnormalCompletion
X13
C24_1.bSet_Req_ModemInitialization
Y10
C24_1.bSet_Req_Connection
Y11
bConnectionPossibleFlag
M40
C24_1.bSts_ModemInitializationCompletion
X10
C24_1.bSts_ConnectionProcessing
X12
bDataTransmissionPossibleFlag
M50
(15)
bDataTransmissionPossibleFlag
M50
bSendingFlag
M93
bTransmissionPossibleFlag
M51
C24_1.stnCH1.bSet_Req_ReceptionRead
X3
C24_1.stnCH1.bSts_ReceptionAbnormalDetection
X4
C24_1.bSet_Req_Connection
Y11
bReceivingFlag
M92
bReceivingFlag
M92
bDisconnectionPossibleFlag
M70
(26)
bReq_Connection
X21
PLS
bReqPulse_Initialization
M31
(29)
bReqPulse_Initialization
M31
bConnectionPossibleFlag
M40
SET
bReq_InitializationDevice
M42
(32)
bReq_InitializationDevice
M42
CALL
P4
(36)
bReq_DataTransmission
X22
SET
bReqPulse_Transmission
M52
(38)
bReqPulse_Transmission
M52
bTransmissionPossibleFlag
M51
SET
bReq_TransmissionDevice
M53
(41)
bReq_TransmissionDevice
M53
CALL
P51
(45)
C24_1.stnCH1.bSet_Req_ReceptionRead
X3
bINPUTInstructionExecutionDevice
M160
SET
bReq_ReadingReceiveData
M55
C24_1.stnCH1.bSts_ReceptionAbnormalDetection
X4
(49)
bReq_ReadingReceiveData
M55
CALL
P52
(53)
bReq_Disconnection
X23
PLS
bReqPulse_Disconnection
M71
(56)
C24_1.bSts_ConnectionProcessing
X12
PLF
bDisconnectionOccur
M80
(59)
bReqPulse_Disconnection
M71
bDisconnectionOccur
M80
SET
bReq_DisconnectionDevice
M72
bDisconnectionOccur
M80


(63)
bReq_DisconnectionDevice
M72
CALL
P7
(67)
FEND
P4
(69)
bReqPulse_Initialization2
M41
SET
C24_1.bSet_Req_ModemInitialization
Y10
(72)
C24_1.bSet_Req_ModemInitialization
Y10
C24_1.bSts_ModemInitializationCompletion
X10
SET
bInitializationCompletion2
M3
RST
C24_1.bSet_Req_ModemInitialization
Y10
PLS
bReqPulse_Connection
M45
C24_1.bSts_Initialization_Connection_AbnormalCompletion
X13
bUnitAccessibleFlag2
M0
FROMP
H0
H221
wInitializationErrorCode
D100
K1
RST
C24_1.bSet_Req_ModemInitialization
Y10
(91)
bReqPulse_Connection
M45
SET
C24_1.bSet_Req_Connection
Y11
(93)
C24_1.bSet_Req_Connection
Y11
C24_1.bSts_ModemInitializationCompletion
X10
C24_1.bSts_ConnectionProcessing
X12
RST
C24_1.bSet_Req_Connection
Y11
C24_1.bSts_Initialization_Connection_AbnormalCompletion
X13
RST
bReq_InitializationDevice
M42
(100)
C24_1.bSet_Req_Connection
Y11
C24_1.bSts_Initialization_Connection_AbnormalCompletion
X13
MOVP
C24_1.uErrorCode_ModemFunction_DU0¥G545
wConnectionErrorCode
D101
(107)
RET
P51
(108)
bReqPulse_Transmission
M52
MOV
K1
wTransmissionChannel
D151
MOV
K6
wTransmissionDataCount
D152
$MOV
"NO2 IO NO1"
wTransmissionData[0]
D51
MOV
H0A0D
wTransmissionData[5]
D56
SET
bReq_OUTPUTInstructionTransmission
M140
SET
bSendingFlag
M93


(126)
M_RJ71C24_Output_00C_1 (M+RJ71C24_Output_00C)
Output FB
bReq_OUTPUTInstructionTransmission
M140
B:i_bEN
o_bENO:B
bOUTPUTInstructionExecutionStatusFlag
M152
C24_1
DUT:i_stModule
o_bOK:B
bOUTPUTInstructionNormalCompleteFlag
M150
wTransmissionChannel
D151
UW:i_uCh
o_bErr:B
bOUTPUTInstructionAbnormalCompleteFlag
M151
wTransmissionDataCount
D152
UW:i_uSendDataLength
o_uErrId:UW
wOUTPUTInstructionErrorCode
D154
wTransmissionData
D51
UW:i_uSendData
(171)
C24_1.stnCH1.bSts_TransmissionAbnormalEnd
X1
INC
cTransmissionAbnormalCompleteCount
C90
(174)
C24_1.stnCH1.bSts_TransmissionNormalCompletion
X0
INC
cTransmissionNormalCompleteCount
C100
(177)
bOUTPUTInstructionNormalCompleteFlag
M150
INC
cOUTPUTInstructionNormalCompleteCount
C50
(180)
bOUTPUTInstructionAbnormalCompleteFlag
M151
INC
cOUTPUTInstructionAbnormalCompleteCount
C51
(183)
bOUTPUTInstructionExecutionStatusFlag
M152
RST
bSendingFlag
M93
RST
bReq_OUTPUTInstructionTransmission
M140
RST
bReq_TransmissionDevice
M53
(187)
RET
P52
(188)
bReq_ReadingReceiveData
M55
MOVP
K1
wRecieveChannel
D161
MOVP
K10
wMaxRecvDataLength
D162
SET
bINPUTInstructionExecutionDevice
M160


(197)
M_RJ71C24_Input_00C_1 (M+RJ71C24_Input_00C)
Input FB
bINPUTInstructionExecutionDevice
M160
B:i_bEN
o_bENO:B
bINPUTInstructionExecutionStatusFlag
M172
C24_1
DUT:i_stModule
o_bOK:B
bINPUTInstructionNormalCompleteDevice
M170
wRecieveChannel
D161
UW:i_uCh
o_bErr:B
bINPUTInstructionAbnormalCompleteDevice
M171
wMaxRecvDataLength
D162
UW:i_uMaxRecvDataLength
o_uErrId:UW
wReceiveErrorCode
D104
o_uRecvDataLength:UW
wReceiveDataCount
D60
o_uRecvData:UW
wReceiveData
D61
(243)
bINPUTInstructionNormalCompleteDevice
M170
INC
cINPUTInstructionNormalCompleteCount
C70
(246)
bINPUTInstructionAbnormalCompleteDevice
M171
INC
cINPUTInstructionAbnormalCompleteCount
C71
P7
(249)
bReq_DisconnectionDevice
M72
PLS
bReq_Disconnection2
M74
(253)
bReq_Disconnection2
M74
SET
C24_1.bSet_Req_Disconnection
Y12
(255)
C24_1.bSet_Req_Disconnection
Y12
C24_1.bSts_ConnectionProcessing
X12
C24_1.bSts_DisconnectionCompletion
X14
MOVP
U0¥G545
wDisconnectionErrorCode
D105
SET
bDisconnectionCompletionFlag
M73
(264)
bDisconnectionCompletionFlag
M73
C24_1.bSts_DisconnectionCompletion
X14
RST
C24_1.bSet_Req_Disconnection
Y12
C24_1.bSts_DisconnectionCompletion
X14
RST
bDisconnectionCompletionFlag
M73
RST
bReq_DisconnectionDevice
M72
(272)
RET


(273)
bINPUTInstructionNormalCompleteDevice
M170
INC
cINPUTInstructionNormalCompleteCount2
C60
(276)
bINPUTInstructionAbnormalCompleteDevice
M171
INC
cINPUTInstructionAbnormalCompleteCount2
C61
(279)
bINPUTInstructionExecutionStatusFlag
M172
RST
bSendingFlag
M93
RST
bReq_ReadingReceiveData
M55
RST
bINPUTInstructionExecutionDevice
M160
(283)
RET
(284)
END

(0) Accessible

(3) Modem initialization
Data communication available

(15) Data transmission available
Modem disconnection available

(26) Convert the modem initialization and line connection command into pulse.

(32) Jump to the subroutine of the modem initialization and line connection processing.

(36) Convert the data communication (transmission) command into pulse.

(41) Jump to the subroutine of the data reception processing.

(45) Jump to the subroutine of the receive data.

(49) Jump to the subroutine of the data reception processing.

(53) Convert the line disconnection command into pulse.

(56) Convert the connection signal OFF into pulse.

(63) Jump to the subroutine of the line disconnection processing.

(69) Set the initialization request signal.

(91) Set the connection request signal.

(93) Reset the request signal when the initialization completion signal and connection signal are turned ON.
Read the error code when initialization/connection abnormal completion signal is turned ON, and reset the request signal.

(108) Set the transmission channel to CH1.
Set the send data quantity.
Set the send data.

(177) Transmission normal completion

(180) Transmission abnormal completion

(188) Set the data reception channel CH1.

(243) Received data read normal completion

(246) Received data read abnormal completion

(253) Set the modem disconnection request signal.

(255) Read the error code when modem disconnection completion signal is turned ON.
Set the modem disconnection completion flag when normal completion signal is turned ON.

(264) Reset the modem disconnection request signal.

Connection receiving station side (RJ71C24-R2❷) program example

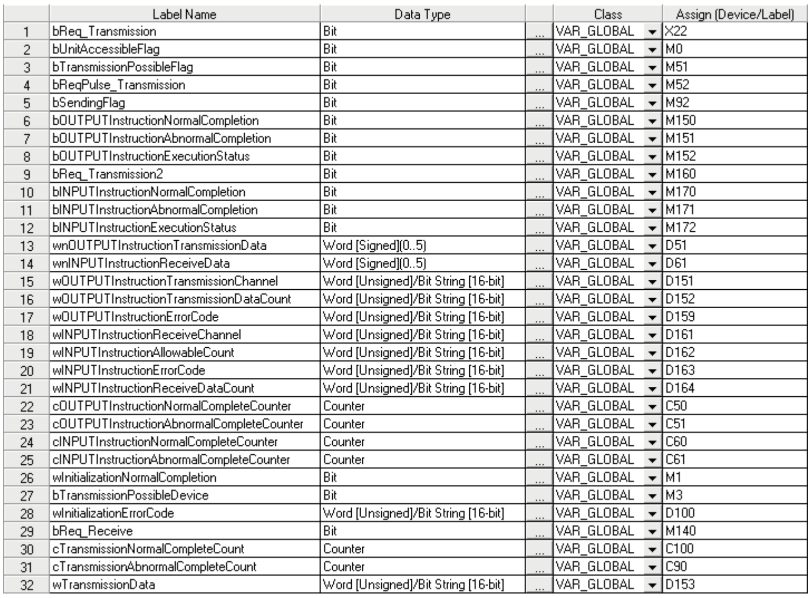
Perform data communication using the nonprocedural protocol by the command from the user after the 'Connection' (X12) turns ON.

■Parameter list

Category	Setting item			Setting content
Basic settings	Communication protocol setting			Nonprocedural protocol
	Communication speed setting			19200 bps
	Transmission setting	Data bit		8
		Parity bit		None
		Odd/even parity		Odd
		Stop bit		1
		Sum check code		Yes
		Online change		Enable
		Setting change		Enable
	Station number setting (CH1, 2 in common: 0 to 31)			0
Application settings	Modem function setting	Modem function specification 1	Modem connection CH specification	1CH
			Data No. for initialization designation	07D5H
		Modem function specification 2	Auto modem initialization designation	Automatically initialize

■Program example

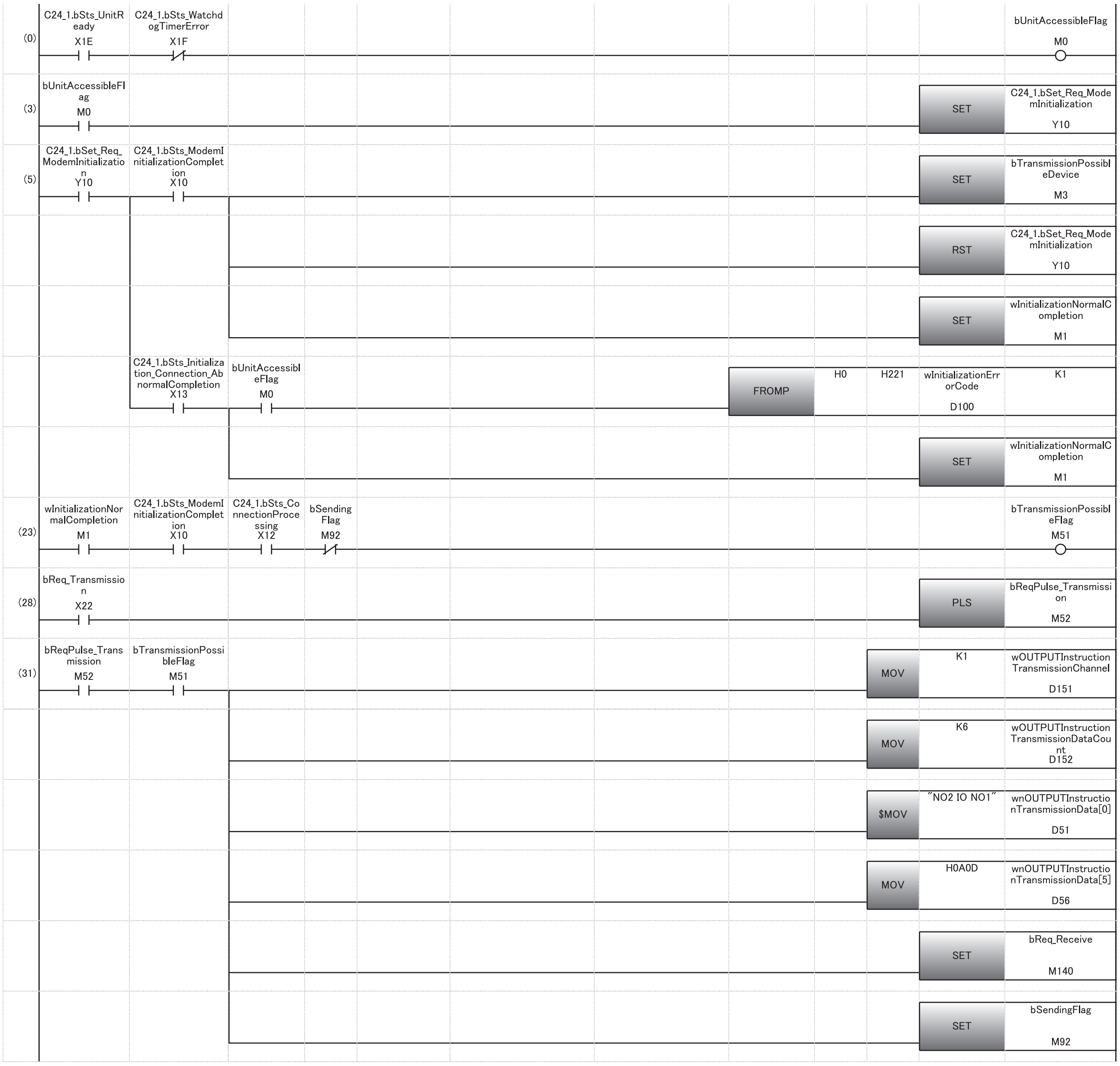
Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.bSts_TransmissionNormalCompletion	CH1 Transmission normal completion	X0
	C24_1.stnCH1.bSts_TransmissionAbnormalEnd	CH1 Transmission abnormal completion	X1
	C24_1.bSts_ModemInitializationCompletion	Modem initialization completion	X10
	C24_1.bSts_DialProcessing	Dialing	X11
	C24_1.bSts_ConnectionProcessing	Connection	X12
	C24_1.bSts_Initialization_Connection_AbnormalCompletion	Initialization/connection abnormal completion	X13
	C24_1.bSts_DisconnectionCompletion	Modem disconnection completion	X14
	C24_1.bSts_UnitReady	Module READY	X1E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X1F
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection	CH1 Reception abnormal detection	X4
	C24_1.stnCH1.bSet_Req_Transmission	CH1 Transmission request	Y0
	C24_1.stnCH1.bSet_Req_ReceptionReadCompletion	CH1 Reception data read completion	Y1
	C24_1.bSet_Req_ModemInitialization	Modem initialization request	Y10
	C24_1.bSet_Req_Connection	Connection request	Y11
	C24_1.bSet_Req_Disconnection	Modem disconnection request	Y12

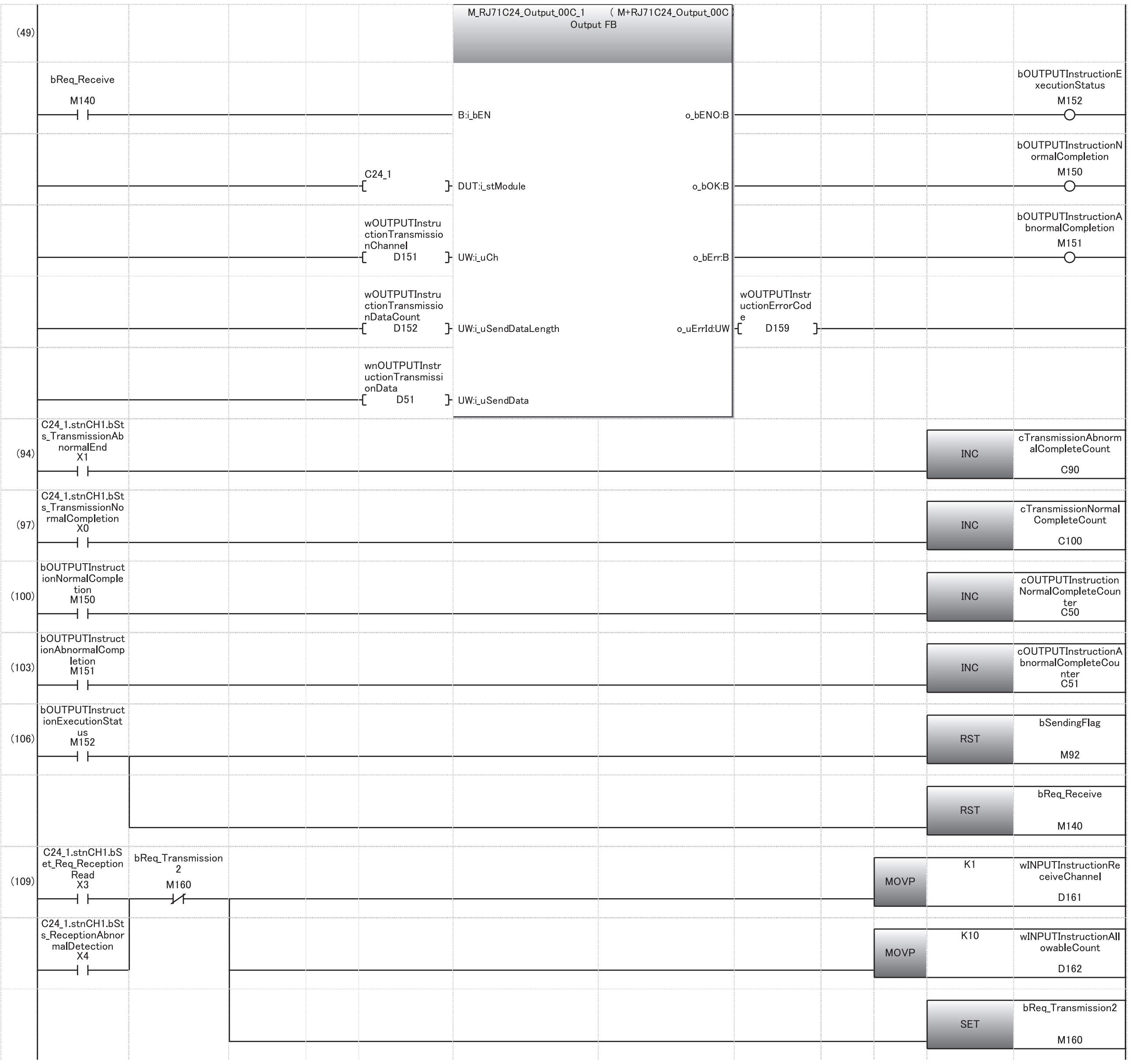
Category	Label name/FB name	Description	Device
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_Output	Sends data for specified data points.	—
	M+RJ71C24_Input	Reads the received data.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_Transmission	Bit	...	VAR_GLOBAL	X22
2	bUnitAccessibleFlag	Bit	...	VAR_GLOBAL	M0
3	bTransmissionPossibleFlag	Bit	...	VAR_GLOBAL	M51
4	bReqPulse_Transmission	Bit	...	VAR_GLOBAL	M52
5	bSendingFlag	Bit	...	VAR_GLOBAL	M92
6	bOUTPUTInstructionNormalCompletion	Bit	...	VAR_GLOBAL	M150
7	bOUTPUTInstructionAbnormalCompletion	Bit	...	VAR_GLOBAL	M151
8	bOUTPUTInstructionExecutionStatus	Bit	...	VAR_GLOBAL	M152
9	bReq_Transmission2	Bit	...	VAR_GLOBAL	M160
10	bINPUTInstructionNormalCompletion	Bit	...	VAR_GLOBAL	M170
11	bINPUTInstructionAbnormalCompletion	Bit	...	VAR_GLOBAL	M171
12	bINPUTInstructionExecutionStatus	Bit	...	VAR_GLOBAL	M172
13	wnOUTPUTInstructionTransmissionData	Word [Signed](0..5)	...	VAR_GLOBAL	D51
14	wnINPUTInstructionReceiveData	Word [Signed](0..5)	...	VAR_GLOBAL	D61
15	wOUTPUTInstructionTransmissionChannel	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D151
16	wOUTPUTInstructionTransmissionDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D152
17	wOUTPUTInstructionErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D159
18	wINPUTInstructionReceiveChannel	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D161
19	wINPUTInstructionAllowableCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D162
20	wINPUTInstructionErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D163
21	wINPUTInstructionReceiveDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D164
22	cOUTPUTInstructionNormalCompleteCounter	Counter	...	VAR_GLOBAL	C50
23	cOUTPUTInstructionAbnormalCompleteCounter	Counter	...	VAR_GLOBAL	C51
24	cINPUTInstructionNormalCompleteCounter	Counter	...	VAR_GLOBAL	C60
25	cINPUTInstructionAbnormalCompleteCounter	Counter	...	VAR_GLOBAL	C61
26	wInitializationNormalCompletion	Bit	...	VAR_GLOBAL	M1
27	bTransmissionPossibleDevice	Bit	...	VAR_GLOBAL	M3
28	wInitializationErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D100
29	bReq_Receive	Bit	...	VAR_GLOBAL	M140
30	cTransmissionNormalCompleteCount	Counter	...	VAR_GLOBAL	C100
31	cTransmissionAbnormalCompleteCount	Counter	...	VAR_GLOBAL	C90
32	wTransmissionData	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D153

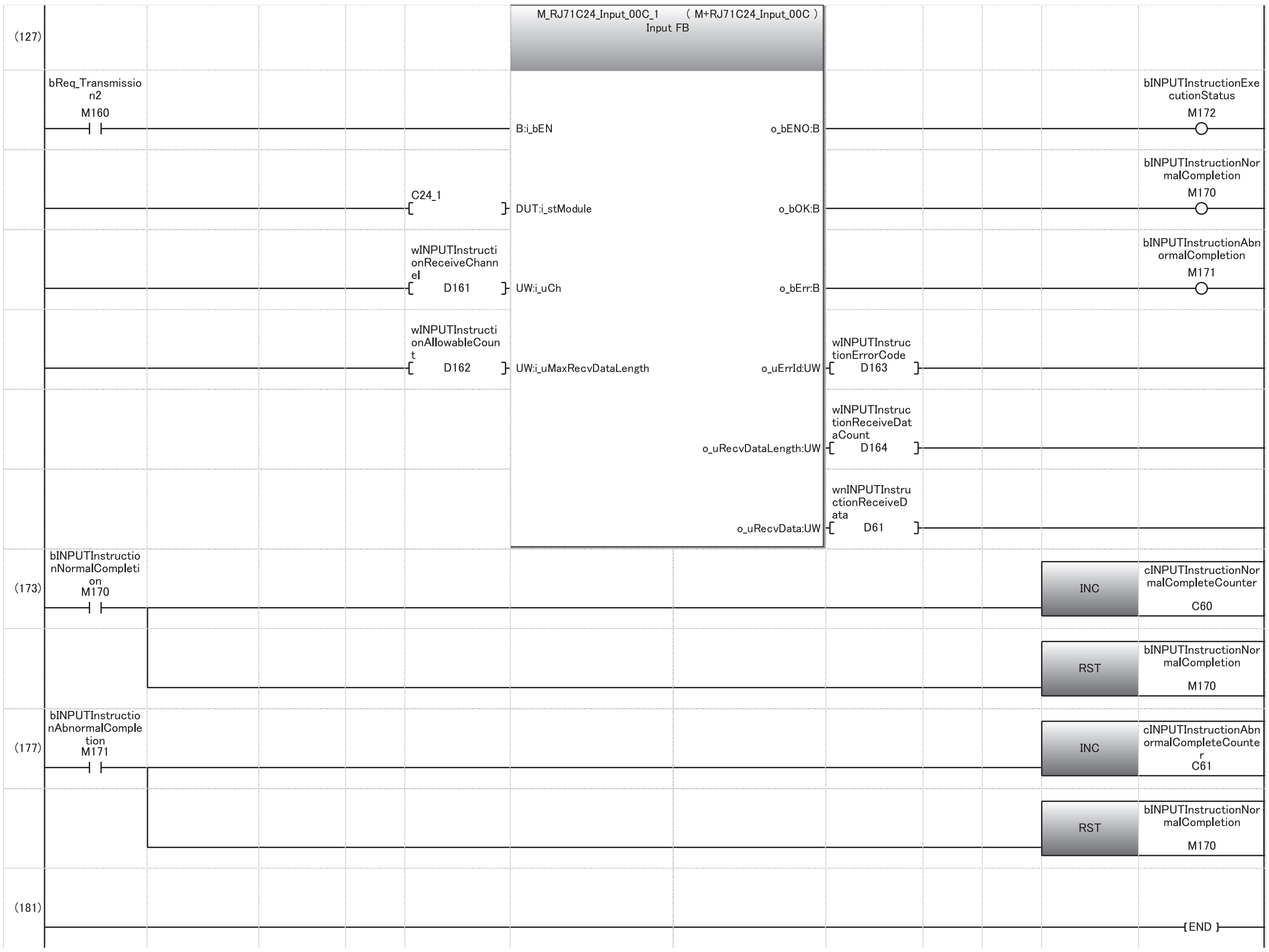
For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
C24_1.bSts_UnitReady
X1E
C24_1.bSts_WatchdogTimerError
X1F
bUnitAccessibleFlag
M0
(3)
bUnitAccessibleFlag
M0
SET
C24_1.bSet_Req_ModemInitialization
Y10
(5)
C24_1.bSet_Req_ModemInitialization
Y10
C24_1.bSts_ModemInitializationCompletion
X10
SET
bTransmissionPossibleDevice
M3
RST
C24_1.bSet_Req_ModemInitialization
Y10
SET
wInitializationNormalCompletion
M1
C24_1.bSts_Initialization_Connection_AbnormalCompletion
X13
bUnitAccessibleFlag
M0
FROMP
H0
H221
wInitializationErrorCode
D100
K1
SET
wInitializationNormalCompletion
M1
(23)
wInitializationNormalCompletion
M1
C24_1.bSts_ModemInitializationCompletion
X10
C24_1.bSts_ConnectionProcessing
X12
bSending Flag
M92
bTransmissionPossibleFlag
M51
(28)
bReq_Transmission
X22
PLS
bReqPulse_Transmission
M52
(31)
bReqPulse_Transmission
M52
bTransmissionPossibleFlag
M51
MOV
K1
wOUTPUTInstructionTransmissionChannel
D151
MOV
K6
wOUTPUTInstructionTransmissionDataCount
D152
$MOV
"NO2 IO NO1"
wnOUTPUTInstructionTransmissionData[0]
D51
MOV
H0A0D
wnOUTPUTInstructionTransmissionData[5]
D56
SET
bReq_Receive
M140
SET
bSendingFlag
M92



(49)
M_RJ71C24_Output_00C_1 （M+RJ71C24_Output_00C）
Output FB
bReq_Receive
M140
B:i_bEN
o_bENO:B
bOUTPUTInstructionExecutionStatus
M152
C24_1
DUT:i_stModule
o_bOK:B
bOUTPUTInstructionNormalCompletion
M150
wOUTPUTInstructionTransmissionChannel
D151
UW:i_uCh
o_bErr:B
bOUTPUTInstructionAbnormalCompletion
M151
wOUTPUTInstructionTransmissionDataCount
D152
UW:i_uSendDataLength
o_uErrId:UW
wOUTPUTInstructionErrorCode
D159
wnOUTPUTInstructionTransmissionData
D51
UW:i_uSendData
(94)
C24_1.stnCH1.bSts_TransmissionAbnormalEnd
X1
INC
cTransmissionAbnormalCompleteCount
C90
(97)
C24_1.stnCH1.bSts_TransmissionNormalCompletion
X0
INC
cTransmissionNormalCompleteCount
C100
(100)
bOUTPUTInstructionNormalCompletion
M150
INC
cOUTPUTInstructionNormalCompleteCounter
C50
(103)
bOUTPUTInstructionAbnormalCompletion
M151
INC
cOUTPUTInstructionAbnormalCompleteCounter
C51
(106)
bOUTPUTInstructionExecutionStatus
M152
RST
bSendingFlag
M92
RST
bReq_Receive
M140
(109)
C24_1.stnCH1.bSet_Req_ReceptionRead
X3
bReq_Transmission2
M160
MOVP
K1
wINPUTInstructionReceiveChannel
D161
C24_1.stnCH1.bSts_ReceptionAbnormalDetection
X4
MOVP
K10
wINPUTInstructionAllowableCount
D162
SET
bReq_Transmission2
M160


(127)
M_RJ71C24_Input_00C_1 (M+RJ71C24_Input_00C)
Input FB
bReq_Transmission2
M160
B:i_bEN
o_bENO:B
bINPUTInstructionExecutionStatus
M172
C24_1
DUT:i_stModule
o_bOK:B
bINPUTInstructionNormalCompletion
M170
wINPUTInstructionReceiveChannel
D161
UW:i_uCh
o_bErr:B
bINPUTInstructionAbnormalCompletion
M171
wINPUTInstructionAllowableCount
D162
UW:i_uMaxRecvDataLength
o_uErrId:UW
wINPUTInstructionErrorCode
D163
o_uRecvDataLength:UW
wINPUTInstructionReceiveDataCount
D164
o_uRecvData:UW
wnINPUTInstructionReceiveData
D61
(173)
bINPUTInstructionNormalCompletion
M170
INC
cINPUTInstructionNormalCompleteCounter
C60
RST
bINPUTInstructionNormalCompletion
M170
(177)
bINPUTInstructionAbnormalCompletion
M171
INC
cINPUTInstructionAbnormalCompleteCounter
C61
RST
bINPUTInstructionNormalCompletion
M170
(181)
END

(0) Accessible

(3) Initialize the modem.

(31) Set the transmission channel.
Set the send data quantity.
Set the send data.
Issue the transmission execution direction.

(100) Normal completion

(103) Abnormal completion

(109) Set the reception channel to CH1.
Issue the reception execution direction.

(173) Normal completion

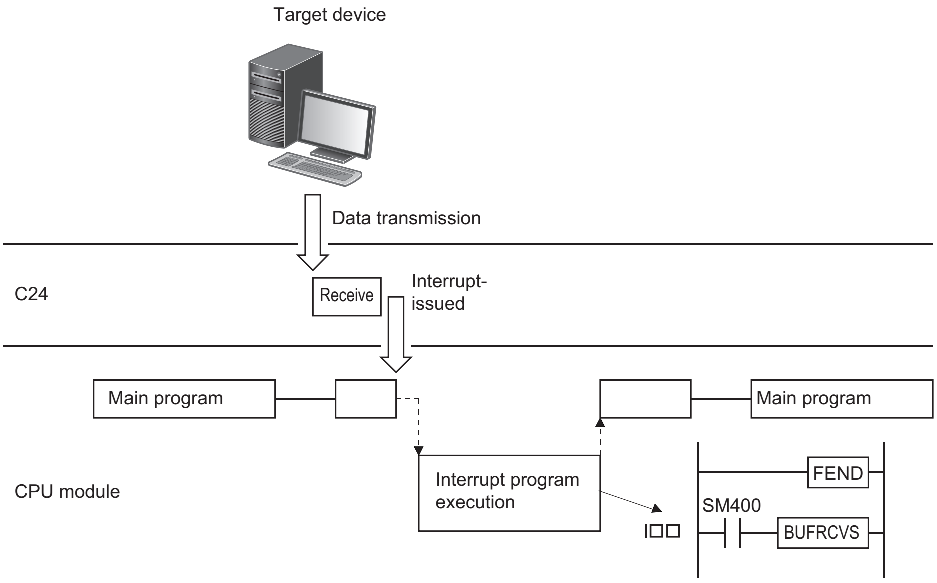
(177) Abnormal completion

10 RECEIVING DATA WITH AN INTERRUPT PROGRAM

In data communications between C24 and a target device, an interrupt program can be used to receive data for the following data communication functions.

- Data reception during communication using the nonprocedural protocol
- Data reception during communication using the bidirectional protocol

This chapter explains data reception with the following data communication function by an interrupt program.



Point

Receiving data with an interrupt program expedites the process of retrieving received data to the CPU module.

10.1 Settings for Receiving Data Using an Interrupt Program

The following explains the settings for performing data reception with an interrupt program during communication using the nonprocedural protocol or bidirectional protocol.

For setting for data reception with an interrupt program, set "Interrupt factor" to "BUFRCVS instruction data reception", and set the interrupt pointer to be used in the program to "Interrupt pointer".

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ Module Name ⇨ [Module Parameter] ⇨"Interrupt Settings"

10.2 Interrupt Program Startup Timing

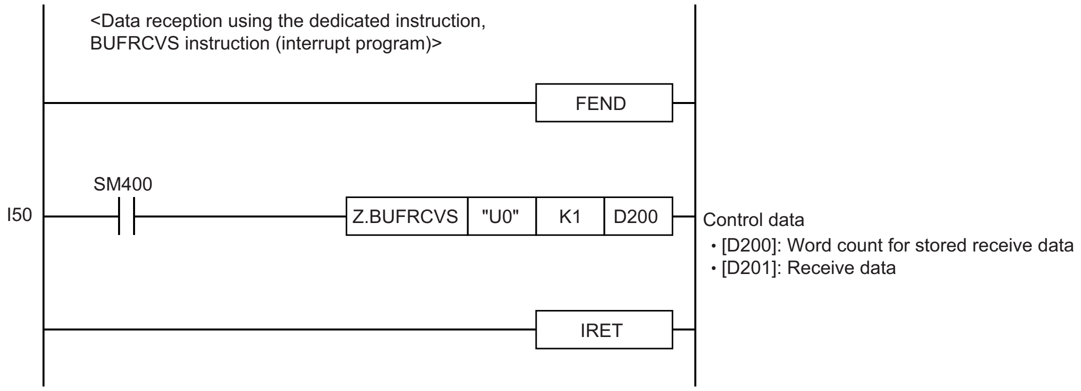
The following explains the startup timing for interrupt program when performing data reception with an interrupt program during communication using the nonprocedural protocol or bidirectional protocol.

- The startup timing is the same for communication using either the nonprocedural protocol or bidirectional protocol.
- Received data from the target device is stored in the receive area of the buffer memory. When the next input signal rises, the interrupt program is started.

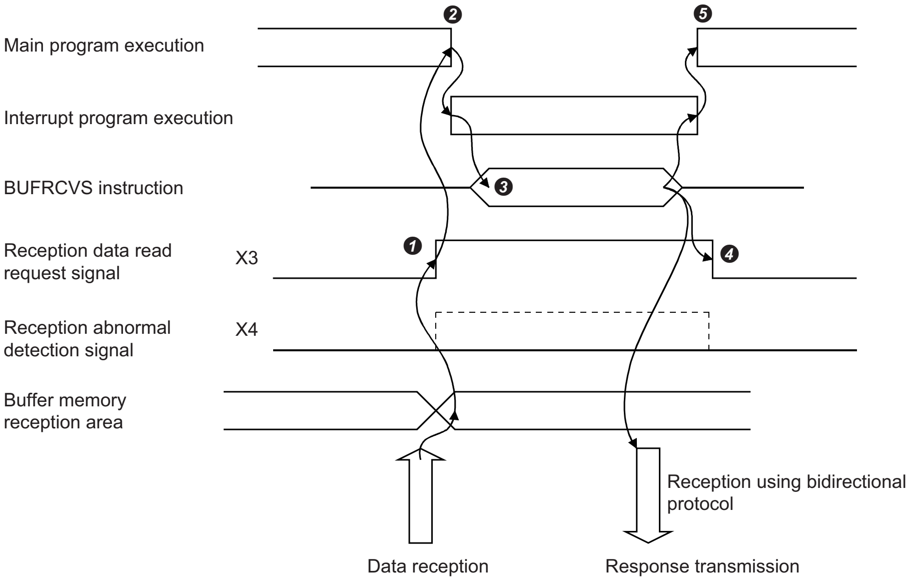
Input signal name	CH1 side	CH2 side
CHn reception data read request signal	X3	XA
CHn reception abnormal detection signal	X4	XB

10.3 Reception Control Method Using an Interrupt Program

The following explains the reception control method when receiving data with an interrupt program during communication using the nonprocedural protocol or bidirectional protocol.


<Data reception using the dedicated instruction, BUFRCVS instruction (interrupt program)>
FEND
SM400
I50
Z.BUFRCVS
"U0"
K1
D200
IRET

Control data
- [D200]: Word count for stored receive data
- [D201]: Receive data


❷
❺
Main program execution
Interrupt program execution
BUFRCVS instruction
❸
Reception data read request signal
X3
❶
❹
Reception abnormal detection signal
X4
Buffer memory reception area
Reception using bidirectional protocol
Data reception
Response transmission

❶ When data is received from the target device, the receive data is stored in the buffer memory and the 'CH1 reception data read request' signal turns ON.

❷ The main program stops executing and the interrupt program starts.

❸ The data reception dedicated instruction, BUFRCVS, for the interrupt program is executed and data is received.*1

❹ When the execution of the BUFRCVS instruction is complete, the 'CH1 reception data read request' signal turns OFF.

❺ The execution of the interrupt program is completed, and execution of the main program restarts.

*1 When the reading of received data using the BUFRCVS instruction is finished, the following processes are performed.
· At normal completion
CPU module error flag (SM0) turns OFF.
· At abnormal completion
CPU module error flag (SM0) turns ON.
The error code is stored in the CPU module error code (SD0).
For more details on the error flag (SM0) and error codes (SD0) of the CPU module, refer to the following manual.
📖MELSEC iQ-R CPU Module User's Manual (Application)

10.4 Program Example

This section shows programming examples for data reception with an interrupt program in data communications using the nonprocedural protocol or bidirectional protocol.

Program example for data reception

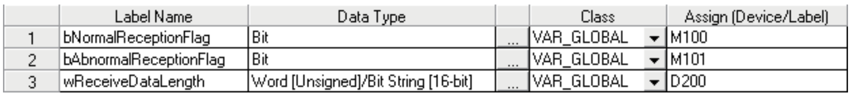
The following shows a program example for data reception with an interrupt program in data communication using the nonprocedural protocol.

■Parameter list

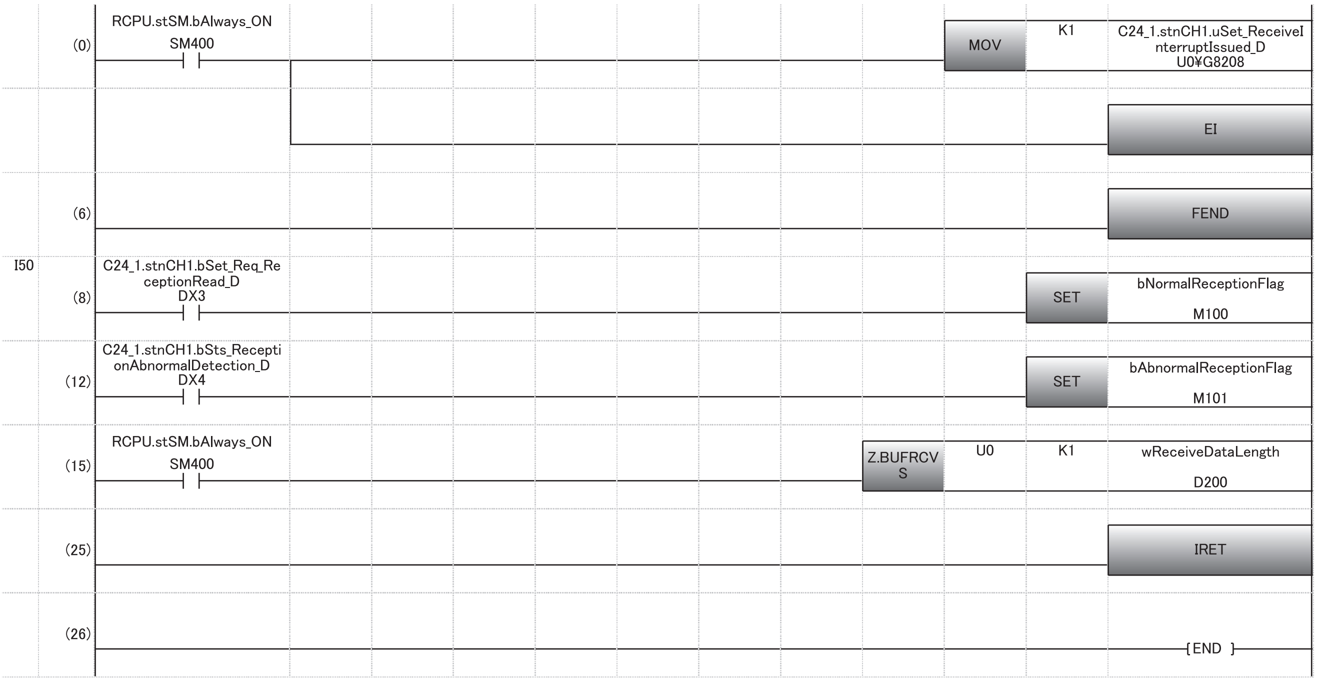
The following shows the parameters for this program.

Setting item			Setting content
Interrupt settings	CH1	Interrupt factor	BUFRCVS instruction data reception
		Interrupt pointer	I50

■Program example

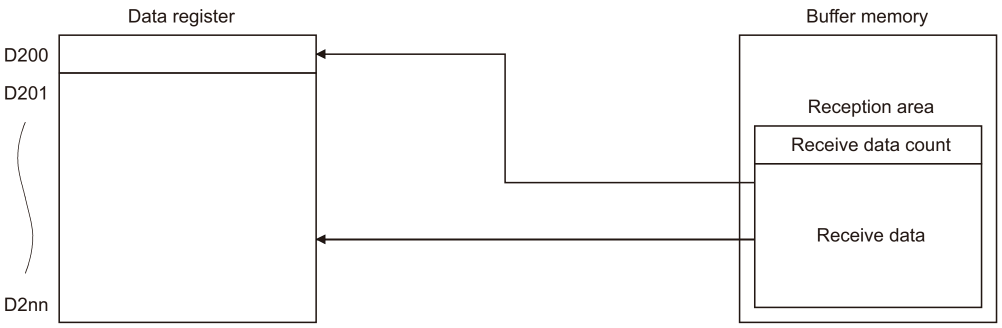
Category	Label name	Description	Device
Module label	C24_1.stnCH1.bSet_Req_ReceptionRead_D	CH1 Reception data read request	DX3
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection_D	CH1 Reception abnormal detection	DX4
	C24_1.stnCH1.uSet_ReceiveInterruptIssued_D	CH1_Receive interrupt-issued designation	U0\G8208
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bNormalReceptionFlag	Bit	...	VAR_GLOBAL	M100
2	bAbnormalReceptionFlag	Bit	...	VAR_GLOBAL	M101
3	wReceiveDataLength	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D200



(0) For the CH1 side, designate the receive interrupt-issued designation.

(15) Receive the data.



Point

- For communication using either the nonprocedural protocol or bidirectional protocol, BUFRCVS instruction is used to receive data with an interrupt program. (☞ Page 580 Dedicated Instructions)
- To startup an interrupt program, create a program to allow/prohibit interrupt during the main program. Instruction to be used is EI, DI, and IMASK.

10.5 Considerations when Receiving Data with an Interrupt Program

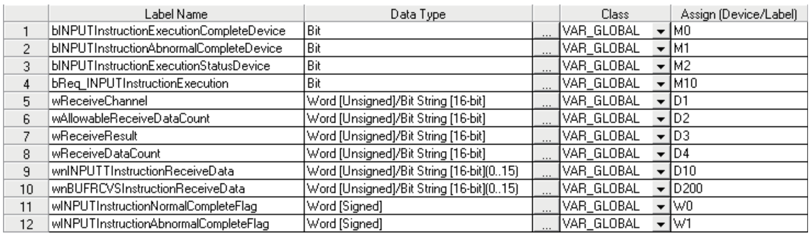
This section describes the considerations when receiving data with an interrupt program.

- Create an interrupt program for data reception for each interface.
- Do not use the INPUT and BIDIN instructions during execution of the interrupt program. Always use the BUFRCVS instruction to receive data.
- Do not turn the 'CH□ reception data read completion' (Y1/Y8) ON/OFF during execution of the interrupt program since C24 turns OFF the 'CH□ reception data read request' (X3/XA) and the 'CH□ reception abnormal detection' (X4/XB) for data reception with an interrupt program.
- Use always ON (SM400) or direct input signal (DX3, DX4) as the contact signal when executing the BUFRCVS instruction.
- After the power supply turns OFF ⇨ ON or the CPU module is reset, data cannot be received because the interrupt program is invalid during the initial processing of the CPU module. For asynchronous data communication with C24 from the target device without communication procedure setting, refer to the program example shown below.

Program example

The following shows an example of program for asynchronous data communications from the target device side to C24 without communication procedure settings on the target device side.

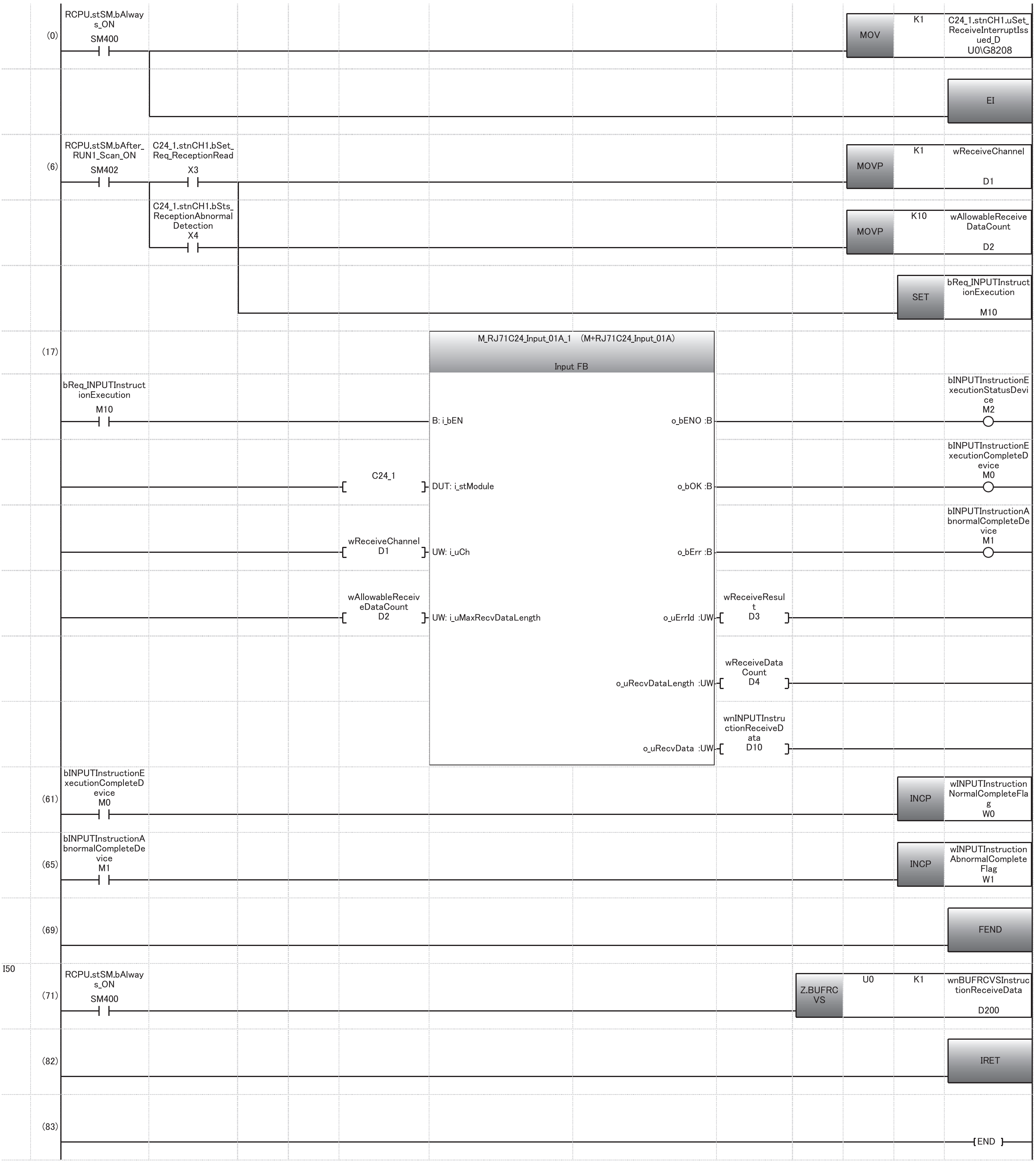
<table>
<tr><th>Category</th><th>Label name/FB name</th><th>Description</th><th>Device</th></tr>
<tr><td>Module label</td><td>C24_1</td><td>Target module</td><td>—</td></tr>
<tr><td></td><td>C24_1.stnCH1.bSet_Req_ReceptionRead</td><td>CH1 Reception data read request</td><td>X3</td></tr>
<tr><td></td><td>C24_1.stnCH1.bSts_ReceptionAbnormalDetection</td><td>CH1 Reception abnormal detection</td><td>X4</td></tr>
<tr><td></td><td>C24_1.stnCH1.uSet_ReceiveInterruptIssued_D</td><td>CH1_Receive interrupt-issued designation</td><td>U0\G8208</td></tr>
<tr><td>Label to be defined</td><td colspan="3">Define the global label as shown below.



<table>
<tr><th></th><th>Label Name</th><th>Data Type</th><th></th><th>Class</th><th>Assign (Device/Label)</th></tr>
<tr><td>1</td><td>bINPUTInstructionExecutionCompleteDevice</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M0</td></tr>
<tr><td>2</td><td>bINPUTInstructionAbnormalCompleteDevice</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M1</td></tr>
<tr><td>3</td><td>bINPUTInstructionExecutionStatusDevice</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M2</td></tr>
<tr><td>4</td><td>bReq_INPUTInstructionExecution</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M10</td></tr>
<tr><td>5</td><td>wReceiveChannel</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D1</td></tr>
<tr><td>6</td><td>wAllowableReceiveDataCount</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D2</td></tr>
<tr><td>7</td><td>wReceiveResult</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D3</td></tr>
<tr><td>8</td><td>wReceiveDataCount</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D4</td></tr>
<tr><td>9</td><td>wnINPUTTInstructionReceiveData</td><td>Word [Unsigned]/Bit String [16-bit](0..15)</td><td>...</td><td>VAR_GLOBAL</td><td>D10</td></tr>
<tr><td>10</td><td>wnBUFRCVSInstructionReceiveData</td><td>Word [Unsigned]/Bit String [16-bit](0..15)</td><td>...</td><td>VAR_GLOBAL</td><td>D200</td></tr>
<tr><td>11</td><td>wINPUTInstructionNormalCompleteFlag</td><td>Word [Signed]</td><td>...</td><td>VAR_GLOBAL</td><td>W0</td></tr>
<tr><td>12</td><td>wINPUTInstructionAbnormalCompleteFlag</td><td>Word [Signed]</td><td>...</td><td>VAR_GLOBAL</td><td>W1</td></tr>
</table>
</td></tr>
<tr><td>FB</td><td>M+RJ71C24_Input</td><td>Reads the received data.</td><td>—</td></tr>
</table>

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
RCPU.stSM.bAlways_ON
SM400
MOV
K1
C24_1.stnCH1.uSet_ReceiveInterruptIssued_D
U0\G8208
EI
(6)
RCPU.stSM.bAfter_RUN1_Scan_ON
SM402
C24_1.stnCH1.bSet_Req_ReceptionRead
X3
MOVP
K1
wReceiveChannel
D1
C24_1.stnCH1.bSts_ReceptionAbnormalDetection
X4
MOVP
K10
wAllowableReceiveDataCount
D2
SET
bReq_INPUTInstructionExecution
M10
(17)
M_RJ71C24_Input_01A_1 (M+RJ71C24_Input_01A)
Input FB
bReq_INPUTInstructionExecution
M10
B: i_bEN
o_bENO :B
bINPUTInstructionExecutionStatusDevice
M2
C24_1
DUT: i_stModule
o_bOK :B
bINPUTInstructionExecutionCompleteDevice
M0
wReceiveChannel
D1
UW: i_uCh
o_bErr :B
bINPUTInstructionAbnormalCompleteDevice
M1
wAllowableReceiveDataCount
D2
UW: i_uMaxRecvDataLength
o_uErrId :UW
wReceiveResult
D3
o_uRecvDataLength :UW
wReceiveDataCount
D4
o_uRecvData :UW
wnINPUTInstructionReceiveData
D10
(61)
bINPUTInstructionExecutionCompleteDevice
M0
INCP
wINPUTInstructionNormalCompleteFlag
W0
(65)
bINPUTInstructionAbnormalCompleteDevice
M1
INCP
wINPUTInstructionAbnormalCompleteFlag
W1
(69)
FEND
I50
(71)
RCPU.stSM.bAlways_ON
SM400
Z.BUFRCVS
U0
K1
wnBUFRCVSInstructionReceiveData
D200
(82)
IRET
(83)
END

(0) For the CH1 side, designate the receive interrupt-issued designation.

(61) Processing at normal completion

(65) Processing at abnormal completion

(71) Execute the receive interrupt program.

11 CHANGING DATA LENGTH UNITS TO BYTE UNITS (WORD/BYTE UNITS SETTING)

The word units are used for the data length (count) of the amount of data transmitted/received using the following data communication functions in data communication between C24 and target devices.

This chapter explains how to change the units (word → byte, byte → word) of the data length (count) transmitted/received with the following data communication functions.

The data length units can be set for each C24 interface. C24 controls the number of data to be transmitted to the target device and the number of data when it requests the CPU module to read the data received from the target device according to the units set by the user.

Data communication functions and buffer memory affected by data length units

The following lists the data communication functions and buffer memory which are affected by the data length units.

(The buffer memory addresses in the table are the default value.)

Data communication function		Name of buffer memory related to data length units	Reference
MC protocol	On-demand function	Data length designation (Un\G161/321)	MELSEC Communication Protocol Reference Manual
Nonprocedural protocol	Data transmission function	Send data quantity designation (Un\G1024/2048)	Page 56 DATA COMMUNICATION USING NONPROCEDURAL PROTOCOL
	Data reception function	Receive end data quantity designation (Un\G164/324) Receive data quantity (Number of data for which read is requested) (Un\G1536/2560)	
Bidirectional protocol	Data transmission function	Send data quantity designation (Un\G1024/2048)	Page 89 DATA COMMUNICATION USING BIDIRECTIONAL PROTOCOL
	Data reception function	Receive end data quantity designation (Un\G164/324) Receive data quantity (Number of data for which read is requested) (Un\G1536/2560)	

How to change the units of the data length (count)

Change the units of the data length (count) by the either of the following methods.

- Changing with Engineering tool

  Set "Word/byte units designation" to "Byte units".

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

- Changing by the CPU module

  The units can be changed with the CSET instruction. (☞ Page 580 Dedicated Instructions)

MEMO

12 CHANGING DATA COMMUNICATION MONITORING TIME

The monitoring time is time used by C24 to monitor the receiving interval time between each byte when receiving data from the target device, the programmable controller CPU processing time, and the time it takes to transmit to the target device.

The monitoring time can be set for each interface. C24 uses the monitoring time set by the user to control data transmission to and reception from the target device.

Set the monitoring time in accordance with the specifications of the target device.

C24 monitoring times are listed below.

○: Configurable ×: Non-configurable, —: Not used

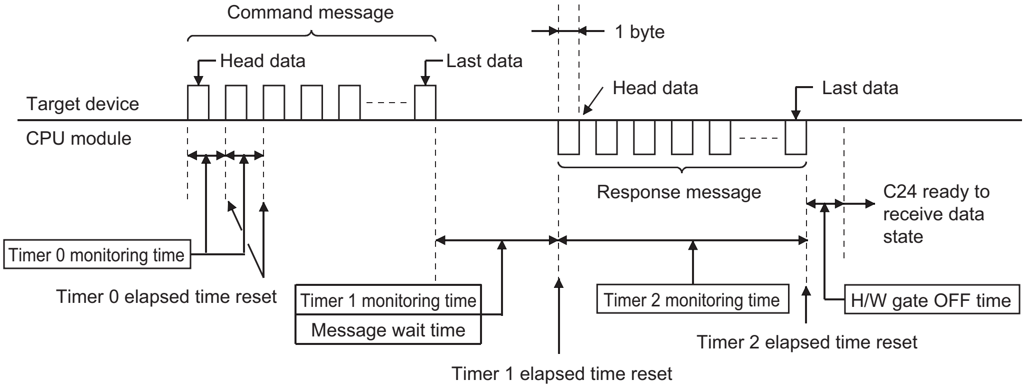
Monitoring time		C24 default value	Protocol that can monitor the time				Remarks
			MC	Non	Bi	Pd	
No-reception monitoring time (timer 0)	Method 0[*1]	0 bytes (Unlimited wait)	○	○	○	—	Transmission time for the set number of byte (depending on the transmission speed)
	Method 1[*1]		×	○	×	—	
Response monitoring time (timer 1)		5 seconds	○	—	○	○	Transmission only for the bidirectional protocol.
Send monitoring time (timer 2)		3 minutes	○	○	○	—	—
Transmission wait time		0 ms	○	—	—	—	No wait time

[*1] Nonprocedural and no-reception monitoring time method
The C24 input signal which turns ON when the no-reception monitoring time (timer 0) timed out differs.

Nonprocedural and no-reception monitoring time method	Input signal which turns ON
Method 0	Reception abnormal detection (X4, XB)
Method 1	Reception data read request (X3, XA)

Ex.

Data communication using MC protocol

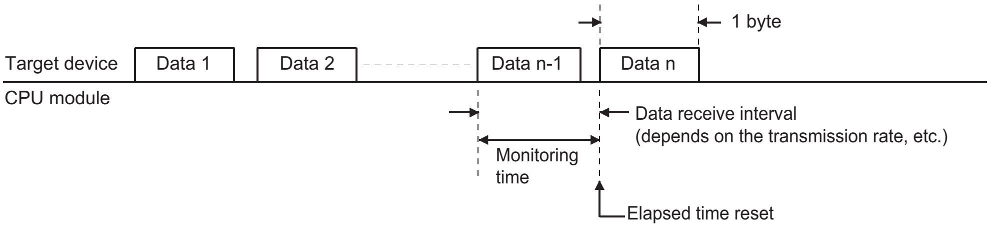


12.1 No-reception Monitoring Time (timer 0) Setting

The no-reception monitoring time (timer 0) is a monitoring time for clearing the state of C24 when it is placed into the data receive wait state by trouble on the target device side.

C24 monitors the reception interval in byte units at the start of data reception from the target device and ends monitoring when the preset last data is received, and it repeats this operation.

The following explains the no-reception monitoring time (timer 0) operation.



One byte is always handled as 12 bits in the no-reception monitoring time (timer 0), regardless of the transmission setting.

When the no-reception monitoring time (timer 0) is changed by a program, perform any of the following to enable the changed value.

- Mode switching (☞ Page 409 SWITCHING THE MODE AFTER STARTING)
- UINI instruction (📖MELSEC iQ-R Programming Manual (Module Dedicated Instructions))
- Programmable controller CPU information clear (☞ Page 498 How to Clear Programmable Controller CPU Information)

C24 operation by no reception monitoring time (timer 0)

C24 monitors the reception interval in byte units and returns the elapsed time to 0 each time one byte is received.

At time-out, C24 performs the following processing.

Data communication using the MC protocol

- C24 stores the error code to the 'MC protocol send error code' (Un\G602/618) for the target interface side.
- C24 transmits a NAK message to the target device and waits to receive the next command message.

Data communications using the nonprocedural protocol (Method 0)

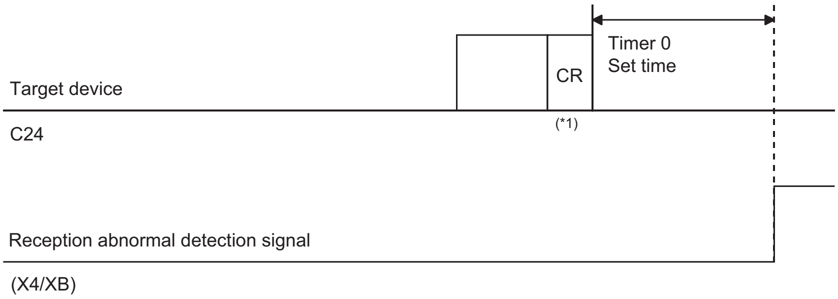
■Data communications not using user frames

- C24 retrieves the received data up to time-out to C24.
- C24 stores the error code to the 'Data reception result' (Un\G600/616) for the target interface and turns ON the 'CH□ reception abnormal detection' (X4, XB), and waits to receive the next data.

Ex.

Receiving by the receive end code (Receive end code: CR + LF (0D0AH))

When the LF is not received within the set time for timer 0 after reception of the CR, the received data up to the CR is stored in the receive data storage area of the buffer memory and the reception abnormal detection signal to the CPU module turns ON.



*1 CR is treated as 1 byte of data included in the message.

■Data communications using user frames

- When designating the final frame, C24 retrieves an arbitrary portion of data in the received data between the start of the reception of the current message and the time-out, and ignores (deletes) the data in the final frame portion.
- C24 stores the error code to the 'Data reception result' (Un\G600/616) for the target interface and turns ON the 'CH□ reception abnormal detection' (X4, XB) and waits to receive the next data.

Data communication using the nonprocedural protocol (Method 1)

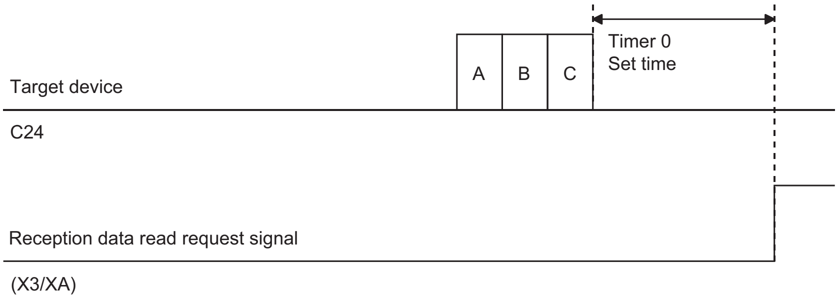
The reception monitoring using Method 1 of the no-reception monitoring time (timer 0) is used to receive a message of which receive end code and receive end data quantity are not determined due to time-out of the no-reception monitoring time (timer 0) in data communication using the nonprocedural protocol.

- Data is received by C24 until the time is out, the 'CH□ reception data read request' (X3, XA) turns ON, and waits to receive the next data.

Ex.

When receiving using the receive end data quantity (Receive end data quantity: 511 bytes)

After 16 bytes of data is received, the 16-byte data is stored in the receive data storage area of the buffer memory by the time out (timer 0) and the reception data read request signal to the CPU module turns ON.



Data communications using the bidirectional protocol

- C24 ignores (deletes) the received data from the start of reception of the current message to time-out.
- C24 stores the error code to the 'Data reception result' (Un\G600/616) for the target interface.
- When reading received data is completed, C24 transmits a NAK message to the target device, and waits to receive the next data.

Changing the no-reception monitoring time (timer 0)

Changing the no-reception monitoring time (timer 0)

For the no-reception monitoring time (timer 0), set the number of transmission characters (number of bytes) corresponding to the data transmission speed set to the target interface to "No-reception monitoring time (timer 0) designation".

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

Adjust or set the value based on the specifications of the target device.

Nonprocedural no-reception monitoring time method designation (Method 0/1)

For the nonprocedural and no-reception monitoring time method designation, set the method to use the no-reception monitoring time (timer 0) in data communications using the nonprocedural protocol to "Nonprocedural and no-reception monitoring time method designation".

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

Remarks

■When changing the no-reception monitoring time (timer 0)

Set the no-reception monitoring time (timer 0) to the following number of bytes or greater.

$$\text{No-reception monitoring time (timer 0)} = 1 + \frac{Td \times Vbps}{12000}$$

(Round up fractions below decimal point.)

Td : Maximum delay time for target device output processing (ms)

Vbps: Transmission rate (bps)

Ex.

Calculation of no-reception monitoring time (timer 0)

Transmission rate (Vbps) : 9600 bps

Maximum delay time for target device output processing (Td) : 50 ms

$$\text{No-reception monitoring time (timer 0)} = 1 + \frac{50 \times 9600}{12000} = 41 \text{ bytes}$$

In this case, actual monitoring time is as shown below:

41 bytes × 12[*1] ÷ 9600 × 1000 = 51.25 ms

[*1] Number of transmit bits per byte (Fixed)

■When changing the no-reception monitoring time (timer 0) for data communication with a target device through C24 RS-422/485 interface

Set the no-reception monitoring time (timer 0) to the following number of bytes or greater.

$$\text{No-reception monitoring time (timer 0)} = 1 + \frac{(Td + T1) \times Vbps}{12000}$$

(Round up fractions below decimal point.)

Td : Maximum delay time for target device output processing (ms)

T1 : Target device side hardware gate OFF time (ms)

Vbps: Transmission rate (bps)

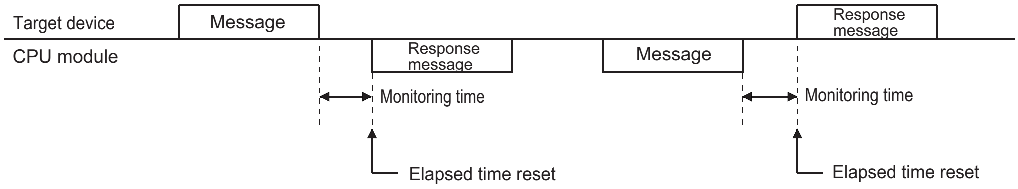
12.2 Response Monitoring Time (timer 1) Setting

The response monitoring time (timer 1) is a monitoring time for clearing the receive wait state of the device that receives a response message when any trouble occurs on the side of device which received a message and the response message (result) cannot be returned to the target device.

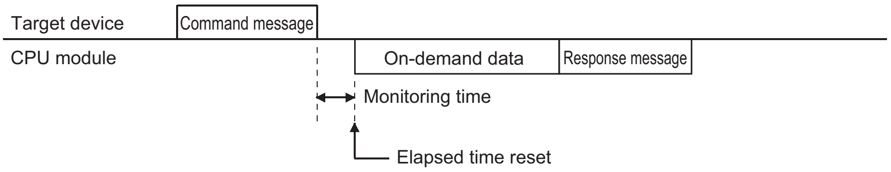
When C24 receives a message from the target device, it monitors the CPU module processing time up to the start of transmission of the response message to the target device.

When a message was transmitted, it monitors the target device processing time up to the start of reception of the response message from the target device.

The following describes the response monitoring time (timer 1) operation.



If on-demand data is transmitted before a response message during data communications using the MC protocol, the time up to the start of transmission of the on-demand data is monitored.



C24 operation by response monitoring time (timer 1)

When response monitoring time (timer 1) is set to 0 ms

- After receiving a message, C24 does not monitor the time up to the start of transmission of the response message to the target device, but waits infinitely.
- After transmitting a message, C24 does not monitor the time up to the start of reception of the response message from the target device, but waits infinitely.

When response monitoring time (timer 1) is set to 100 ms or longer

- After receiving a message, C24 monitors the time up to the start of transmission of the response message to the target device and returns the elapsed time to 0 at the start of transmission.
- After transmitting a message, C24 monitors the time up to the start of reception of the response message from the target device and returns the elapsed time to 0 at the start of reception.

At time-out, C24 performs the following processing.

■Data communications using the MC protocol

- C24 stores the error code to the 'MC protocol send error code' (Un\G602/618) for the target interface.
- C24 transmits a response message (NAK message) to the target device and waits to receive the next command message.

■Data communications using the bidirectional protocol

- C24 stores the error code to the 'Data transmission result' (Un\G599/615) for the target interface and completes the transmission processing abnormally.
- While waiting to transmit a response message, C24 does not check the response monitoring time.

Changing the response monitoring time (timer 1)

The response monitoring time (timer 1) is registered on "Response monitoring time (timer 1) designation" of the parameter.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

During data communications using the MC protocol, set the response monitoring time so as to be as long as the transmission wait time or longer.

The transmission wait time is designated in the following:

- 1C frame: Designate it is a command message.
- 2C/3C/4C frame: "Transmission wait time designation"

Considerations for data communications using the MC protocol

When changing the response monitoring time (timer 1) default value (5 sec), take into account the following.

Set it as to be as long as the transmission wait time or longer in either case of the following. (☞ Page 267 Transmission Wait Time Setting)

■When specifying monitor condition

When designating the monitor conditions with the following functions, set the maximum time according to the system operation.

- Random read in word units
- Device memory monitor

■When accessing by other than above

- When accessing a station connected (including multidrop connection) to a target device, set the following value, or longer.
  Response monitoring time ≥ Maximum number of scans required to process the command used × Connected station scan time
- When accessing another station over a network system, set the default value to 'Unlimited wait', or the following time or longer.
  Response monitoring time ≥ maximum number of scans required to process the command used × communications time

When setting the default value to 'Unlimited wait', check the target device response wait time, and initialize C24 transmission sequence when time-out occurs.

For the maximum number of scans and the communication time required by processing, refer to the following manual.

📖MELSEC Communication Protocol Reference Manual

Considerations for data communications using bidirectional protocol

when changing the default value (5 seconds) for the response monitoring time (timer 1), set it as to be as long as the following time, or longer.

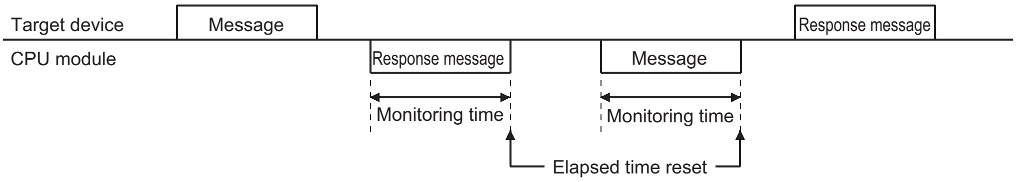
- (Sequence scan time × 2) + 100 ms

12.3 Send Monitoring Time (timer 2) Setting

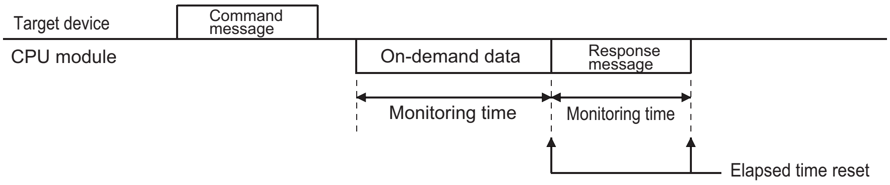
The send monitoring time (timer 2) is a monitoring time for clearing the wait state of C24 when C24, which is to transmit a message or response message (result), is placed into the data transmit wait state due to any trouble of the target device side When C24 transmits a message, it monitors the wait time up to the end of transmission of the message.

When C24 received a message from a target device, it monitors the wait time up to the end of transmission of the response message.

The following explains the send monitoring time (timer 2) operation.



If on-demand data is transmitted before a response message during data communications using the MC protocol, each time is monitored.



C24 operation by send monitoring time (timer 2)

When send monitoring time (timer 2) is set to 0 ms

- The time until transmission of a message or response message has been completed is not monitored.
- If C24 cannot transmit, it waits infinitely.

When send monitoring time (timer 2) is set to 100 ms or longer

- C24 monitors the time from completion of message or response message transmission preparations to the end of transmission, and returns the elapsed time to 0 at the end of transmission.

At time-out, C24 performs the following processing.

■Data communications using MC protocol

- While waiting for the end of transmission of the response message, C24 stores the error code to the 'Data transmission result' (Un\G599/615) for the target interface.
  C24 enters the state in which it waits to receive the next command message without sending a response message (NAK message) to the target device.
- During on-demand data transmission, C24 stores the error code to the 'On-demand execution results' (Un\G598/614) for the target interface.
- If transmission was terminated midway in either of the cases above, C24 does not transmit the remaining data.

■Data communications using nonprocedural protocol or bidirectional protocol

- While waiting for the end of transmission of a message, C24 stores the error code to the 'Data transmission result' (Un\G599/615) for the target interface and completes the transmission processing abnormally.
  If message transmission was terminated midway, C24 does not transmit the remaining data.
- If waiting for the end of transmission of a response message, C24 stores the error code to the 'Data reception result' (Un\G600/616) for the target device and turns ON the 'CH□ reception abnormal detection' (X4, XB).[*1]
  When the receive data read processing completes, C24 turns OFF the 'CH□ reception abnormal detection' (X4, XB) and waits to receive the next command.
  If transmission of the response message was terminated midway, C24 does not transmit the remaining data.

[*1] When communicating using bidirectional protocol, it stores the error code in the data reception result storage area for the target interface. (The reception abnormal detection signal does not turn ON.)

Changing the send monitoring time (timer 2)

The send monitoring time (timer 2) is set in "Send monitoring time (timer 2) designation" of the parameter.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

The send monitoring time (timer 2) monitors the transmission termination time in the condition below.

- When DTR/DSR signal control is used and the DR (DSR) signal is turned OFF (☞ Page 269 Control Contents of DTR/DSR (ER/DR) Signal Control)
- When DC1/DC3 receive control is used and DC3 is received (☞ Page 271 Control Contents of DC Code Control)
- When the RS-232 interface CS (CTS) signal is turned OFF (📖MELSEC iQ-R Serial Communication Module User's Manual(Startup))

Remarks

■When changing the send monitoring time (time 2) setting

Find the send monitoring time (timer 2) from the maximum delay time for the message reception processing or response message transmission processing on the target device side and the transmission time per one-byte (t), and change the setting value. However, the value should be rounded to the nearest 100 ms above.

- Number of bytes transmitted per second (n) = Transmission rate/number of bits transmitted per byte
- Transmission time per byte (t) = 1000 (ms)/number of bytes transmitted per second (n)
- Send monitoring time (timer 2) = (Maximum processing delay time of target device) + (Transmission time per one-byte (t) × Number of transmission bytes)

Under the following conditions, the send monitoring time (timer 2) is set to 300 ms.

- Transmission rate : 9600 bps
- Number of transmit bits/byte : 11 (start bit: 1, data bits: 8, stop bits: 2)
- Maximum processing delay time : 200 ms
- Number of transmission bytes: 3 bytes

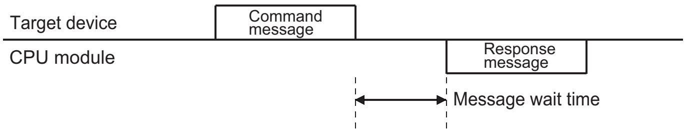
12.4 Transmission Wait Time Setting

The transmission wait time is used during data communications using the MC protocol.

It is the time for a target device that cannot receive the data immediately after data has been transmitted.

When C24 transmits a response message in reply to a command message received from the target device, transmission of the response message is delayed by the transmission wait time, or longer.

The following explains the transmission wait time operation for data communications using 2C/3C/4C frame. (For 1C frame, the transmission wait time is designated in the command message.)



C24 operation by transmission wait time

When transmission wait time is set to 0 ms

If a response message can be transmitted, C24 immediately transmits the response message. A transmission wait time is not set.

When the transmission wait time is set to 10 ms or longer

If a response message can be transmitted, and the transmission wait time after reception of the command message has elapsed, C24 transmits the response message.

Changing the transmission wait time

The transmission wait time is registered on "Transmission wait time designation" of the parameter.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

- If the target device that must wait a certain time before it can receive a response message after a command message was transmitted, set the transmission wait time as explained above.
  Especially, for data communications with a target device connected to C24 RS-422/485 interface, set the transmission wait time to the hardware gate OFF time or longer of the target device.
- The transmission wait time described here is the time for data communications using the 2C/3C/4C frame.

13 DATA COMMUNICATIONS USING DC CODE TRANSMISSION CONTROL

The transmission control function controls (suspends, restarts) the transmission and reception of data between C24 and target device by turning transmission control signals ON and OFF or by transmitting and receiving DC codes (DC1, DC2, DC3, DC4), or informs the valid range for the data to the target device.

The transmission control function can be set for each C24 interface. C24 uses the transmission control function set by the user to control data communications with target devices.

Set the transmission control function to match the specifications of the target device.

The table below lists C24 transmission control functions.

The meanings of abbreviations in the table are as follows:

- MC: MC protocol
- Non: Nonprocedural protocol
- Bi: Bidirectional protocol
- Pd: Predefined protocol
- MD: MODBUS (slave function)
- S: Simple CPU communication

(○: Valid, —: Invalid)

Transmission control function	Type of control	Interface that can be controlled		Protocol that can be controlled						Remarks
		232	422/485	MC	Non	Bi	Pd	MD	S	
DTR/DSR signal control[1]	DTR control	○	— (ignore)	—	○	—	○	—	○	Cannot be used simultaneously with DC control. Select either one.
	DSR control			○	○	—	○	—	○	
RS/CS signal control[2]		○	—	○	○	○	○	○	○	Normally controlled.
CD signal control[2]		○	—	○	○	○	○	—	○	The cable wiring depends on whether or not control is used. With half-duplex communications, control is necessary.
DC code control[1]	DC1/DC3 transmission control	○	○	—	○	—	○	—	○	Cannot be used simultaneously with DTR/DSR signal control. Select either one.
	DC1/DC3 reception control			○	○	○	○	—	○	
	DC2/DC4 transmission control			○	○	○	○	—	○	
	DC2/DC4 reception control			○	○	○	○	—	○	

[1] Refer to the following section when the full-duplex communication is used to communicate data with the bidirectional protocol.
☞ Page 107 Processing when Simultaneous Transmission is Performed during Full-Duplex Communications

[2] Check the operation on C24 by the RS(RTS) signal, control contents of CS(CTS) signal, or CD terminal check designation.
📖MELSEC iQ-R Serial Communication Module User's Manual(Startup)

Precautions

- When C24 is started, DTR/DSR signal control and RS/CS signal control are enabled.
- For simple CPU communication, a setting for communication depends on the target device. Set it according to the target device. (☞ Page 121 Data Communication Procedure)

How to change the transmission control method and DC code

Switching between DC code control and DTR/DSR control as well as changing of the DC code are set on "Transmission control specification" of the parameter.

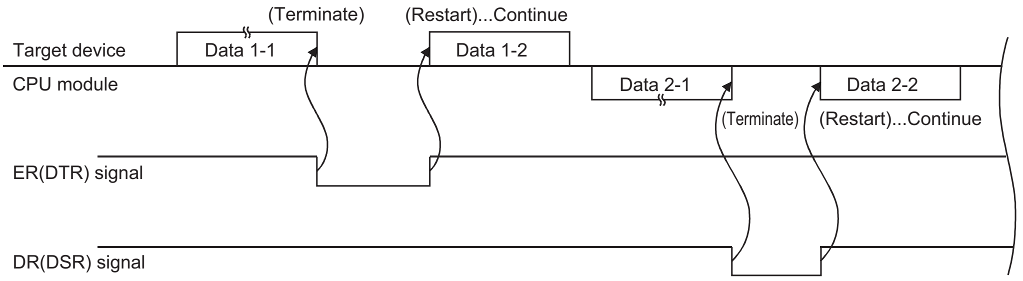
Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

13.1 Control Contents of DTR/DSR (ER/DR) Signal Control

This control uses the RS-232 interface DTR/DSR signals to inform the target device whether or not the host station is ready to receive data.

C24 uses the ER (DTR) signal to inform the target device whether or not the host station is ready to receive data, and uses the DR (DSR) signal to check if the target device is ready to receive data.

Ex.



C24 DTR control contents and free OS area specification

C24 DTR control contents

C24 uses the ER (DTR) signal to inform the target device whether or not it is ready to receive data.

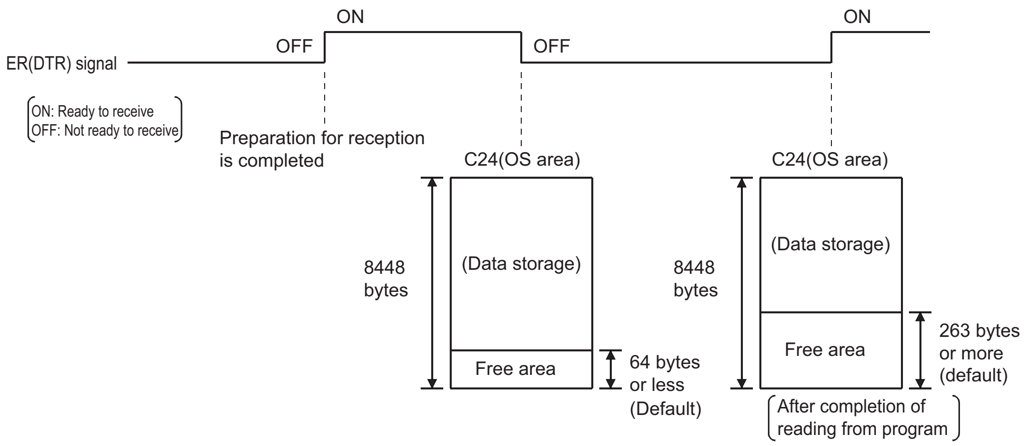
The data transmitted from the target device using the nonprocedural protocol is stored in the receive data storage area in the buffer memory through the OS area. (☞ Page 57 Receiving Data from Target Device)

Under the following conditions, received data is temporarily stored in the OS area and is transferred to the receive data storage area when the current received data read processing is completed.

- Received data exceeding the size of the receive data storage area on the buffer memory when data, which is "receive data storage area < receive data length", was received.
- Data received before the program reads the previously received data.

C24 turns the ER (DTR) signal ON/OFF as listed below, depending on the size of the free OS area.

- Free area is 64 bytes (default) or less: OFF
- Free area is 263 bytes (default) or more: ON


ER(DTR) signal
ON: Ready to receive
OFF: Not ready to receive
OFF
ON
OFF
ON
Preparation for reception is completed
C24(OS area)
8448 bytes
(Data storage)
Free area
64 bytes or less (Default)
C24(OS area)
8448 bytes
(Data storage)
Free area
263 bytes or more (default)
After completion of reading from program

Free OS area specification

For the DTR/DSR (ER/DR) signal control, an available capacity in the OS area that informs data cannot be received can be set in the parameter below.

🖱 Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

Set the setting value as to be "Transmission control start free space < Transmission control end free space".

● "Transmission control start free space designation"

Specify an available capacity in the OS area in order to notify that data cannot be received (ER (DTR) signal is OFF).

- Transmission control start free space: 64 to 4,095 (Default: 64)

● "Transmission control end free space designation"

Specify an available capacity in the OS area in order to notify that data can be received (ER (DTR) signal is ON).

- Transmission control end free space: 263 to 4,096 (Default: 263)

■Remarks

- The receive data clear processing clears the data stored in the OS area. (The receive area in the buffer memory is not cleared.) (☞ Page 70 Receive data clear)
- If more data is received when the above free OS area is 0 bytes, an SIO error occurs and the received data is ignored until a free area is available. In this case, the SIO signal turns ON. (☞ Page 511 Hardware information communication error "SIO" occurs)

C24 DSR control

C24 uses the DSR signal to detect whether or not the target device is ready to receive data and to control data transmission to the target device as shown below, depending on whether the DSR is ON/OFF.

- If the DSR signal is ON and there is data to be transmitted, C24 transmits it to the target device.
- If the DSR signal is OFF, even if there is data to be transmitted, C24 does not transmit it to the target device.
  When the DSR signal is turned ON, C24 transmits the data to the target device.

13.2 Control Contents of DC Code Control

This control uses C24 transmission control data to inform a target device whether or not host station is ready to receive data and the valid range of the transmitted/received data.

The four types of C24 DC code control shown below are available. These control functions can be used simultaneously.

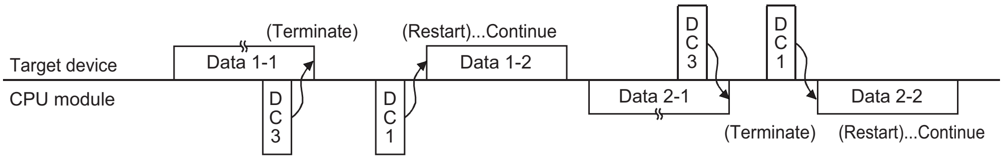
- DC1/DC3 transmission control
- DC1/DC3 reception control
- DC2/DC4 transmission control
- DC2/DC4 reception control

DC1/DC3 transmission control, DC1/DC3 reception control

C24 informs the target device whether or not host station is ready to receive data by transmitting the DC1 and DC3 signals, and checks whether or not the target device is ready to receive data by receiving the DC1 and DC3 signals.

- DC1: Control data that notifies the target device that C24 is ready to receive data
- DC3: Control data that notifies the target device that C24 is not ready to receive data

Ex.

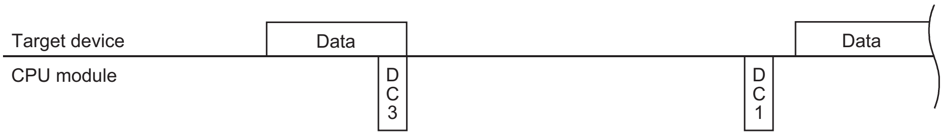


C24 DC1/DC3 transmission control contents and free OS area specification

The control contents and the free OS area specification are the same as those of the DTR control. (☞ Page 269 C24 DTR control contents and free OS area specification)

C24 operates depending on the capacity of free OS area as follows:

- Free area is 64 bytes (default) or less: DC3 transmission (reception disabled)
- Free area is 263 bytes (default) or more: DC1 transmission (reception enabled)

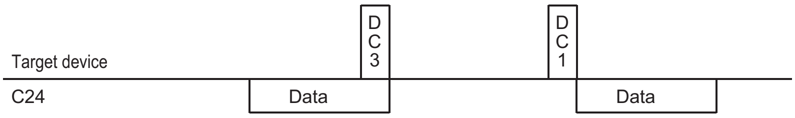


■Remarks

- The receive data clear processing clears the data stored in the OS area. (The receive area in the buffer memory is not cleared.) (☞ Page 70 Receive data clear)
- If more data is received when the above free OS area is 0 bytes, an SIO error occurs and the received data is ignored until a free area is available. In this case, the SIO signal turns ON. (☞ Page 511 Hardware information communication error "SIO" occurs)

C24 DC1/DC3 reception control contents

- When C24 receives DC3 from the target device, it terminates data transmission.
  The program cannot read the received DC3 signal.
- When C24 receives DC1 from the target device, it restarts data transmission. (C24 resumes transmission from the data terminated by the DC3 reception.)
  The program cannot read the received DC1 signal.



- Once DC1 is received, subsequent DC1 is ignored and removed from the received data.

■Remarks

In the DC1/DC3 transmission/reception control, the state of C24 is as follows at the time of power-on, reset or mode switching of the CPU, or the UINI instruction execution.

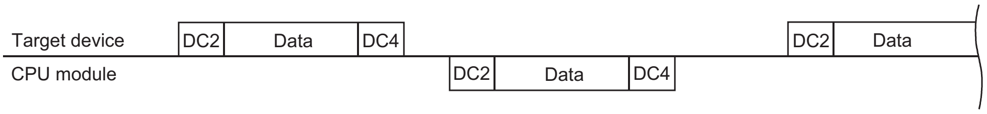
- C24 does not transmit DC1 to the target device.
- C24 is in the same state as when it transmits DC1.
- C24 is in the same state as when it received DC1 even if DC1 is not received from the target device.

DC2/DC4 transmission control, DC2/DC4 reception control

C24 encloses the data transmitted from the host station in the DC2 and DC4 codes and transmits it to the target device, and processes the data received from the target device enclosed in the DC2 and DC4 codes as valid data.

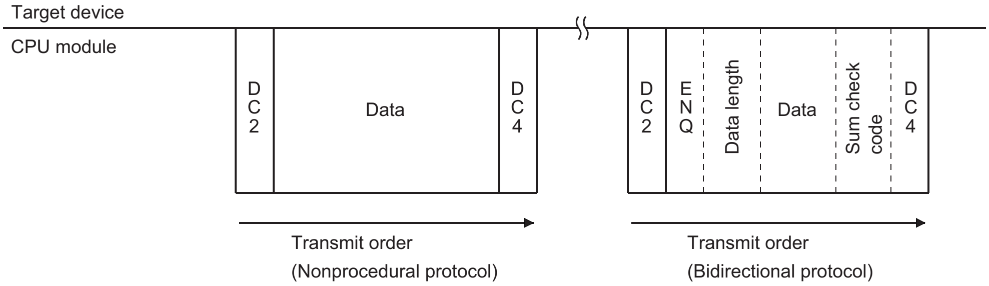
- DC2: Control data that informs the target device that the data following it is the start of the valid data
- DC4: Control data that informs the target device that the data immediately preceding it is the end of the valid data

Ex.



C24 DC2/DC4 transmission control contents

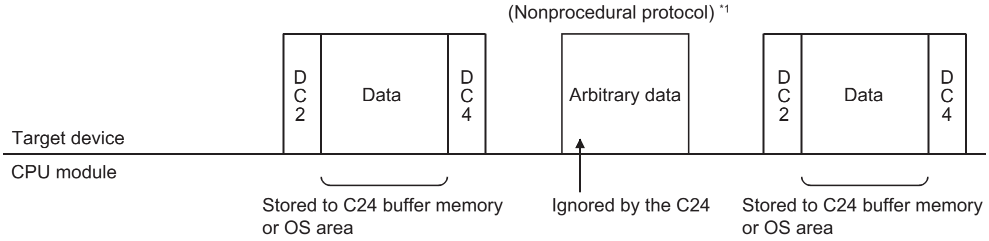
When transmitting data to a target device, C24 adds the DC2 code to the head of the data to be transmitted and the DC4 code to the end of the data to be transmitted.



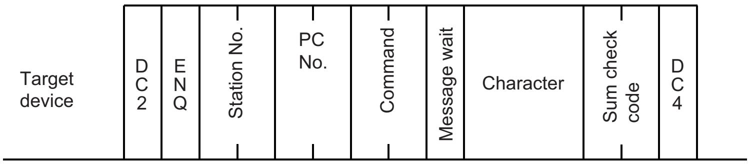
* The DC2 and DC4 code are also added for communication using MC protocol.

C24 DC2/DC4 reception control contents

- When DC2 is received from the target device, C24 processes the received data up to DC4 as valid data. The received DC2 cannot be read by the program.
- When DC4 is received from the target device, C24 ignores the received data up to the one immediately before DC2 as invalid data. The received DC4 cannot read by the program.


(Nonprocedural protocol) *1
DC2
Data
DC4
Arbitrary data
DC2
Data
DC4
Target device
CPU module
Stored to C24 buffer memory or OS area
Ignored by the C24
Stored to C24 buffer memory or OS area

*1 MC protocol 1C frame



- Once DC2 is received, subsequent DC2 is ignored and removed from the received data.

13.3 Considerations for Using Transmission Control Functions

The following describes the considerations for using the transmission control function of C24.

Agreement between target device and CPU module

Determine the following between the target device and the CPU module.

- Whether or not the transmission control function is to be used. When it is used, which control is to be used for data communications.
- Control timing
- DC1 to DC4 codes when DC code control is performed
  (The DC1 to DC4 codes to be used can be arbitrarily changed.)

Conditions to use the transmission control function

- DTR/DSR control and DC code control cannot be used at the same time.
- When using DTR/DSR control, connect the ER (DTR) and DR (DSR) signals on the C24 side to the target device.

Transmission control function setting

Set the transmission control function that can control the target interface.
If a function that cannot control the target interface is set, the set contents are invalid.

Setting of transmission control function during interlink operation

When the two interfaces of C24 is used as interlink operation, set the transmission control function to either one of interfaces of which transmission control is required.
Set the other interface to "Do not use transmission control function" (the setting values to directly set to the buffer memory is 0001H).

DC code control

- DC1/DC3 transmission control and DC1/DC3 reception control are possible when full-duplex communications is used to communicate data between C24 and target devices.
  Do not use DC1/DC3 control for half-duplex communications.
- The same data as the DC1 to DC4 codes cannot be included in the user data.
  To handle the same data as a DC code as user data, perform any of the following:
  Use DTR/DSR control.
  Change the DC code. Do not use the transmission control function.

■Remarks

If the user data received from the target device includes the relevant DC code when DC1/DC3 reception control and DC2/DC4 reception control are used, C24 performs the corresponding DC code control.
If the user data transmitted from the CPU module includes a DC code, it is sent unchanged.

Handling of ER (DTR) and DR (DSR) signals when DTR/DSR control is not used

When the DTR/DSR control is not used, C24 handles the ER (DTR) and DR (DSR) signals as described below.

- ER (DTR) signal is always ON.
- The ON/OFF status of DR (DSR) signal is ignored.

14 DATA COMMUNICATIONS USING HALF-DUPLEX COMMUNICATIONS

This chapter explains the settings to prevent simultaneous transmission from C24 and the target device for data communication between them using the RS-232 interface.

The setting for each interface is available for RJ71C24-R2.

When C24 is started, full-duplex communications is set. The communication method can be changed by the user according to the specifications of the target device.

Point

Since the setting for half-duplex communications is not required in the following cases, this chapter can be skipped.

- When data is only transmitted or received during data communications using the nonprocedural protocol.
- When no data is transmitted from the target device to C24 without direction from C24 by the specifications of the target device.

14.1 Half-duplex Communications

The following describes the differences between full-duplex communications and half-duplex communications.

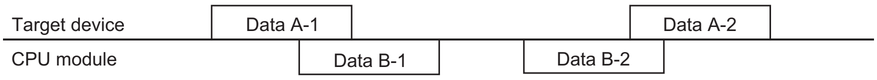
Since C24 performs the communication controls on the C24 side with the communication method set by the parameter as shown below, control by a program is not necessary.

Full-duplex communications

This is a communication method to communicate data with a target device like a conversation pattern with a telephone.

C24 can receive data while transmitting data to the target device.

It can also transmit data while receiving data from the target device.

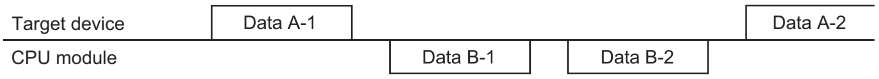


Half-duplex communications

This is a communication method to communicate data with a target device like a conversation pattern with a transceiver.

When C24 receives data from the target device while transmitting data to the target device, it controls data transmission and reception according to 'simultaneous transmission priority/non-priority designation'.

C24 does not transmit data while it is receiving data from the target device.



14.2 Data Transmission and Reception Timing

Half-duplex communications uses C24 RS-232 interface CD (DCD) and RS (RTS) signals to control communications.
If the target device can transmit and receive data according to ON/OFF of C24 RS (RTS) and CD (DCD) signals as shown below, half-duplex communications is possible.

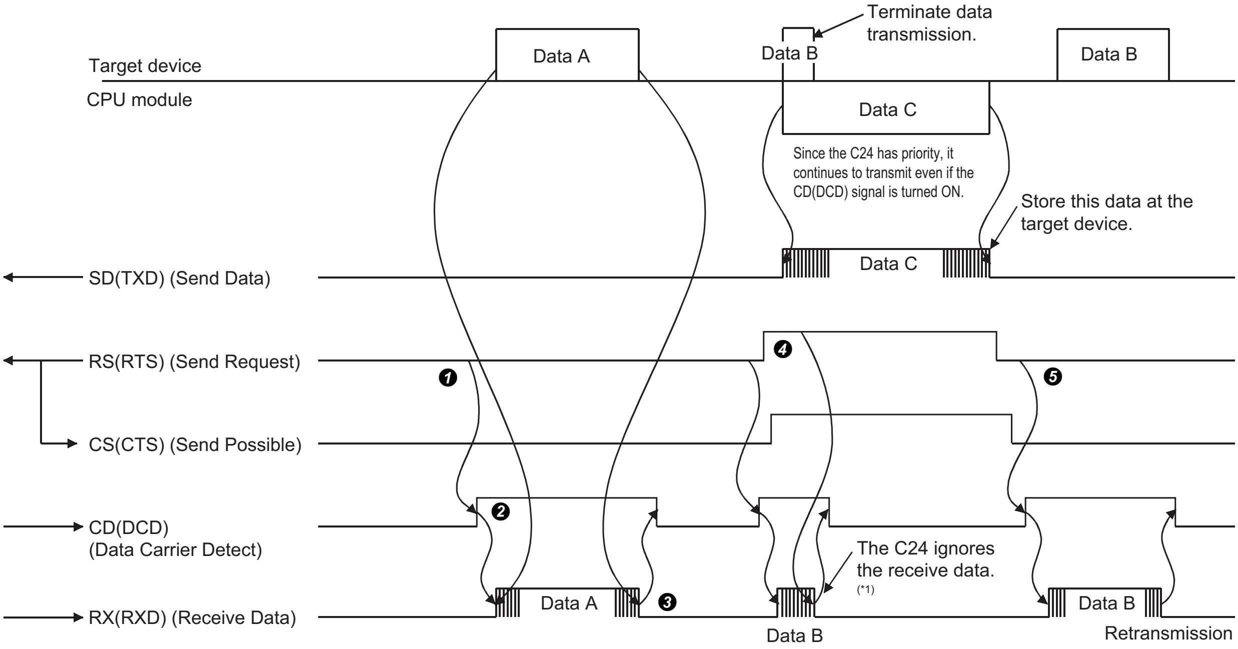
- RS (RTS) signal : C24 turns ON/OFF this signal as shown below.
  When data is transmitted from C24, this signal is turned ON. When transmission is completed, this signal is turned OFF.
- CD (DCD) signal: Turn ON/OFF this signal on the target device side as shown below.
  To transmit data from the target device, turn ON this signal. When transmission is complete, turn OFF this signal.

The data transmission/reception timings for data communications with half-duplex communications are shown by the CD (DCD) signal and RS (RTS) signal of C24.

Timing when transmitting data from target device

Transmit data by controlling C24 CD (DCD) signal according to the setting value for "simultaneous transmission priority/non-priority designation" of the parameter.

■When C24 is designated to "Priority"



❶ When transmitting data, check the RS (RTS) signal. If the RS (RTS) signal is OFF, turn ON the CD (DCD) signal. If the RS (RTS) signal is ON, wait until it is turned OFF, then turn ON the CD (DCD) signal.

❷ Transmit data after turning ON the CD (DCD) signal.

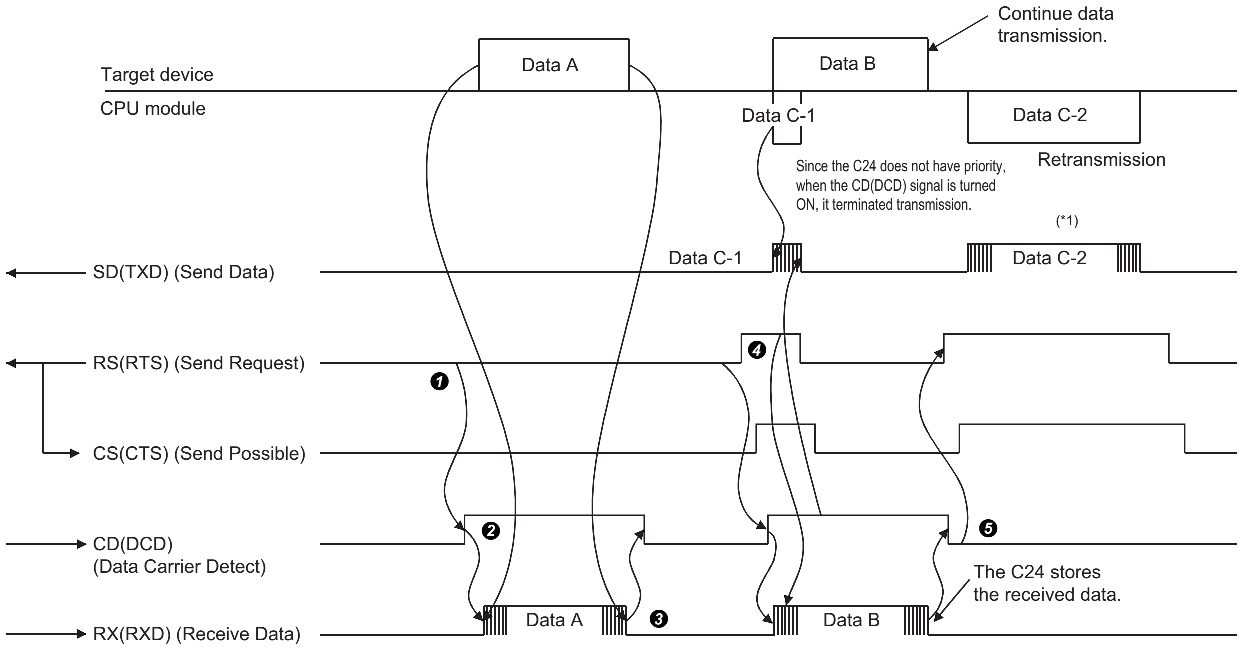
❸ After data transmission is completed, turn OFF the CD (DCD) signal.

❹ If the RS (RTS) signal is turned ON during data transmission, terminate data transmission and turn OFF the CD (DCD) signal, and perform data reception processing. (Simultaneous transmission occurred.)

❺ After transmission from C24 is completed, retransmit all the data terminated at the step ❹.

*1 Take the following measures between the devices communicating data as a countermeasure against ignoring received data by C24.
· Transmission and reception of a response message in reply to data transmission
· Time-out check of a response message and retransmission of data due to time-out error (target device side)

■When C24 is designated to "Non-priority"



The contents of steps ❹ and ❺ below are different from those of the case "Priority" is designated.

❶ When transmitting data, check the RS (RTS) signal. If the RS (RTS) signal is OFF, turn ON the CD (DCD) signal. If the RS (RTS) signal is ON, wait until it is turned OFF, then turn ON the CD (DCD) signal.

❷ Transmit data after turning ON the CD (DCD) signal.

❸ After data transmission is completed, turn OFF the CD (DCD) signal.

❹ Keep transmitting data to C24 even if the RS (RTS) signal is turned ON during data transmission. (Simultaneous transmission occurred.)

❺ After transmission from the target device is completed, transmit data from C24 to the target device. (☞ Page 278 Timing when transmitting data from C24)

*1 For the transmission content, refer to the following section.
☞ Page 278 Timing when transmitting data from C24

■Remarks

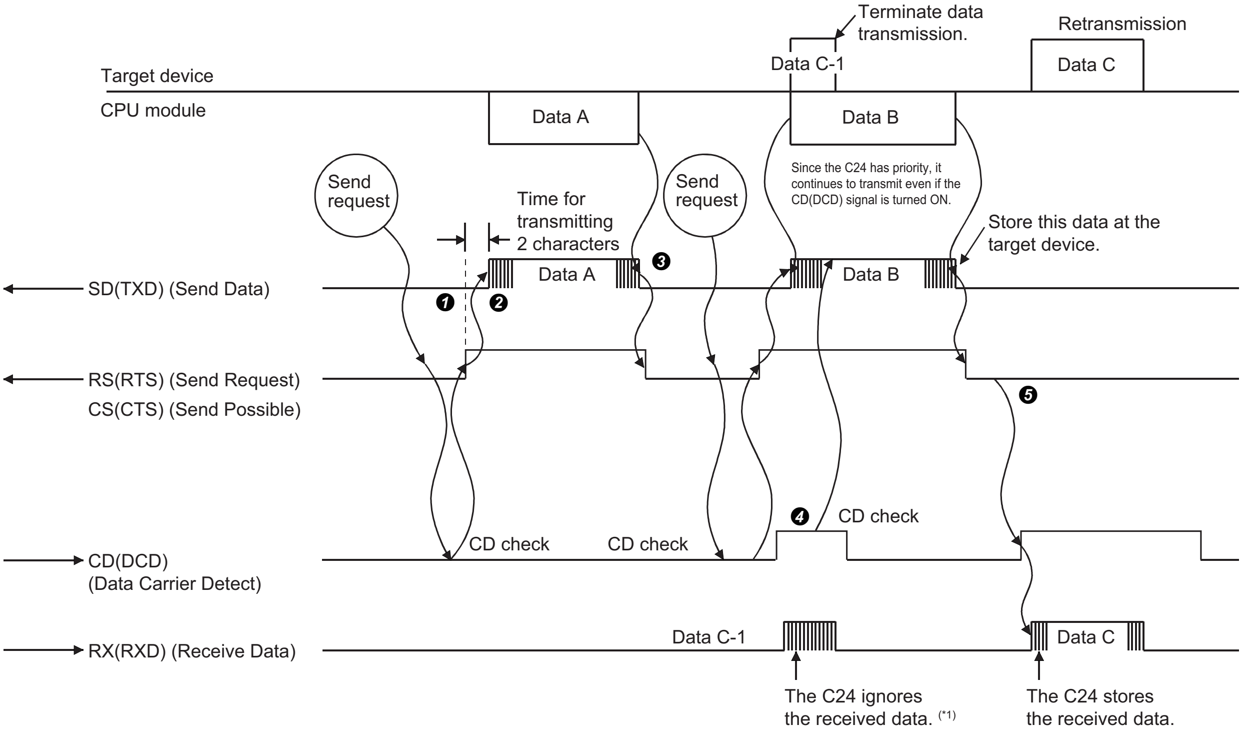
When using the DTR/DSR transmission control function, transmit data from the target device to C24 as shown below in both cases above. (☞ Page 268 DATA COMMUNICATIONS USING DC CODE TRANSMISSION CONTROL)

- When C24 ER (DTR) signal is turned OFF, terminate data transmission.
- When C24 ER (DTR) signal is turned ON after data transmission is terminated, restart data transmission (transmit from the terminated part of the data).

Timing when transmitting data from C24

Transmit data by controlling C24 RS (RTS) signal according to the setting value for "simultaneous transmission priority/non-priority designation" of the parameter.

■When C24 is designated to "priority"



Terminate data transmission.
Retransmission
Data C-1
Data C
Target device
CPU module
Data A
Data B
Send request
Time for transmitting 2 characters
Send request
Since the C24 has priority, it continues to transmit even if the CD(DCD) signal is turned ON.
Store this data at the target device.
Data A
Data B
SD(TXD) (Send Data)
❶
❷
❸
RS(RTS) (Send Request)
CS(CTS) (Send Possible)
❺
❹
CD check
CD check
CD check
CD(DCD)
(Data Carrier Detect)
Data C-1
Data C
RX(RXD) (Receive Data)
The C24 ignores the received data. (*1)
The C24 stores the received data.

❶ When transmitting data, check the C24 CD (DCD) signal. If the CD (DCD) signal is OFF, turn ON the C24 RS (RTS) signal. If the CD (DCD) signal is ON, wait until it is turned OFF, and then turn ON the RS (RTS) signal.

❷ Transmit the data after turning ON the RS (RTS) signal.

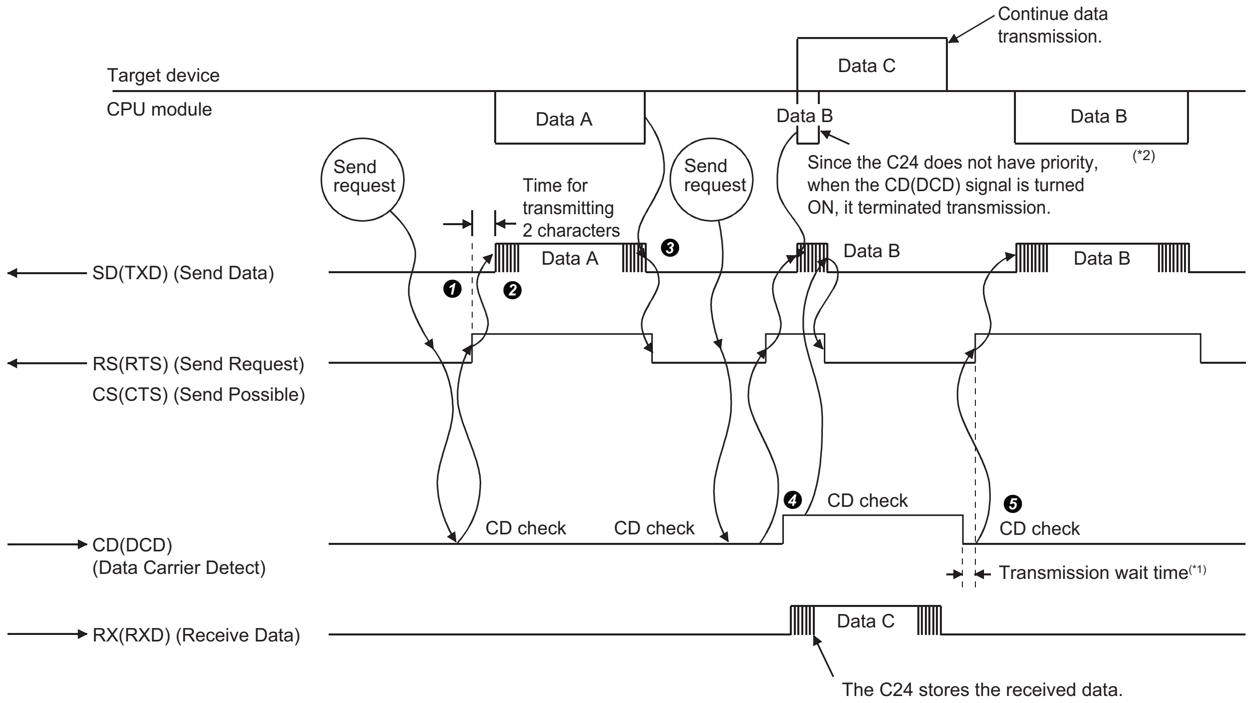
❸ After data transmission is completed, turn OFF the RS (RTS) signal.

❹ Keep transmitting data from C24 to the target device even if the CD (DCD) signal is turned ON during data transmission. (Simultaneous transmission occurred.)

❺ After transmission from C24 is completed, transmit all the data terminated at step ❹ from the target device to C24.

[*1] As a countermeasure against the disregard for the received data by C24, perform the following between the communicating devices.
· Transmission and reception of a response message in reply to data transmission.
· Time-out check of a response message and retransmission of data due to time-out error (target device side)

■When C24 is designated to "Non-priority"



The contents of steps ❹ and ❺ below are different from those of the case "Priority" is designated.

❶ When transmitting data, check the C24 CD (DCD) signal. If the CD (DCD) signal is OFF, turn ON the C24 RS (RTS) signal. If the CD (DCD) signal is ON, wait until it is turned OFF, then turn ON the RS (RTS) signal.

❷ Transmit the data after turning ON the RS (RTS) signal.

❸ After data transmission is completed, turn OFF the RS (RTS) signal.

❹ If the CD (DCD) signal is turned ON during data transmission, terminate data transmission and turn OFF the RS (RTS) signal, and perform the data receive processing. (Simultaneous transmission occurred.)

❺ After transmission from the target device is completed, transmit the data terminated at step ❹ from the beginning or terminated part of the data.[*2]

*1 The data equivalent to the setting value of "simultaneous transmission priority/non-priority" is not transmitted.

*2 The data is transmitted from the beginning or from the part where the last transmission has been terminated according to the settings of "Retransmission time transmission method".

■Remarks

When using the DTR/DSR transmission control function, transmit data from the target device to C24 as shown below in both cases above. (☞ Page 268 DATA COMMUNICATIONS USING DC CODE TRANSMISSION CONTROL)

- When the C24 DR (DSR) signal is turned OFF, terminate data transmission.
- When the C24 DR (DSR) signal is turned ON after data transmission is terminated, restart data transmission (transmit from the terminated part of the data).

14.3 Changing the Communication Method

To change the data communication method from full-duplex communication to half-duplex communication, setting the following parameters is required.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

RS-232 communication method designation

Set to "half-duplex communications".

Simultaneous transmission priority/non-priority designation

Set whether to continue ("priority") or stop ("non-priority") transmission from C24 when C24 and the target device start data transmission simultaneously in half-duplex communications.

Setting values "1" to "255" to set to "Non-priority" will be the transmission wait time that elapses from when the resumption of data transmission becomes available until data is actually sent.

Retransmission time transmission method designation

When "half-duplex communication" and "non-priority" are set as indicated above, this setting becomes valid. Set whether the terminated message will be transmitted from the beginning ('Resend') or from the terminated part of the message ('Do not resend') when C24 and the target device start transmission simultaneously, and C24 stops and then restarts transmitting.

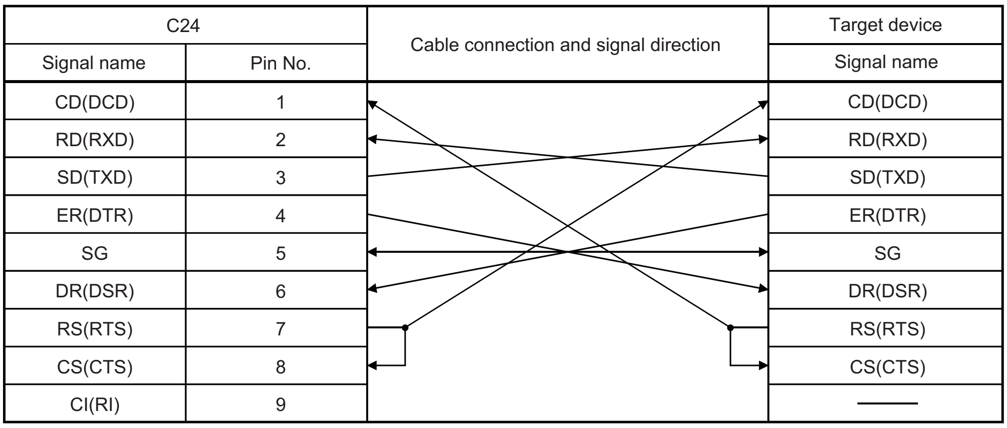
RS-232 CD terminal check designation

Set the CD terminal check designation to "Check".

14.4 Connector Connections for Half-duplex Communications

The following explains the connector that connects C24 and target device when half-duplex communications are used. Connect to the target device in accordance with the following.

- Connect the C24 RS (RTS) signal to one of the target device signals (CS (CTS), DR (DSR), or CD (DCD) signal) for half-duplex communications.
- Connect the C24 CD (DCD) signal to one of the target device signals (RS (RTS) or DR (DSR) signal) for half-duplex communications.
- The half-duplex communications cannot be performed when an RS-232 and RS-422 converter is used.



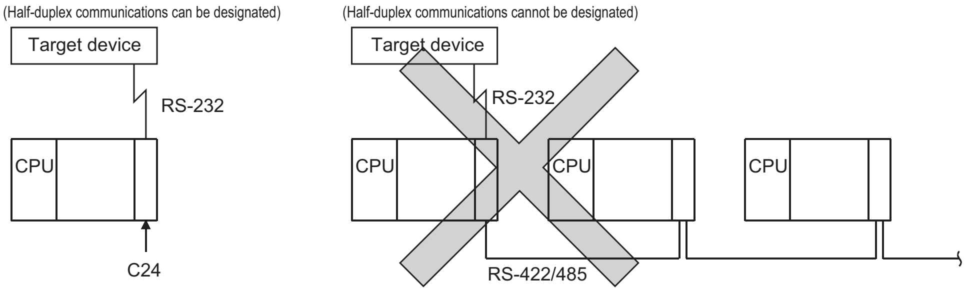
C24		Cable connection and signal direction	Target device
Signal name	Pin No.		Signal name
CD(DCD)	1		CD(DCD)
RD(RXD)	2		RD(RXD)
SD(TXD)	3		SD(TXD)
ER(DTR)	4		ER(DTR)
SG	5		SG
DR(DSR)	6		DR(DSR)
RS(RTS)	7		RS(RTS)
CS(CTS)	8		CS(CTS)
CI(RI)	9		———

14.5 Considerations for Half-duplex Communications

The following describes the considerations for using half-duplex communications.

Half-duplex communications system configuration and functions

Half-duplex communications can be used only in a system configuration where the CPU module and a target device are connected on a 1:1 basis.


(Half-duplex communications can be designated)
Target device
RS-232
CPU
C24
(Half-duplex communications cannot be designated)
Target device
RS-232
CPU
CPU
CPU
RS-422/485

Agreement and confirmation between target device and CPU module

Determine and confirm the following items between the target device and the CPU module.

- Whether or not half-duplex communications can be performed by C24 RS (RTS) signal and CD (DCD) signal.
- C24 RS (RTS) signal and CD (DCD) signal ON/OFF timing
- C24 and target device data transmission timing
- RS-232 cable connection method

Transmission control

When the transmission control function is used, DC1/DC3 transmission control and DC1/DC3 reception control of the DC code control cannot be used with half-duplex communications. Therefore, do not designate them.(☞ Page 268 DATA COMMUNICATIONS USING DC CODE TRANSMISSION CONTROL)

15 CONTENTS AND REGISTRATION OF USER FRAMES FOR DATA COMMUNICATION

User frames are used to register some part, or all of messages exchanged between a target device and C24 in advance and use them to check the data to be transmitted or received.

The following functions can use C24 user frames to transmit and receive data.

- MC protocol on-demand function.(☞ Page 299 ON-DEMAND DATA COMMUNICATIONS USING USER FRAMES)
- The data transmit/receive function using the nonprocedural protocol (☞ Page 308 DATA COMMUNICATIONS USING USER FRAMES)

Data can be transmitted and received by registering the corresponding user frames to C24 in advance to match the data contents that are transmitted and received between the target device and C24.

This chapter explains the data that can be registered, the data contents that are transmitted and received, and registering to C24 of user frames that can be used in data communications with the target device.

For information on how to use user frames in each data communication function, refer to the following sections.

☞ Page 299 ON-DEMAND DATA COMMUNICATIONS USING USER FRAMES

☞ Page 308 DATA COMMUNICATIONS USING USER FRAMES

15.1 User Frame Types and Contents During Communication

This section explains the types of user frames handled by C24 and the data contents that are transmitted and received.

The following two types of user frames are available. Either type can be used as a user frame.

Frame type	Overview	Reference
User frame	A frame that contains arbitrary data arranged according to the specifications of a target device. The data arrangement can be defined by a user.	Page 282 User frames to be registered and used by the user
Default registration frame	A frame that is registered in C24 in advance. The frame cannot be changed by a user.	Page 290 Default registration frame (read only)

User frames to be registered and used by the user

The following explains data to be registered, data contents that are transmitted and received, and how C24 handles user frames registered C24 flash ROM or buffer memory by the user.

Overview

User frames registered by the user are frames that contain arbitrary data according to the specifications of the target device, and the data arrangement are determined by the user.

User frame registration

Up to 80 bytes (80 characters) of data can be registered as 1 user frame.

Up to 231 user frames can be registered (written), read, and deleted to/from C24.

Registration destination		Maximum number of frames can be registered	Frame No.	Application	Remarks
CPU module	Built-in memory[*1]	31 frames	8001H to 801FH	A user frame of which contents are changed after data communication has started	—
	SD memory card[*1]	31 frames	8001H to 801FH	A user frame of which contents are changed after data communication has started	—
C24	Flash ROM[*1]	200 frames	03E8H to 04AFH	A user frame of which contents is not changed after data communication has started	—
	Buffer memory	31 frames	8001H to 801FH	A user frame of which contents are changed after data communication has started	Registration by overwriting with MOV instruction is available.

[*1] For the priority of the registered data, refer to the following section.
☞ Page 434 PARAMETER REGISTRATION TO CPU MODULE

Registration data to handle variable data (Sum check code, C24 station number, etc.) as a part of the user frame can be included in the user frame. (☞ Page 283 Variable data)

User frames can be registered by overwriting to C24 buffer memory. (The preceding contents are destroyed.)

Data that can be registered as user frame

Up to 80 bytes of data can be registered by combining 1 byte data of which registered code is 01H to FEH and 2 bytes data of which registered code is FFH + 00H to FFH + FFH.

■One byte data of which registered code is (01H to FEH)

This is the registered code for transmitting and receiving the data of which registered code is (01H to FEH).

■Two bytes data of which registered code is (FFH + 00H) to (FFH + FFH)

This is the registered code for transmitting and receiving the variable data (sum check code, C24 station number, etc.) as part of the user frame. (☞ Page 283 Variable data)

FFH is the registered code of the first byte for handling variable data.

Variable data

"Variable data" is a generic term for the data shown below.

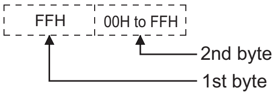
These variable data can be handled as part of a user frame.

- Sum check code covering a determined range of messages to be transmitted/received
- Horizontal parity code covering a determined range of message to be transmitted/received
- Two's complement sum check code covering a determined range of messages to be transmitted/received
- C24 station number
- One byte data in data transmission (00H (NULL), FFH)
- One byte of arbitrary data in data reception. (Used to handle an arbitrary byte of data as part of the user frame during reception check by C24.)

■Variable data designation method

Variable data is designated by combining registered code FFH and the data codes shown in the table below.

The sum check code, C24 station No., and other variable data can be handled according to FFH of the first byte and 00H to FFH of the second byte.



■Variable data designation contents, data contents transmitted and received, and handling by C24

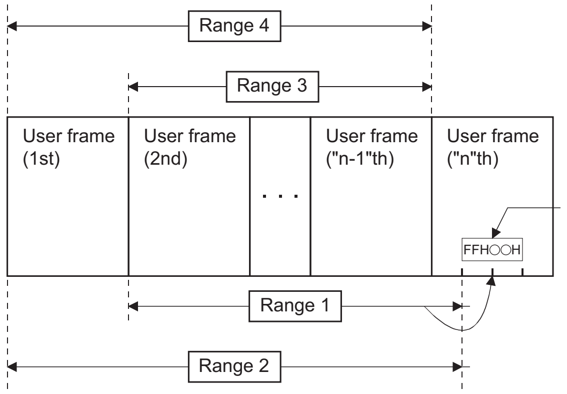
The table below shows the registered codes (FFH+00H) to (FFH + FFH) combinations for handling variable data, the data contents that are transmitted and received, and how C24 handles the data.

Variable data registered code		Data contents transmitted and received/C24 handling	Calculating range	Detailed explanation
1st byte	2nd byte			
FFH	00H	When transmitting: Transmits the data code 00H (NULL) data (1 byte). When receiving: Skips the given part (1 byte) of the receive user frame. (Skips the check and performs receive processing.)	—	☞ Page 285 C24 processing to the registered code FFH+00H
	01H	Transmits and receives the host station number as 1-byte binary code data (00H to 1FH).		☞ Page 285 C24 processing to the registered code FFH+01H
	04H	Transmits and receives the horizontal parity code in the calculating range using 1 byte of binary code data.	Range 1	☞ Page 286 C24 processing to registered codes FFH+04H, FFH+05H, FFH+0AH, and FFH+0BH
	05H	Transmits and receives the horizontal parity code in the calculating range using 2-byte ASCII code data		
	0AH	Transmits and receives the horizontal parity code in the calculating range using 1 byte of binary code data.	Range 2	
	0BH	Transmits and receives the horizontal parity code in the calculating range using 2-byte ASCII code data		
	11H	Transmits and receives the two's complement sum check code in the calculating range.	Range 1	☞ Page 287 C24 processing to registered codes FFH+11H and FFH+17H
	17H		Range 2	
	E5H	Transmits and receives the sum check code in the calculating range. (Excluding the last one frame)[*1]	Range 3	☞ Page 289 C24 processing to registered code FFH+E5H
	EBH		Range 4	☞ Page 289 C24 processing to registered code FFH+EBH
	EEH	Transmits and receives the sum check code in the calculating range.[*1]	Range 1	☞ Page 288 C24 processing to registered codes FFH+EEH to FFH+F9H
	F0H			
	F1H			
	F3H			
	F4H		Range 2	
	F6H			
	F7H			
	F9H			
	FFH	Transmits and receives the registered code FFH data (1 byte).	—	—

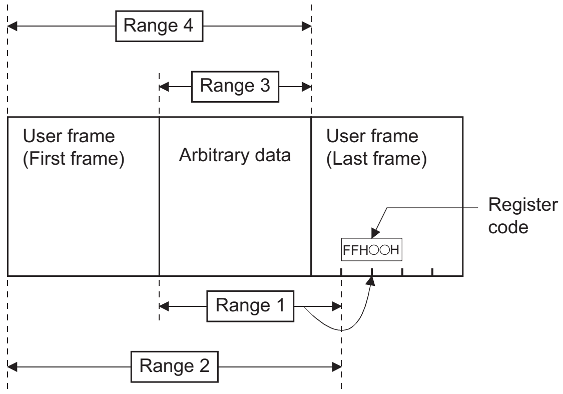
[*1] During data communication using a user frame that handles the final frame, the contents of the "sum check code" in the transmission setting set by the parameter will be ignored.

■Calculating ranges for the registered codes

The following shows the calculating ranges for the registered codes.



(Calculation range when transmitting data)



(Calculation range when receiving data)

Calculating range		Target range[*1]
Range 1	When transmitting	Range from the data following the start frame (the user frame (1st) when multiple user frames are designated) to the one immediately before the registered code
	When receiving	Range from the data following the start frame to the one immediately before the registered code
Range 2	When transmitting/ receiving	Range from the head of a message to the data immediately before the registered code
Range 3	When transmitting	Range from the data following the start frame (the user frame (1st) when multiple frames are designated) to the one immediately before the final frame (the user frame (nth) when multiple frames are designated)
	When receiving	Range from the data following the start frame to the one preceding the final frame
Range 4	When transmitting/ receiving	Range from the head of a message to the data preceding the final frame

[*1] The additional code data for the transparent code designation shown below is excluded.
☞ Page 358 TRANSPARENT CODES AND ADDITIONAL CODES

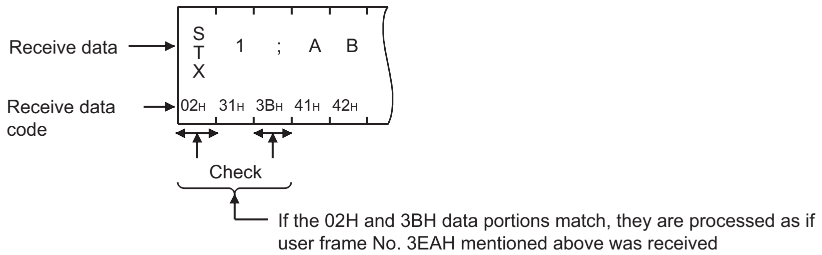
Variable data registered code

■C24 processing to the registered code FFH+00H

The following shows an example to describe the processing performed by C24 when it receives a user frame portion corresponding to registered codes FFH and 00H.

When setting a user frame containing the data codes 02H, FFH, 00H, and 3BH for a receive user frame as user frame number, 3EAH

- C24 performs the reception processing as the user frame, 3EAH has been received when it receives 3 bytes of data, "STX, arbitrary data (for 1 byte), ;".
- C24 does not check the 2 bytes described above.


Receive data
Receive data code
S T X
1
;
A
B
02H
31H
3BH
41H
42H
Check
If the 02H and 3BH data portions match, they are processed as if user frame No. 3EAH mentioned above was received

■C24 processing to the registered code FFH+01H

C24 transmits and receives the user frame portion corresponding to registered code, FFH+01H by representing the station number set in the parameter as 1-byte binary data.

For the usage example, refer to the start frame portion in the figures below.

☞ Page 286 C24 processing to registered codes FFH+04H, FFH+05H, FFH+0AH, and FFH+0BH

■C24 processing to registered codes FFH+04H, FFH+05H, FFH+0AH, and FFH+0BH

- For FFH + 04H and FFH + 0AH

  The horizontal parity code that calculates the range for the data to be transmitted/received (message) is expressed as 1 byte binary data and then transmitted and received.

  The difference between FFH + 04H and FFH + 0AH is the difference in the calculating range.

- For FFH + 05H and FFH + 0BH

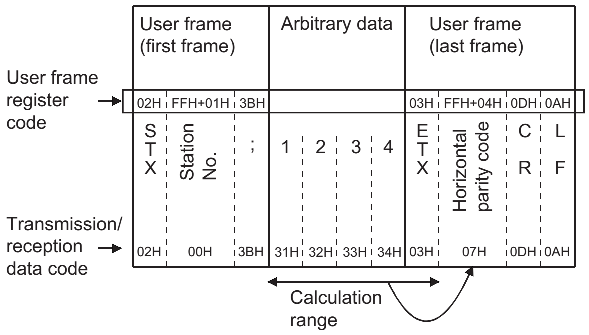
  The horizontal parity code that calculates the range for the data to be transmitted/received (message) is expressed as 2-byte ASCII code data and then transmitted and received from the upper digit.

  The difference between FFH + 05H and FFH + 0BH is the difference in the calculating range.

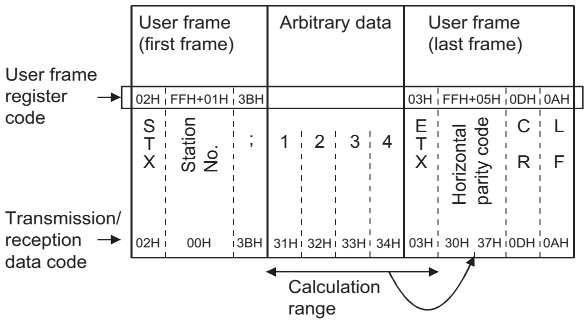
Ex.

Examples of data arrangement when the horizontal parity code is transmitted/received are shown below. (The start frame and final frame are equivalent to one frame respectively.)

- For registered code FFH + 04H



- For registered code FFH + 05H



- How to calculate the horizontal parity code

  This is a numeric value obtained by calculating the XOR for the subject data and then converting it to ASCII code.

Ex.

For message example above

```
  "1" (31H)  0011 0001
                 XOR
  "2" (32H)  0011  0010 = 0000  0011
                              XOR
  "3" (33H)                 0011  0011 = 0011  0000
                                             XOR
  "4" (34H)                              0011  0100 = 0000  0100
                                                          XOR
"ETX" (03H)                                           0000  0011 = 0000  0111
                                                                     ↓     ↓
                                                                    "0"   "7"
                                                     ASCII code    (30H) (37H)
```

■C24 processing to registered codes FFH+11H and FFH+17H

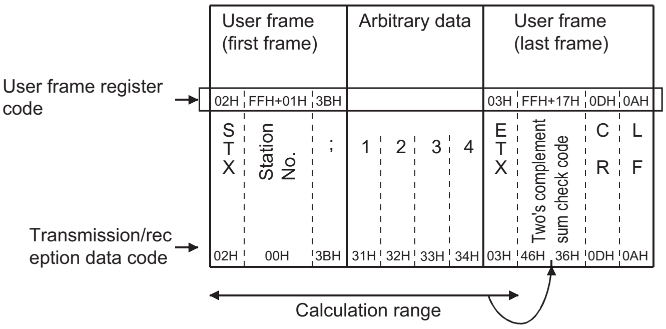
The two's complement sum check code that calculates the range for data (message) to be transmitted/received is expressed as two character data in ASCII code, then transmitted and received from the upper digit.

The difference between (FFH+11H) and (FFH+17H) is the difference in the calculating ranges.

An example of the contents (arrangement) of the transmission and reception of the two's complement sum check code is shown below.

Ex.

Arrangement of data transmitted and received with the registered code FFH+17H (The start frame and final frame are equivalent to one frame respectively.)



- How to calculate the two's complement sum check code

  The lower 1 byte of the value obtained by adding the subject data as binary data is converted to a two's complement, then converted to a hexadecimal ASCII code.

Ex.

For message example above

02H+00H+3BH+31H+32H+33H+34H+03H=10AH ------● Data in the subject range are added as binary data (hexadecimal)

↓

00001010 ------● The lower 1 byte of the above numeric value is expressed as a binary number

↓

11110110 ------● Reversed, then converted to a two's complement

↓

F6H ------● Converted to a hexadecimal ASCII code

↓

"F" (46H) "6" (36H)

■C24 processing to registered codes FFH+EEH to FFH+F9H

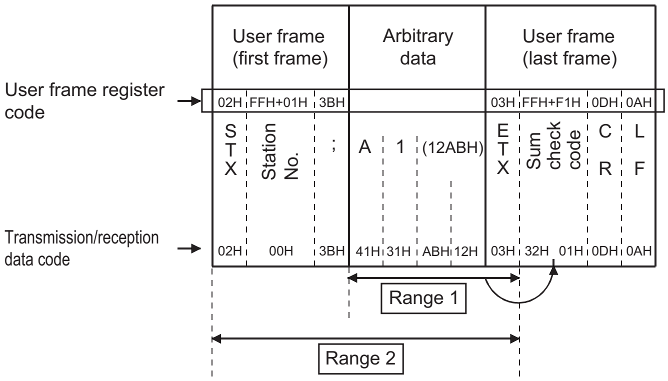
The sum check code that calculates the range for data (message) to be transmitted/received is expressed as the binary code/ASCII code data listed below, then transmitted and received.

Registered code	Data contents transmitted and received
FFH + EEH or FFH + F4H	The lower 2 bytes of the calculated sum check code are transmitted and received as 2-byte binary code data.
FFH + F0H or FFH + F6H	The lower byte of the calculated sum check code is transmitted and received as 1-byte binary code data.
FFH + F1H or FFH + F7H	The lower byte of the calculated sum check code is converted to 2-digit ASCII code, then transmitted and received.
FFH + F3H or FFH + F9H	The lower 4 bits of the calculated sum check code is converted to 1-digit ASCII code, then transmitted and received.

An example of the contents (arrangement) of the transmission and reception of the sum check code is given below.

Ex.

Arrangement of data transmitted and received (one start frame and one final frame)



- How to calculate the sum check code and data contents transmitted and received
  This is a numeric value obtained by adding data in the above range as binary data.

Ex.

For Range 1: 41H+ 31H + ABH + 12H + 03H = 0132H

For Range 2: 02H + 00H + 3BH + 41H + 31H + ABH + 12H + 03H = 016FH

Calculating range	Registered code	Data contents transmitted and received (arrangement)
Range 1	FFH + EEH	01H and 32H are transmitted and received, beginning from 32H.
	FFH + F0H	32H is transmitted and received.
	FFH + F1H	"3" and "2" are transmitted and received, beginning from "3."
	FFH + F3H	"2" is transmitted and received.
Range 2	FFH + F4H	01H and 6FH are transmitted and received, beginning from 6FH.
	FFH + F6H	6FH is transmitted and received.
	FFH + F7H	"6" and "F" are transmitted and received, beginning from "6".
	FFH + F9H	"F" is transmitted and received.

■C24 processing to registered code FFH+E5H

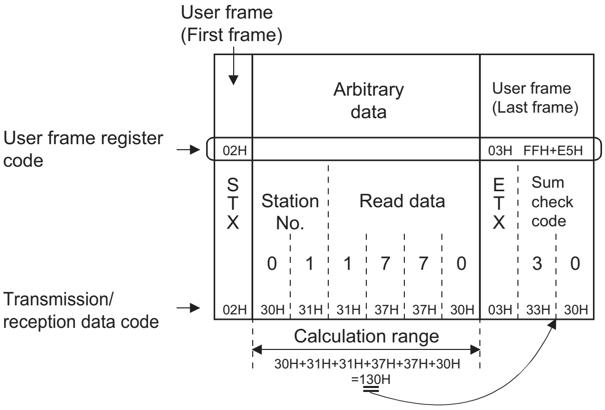
The sum check code that calculates data (message) to be transmitted/received excluding the first one frame and last one frame is expressed as the ASCII code data and transmitted or received.

- How to calculate the sum check code

  Lower one byte of the calculated sum check code is converted into 2-digit ASCII code data, and transmitted and received.

Ex.

Arrangement of data when transmitted/received



■C24 processing to registered code FFH+EBH

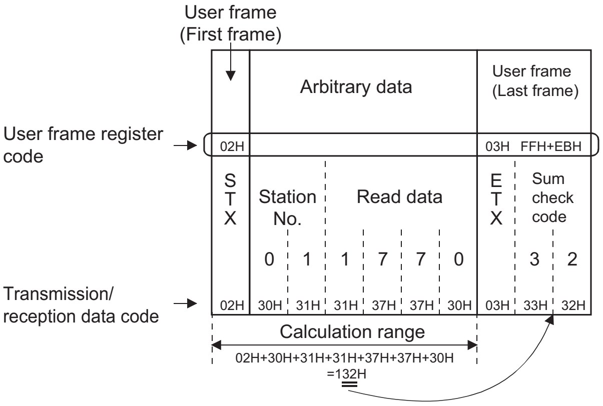
The sum check code that calculates the data (message) to be transmitted/received, including the first one frame and excluding the last one frame, is expressed as the ASCII code data, and transmitted or received.

- How to calculate the sum check code

  Lower one byte of the calculated sum check code is converted into 2-digit ASCII code data, and transmitted or received.

Ex.

Arrangement of data when transmitted/received



Default registration frame (read only)

This frame is registered in C24 in advance and can be used in the same way as the other user frames.

Overview

The default registration frame is registered in the operating system ROM of C24.

The one-byte data (code: 01H to FEH) and a maximum of five-byte data is registered, and can be used for read-only. (Frame No.: 1H to 3E7H)

Each of these frames is treated as a user frame.

Default registration frame registered data and data contents transmitted and received

The codes of the registration data for default registration frame and the data contents that are transmitted and received are shown below.

Default registration frame No. (HEX (DEC))	Registration data code (1st byte to nth byte)	Registered byte count	Frame byte count	Data contents transmitted and received
				Data contents
1H (1)	01H	1	1	(Data of code shown at the left)
2H (2)	02H			STX
⋮	⋮			⋮
FEH (254)	FEH			(Data of code shown at the left)
FFH (255)	—		—	(For variable data designation)
100H (256)	00H	1	1	NUL
101H (257)	FFH	1		(Data of code shown at the left)
102H (258)	0DH, 0AH	2	2	CR, LF
103H (259)	10H, 02H			DLE, STX
104H (260)	10H, 03H			DLE, ETX
105H (261)	00H, FEH	2	2	(Data of codes shown at the left)
106H (262)	00H, 00H, FEH	3	3	(Data of codes shown at the left)
107H (263)	03H, FFH, F1H	3	2	ETX, sum check code[*1]
108H (264)	03H, FFH, F1H, 0DH, 0AH	5	4	ETX, sum check code, CR, LF[*1]
109H (265)	(None)	—	—	—
⋮				
10DH (269)				
10EH (270)	FFH, EEH	2	1	Sum check code[*1]
110H (272)	FFH, F0H			
111H (273)	FFH, F1H			
113H (275)	FFH, F3H			
114H (276)	FFH, F4H			
116H (278)	FFH, F6H			
117H (279)	FFH, F7H			
119H (281)	FFH, F9H			
11FH (287)	FFH, FFH			Registered code FFH data (1 byte)
120H (288)	(None)	—	—	—
⋮				
3E7H (999)				

[*1] The combination of FFH, □□H in the registered code is used to handle variable data (sum check code, C24 station number, etc.) as part of the user frame.
The data contents that are transmitted and received and the byte count depend on the code combined with registered code FFH.
For combination of registered code to handle variable data and the data contents transmitted/received, refer to the following section.
☞ Page 283 Variable data

15.2 Transmission/Reception Processing Using User Frame Registration Data

The following explains how C24 transmits and receives using user frame registration data.
C24 checks the transmission/reception of following data, using registered data.

Transmitting data

When user frame transmission is designated

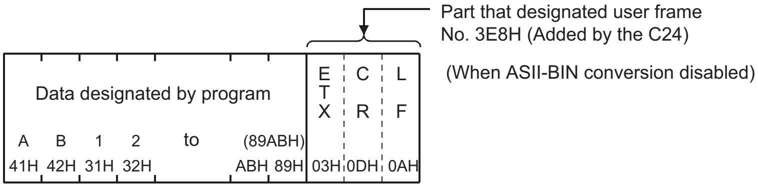
A user frame is converted or not converted from ASCII to binary data and transmitted, depending on the data communications protocol, ASCII-BIN conversion designation, and user frame No. designation, based on the following data as the send data of the given part.

■Registered portion of 1-byte data for code of registration data, 01H to FEH

C24 transmission is based on the data for registered code (01H to FEH).

Ex.

When data codes 03H, 0DH, and 0AH are registered as user frame number, 3E8H
When user frame number, 3E8H is designated during data transmission, C24 transmits the data codes 03H, 0DH, 0AH (ETX, CR, LF) as the data to be transmitted for the given user frame if ASCII-BIN conversion is disabled.
When ASCII-BIN conversion is enabled, C24 converts each of the data above to 2-byte ASCII code data and transmits.


Part that designated user frame
No. 3E8H (Added by the C24)
(When ASII-BIN conversion disabled)
Data designated by program
A
B
1
2
to
(89ABH)
41H
42H
31H
32H
ABH
89H
ETX
CR
LF
03H
0DH
0AH

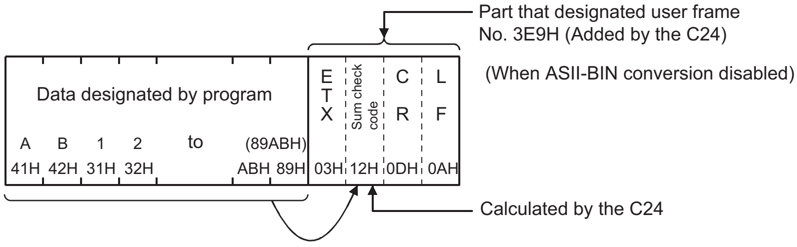
■Registered portion of 2-byte data for code of registration data, FFH+00H to FFH

C24 transmission is based on the variable data corresponding to the combination of the registered codes FFH and 00H to FFH.

For example, if sum check code is registered, C24 will calculate and transmit the sum check code. If C24 station No. is registered, the station number set to C24 is transmitted.

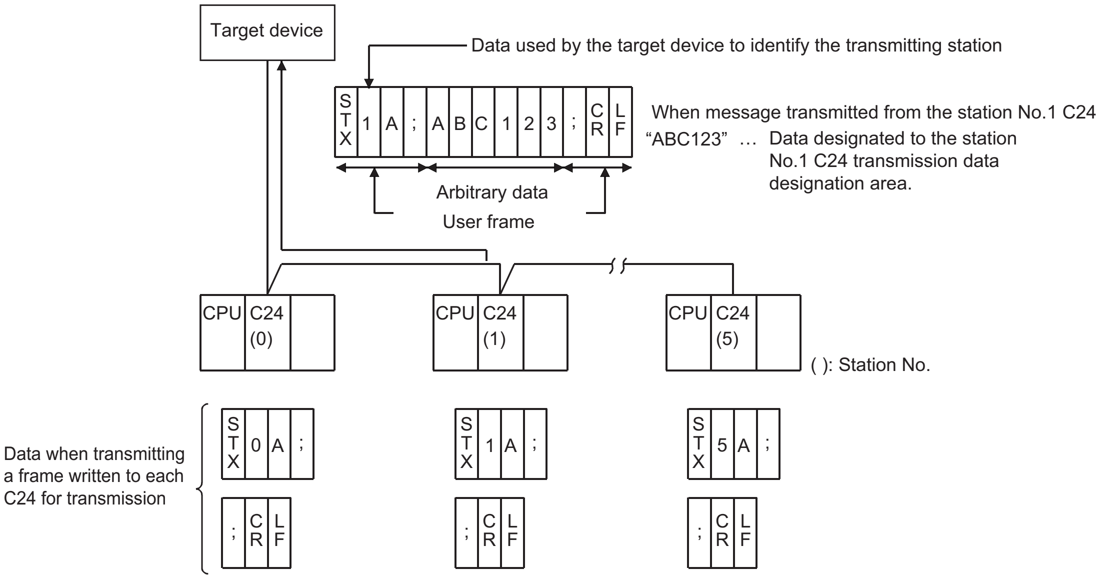
Ex.

When data codes 03H, FFH, F0H, 0DH, 0AH are registered as user frame number, 3E9H
When user frame number, 3E9H is designated during data transmission, C24 calculates the sum check code as the data to be transmitted for the given user frame portion. If ASCII-BIN conversion is disabled, C24 transmits the calculated data. If ASCII-BIN conversion is enabled, C24 converts the calculated data to 2 bytes per byte ASCII code data and transmits.


Part that designated user frame
No. 3E9H (Added by the C24)
(When ASII-BIN conversion disabled)
Data designated by program
A
B
1
2
to
(89ABH)
41H
42H
31H
32H
ABH
89H
ETX
Sum check code
CR
LF
03H
12H
0DH
0AH
Calculated by the C24

■Remarks

With the multidrop connection, including data for the target device to identify which station transmitted a message in the user frame can facilitate creation of arbitrary data to be transmitted.


Target device
Data used by the target device to identify the transmitting station
S T X | 1 | A | ; | A | B | C | 1 | 2 | 3 | ; | C R | L F
When message transmitted from the station No.1 C24
"ABC123" … Data designated to the station No.1 C24 transmission data designation area.
Arbitrary data
User frame
CPU | C24 (0)
CPU | C24 (1)
CPU | C24 (5)
( ): Station No.
Data when transmitting a frame written to each C24 for transmission
S T X | 0 | A | ;
; | C R | L F
S T X | 1 | A | ;
; | C R | L F
S T X | 5 | A | ;
; | C R | L F

Receiving data

When user frame reception is designated

When the start frame is set, C24 receives the message when it receives data with the same arrangement as the designated start frame.

If the final frame is set, C24 issues the read request of received data to the CPU module when it receives data with the same arrangement as the designated final frame.

C24 reception processing by registration data

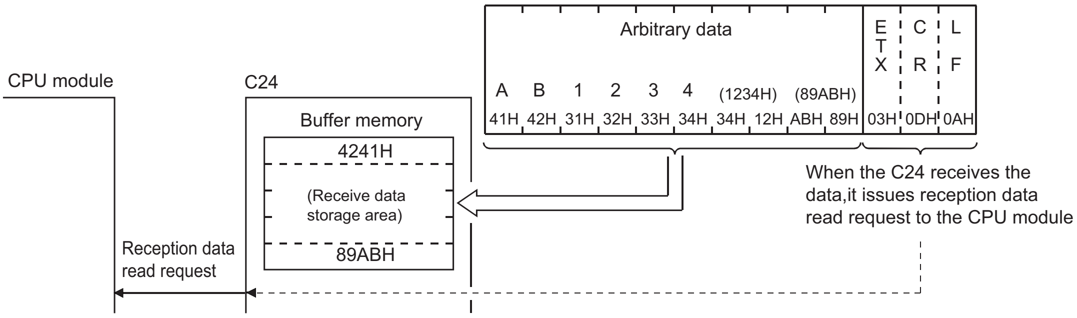
■Registered portion of 1-byte data for code of registration data, 01H to FEH

C24 receives and checks if the received data is data of the same code (01H to FEH) as the registered code.

Ex.

When data codes 03H, 0DH, and 0AH are registered as user frame number, 3E8H

When user frame No. 3E8H is set for data reception, C24 receives and checks data codes 03H, 0DH, 0AH (ETX, CR, LF) as the receive data of that user frame portion.


CPU module
C24
Buffer memory
4241H
(Receive data storage area)
89ABH
Reception data read request
Arbitrary data
A B 1 2 3 4 (1234H) (89ABH)
41H 42H 31H 32H 33H 34H 34H 12H ABH 89H
E T X | C R | L F
03H 0DH 0AH
When the C24 receives the data,it issues reception data read request to the CPU module

■Registered portion of 2-byte data for code of registration data, FFH+00H to FFH

C24 receives and checks if the received data is variable data corresponding to the combination of the registered codes FFH and 00H to FFH.

For example, if a sum check code is registered, C24 calculates the sum check code from the receive data and checks if it is the same as the received sum check code. If the two codes are not the same, C24 performs error processing.

If C24 station number is registered, C24 checks if the received station number is the same as the station number set in C24. If the two station numbers are not the same, C24 processes the data as if normal data was received instead of a user frame.

Ex.

When data code 03H, FFH, F0H, 0DH, 0AH is registered as user frame number, 3E9H

When user frame No. 3E9H is set for data reception, C24 calculates, receives, and checks the sum check code as the receive data of that user frame portion.

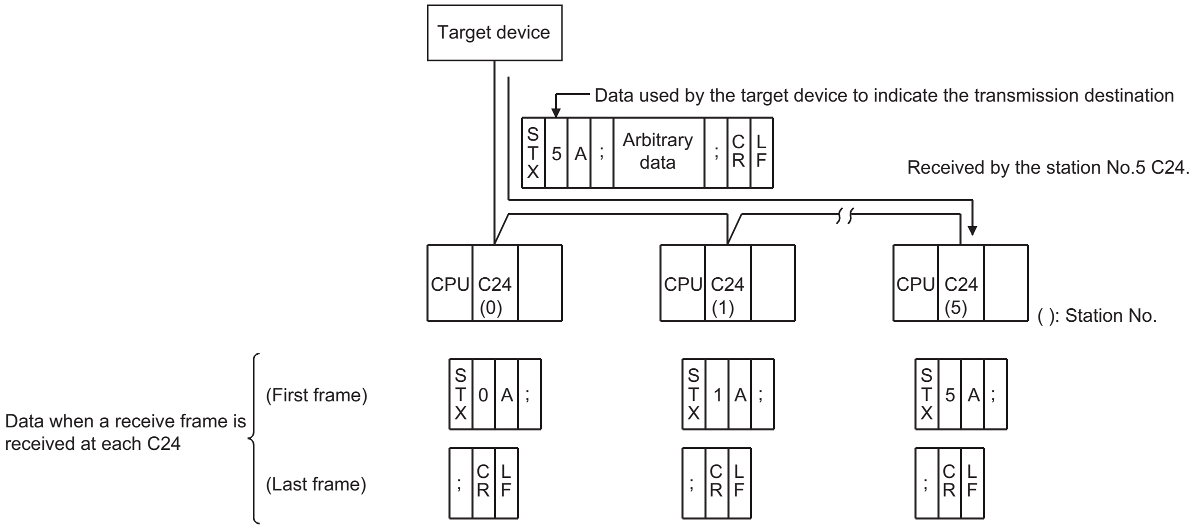
■In a reception message

C24 removes the user frame data portion from the reception message.

(The CPU module cannot read this data.)

■Remarks

With the multidrop connection, the module on a given station stores only arbitrary data portion of a message transmitted by the target device in the receive data storage area by registering each receive user frame specific to respective modules to be connected.



15.3 Considerations for Registering, Reading, and Deleting User Frames

The following shows the considerations for registering user frames and using registered user frames to transmit data to and receive data from the target device normally.

Considerations for registering, reading or deleting user frames

■Registration of user frame

A user frame can be registered using one of the following methods.

- Registering by using the module extended parameters. (☞ Page 297 Registration method using module extended parameters)
- Registering with the dedicated instruction "PUTE" from the CPU module. (📖MELSEC iQ-R Programming Manual (Module Dedicated Instructions))
- Registering from a target device with command "1610" through communications using the MC protocol. (📖 MELSEC Communication Protocol Reference Manual)

When a user frame is registered using the dedicated instruction or the MC protocol after the registration to the CPU built-in memory or SD memory card by module extended parameters, it is replaced with the contents registered using module extended parameters by powering OFF → ON or switching the CPU module STOP → RUN.

■Setting required to register/delete user frame

The following settings are required when registering or deleting user frames.

- Set "Setting change" to "Enable" in the transmission setting of the parameter.

■Registering, reading or deleting user frames from the CPU module

Register/read/delete a user frame when no data is communicated with the target device.

■Registration of sum check code

A user frame only with sum check code for variable data cannot be registered.

To register sum check code, add arbitrary data.

■User frame to be used for reception

Register a receive user frame in the CPU module or the flash ROM of C24.

■Designation of variable data

The variable data (05H to F9H) can be designated at only one place in the final frame.

Considerations for using user frames

■Transmitting/receiving data using user frames

It is necessary to set the user frame number to be used to the buffer memory prior to data transmission/reception (receive user frames must be set at the startup of C24).

■Data communication procedure when the number for a receive user frame is set from the CPU module

The procedure to set a receive user frame number form the CPU module and receive data is shown below.

Receive user frame data by performing the following operations sequentially.

- Restart the CPU module when a user frame to be used for data reception was registered from the CPU module.
- Set the receive user frame number to the buffer memory at the startup of C24, and write "1" in the 'User frame use enable/disable designation' (Un\G173/333) on the buffer memory.
- Start data reception once the value in the user frame use enable/disable designation area on the buffer memory changes to "2".
- Check whether data is received normally from the target device.

■When additional code data is received during data reception using the nonprocedural protocol

C24 does not assume that 1-byte data immediately after it is data for the following controls.

- Data received as the start frame, final frame for user frame (☞ Page 282 User Frame Types and Contents During Communication, ☞ Page 291 Transmission/Reception Processing Using User Frame Registration Data)
- Receive end code data

Therefore, do not set a user frame containing data receive additional code data as a user frame for data reception using the nonprocedural protocol.

■Arbitrary data portion in a message received from a target device

The arbitrary data portion of a message received from a target device cannot include data with the same arrangement (same code) as the final frame.

Ex.

User frame (First frame)	Arbitrary data	User frame (Last frame)

■Data bit length of transmission specification

In the following cases, set the data bit length of the transmission specification to 8 bits.

(Set this in "Transmission Setting" of the parameter with Engineering tool for the C24 side.)

- When transmitting and receiving the sum check code of the variable data as binary data (registered code: FFH+EEH, FFH+F0H, FFH+F4H, FFH+F6H)
- When transmitting and receiving a user frame containing data codes 80H to FFH

15.4 Registering/Reading/Deleting User Frames

This section explains registering, reading, and deleting user frames in/from C24 flash ROM or buffer memory.

Type of user frames

Type		User frame No.	Registration destination	Remarks
Data communication function	Default registration frame	1H to 3E7H (1 to 999)	ROM for C24 operating system	Read enabled
	User frame	3E8H to 4AFH[*1,*2] (1000 to 1199)	C24 flash ROM	Register/read/delete enabled
		8001H to 801FH[*3] (-32767 to -32737)	C24 buffer memory (Un\G6912 to 8182)	
User frame for the programmable controller CPU monitoring function		B001H to B00AH, B061H, B080H to B082H	ROM for C24 operating system	Register/read/delete disabled

*1 A user frame cannot be overwritten for registration. To reregister a user frame having the same number, first delete the current user frame and then reregister.

*2 Check the registration status of the user frame number with Engineering tool.

*3 The data arrangement of user frame for registration or reading is the same as the one for registration or reading to/from the flash ROM. Register or read user frames in accordance with the arrangement above.
Although user frames can registered in the buffer memory, register user frames of the fixed format portion in the flash ROM as much as possible.

Devices that can register/read/delete user frames

Type	User frame No.	Device that can register/read/delete								
		CPU module			Target device			Engineering tool		
		Register	Read	Delete	Register	Read	Delete	Register	Read	Delete
Default registration frame	1H to 3E7H	×			×	○	×	×		
User frame	3E8H to 4AFH	○[*1]	○[*2]	○[*1]	○			○		
	8001H to 801FH	○						×		
User frame for the programmable controller CPU monitoring function	B001H to B00AH, B061H, B080H to B082H	×								

*1 Use the PUTE instruction.

*2 Use the GETE instruction.

Precautions

Register/read/delete user frames by a program while no data is communicated with the target device.

Registration method using module extended parameters

Register the user frame to be used for "Registration frame No. 1000" to "Registration frame No. 1199".

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ Module Name ⇨ [Module Extended Parameter] ⇨ "User frame contents"

Buffer memory to be used

Requirement for reading/writing from CPU module

○: Always performed, △: Performed as required, ×: Unnecessary

Address (Hexadecimal (decimal))	Name		Stored value	Processing		
				Register	Read	Delete
2H (2)	For flash ROM access	Register/read/delete instruction	0: No request 1: Registration request 2: Read request 3: Deletion request	○	○	○
3H (3)		Frame number direction	0: No designation 1000 to 1199 (3E8H to 4AFH): Frame No. to be registered/read/deleted			
4H (4)		Registration/read/delete result storage	0: Normal completion Other than 0: Abnormal completion (error code)			
5H (5)		Registration data byte count designation (☞ Page 298 Registration data byte count designation (Un\G5, 6912, 6953, ..., 8142))	0: When deleting 1 to 80 (1H to 50H): Registered code byte count	○	○	×
6H (6) to 2DH (45)		User frame (☞ Page 298 User frame (Un\G6 to 45, 6913 to 6952, 6954 to 6993, ..., 8143 to 8182))	Code of frame to be registered/read (for 80 bytes)			
204H (516)		Number of registered user frame storage	0: No registration to flash ROM 1 to 200 (1H to C8H): Number of registration to flash ROM	△	△	△
205H (517) to 21DH (541)		User frame registration status storage (☞ Page 298 User frame registration status (For registration number check) (Un\G517 to 541)) (For registration No. check)	0: No registration in the given range Other than 0 : Registration status			
21EH (542)	Number of registered default registration frames storage (ROM for operating system)		n: number of registration (☞ Page 290 Default registration frame (read only))			
1B00H (6912)	Registration No. 8001H	Registration data byte count designation (☞ Page 298 Registration data byte count designation (Un\G5, 6912, 6953, ..., 8142))	• 1 to 80 (1H to 50H): Number of registered code bytes • Registered frame code (for 80 bytes) (Registered area for 31 frames)	○	△	×
1B01H (6913) to 1B28H (6952)		User frame storage (for 40 words) (☞ Page 298 User frame (Un\G6 to 45, 6913 to 6952, 6954 to 6993, ..., 8143 to 8182))				
1B29H (6953)	Registration No. 8002H	Registration data byte count designation (☞ Page 298 Registration data byte count designation (Un\G5, 6912, 6953, ..., 8142))				
1B2AH (6954) to 1B51H (6993)		User frame storage (for 40 words) (☞ Page 298 User frame (Un\G6 to 45, 6913 to 6952, 6954 to 6993, ..., 8143 to 8182))				
1B52H (6994) to 1FCDH (8141)	⋮					
1FCEH (8142)	Registration No. 801FH	Registration data byte count designation (☞ Page 298 Registration data byte count designation (Un\G5, 6912, 6953, ..., 8142))				
1FCFH (8143) to 1FF6H (8182)		User frame storage (for 40 words) (☞ Page 298 User frame (Un\G6 to 45, 6913 to 6952, 6954 to 6993, ..., 8143 to 8182))				

■Registration data byte count designation (Un\G5, 6912, 6953, ..., 8142)

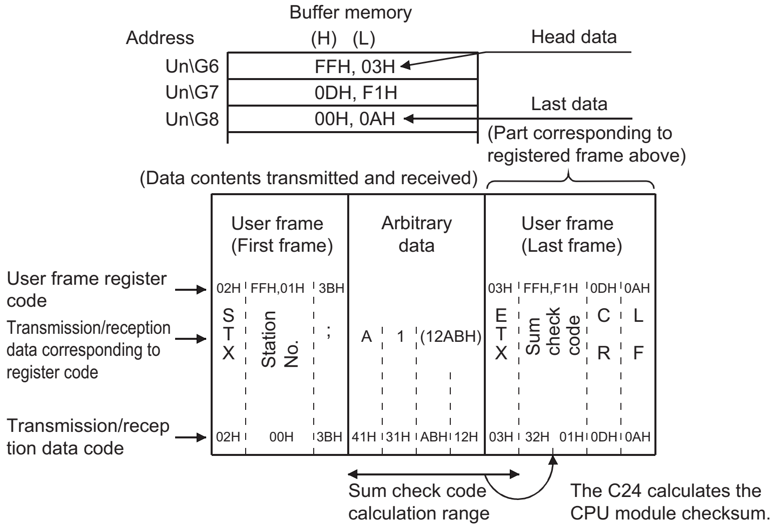
- The total number of bytes of registration data of user frame to be registered (written)/read is indicated.
- Accessing to the flash ROM
  At the time of registration, the user writes the total number of bytes of registration data.
  At the time of reading, the total number of bytes of registration data is stored.
- Accessing to the buffer memory
  At the time of registration, the user writes the total number of bytes of registration data.

■User frame (Un\G6 to 45, 6913 to 6952, 6954 to 6993, ..., 8143 to 8182)

- During the registration, the user sequentially stores the registration data of the user frame to be registered in the order of (L) and (H), beginning from the head area of the given area range.
- During the reading, the registration data of the registered user frame is stored with the same contents and arrangement as when registering.

Ex.

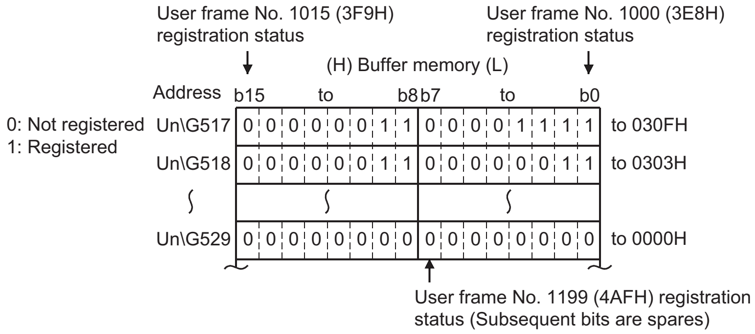
Contents stored in the user frame storage area when a user frame to transmit and receive ETX, sum check code, CR, and LF (registered codes: 03H, FFH, F1H, 0DH, 0AH) is registered to the flash ROM.


Buffer memory
Address
(H) (L)
Head data
Un\G6
FFH, 03H
Un\G7
0DH, F1H
Un\G8
00H, 0AH
Last data
(Part corresponding to registered frame above)
(Data contents transmitted and received)
User frame (First frame)
Arbitrary data
User frame (Last frame)
User frame register code
Transmission/reception data corresponding to register code
Transmission/reception data code
02H FFH,01H 3BH
03H FFH,F1H 0DH 0AH
STX
Station No.
;
A 1 (12ABH)
ETX
Sum check code
CR
LF
02H 00H 3BH 41H 31H ABH 12H 03H 32H 01H 0DH 0AH
Sum check code calculation range
The C24 calculates the CPU module checksum.

■User frame registration status (For registration number check) (Un\G517 to 541)

- The registration status of user frames to the flash ROM is stored with the values shown below.
- The contents of each area that indicates the registration status are shown below. The contents of the area are indicated in one user frame number/1 bit form.

Ex.


User frame No. 1015 (3F9H) registration status
User frame No. 1000 (3E8H) registration status
(H) Buffer memory (L)
Address
b15 to b8 b7 to b0
0: Not registered
1: Registered
Un\G517 0 0 0 0 0 0 1 1 0 0 0 0 1 1 1 1 to 030FH
Un\G518 0 0 0 0 0 0 1 1 0 0 0 0 0 0 1 1 to 0303H
Un\G529 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 to 0000H
User frame No. 1199 (4AFH) registration status (Subsequent bits are spares)

16 ON-DEMAND DATA COMMUNICATIONS USING USER FRAMES

During communications between a target device and a CPU module using the MC protocol, on-demand data can be transmitted from the CPU module to the target device with the on-demand function using user frames.

This chapter explains the transmission of the data specified by the CPU module using a message format other than the formats (1C frame format 1 to 4, 4C frame format 5) described in the following manual.

(📖MELSEC Communication Protocol Reference Manual)

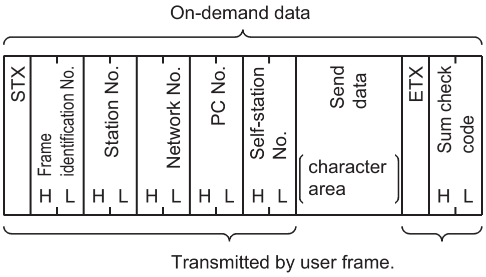
16.1 Data Communication Function Using User Frame

The data communication using user frame means transmitting/receiving the first and last parts of a message in the format selected by a user in data communications between the CPU module and target devices via C24.

By using the function described in this chapter, on-demand data in the order shown below can be transmitted from C24 to the target device.

Ex.

Transmitting data according to the order of 3C frame, format 1



As shown in the left, the user can sort the data except for the "Send data" portion.

Data can be communicated using the user frames by registering in advance (by data codes) message formats that can be transmitted/received by the target device according to its specifications as user frames in C24.

In the above figure, C24 transmits the on-demand data as described below:

- Sum check code of a user frame
  C24 calculates the sum check code according to the contents registered in advance by the user, and transmits it as ASCII/binary data.
- Other than sum check code of a user frame
  C24 transmits data of the code registered in advance by the user. (No conversion)
- Send data (character B portion)
  This is the data that the program requested to transmit with ONDEMAND instruction.
  It is transmitted with the same contents and in the same order as the one transmitted without using a user frame according to the communication protocol and word/byte designation by the parameters of Engineering tool. (📖 MELSEC Communication Protocol Reference Manual)

16.2 User Frame On-Demand Data Transmission and Buffer Memory Used

This section explains the on-demand data transmission processing using a user frame and the arrangement of on-demand data based on the user frame specification to the buffer memory.

Transmission of on-demand data using user frames

The following describes the transmission of on-demand data using user frames.

■Processing on the CPU module side

- Before issuing a transmission request to C24, set the No. of the user frame registered in C24 to the buffer memory shown in the table below.
- The execution procedures and control procedures of the CPU module side excluding the above are the same as transmitting on-demand data without user frame described in the following manual.
  (📖 MELSEC Communication Protocol Reference Manual)

■Processing of the target device side

- When the target device receives the user frame transmitted by C24 as the start frame, receive it as on-demand data.

Buffer memory to be used and the order of on-demand data

■Buffer memory to be used

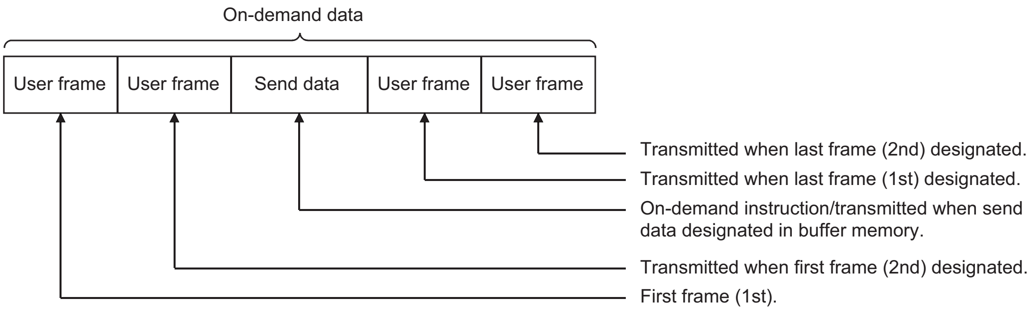
For on-demand data transmission using a user frame, the user frame to be transmitted is designated in the buffer memories shown below.

Address		Name			Description
CH1	CH2				
A9H	149H	On-demand user frame specification	Start frame No. designation	(1st)	Designates the number of the user frame to be transmitted as the start frame. 0H: No designation Others : Designated*1 For the 1st frame, other than 0 must be designated. When designating the 1st frame, the 2nd frame can also be designated.
AAH	14AH			(2nd)	
ABH	14BH		Final frame No. designation	(1st)	Designates the number of the user frame to be transmitted as the final frame. 0H: No designation Others : Designated*1 When designating the 1st frame, the 2nd frame can also be designated.
ACH	14CH			(2nd)	

*1 Designates the registration No. (shown below) of the user frame to be transmitted from among the user frames registered in C24.
1H to 3E7H (1 to 999) : Default registration frame
3E8H to 4AFH (1000 to 1199) : User frame (registered in flash ROM)
8001H to 801FH (-32767 to -32737) : User frame (registered in buffer memory)

■On-demand data arrangement

The following shows the order of on-demand data by user frame specification.


On-demand data
User frame
User frame
Send data
User frame
User frame
Transmitted when last frame (2nd) designated.
Transmitted when last frame (1st) designated.
On-demand instruction/transmitted when send data designated in buffer memory.
Transmitted when first frame (2nd) designated.
First frame (1st).

■Combination

Only the following combinations can be used for arrangement of on-demand data by user frame specification.

○: designated data

Combination	Data name				
	Start frame (1st)	Start frame (2nd)	Send data	Final frame (1st)	Final frame (2nd)
1	○	○	○	○	○
2	○	○	○	○	
3	○	○	○		
4	○	○			
5	○		○	○	○
6	○		○	○	
7	○		○		
8	○				

■Description of data to be transmitted

- The outline of data to be transmitted for on-demand data transmission by user frame specification is shown below.

Contents of data to be transmitted		Communication with ASCII code	Communication wit binary code
User frame • Start frame (1st) • Final frame (Last 1 frame)	The portions of which registered code is from 00H to FEH	The data of the code registered in C24 is transmitted. (No conversion)	
	Combined portions of which registered code is FFH and 00H to FFH	The data according to the user-designated contents, code, and number of bytes is transmitted.	
User frame • Start frame (2nd) • Final frame (Other than last 1 frame)	The portions of which registered code is from 00H to FEH	The data of the code registered in C24 is converted to ASCII data and transmitted.	The data of the code registered in C24 is transmitted. For 10H data, the data of 10H + 10H is transmitted.
	Combined portions of which registered code is FFH and 00H to FFH	The data of the contents, code, and number of bytes designated by the user to ASCII data and transmitted.	The data of the contents, code, and number of bytes designated by the user is transmitted. For 10H data, the data of 10H + 10H is transmitted.
Send data (📖MELSEC Communication Protocol Reference Manual)	—	The designated send data is converted to ASCII data and transmitted.	The designated send data is transmitted by the code unchanged. (No conversion) For 10H data, the data of 10H + 10H is transmitted.

16.3 On-Demand Function Control Procedure During User Frame Use

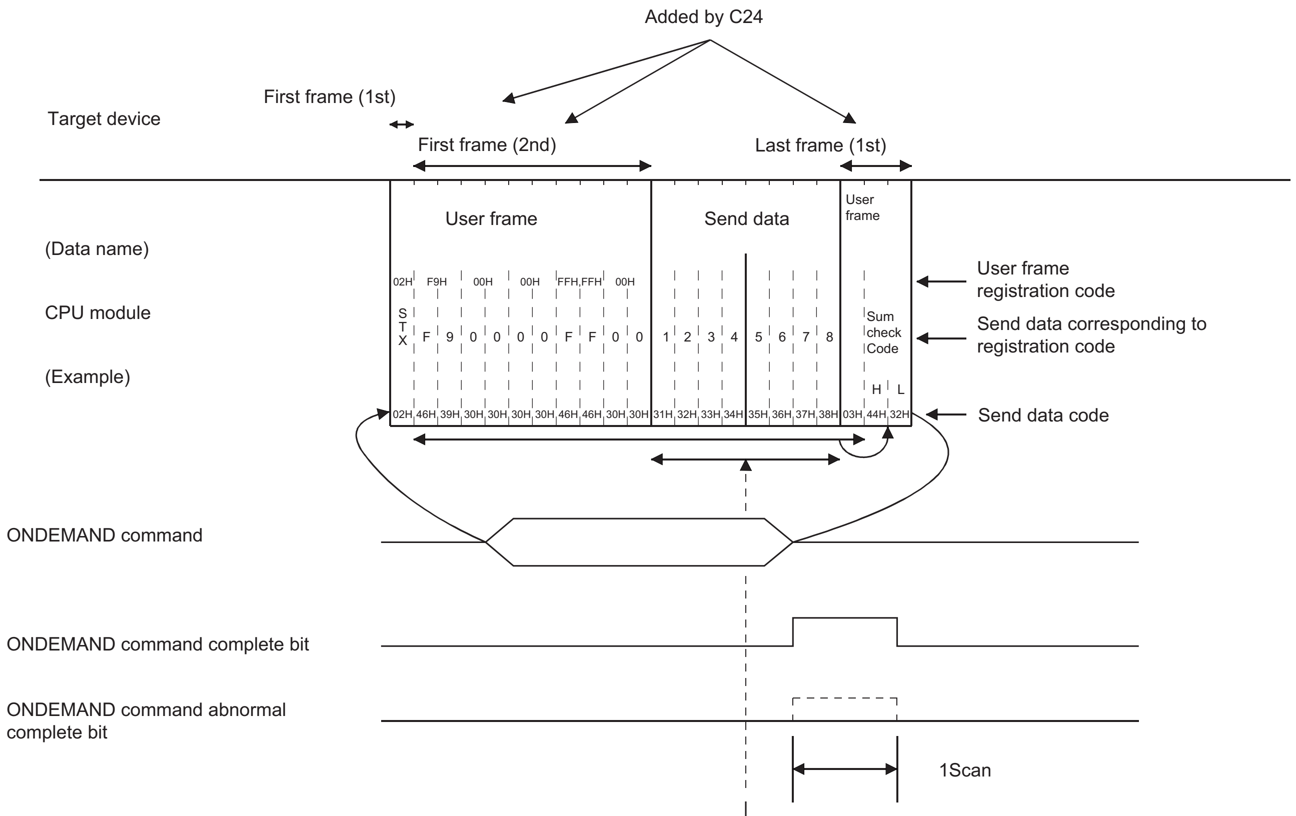
This section explains the control procedure when on-demand data is transmitted to a target device with the on-demand function by using a user frame.

Data communication using the ASCII code

Control procedure when two-word send data (1234H, 5678H) is designated by ONDEMAND instruction

- Set the "Communication protocol setting" to any one of "MC protocol (Format 1)" to "MC protocol (Format 4)".
- Set the "Station number" to "0".
- Set the "Word/byte units designation" to word unit.
- Set the "On-demand user frame specification" items to the following:
  Start frame No. 1st: 02H
  Start frame No. 2nd: 3EBH
  Final frame No. 1st: 401H
  Final frame No. 2nd: 0H (No specification)
- User frame registration

User frame No.	User frame (Registered code)	Contents of user frame registration
02H (2)	02H	Data code from STX to host station number matched to 3C frame, format 1
3EBH (1003)	F9H, 00H, 00H, FFH, FFH, 00H	
401H (1025)	03H, FFH, F1H	Data code of ETX, sum check code equivalent to 3C frame, format 1



Buffer memory

(Word units)	Un\G150	0
(First address)	Un\G160	C00H
(Data length)	Un\G161	2
(1st first frame No.)	Un\G169	02H
(2nd first frame No.)	Un\G170	3EBH
(1st last frame No.)	Un\G171	401H
(2nd last frame No.)	Un\G172	0H
(Result of execution)	Un\G598	0
(Send data)	Un\G3072	1234H
	Un\G3073	5678H
(CH1 side)		

When a transmission error is generated,
other than 0 is stored.

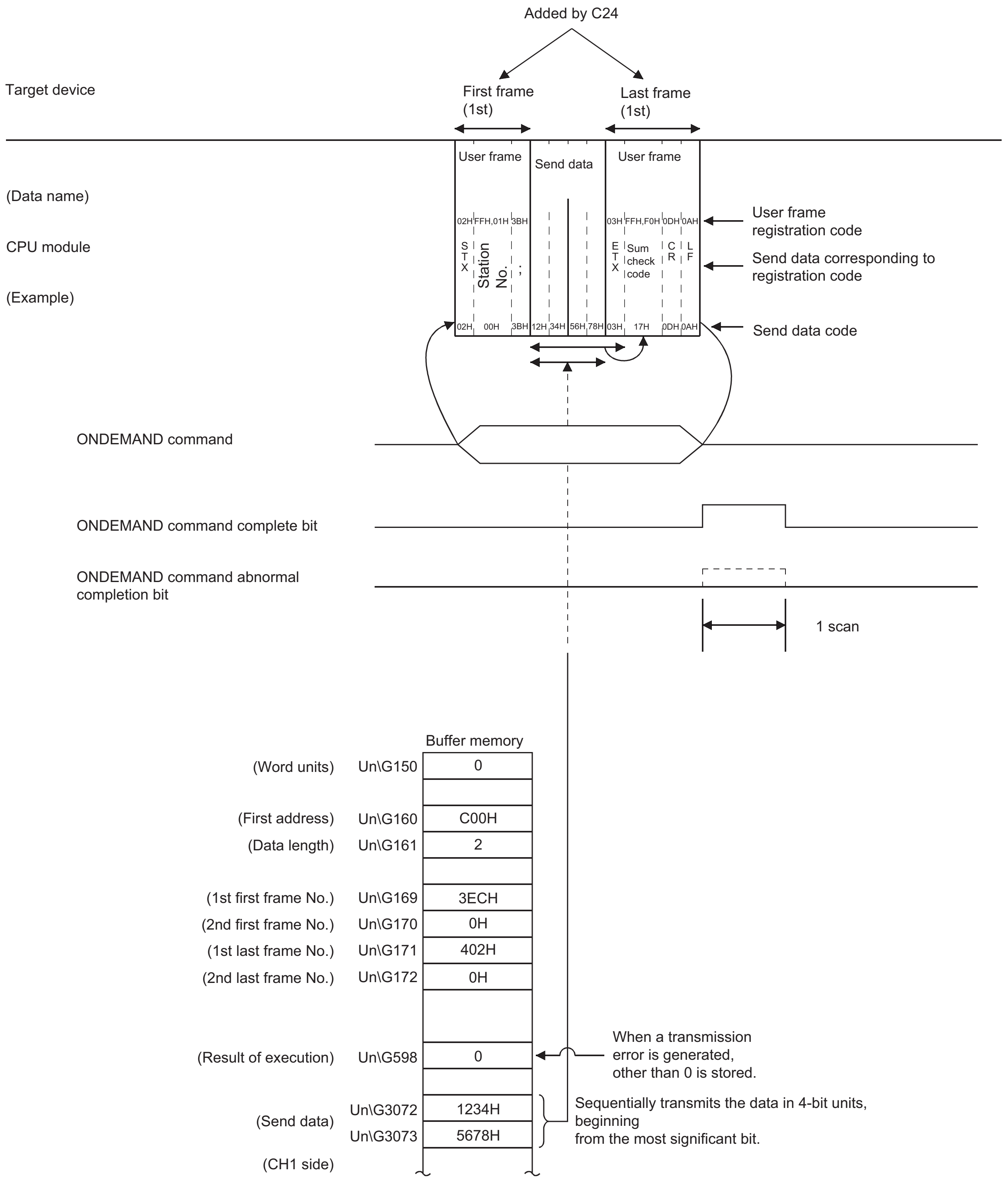
Sequentially transmits the data in 4-bit units, beginning
from the most significant bit.

Data communications using the binary code

Control procedure when two-word send data (1234H, 5678H) is designated by ONDEMAND instruction

- Set the "Communication protocol setting" to "MC protocol (Format 5)".
- Set the "Station number" to "0".
- Set the "Word/byte units designation" to word unit.
- Set the "On-demand user frame specification" items to the following:
  Start frame No. 1st: 3ECH
  Start frame No. 2nd: 0H (No specification)
  Final frame No. 1st: 402H
  Final frame No. 2nd: 0H (No specification)
- User frame registration

User frame No.	User frame (Registered code)	Contents of user frame registration
3ECH (1004)	02H, FFH, 01H, 3BH	STX + C24 station No. +;
402H (1026)	03H, FFH, F0H, 0DH, 0AH	ETX + sum check code + CR + LF Sum check code is designated by one byte binary code.


Added by C24
Target device
First frame
(1st)
Last frame
(1st)
(Data name)
CPU module
(Example)
User frame
Send data
User frame
02H
FFH,01H
3BH
S T X
Station No.
;
03H
FFH,F0H
0DH
0AH
E T X
Sum check code
C R
L F
02H
00H
3BH
12H
34H
56H
78H
03H
17H
0DH
0AH
User frame registration code
Send data corresponding to registration code
Send data code
ONDEMAND command
ONDEMAND command complete bit
ONDEMAND command abnormal completion bit
1 scan
Buffer memory
When a transmission error is generated, other than 0 is stored.
Sequentially transmits the data in 4-bit units, beginning from the most significant bit.
(CH1 side)

		Buffer memory
(Word units)	Un\G150	0
(First address)	Un\G160	C00H
(Data length)	Un\G161	2
(1st first frame No.)	Un\G169	3ECH
(2nd first frame No.)	Un\G170	0H
(1st last frame No.)	Un\G171	402H
(2nd last frame No.)	Un\G172	0H
(Result of execution)	Un\G598	0
(Send data)	Un\G3072	1234H
	Un\G3073	5678H

16.4 Example of On-Demand Data Transmission Program Using User Frames

The following shows a program example when transmitting on-demand data including user frames.

Configure the settings and perform registration with Engineering tool in advance.

On-demand data transmission program using user frames

■Input/output signal of C24

Install C24 in the position where the I/O signal is X/Y00 to X/Y1F.

■C24 interface used for communication with target device

CH1 side RS-232 interface of C24

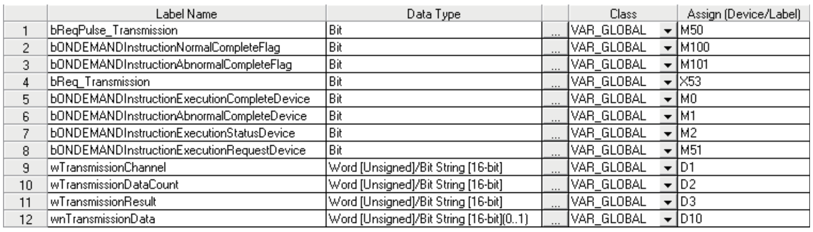
■Parameter settings

Setting item				Setting value
Basic settings	Various control specification	Communication protocol setting		MC protocol (format 1)
		Communication speed setting		19200 bps
		Transmission setting	Data bit	8 bits
			Parity bit	None
			Odd/even parity	Odd
			Stop bit	1 bit
			Sum check code	None
			Online change	Disable
			Setting change	Disable
		Station number setting (CH1, 2 in common: 0 to 31)		0
		Communication control specification	Word/byte units designation	Word
		Send wait time specification	Transmission wait time designation	0H
		On-demand function specification	Buffer memory start address designation	400H
			Data length designation	0H
Application settings	User frame specification	On-demand user frame specification	Start frame No. designation (1st)	2H
			Start frame No. designation (2nd)	3EBH
			Final frame No. designation (1st)	401H
			Final frame No. designation (2nd)	0H

■Program example

Transmit the on-demand data with the on-demand function, and designate the two-word data to be transmitted with the ONDEMAND instruction.

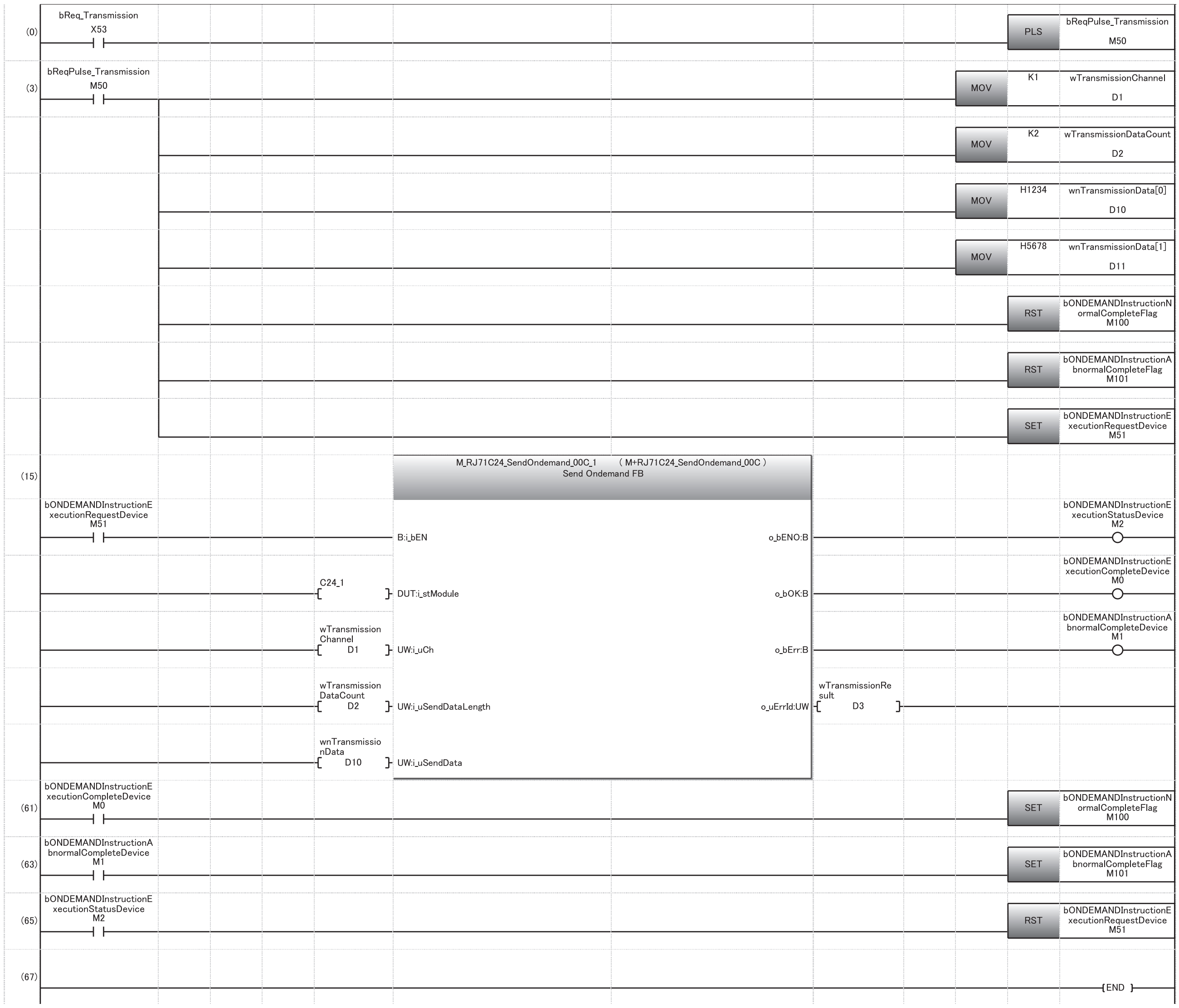
The data in the user frame portion of the on-demand data to be transmitted is the registered data of the user frame number registered with Engineering tool.

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_SendOndemand	Sends data using the on-demand function of MC protocol.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReqPulse_Transmission	Bit	...	VAR_GLOBAL	M50
2	bONDEMANDInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
3	bONDEMANDInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101
4	bReq_Transmission	Bit	...	VAR_GLOBAL	X53
5	bONDEMANDInstructionExecutionCompleteDevice	Bit	...	VAR_GLOBAL	M0
6	bONDEMANDInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
7	bONDEMANDInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M2
8	bONDEMANDInstructionExecutionRequestDevice	Bit	...	VAR_GLOBAL	M51
9	wTransmissionChannel	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
10	wTransmissionDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2
11	wTransmissionResult	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D3
12	wnTransmissionData	Word [Unsigned]/Bit String [16-bit](0..1)	...	VAR_GLOBAL	D10

For information on the module FB to be used, refer to the following manual.

MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
bReq_Transmission
X53
PLS
bReqPulse_Transmission
M50
(3)
bReqPulse_Transmission
M50
MOV
K1
wTransmissionChannel
D1
MOV
K2
wTransmissionDataCount
D2
MOV
H1234
wnTransmissionData[0]
D10
MOV
H5678
wnTransmissionData[1]
D11
RST
bONDEMANDInstructionNormalCompleteFlag
M100
RST
bONDEMANDInstructionAbnormalCompleteFlag
M101
SET
bONDEMANDInstructionExecutionRequestDevice
M51
(15)
M_RJ71C24_SendOndemand_00C_1 (M+RJ71C24_SendOndemand_00C)
Send Ondemand FB
bONDEMANDInstructionExecutionRequestDevice
M51
B:i_bEN
o_bENO:B
bONDEMANDInstructionExecutionStatusDevice
M2
C24_1
DUT:i_stModule
o_bOK:B
bONDEMANDInstructionExecutionCompleteDevice
M0
wTransmission Channel
D1
UW:i_uCh
o_bErr:B
bONDEMANDInstructionAbnormalCompleteDevice
M1
wTransmission DataCount
D2
UW:i_uSendDataLength
o_uErrId:UW
wTransmissionResult
D3
wnTransmissionData
D10
UW:i_uSendData
(61)
bONDEMANDInstructionExecutionCompleteDevice
M0
SET
bONDEMANDInstructionNormalCompleteFlag
M100
(63)
bONDEMANDInstructionAbnormalCompleteDevice
M1
SET
bONDEMANDInstructionAbnormalCompleteFlag
M101
(65)
bONDEMANDInstructionExecutionStatusDevice
M2
RST
bONDEMANDInstructionExecutionRequestDevice
M51
(67)
END

(0) On-demand transmission direction pulse

(3) Set the transmission channel to the CH1 side.
Set the send data quantity to two-word.
Set the data to be transmitted.
Reset the normal completion flag.
Reset the abnormal completion flag.
Set the execution direction.

(61) Normal completion

(63) Abnormal completion

(65) Reset the execution direction.

Point

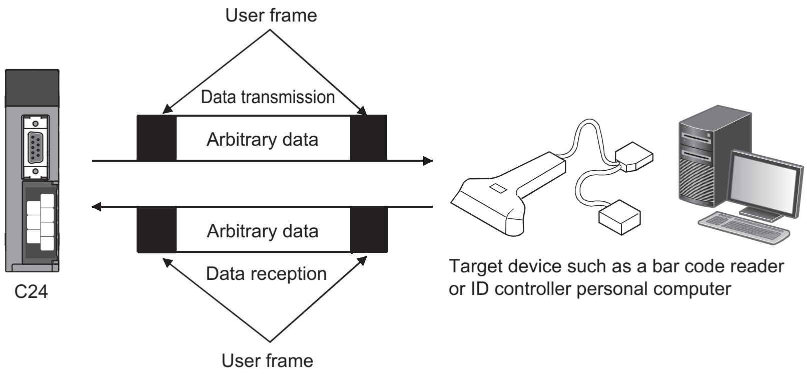
- The SPBUSY instruction is used to read the communication status by the dedicated instruction. (☞ Page 580 Dedicated Instructions)
- Designate the storage capacity for send data (stored in D10 to D11 in the above program example) and data length (stored in D2 in the above program example) so that they do not exceed the range of buffer memory assigned by the user for the on-demand function.

17 DATA COMMUNICATIONS USING USER FRAMES

Registering the fixed format portion of the message transmitted/received by a target device and C24 as a user frame beforehand allows data transmission/reception using a user frame.

Data transmission/reception using a user frame simplifies the creation of data to be transmitted on the CPU module side, and the program for checking received data.

This chapter explains the data transmission/reception methods and procedures when communication data using user frames with the nonprocedural protocol of C24.



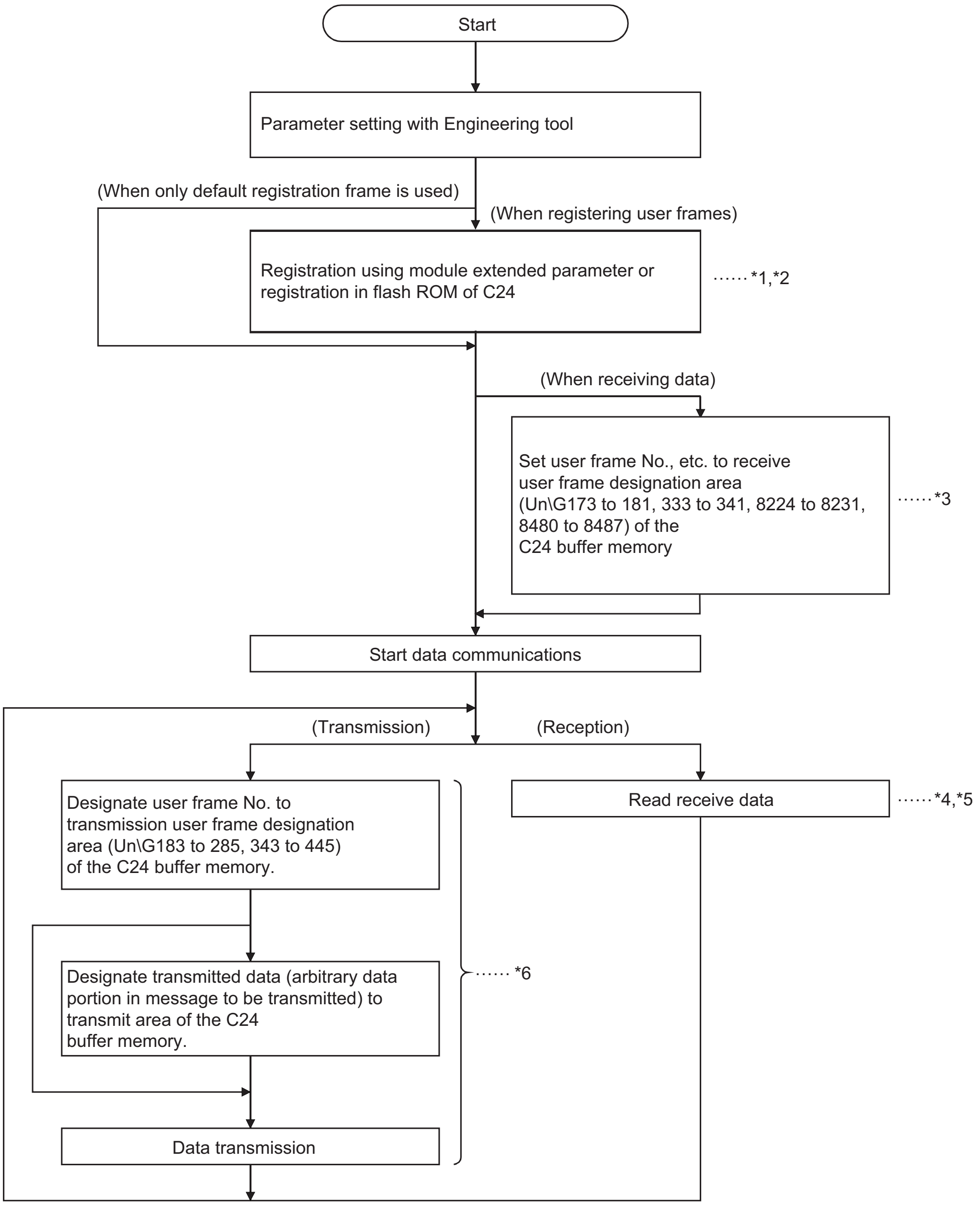
Point

For details on "Transparent code" and "ASCII-BIN conversion", refer to the following chapters.

- When using transparent codes (☞ Page 358 TRANSPARENT CODES AND ADDITIONAL CODES)
- When performing data communication using ASCII-BIN conversion (☞ Page 379 COMMUNICATING WITH ASCII CODE (ASCII-BIN CONVERSION))

17.1 Procedure of Data Communication

This section explains an overview of the procedure when performing data communication between a target device and a CPU module using a user frame.


Start
Parameter setting with Engineering tool
(When only default registration frame is used)
(When registering user frames)
Registration using module extended parameter or registration in flash ROM of C24
······*1,*2
(When receiving data)
Set user frame No., etc. to receive user frame designation area (Un\G173 to 181, 333 to 341, 8224 to 8231, 8480 to 8487) of the C24 buffer memory
······*3
Start data communications
(Transmission)
(Reception)
Designate user frame No. to transmission user frame designation area (Un\G183 to 285, 343 to 445) of the C24 buffer memory.
Designate transmitted data (arbitrary data portion in message to be transmitted) to transmit area of the C24 buffer memory.
Data transmission
······*6
Read receive data
······*4,*5

*1 ☞ Page 454 PARAMETER SETTING

*2 ☞ Page 282 CONTENTS AND REGISTRATION OF USER FRAMES FOR DATA COMMUNICATION

*3 ☞ Page 321 Setting of receive user frames

*4 ☞ Page 310 Receive data

*5 ☞ Page 316 Timing for start/completion of data reception

*6 ☞ Page 345 Data Transmission

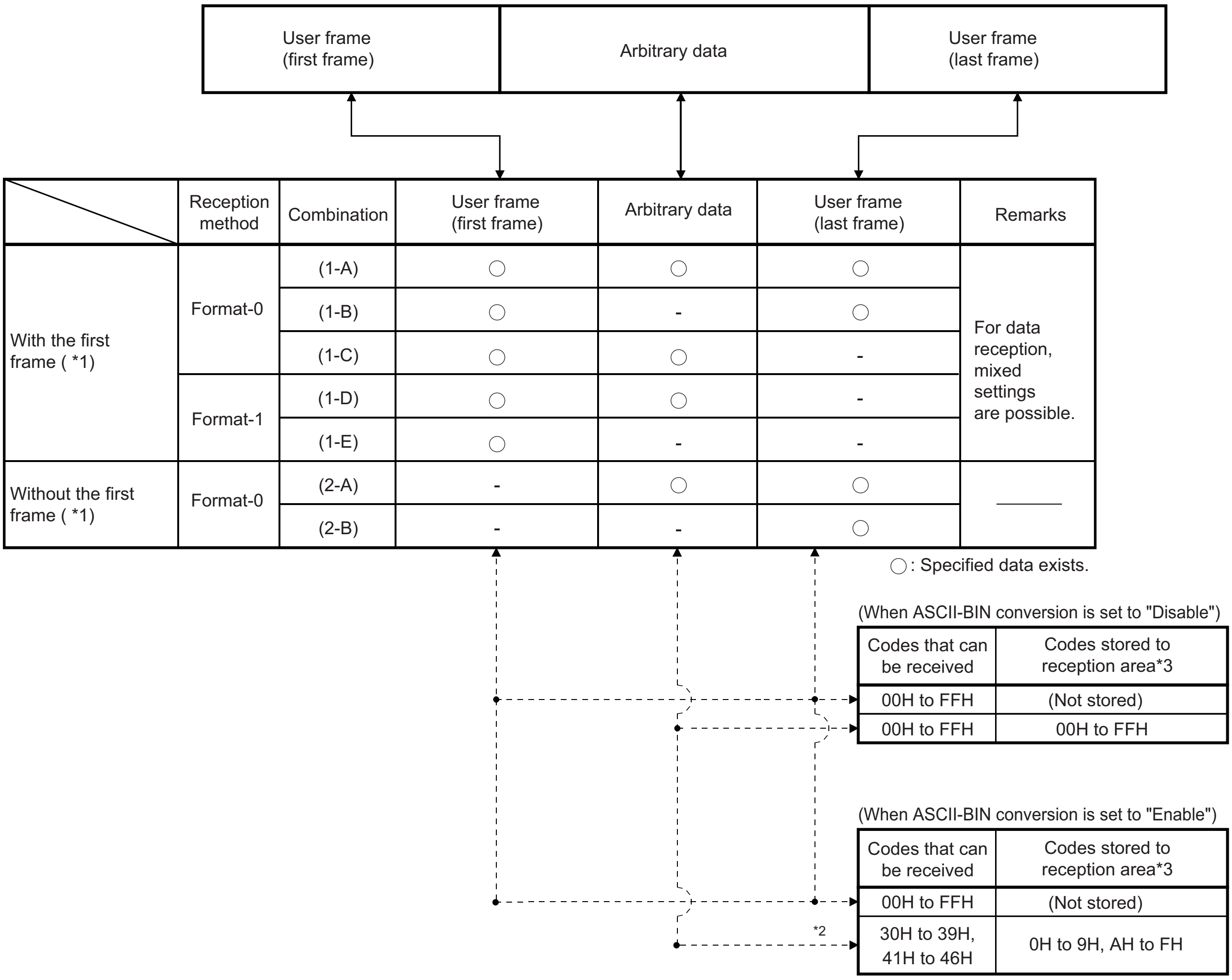
17.2 Data Reception

When receiving data using a user frame, the reception method on C24 side includes Method 0 and Method 1.

The following explains data reception for each method.

Receive data

In data reception using a user frame, data arranged as indicated below can be received.



	Reception method	Combination	User frame (first frame)	Arbitrary data	User frame (last frame)	Remarks
With the first frame (*1)	Format-0	(1-A)	○	○	○	For data reception, mixed settings are possible.
		(1-B)	○	-	○	
		(1-C)	○	○	-	
	Format-1	(1-D)	○	○	-	
		(1-E)	○	-	-	
Without the first frame (*1)	Format-0	(2-A)	-	○	○	———
		(2-B)	-	-	○	

○: Specified data exists.

(When ASCII-BIN conversion is set to "Disable")

Codes that can be received	Codes stored to reception area*3
00H to FFH	(Not stored)
00H to FFH	00H to FFH

(When ASCII-BIN conversion is set to "Enable")

Codes that can be received	Codes stored to reception area*3
00H to FFH	(Not stored)
30H to 39H, 41H to 46H	0H to 9H, AH to FH

*1 User frames for receiving
· The user frame for data reception can be set to a maximum of four combinations of the start frame and the final frame, regardless of whether there is a specification.
· For a combination that specifies the start frame (with a start frame), specifying the start frame is required even in other combinations.
· For a combination that does not specify the start frame (without a start frame), the start frame cannot be specified even in other combinations.
· A combination with the start frame and a combination without the start frame cannot be combined.

*2 If data of other than 30H to 39H and 41H to 46H are received as the data code of the arbitrary data area (including the transparent code data), an error will occur after the C24 ASCII-BIN conversion processing.

*3 Arbitrary data portion of received data
When the arbitrary data portion is stored to the receive area, and the storage byte quantity is an odd number of bytes, the receive data quantity shown below is stored to the receive data quantity storage area. (When ASCII-BIN conversion is enabled, the receive data quantity is the storage byte count when the arbitrary data portion is converted to binary code and stored to the receive area.)
· Word units: Receive data quantity = Number of bytes stored to receive area ÷ 2 (Digits after a decimal points are rounded up.)
· Byte units: Receive data quantity = Number of bytes stored to receive area
(00H is stored to the upper byte of the last data storage location of the receive area.)
When "ASCII-BIN conversion" is enabled, make the arbitrary data part of the receive data an even number of bytes excluding the additional code.

Point

Handling of received data by C24

- When a user frame (start frame, final frame) registered to C24 is received, the reception processing by user frame is performed.
- Of the start frame No. and final frame No. (maximum 4 sets) set in default by the user for the user frame specification for receiving area in buffer memory, the set number (nth) of the user frame received from the target device is stored to the receive user frame storage area.
- When a transparent code for receiving is specified, the additional code data included in the arbitrary data portion is deleted immediately after reception (or, before conversion when ASCII-BIN conversion is performed).

Reception with the start frame (combination 1-A to C) (reception using Method 0)

■Reception of combination (1-A)

This is a method that any data portion of a reception message which can be handled by the CPU module side is enclosed by a start frame and a final frame and transmitted from a target device.

C24 starts the reception processing when data of the same arrangement as the start frame is received.

When data of the same arrangement as the final frame is received, the arbitrary data is stored in the receive area on the buffer memory and a read request is issued to the CPU module.

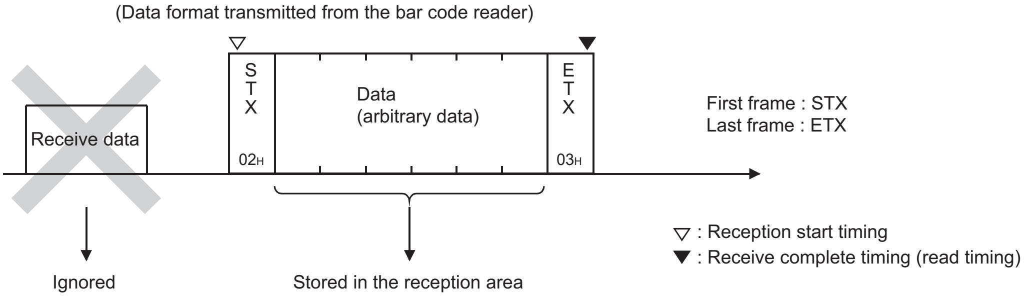
Precautions

- Any received data prior to the start frame will be ignored.
- Set a data quantity that exceeds the size of the arbitrary data portion to be received for the receive end data quantity initially set to C24.
- Reception processing by the receive end code initially set to C24 will not be performed. Received data of a receive end code are treated as an arbitrary data portion.

Ex.

When data transmitted from a bar code reader is received

Register STX as the start frame and register ETX as the final frame according to the message format of the bar code reader.


(Data format transmitted from the bar code reader)
Receive data
S T X
02H
Data
(arbitrary data)
E T X
03H
First frame : STX
Last frame : ETX
Ignored
Stored in the reception area
▽ : Reception start timing
▼ : Receive complete timing (read timing)

■Reception of combination (1-B)

This is a method that message to be received by the CPU module side are all transmitted in a fixed format from a target device.

C24 starts the reception processing when data of the same arrangement as the start frame is received.

When data of the same arrangement as the final frame is received, a read request is issued to the CPU module.

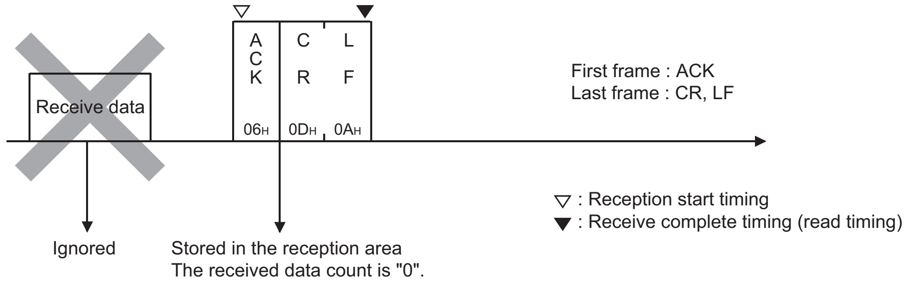
Since there is no arbitrary data portion, the receive data quantity will be "0" when a read request is issued to the CPU module.

Precautions

- Any received data prior to the start frame will be ignored.
- The receive end data quantity initially set to C24 uses the default value.

Ex.

When ACK is registered as the start frame, and CR+LF is registered as the final frame


Receive data
A C K
C R
L F
06H
0DH
0AH
First frame : ACK
Last frame : CR, LF
▽ : Reception start timing
▼ : Receive complete timing (read timing)
Ignored
Stored in the reception area
The received data count is "0".

■Reception of combination (1-C)

This is a method that the start of data transmission from a target device to the CPU module side is notified by a start frame, an arbitrary data portion of a fixed length is repeatedly transmitted from a target decide after that.

After the start frame is received, the subsequent received data will be handled as arbitrary data portion.

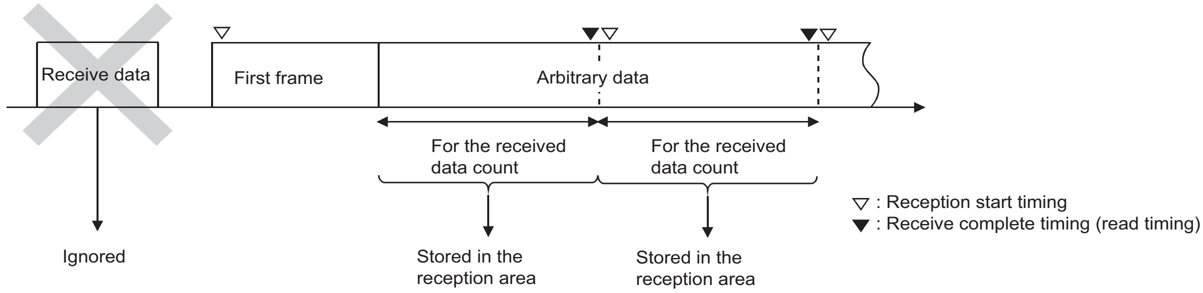
C24 starts the reception processing when data of the same arrangement as the start frame is received.

A read request is repeatedly issued to the CPU module when an arbitrary data portion, which is equivalent to the receive end data quantity initially set to C24, is received.

Precautions

- Any received data prior to the start frame will be ignored.
- The receive end data quantity initially set to C24 should be a data quantity for an arbitrary data (fixed length) transmitted from the target device.
- Reception processing by the receive end code initially set to C24 will not be performed. Received data of a receive end code are treated as an arbitrary data portion.

Ex.


Receive data
First frame
Arbitrary data
For the received data count
For the received data count
Ignored
Stored in the reception area
Stored in the reception area
▽ : Reception start timing
▼ : Receive complete timing (read timing)

Reception with the start frame (combination 1-D, 1-E) (reception using Method 1)

This is a method that an arbitrary data portion, which is equivalent to the 'Method 1 dedicated receive end data quantity' initially set to C24, is transmitted from a target device with a start frame.

C24 starts the reception processing when data of the same arrangement as the start frame is received.

When an arbitrary data, which is equivalent to the specified 'Method 1 dedicated receive end data quantity', is received, the arbitrary data portion is stored in the receive area on the buffer memory and a read request is issued to the CPU module.

Point

- Method 1 dedicated receive end data quantity

The receive end data quantity for data reception by using Method 1 is referred to as 'method 1 dedicated receive end data quantity'.

It can be specified with the number of word/byte of "0" or bigger (the unit depends on the word/byte units designation) for each combination of the start frame and the final frame, which specifies Method 1 (up to 4 combinations).

Precautions

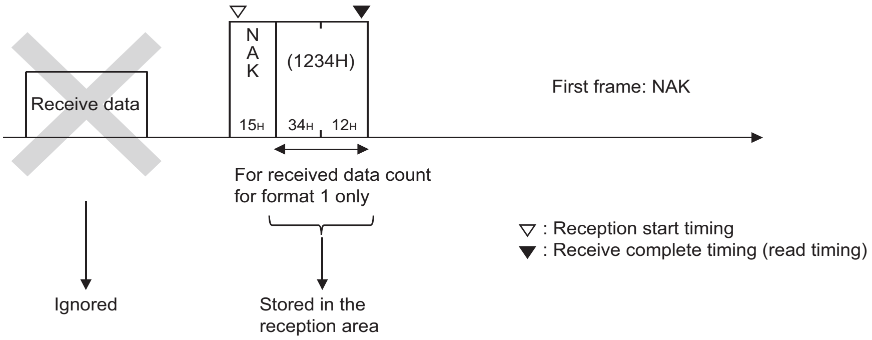
- Any received data prior to the start frame will be ignored.
- The receive end data quantity initially set to C24 is not used.
- Reception processing by the receive end code initially set to C24 will not be performed. Received data of a receive end code are treated as an arbitrary data portion.
- For reception using Method 1, set "No designation" to the final frame.

Ex.

By specifying a user frame, in which only ACK (06H) is registered, as the start frame and also the 'method 1 dedicated receive end data quantity' for an arbitrary data portion as 0 bytes, a read request will be issued to the CPU module upon the reception of a 1-byte ACK.

Ex.

By specifying a user frame, in which only NAK (15H) is registered, as the start frame and also the 'method 1 dedicated receive end data quantity' for an arbitrary data portion as 2 bytes, a read request will be issued to the CPU module upon the reception of NAK + 2-byte data.


Receive data
N A K
(1234H)
15H
34H
12H
First frame: NAK
For received data count for format 1 only
▽ : Reception start timing
▼ : Receive complete timing (read timing)
Ignored
Stored in the reception area

Reception without the start frame (combination 2-A, 2-B) (reception using Method 0)

■Reception of combination (2-A)

This is a method that a user frame is used as the final frame in place of the end code for nonprocedural protocol data reception and transmitted from a target device with an arbitrary data portion.

C24 starts the reception processing when the arbitrary data portion is received.

When data of the same arrangement as the final frame is received, the arbitrary data is stored in the receive area on the buffer memory and a read request is issued to the CPU module.

Precautions

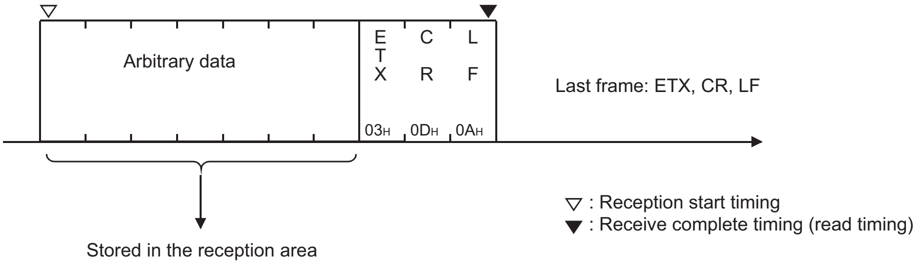
- Any received data prior to the final frame will be all treated as an arbitrary data portion.
- The receive end data quantity initially set to C24 should be a data quantity that exceeds the size of the arbitrary data to be received.
- C24 performs the following processing with respect to the received data that is the same as the receive end code initially set to C24.

  When it is included in the arbitrary data portion: C24 performs the reception processing by the receive end code.

  When it is included in the final frame: C24 does not perform the reception processing by the receive end code.

Ex.

By registering ETX + CR + LF as the final frame, a read request is issued to the CPU module when the end of the received data receives a message of ETX + CR+ LF.


▽
▼
Arbitrary data
E T X
C R
L F
03H
0DH
0AH
Last frame: ETX, CR, LF
Stored in the reception area
▽ : Reception start timing
▼ : Receive complete timing (read timing)

■Reception of combination (2-B)

This is a method that a user frame is used as the final frame in place of the end code for nonprocedural protocol data reception and data in a fixed format is transmitted from a target device.

C24 issues a read request to the CPU module when data of the same arrangement as the final frame is received.

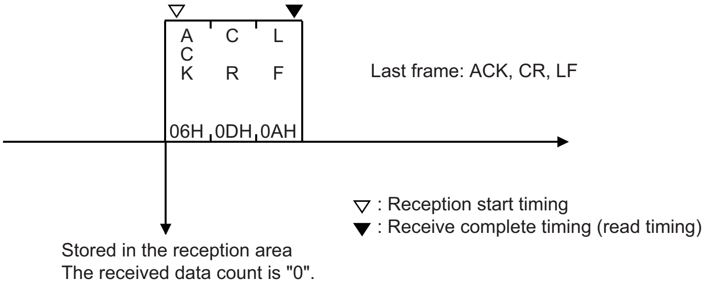
Upon data reception of this combination which contains no arbitrary data portion, the receive data quantity will be "0" when a read request is issued to the CPU module.

Precautions

- Any received data prior to the final frame will be all treated as an arbitrary data portion.
- The receive end data quantity initially set to C24 uses the default value.
- When received data that is the same as the receive end code initially set to C24 is included in the final frame, C24 does not perform the reception processing by the receive end code.

Ex.

By registering ACK + CR + LF as the final frame, a read request is issued to the CPU module when the end of the received data receives a message of ACK + CR + LF.

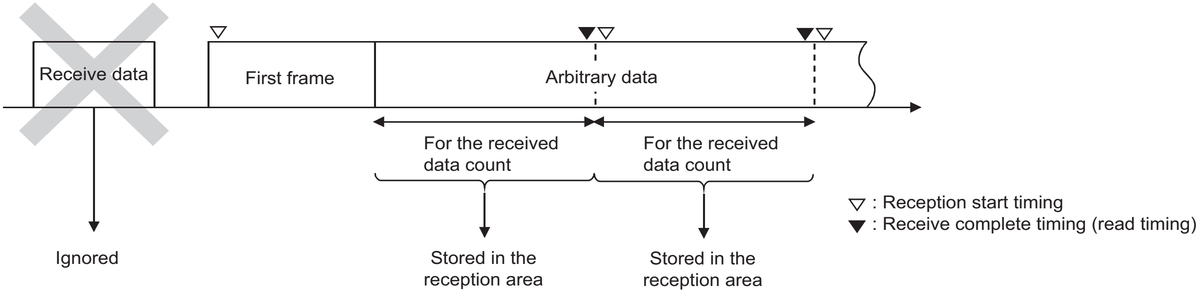

▽
▼
A C K
C R
L F
06H
0DH
0AH
Last frame: ACK, CR, LF
Stored in the reception area
The received data count is "0".
▽ : Reception start timing
▼ : Receive complete timing (read timing)

Remarks

The following shows the difference between the reception methods (Method 0 and Method 1) in regard to how to treat the received data when data in combination of (a start frame + and arbitrary data portion) is received.

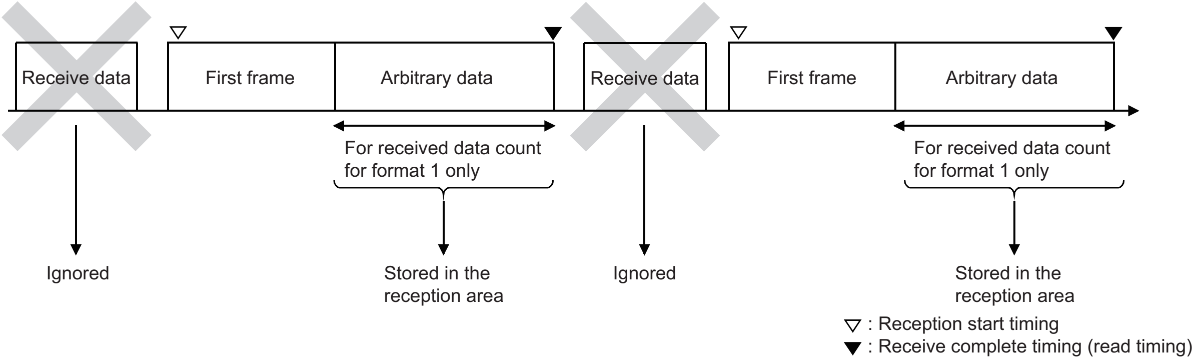
■When data is received using Method 0 (combination (1-C))

- C24 regards all arbitrary data portions as valid data after a start frame has been received, and stores it sequentially in the receive area.
- C24 issues a read request to the CPU module each time arbitrary data portions, which are equivalent to the receive end data quantity, are received, and repeats this process.


Receive data
First frame
Arbitrary data
For the received data count
For the received data count
Ignored
Stored in the reception area
Stored in the reception area
▽ : Reception start timing
▼ : Receive complete timing (read timing)

■When data is received using Method 1 (combination (1-D, 1-E))

- C24 regards arbitrary data portions, which are equivalent to the 'method 1 dedicated receive end data quantity' specified for combination of the received start frames, as valid data after a start frame has been received, and stores it in the receive area. It then issues a read request to the CPU module.
- Any received data after data, which is equivalent to the 'method 1 dedicated receive end data quantity', has been received up to the next start frame will be ignored. (Data will not be stored in the receive area.)


Receive data
First frame
Arbitrary data
Receive data
First frame
Arbitrary data
For received data count for format 1 only
For received data count for format 1 only
Ignored
Stored in the reception area
Ignored
Stored in the reception area
▽ : Reception start timing
▼ : Receive complete timing (read timing)

Timing for start/completion of data reception

This section explains the reading of received data based on the user frame and other factors (such as the receive end code and receive end data quantity) during data reception using a user frame.

Timing for start/completion of data reception

The following describes the timing for start/completion of the data reception processing with C24.

- Data reception using a user frame
- Data reception using the receive end code and receive end data quantity initially set in C24
- Data reception using 'method 1 dedicated receive end data quantity'

Type	Setting of the receive user frame	When using Method 0	When using Method 1
Data reception start	With a start frame	When the start frame is received.	
	Without a start frame	When the first data of arbitrary data is received.	—
Data reception completion	—	The factors of data reception completion (timing of reading received data to the CPU module) is one of the following:	
		(Depends on the conditions of the prior settings. ☞ Page 317 Timing chart for reception processing of C24)	
		• When the final frame is received. • When data of the receive end code is received. (In the case of a combination without the start frame) • When data which is equivalent to the receive end data quantity is received. • When a reception error (time out for the no-reception monitoring time (timer 0)) occurs.	• When the start frame is received while "0" is specified for the 'method 1 dedicated receive end data quantity'. • When data which is equivalent to the 'method 1 dedicated receive end data quantity' is received while "1" or more is specified for the 'method 1 dedicated receive end data quantity'. • When a reception error (time out for the no-reception monitoring time (timer 0)) occurs.
		Arbitrary data portion up to the reception of any listed above or the occurrence of a reception error is stored in the receive area in the buffer memory.	

Timing chart for reception processing of C24

The charts indicating the timing of the reception processing for data reception using a user frame, including the reception processing by the receive end data quantity, are shown below.

The numbers in the table indicate the timing of a received data read request to the CPU module.

<table>
<tr><th colspan="2" rowspan="3">Combination</th><th rowspan="3">Amount of received data of arbitrary data</th><th colspan="8">Timing pattern number</th></tr>
<tr><th colspan="8">Received data</th></tr>
<tr><th>Received data prior to the reception of the start frame</th><th colspan="2">When receiving the start frame</th><th colspan="2">When receiving arbitrary data in the reception message</th><th colspan="2">When receiving the final frame</th><th>When receiving the end code[*1]</th></tr>
<tr><td rowspan="7">Combination with the start frame</td><td rowspan="2">1-A</td><td>Receive data quantity < receive end data quantity</td><td rowspan="7">Deleted</td><td colspan="2" rowspan="6">Data reception start</td><td colspan="2">—</td><td colspan="2">1-A(1)</td><td rowspan="2">Data of the end code is treated as part of arbitrary data.</td></tr>
<tr><td>Receive data quantity ≥ Receive end data quantity</td><td colspan="4">1-A(2)</td></tr>
<tr><td>1-B</td><td>Receive data quantity = 0</td><td colspan="2">—</td><td colspan="2">1-B</td><td>—</td></tr>
<tr><td rowspan="2">1-C</td><td>Receive data quantity ≤ Receive end data quantity</td><td colspan="2">1-C(1)</td><td colspan="2" rowspan="4">—</td><td rowspan="3">Data of the end code is treated as part of arbitrary data.</td></tr>
<tr><td>Receive data quantity > Receive end data quantity</td><td colspan="2">1-C(2)</td></tr>
<tr><td>1-D</td><td>Receive data quantity
Method 1 dedicated receive end data quantity > 0</td><td colspan="2">1-D</td></tr>
<tr><td>1-E</td><td>Receive data quantity
Method 1 dedicated receive end data quantity = 0</td><td>Data reception start</td><td>1-E</td><td colspan="2">—</td><td>—</td></tr>
<tr><td rowspan="3">Combination without the start frame</td><td rowspan="2">2-A[*2]</td><td>Receive data quantity < Receive end data quantity</td><td rowspan="3">—</td><td colspan="2" rowspan="3">—</td><td colspan="2">Data reception start</td><td colspan="2">2-A(1)</td><td>2-A(3)</td></tr>
<tr><td>Receive data quantity ≥ Receive end data quantity</td><td>Data reception start</td><td colspan="3">2-A(2)</td><td>2-A(4)</td></tr>
<tr><td>2-B[*2]</td><td>Receive data quantity = 0</td><td colspan="2">—</td><td>Data reception start</td><td>2-B</td><td>—</td></tr>
</table>

[*1] The reception processing by the receive end code is not performed when the same data as the receive end code has been registered in the final frame.

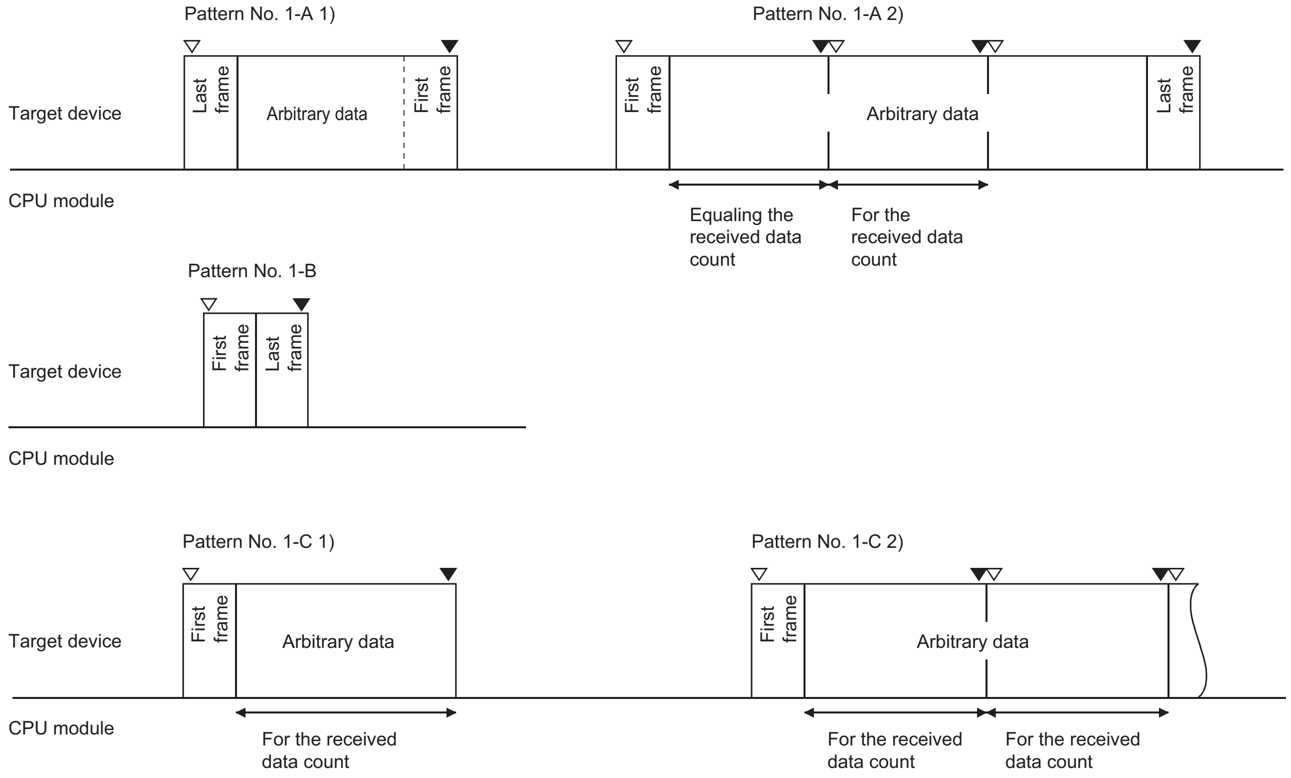
[*2] The data reception interval is not monitored with the no-reception monitoring time (timer 0) when only the final frame is set.

Point

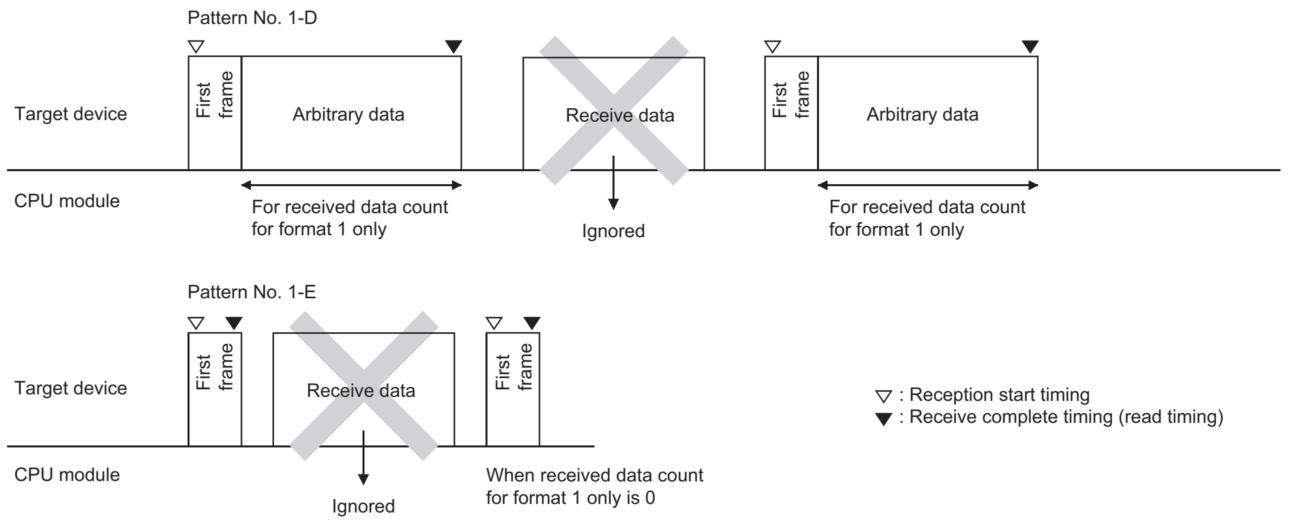
When a reception error is detected, C24 stores arbitrary data portion of the data received immediately before the error in the receive area on the buffer memory, then turns the 'CH1 reception abnormal detection' (X4) ON.

■Combination with the start frame (Timing patterns for data reception start and data reception complete (read))

(For data reception with format-0)



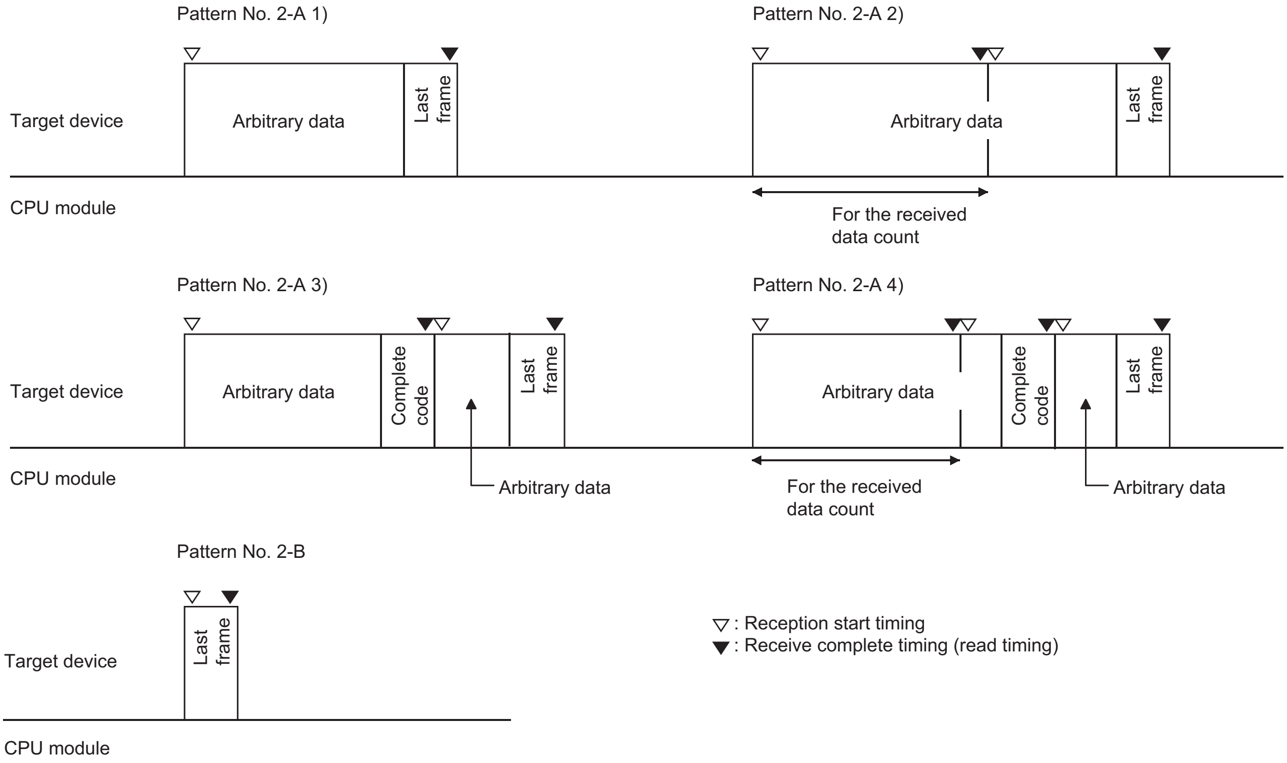
(For data reception with format-1)


Pattern No. 1-D
Target device
CPU module
First frame
Arbitrary data
Receive data
First frame
Arbitrary data
For received data count for format 1 only
Ignored
For received data count for format 1 only
Pattern No. 1-E
Target device
CPU module
First frame
Receive data
First frame
Ignored
When received data count for format 1 only is 0
▽ : Reception start timing
▼ : Receive complete timing (read timing)

Point

When data is received using Method 1, C24 checks again whether the start frame has been received after receiving data equaling the 'method 1 dedicated receive end data quantity'. Reception data during that time is ignored.

■Combination without the start frame (Timing patterns for data reception start and data reception complete (read))



Reception procedure

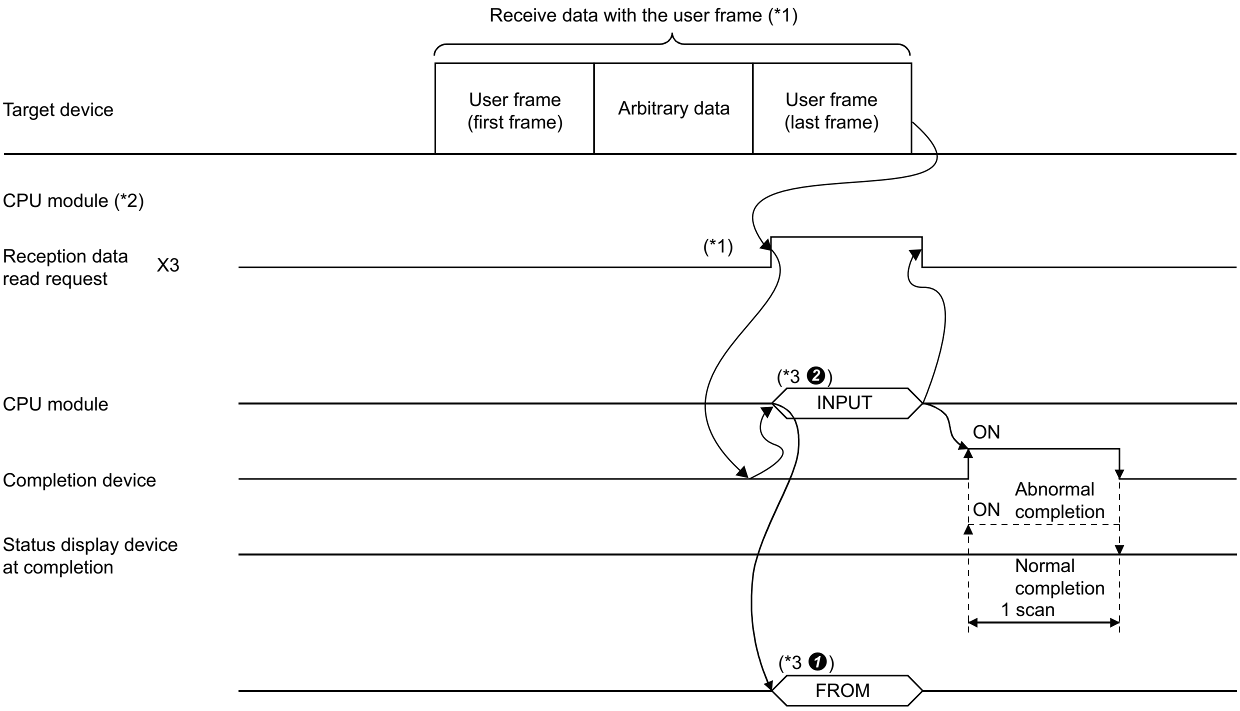
The following shows the reception procedure when a message, including data with the same arrangement as the specified user frame, is received and the arbitrary data is read to the CPU module.

Procedure

(When the CH1 side interface of C24 receives data)

➊ Read the user frame setting number received from the target device (□-th) out of the start frame number and final frame number (up to 4 combinations) initially set in the user frame specification for receiving area on the buffer memory from the 'Receive user frame (nth)' (Un\G603).

➋ Read arbitrary data portion in the reception message from the receive area (Un\G1536 to 2047 (default))



*1 For received data using a user frame and the timing of reading the received data to the CPU module, refer to the timing for start/ completion of data reception (☞ Page 310 Receive data, ☞ Page 316 Timing for start/completion of data reception)

*2 The receive user frame registration number and others should have been set with Engineering tool.
At the time of debugging, the initial setting value for data reception shall be set prior to data reception, but after debugging is completed, it shall be set to the startup of C24. (☞ Page 321 Setting of receive user frames)

*3 Reading (➊) for the CPU module to check which user frame was received by C24 and reading of the received data (➋).

Setting of receive user frames

Setting of the receive user frame

This setting is to receive data from a target data using the nonprocedural protocol with a user frame.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ Module Name ⇨ [Module Parameter] ⇨ "Application Settings" ⇨ "User frame specification"

Setting item		**Description of setting value**	
		Method 0	**Method 1**
Receive end data quantity designation		Designates the data quantity that exceeds the size of the arbitrary data to be received or the data count of the received data size.	The specified value is invalid. (The 'method 1 dedicated receive end data quantity' becomes valid.)
Receive end code designation		• Reception of combination with the start frame: The specified value is invalid • Reception of combination without the start frame: Designate the code for the last data in the reception message for performing a read request to the CPU module.	
User frame use enable/disable designation		Designates "Use".	
Start frame No. designation	1st to 4th	Designate the user frame No. ('0' or '1' or higher).	Designate the user frame No. ('1' or higher).
Final frame No. designation	1st to 4th	Designate the user frame No. ('0' or '1' or higher).	Designate '0H' for all.
User frame receiving method designation	1st to 4th	Designate "Method 0".	Designate "Method 1".
Method 1 dedicated receive end data quantity	1st to 4th	The specified value is invalid.	Designate the data quantity for the arbitrary data to be received.

Examples of registering a receive user frame

The following are examples of pre-registering a receive user frame on the CH1 side with Engineering tool.

■When the start frame is specified

In the example, the following three combinations are registered for the receive user frame.

- Setting conditions

User frame		User frame receiving method designation	Method 1 dedicated receive end data quantity	Remarks
1st combination	(Start frame + Final frame)	Method 0	—	The receive end data quantity becomes valid.
2nd combination	(Start frame only)	Method 1	0H	Method 1 dedicated receive end data quantity becomes valid.
3rd combination	(Start frame only)	Method 1	2H	

- Setting value

Setting item		Setting value	Remarks
Receive end data quantity designation		1FFH	For Method 0
Receive end code designation		FFFFH	Set to "No designation" for receive end code.
User frame specification for receiving User frame use enable/disable designation		Use	Always specify "Use".
User frame specification for receiving Start frame No. designation	1st	3E8H	0H: No designation 1H or higher: Designated In this setting, only three combinations become valid.
	2nd	3E9H	
	3rd	3EAH	
	4th	0H	
User frame specification for receiving Final frame No. designation	1st	41DH	
	2nd	0H	
	3rd	0H	
	4th	0H	
User frame receiving method designation	1st	Method 0	The receive end data quantity becomes valid.
	2nd	Method 1	Method 1 dedicated receive end data quantity becomes valid.
	3rd	Method 1	
	4th	Method 0	—
Method 1 dedicated receive end data quantity designation	1st	0H	For Method 1
	2nd	0H	
	3rd	2H	
	4th	0H	

■When the start frame is not specified

In the example, the following three combinations are registered for the receive user frame.

- Setting conditions

User frame		User frame receiving method designation	Method 1 dedicated receive end data quantity	Remarks
1st combination	(Final frame only)	Method 0	—	The receive end data quantity becomes valid.
2nd combination	(Final frame only)			
3rd combination	(Final frame only)			

- Setting value

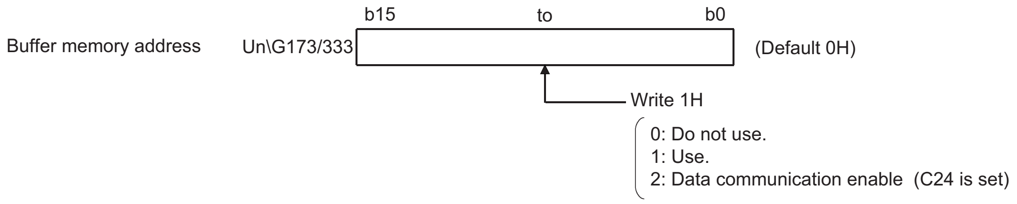
Setting item		Setting value	Remarks
Receive end data quantity designation		1FFH	For Method 0
Receive end code designation		□□H	Designate any receive end code.
User frame specification for receiving User frame use enable/disable designation		Use	Always specify "Use".
User frame specification for receiving Start frame No. designation	1st	0H	0H: No designation
	2nd	0H	
	3rd	0H	
	4th	0H	
User frame specification for receiving Final frame No. designation	1st	41DH	0H: No designation 1H or higher: Designated In this setting, only three combinations become valid.
	2nd	41EH	
	3rd	41FH	
	4th	0H	
User frame receiving method designation	1st	Method 0	The receive end data quantity becomes valid.
	2nd	Method 0	'Method 0' is applied to all since the start frame is not designated.
	3rd	Method 0	
	4th	Method 0	
Method 1 dedicated receive end data quantity designation	1st	0H	Setting value for Method 1. This setting is not required since 'Method 0' is applied to all.
	2nd	0H	
	3rd	0H	
	4th	0H	

Receive user frame setting details

The following explains the purpose and designated value/stored value of each setting data for data reception using a user frame.

■User frame use enable/disable designation (Un\G173/333)

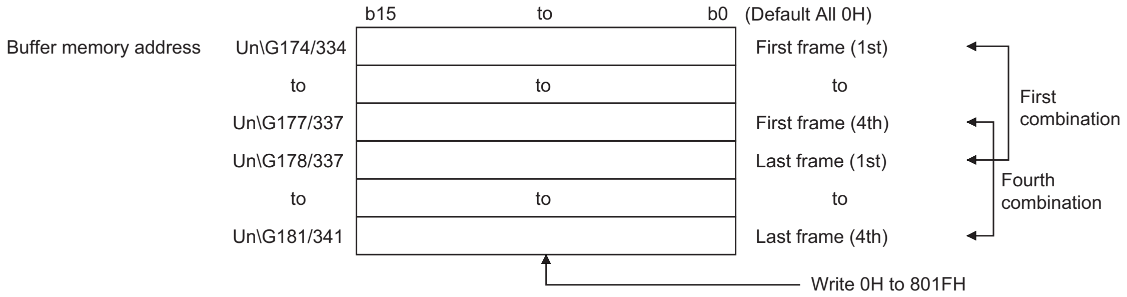
Designate "Use" when using user frames to receive data.



- Write '1' in the user frame use enable/disable designation area.
- After the preparation for receiving data using the user frames is completed, '2' is written in the user frame use enable/disable designation area. (Set by C24)
- After the value in the user frame use enable/disable designation area is changed from '1' to '2', start receiving data used by the user frame.
  Until '2' is written in the user frame use enable/disable designation area, data transmission cannot be started.

■Start frame No. designation area and final frame No. designation area (Un\G174 to 181/334 to 341)

From among the user frames registered in C24, designate the user frame No. to be used by following the designation method as shown below.



- 0H (0): No designation
- 1H to 3E7H (1 to 999): Designate the default registration frame (for OS ROM registration).
- 3E8H to 4AFH (1000 to 1199): Designate the user frame (for flash ROM registration).
- 8001H to 801FH (-32767 to -32737): Designate the user frame (for buffer memory registration).

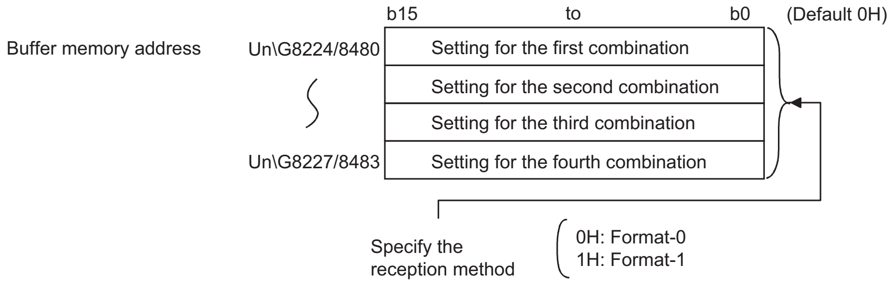
Precautions

- For the receive user frames, the start frame and final frame are set as a pair, regardless of whether the target device transmits these frames.
  (If the target device does not transmit either one of the frames, set the frame No. that is not transmitted to "0" (No designation).)
- A maximum of four combinations of start and final frames can be set for the nonprocedural protocol.
  However, if there is a combination that specifies the start frame when multiple combinations are set, specify the start frame for all other combinations.
  In addition, set in the following order starting from the first buffer memory.
  When specifying the start frame
  Each frame number for combinations that specify the start frame and the final frame
  Each frame number for combinations that specify the start frame but not the final frame
  When setting multiple combinations
  For the combination that specifies only the final frame without specifying the start frame, the start frame for all of the combinations cannot be specified. Set the frame number for the final frame to be used from the first buffer memory in order.
- When setting more than one combination, the start frame having the registered code in the same arrangement or the same frame No. cannot be designated.
- Use the user frame No. of the following user frames to set the receive user frames.
  Default registration frame No.: 1H to 3E7H
  User frame No. registered in C24 flash ROM: 3E8H to 4AFH
  User frame No. registered in C24 buffer memory: 8001H to 801FH
- Do not set the frame number of a user frame contains the additional code data by the transparent code for receiving specification as a user frame for data reception using the nonprocedural protocol.

■User frame receiving method designation (Un\G8224 to 8227/8480 to 8483)

In data reception using a user frame, designate the reception method for each combination of receive user frames.

This setting is valid when the setting is configured in a combination that specifies a start frame. (☞ Page 325 Start frame No. designation area and final frame No. designation area (Un\G174 to 181/334 to 341))

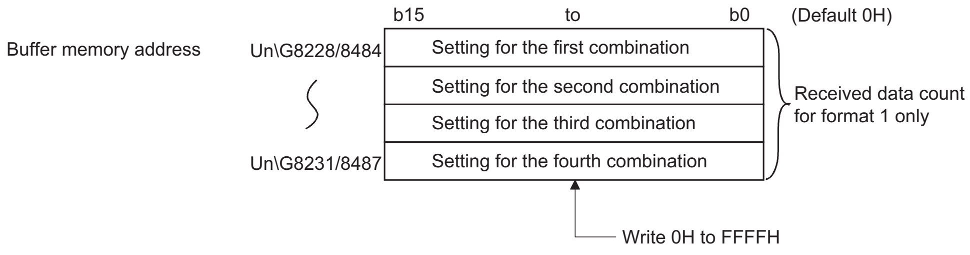

b15
to
b0
(Default 0H)
Buffer memory address
Un\G8224/8480
Un\G8227/8483
Setting for the first combination
Setting for the second combination
Setting for the third combination
Setting for the fourth combination
Specify the reception method
0H: Format-0
1H: Format-1

Point

For combinations other than those with the start frame only, data reception is performed using 'Method 0' even if 'Method 1' is designated through the above reception method setting.

■Method 1 dedicated receive end data quantity designation (Un\G8228 to 8231/8484 to 8487)

- For a combination specifying Method 1 in the user frame receiving method designation, designate the arbitrary data word/byte count (size for performing a reception data read request to the CPU module) when the applicable start frame is received.
  Specify this for each combination specifying Method 1.
- Designate this within the size of the receive data storage area on the buffer memory.
- The unit of setting values is in accordance with the setting value set for word/byte units designation.


b15
to
b0
(Default 0H)
Buffer memory address
Un\G8228/8484
Un\G8231/8487
Setting for the first combination
Setting for the second combination
Setting for the third combination
Setting for the fourth combination
Received data count for format 1 only
Write 0H to FFFFH

17.3 Program for Data Reception with User Frame

This section shows examples of the program to read the received data stored in C24 buffer memory to the CPU module, when data including the user frame is received.

Reception program example

Data reception using a user frame in the following conditions and settings is shown below.

■Input/output signal of C24

Install C24 in the position where the I/O signal is X/Y00 to X/Y1F.

■C24 interface used for communication with target device

CH1 side RS-232 interface of C24

■Parameter settings

Setting item				Setting value
Basic settings	Various control specification	Test mode setting		—
		Communication protocol setting		Nonprocedural protocol
		Communication speed setting		19200 bps
		Transmission setting	Operation setting	Independent
			Data bit	8
			Parity bit	None
			Odd/even parity	Odd
			Stop bit	1
			Sum check code	None
			Online change	Disable
			Setting change	Disable

■Setting content of buffer memory

For other than the following items, use default values.

Address DEC (HEX)	Name	Setting value
CH1		
173 (ADH)	User frame use enable/disable designation	1H: Use
174 (AEH)	Start frame No. designation, 1st[*1]	3E8H[*2]
		8001H[*3]
175 (AFH)	Start frame No. designation, 2nd[*1]	3E9H[*2]
		8002H[*3]
176 (B0H)	Start frame No. designation, 3rd[*1]	6H
177 (B1H)	Start frame No. designation, 4th[*1]	15H
178 (B2H)	Final frame No. designation, 1st[*1]	41BH[*2]
		8005H[*3]
179 (B3H)	Final frame No. designation, 2nd	0H: None
180 (B4H)	Final frame No. designation, 3rd	0H: None
181 (B5H)	Final frame No. designation, 4th	0H: None
8224 (2020H)	User frame receiving method designation, 1st	0H: Method 0
8225 (2021H)	User frame receiving method designation, 2nd	1H: Method 1
8226 (2022H)	User frame receiving method designation, 3rd	1H: Method 1
8227 (2023H)	User frame receiving method designation, 4th	1H: Method 1
8229 (2025H)	Method 1 dedicated receive end data quantity designation, 2nd	12 words
8230 (2026H)	Method 1 dedicated receive end data quantity designation, 3rd	0 word
8231 (2027H)	Method 1 dedicated receive end data quantity designation, 4th	2 words

[*1] It indicates the registered contents of the user frame No. specified to the start and final frame No., respectively.

[*2] When the setting is configured with Engineering tool. (☞ Page 328 Program example when setting with Engineering tool)

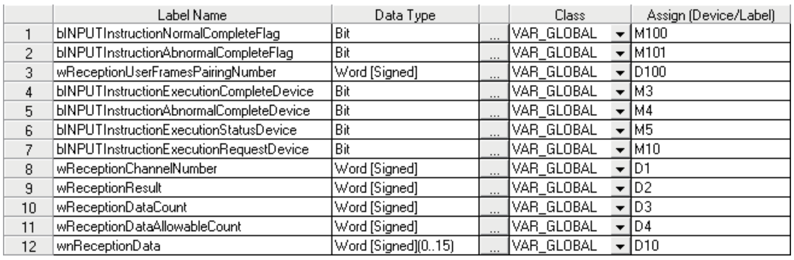
[*3] When the setting is configured by a program. (☞ Page 331 Program example when setting by a program)

■Registration contents in user frame

User frame No.	Registered code	Registered data content
3E8H/8001H	02H, 51H, 20H, 0AH, 3BH	STX, Q, (SP), Station No. of target device, ;
3E9H/8002H	02H, 41H, 3BH	STX, A, ;
6H	06H	ACK
15H	15H	NAK
41BH/8005H	03H, FFH, F0H	ETX, Sum check code

Program example when setting with Engineering tool

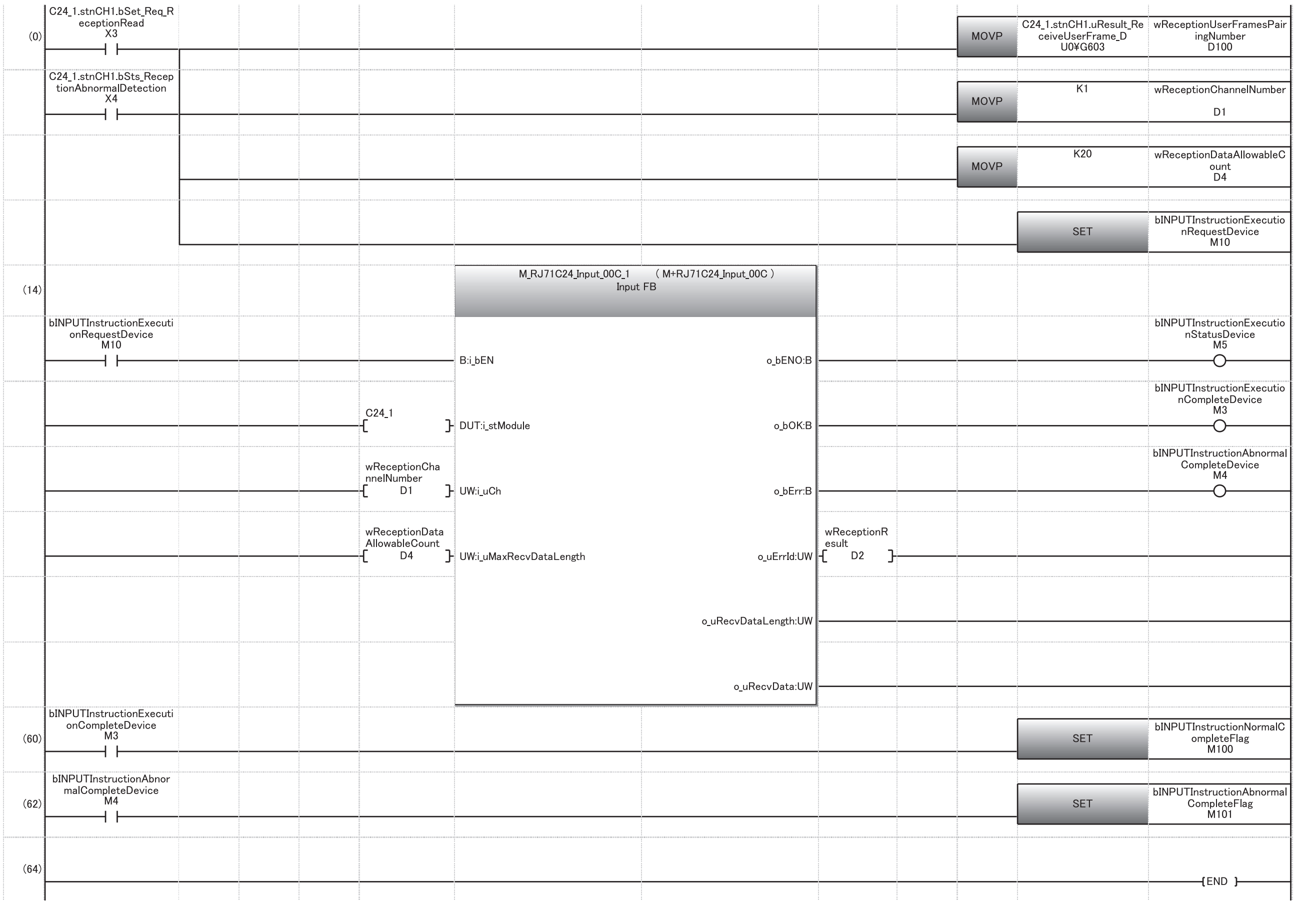
■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection	CH1 Reception abnormal detection	X4
	C24_1.stnCH1.uResult_ReceiveUserFrame_D	CH1_Receive user frame (nth)	U0\G603
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_Input	Reads the received data.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bINPUTInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
2	bINPUTInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101
3	wReceptionUserFramesPairingNumber	Word [Signed]	...	VAR_GLOBAL	D100
4	bINPUTInstructionExecutionCompleteDevice	Bit	...	VAR_GLOBAL	M3
5	bINPUTInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M4
6	bINPUTInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M5
7	bINPUTInstructionExecutionRequestDevice	Bit	...	VAR_GLOBAL	M10
8	wReceptionChannelNumber	Word [Signed]	...	VAR_GLOBAL	D1
9	wReceptionResult	Word [Signed]	...	VAR_GLOBAL	D2
10	wReceptionDataCount	Word [Signed]	...	VAR_GLOBAL	D3
11	wReceptionDataAllowableCount	Word [Signed]	...	VAR_GLOBAL	D4
12	wnReceptionData	Word [Signed](0..15)	...	VAR_GLOBAL	D10

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference

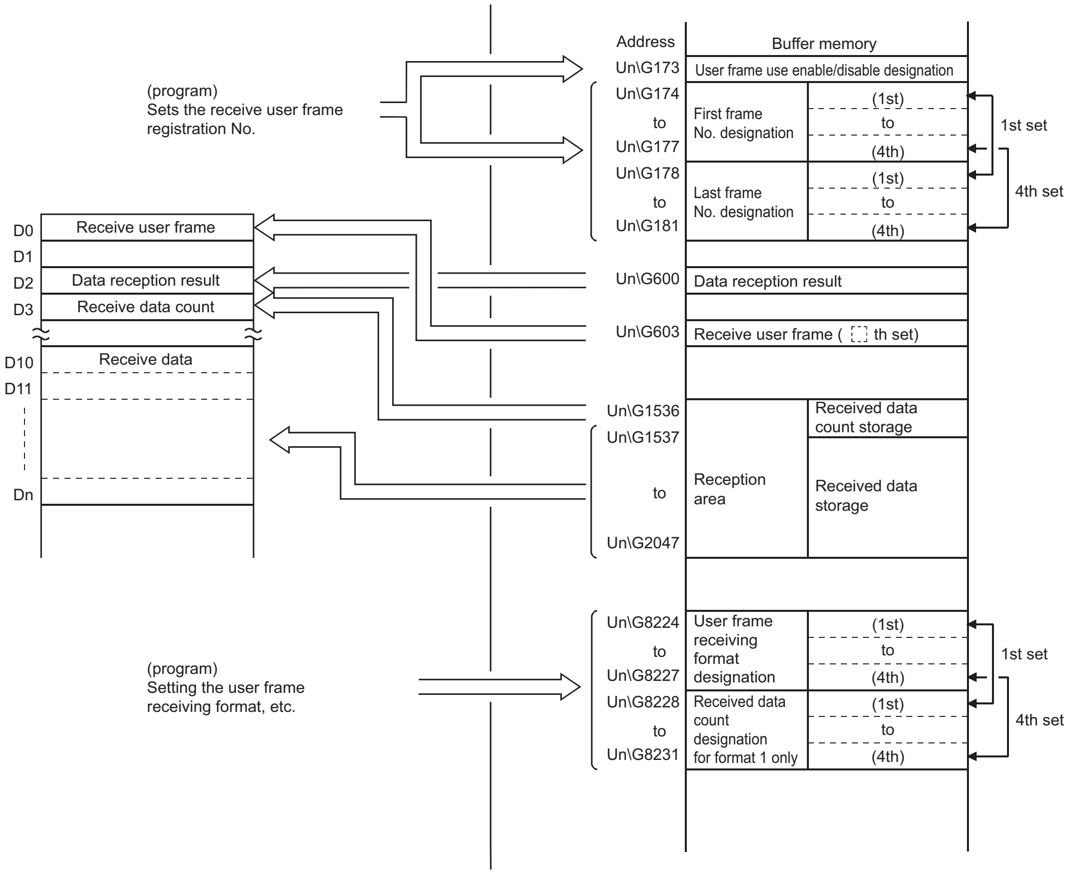

(0)
C24_1.stnCH1.bSet_Req_ReceptionRead
X3
MOVP
C24_1.stnCH1.uResult_ReceiveUserFrame_D
U0¥G603
wReceptionUserFramesPairingNumber
D100
C24_1.stnCH1.bSts_ReceptionAbnormalDetection
X4
MOVP
K1
wReceptionChannelNumber
D1
MOVP
K20
wReceptionDataAllowableCount
D4
SET
bINPUTInstructionExecutionRequestDevice
M10
(14)
M_RJ71C24_Input_00C_1 （M+RJ71C24_Input_00C）
Input FB
bINPUTInstructionExecutionRequestDevice
M10
B:i_bEN
o_bENO:B
bINPUTInstructionExecutionStatusDevice
M5
C24_1
DUT:i_stModule
o_bOK:B
bINPUTInstructionExecutionCompleteDevice
M3
wReceptionChannelNumber
D1
UW:i_uCh
o_bErr:B
bINPUTInstructionAbnormalCompleteDevice
M4
wReceptionDataAllowableCount
D4
UW:i_uMaxRecvDataLength
o_uErrId:UW
wReceptionResult
D2
o_uRecvDataLength:UW
o_uRecvData:UW
(60)
bINPUTInstructionExecutionCompleteDevice
M3
SET
bINPUTInstructionNormalCompleteFlag
M100
(62)
bINPUTInstructionAbnormalCompleteDevice
M4
SET
bINPUTInstructionAbnormalCompleteFlag
M101
(64)
END

(0) Read the combinations of user frames received.
Set the reception channel to CH1.
Set the allowable number of data to be received to 20.
Specify the execution instruction.

(14) Execute reception

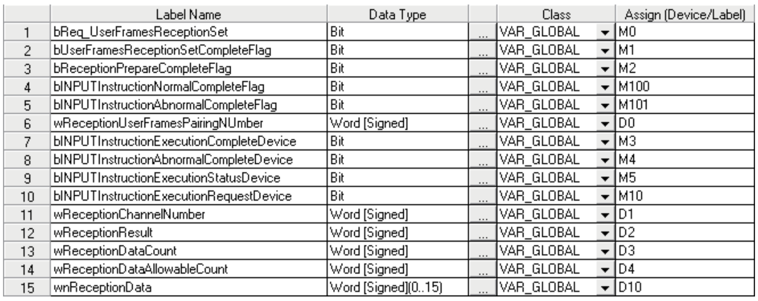
(60) Set normal completion flag.

(62) Set abnormal completion flag.


Address
Buffer memory
(program)
Sets the receive user frame registration No.
Un\G173
User frame use enable/disable designation
Un\G174
to
Un\G177
First frame No. designation
(1st)
to
(4th)
1st set
Un\G178
to
Un\G181
Last frame No. designation
(1st)
to
(4th)
4th set
D0
Receive user frame
D1
D2
Data reception result
D3
Receive data count
D10
Receive data
D11
Dn
Un\G600
Data reception result
Un\G603
Receive user frame (th set)
Un\G1536
Received data count storage
Un\G1537
to
Un\G2047
Reception area
Received data storage
(program)
Setting the user frame receiving format, etc.
Un\G8224
to
Un\G8227
User frame receiving format designation
(1st)
to
(4th)
1st set
Un\G8228
to
Un\G8231
Received data count designation for format 1 only
(1st)
to
(4th)
4th set

Program example when setting by a program

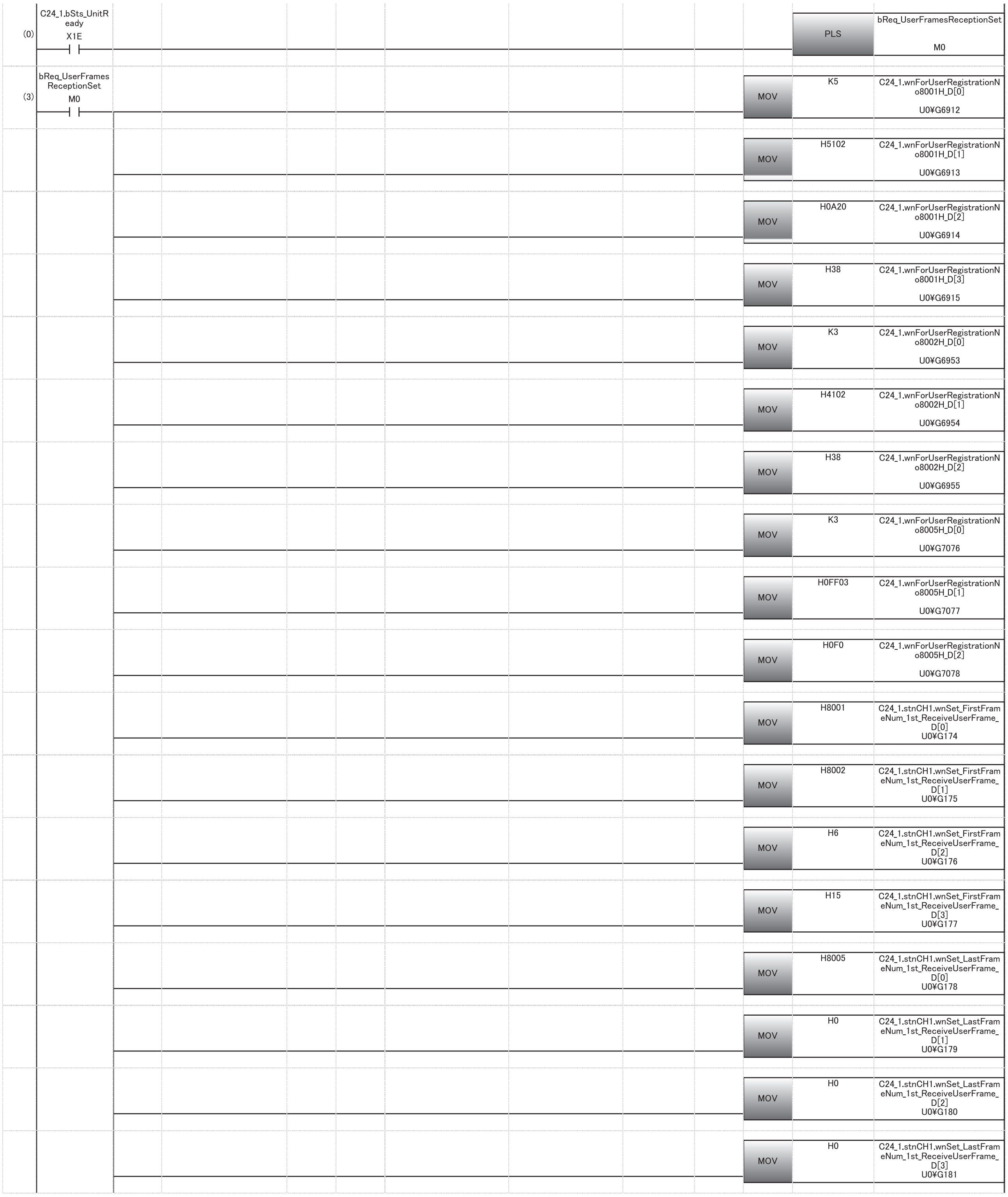
■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.bSts_UnitReady	Module READY	X1E
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection	CH1 Reception abnormal detection	X4
	C24_1.stnCH1.uSet_UserFrameUse_Enable_disable_ReceiveUserFrame_D	CH1_Receive user frame use enable/disable designation	U0\G173
	C24_1.stnCH1.wnSet_FirstFrameNum_1st_ReceiveUserFrame_D[0]	CH1_Receive user frame_Start frame No. designation [0]	U0\G174
	C24_1.stnCH1.wnSet_FirstFrameNum_1st_ReceiveUserFrame_D[1]	CH1_Receive user frame_Start frame No. designation [1]	U0\G175
	C24_1.stnCH1.wnSet_FirstFrameNum_1st_ReceiveUserFrame_D[2]	CH1_Receive user frame_Start frame No. designation [2]	U0\G176
	C24_1.stnCH1.wnSet_FirstFrameNum_1st_ReceiveUserFrame_D[3]	CH1_Receive user frame_Start frame No. designation [3]	U0\G177
	C24_1.stnCH1.wnSet_LastFrameNum_1st_ReceiveUserFrame_D[0]	CH1_Receive user frame_Final frame No. designation [0]	U0\G178
	C24_1.stnCH1.wnSet_LastFrameNum_1st_ReceiveUserFrame_D[1]	CH1_Receive user frame_Final frame No. designation [1]	U0\G179
	C24_1.stnCH1.wnSet_LastFrameNum_1st_ReceiveUserFrame_D[2]	CH1_Receive user frame_Final frame No. designation [2]	U0\G180
	C24_1.stnCH1.wnSet_LastFrameNum_1st_ReceiveUserFrame_D[3]	CH1_Receive user frame_Final frame No. designation [3]	U0\G181
	C24_1.stnCH1.uResult_ReceiveUserFrame_D	CH1_Receive user frame (nth)	U0\G603
	C24_1.wnForUserRegistrationNo8001H_D[0]	For user registration number 8001H [0]	U0\G6912
	C24_1.wnForUserRegistrationNo8001H_D[1]	For user registration number 8001H [1]	U0\G6913
	C24_1.wnForUserRegistrationNo8001H_D[2]	For user registration number 8001H [2]	U0\G6914
	C24_1.wnForUserRegistrationNo8001H_D[3]	For user registration number 8001H [3]	U0\G6915
	C24_1.wnForUserRegistrationNo8002H_D[0]	For user registration number 8002H [0]	U0\G6953
	C24_1.wnForUserRegistrationNo8002H_D[1]	For user registration number 8002H [1]	U0\G6954
	C24_1.wnForUserRegistrationNo8002H_D[2]	For user registration number 8002H [2]	U0\G6955
	C24_1.wnForUserRegistrationNo8005H_D[0]	For user registration number 8005H [0]	U0\G7076
	C24_1.wnForUserRegistrationNo8005H_D[1]	For user registration number 8005H [1]	U0\G7077
	C24_1.wnForUserRegistrationNo8005H_D[2]	For user registration number 8005H [2]	U0\G7078
	C24_1.stnCH1.wnSet_UserFrameReceiveFormat_D[1]	CH1_User frame receiving method designation [1]	U0\G8225
	C24_1.stnCH1.wnSet_UserFrameReceiveFormat_D[2]	CH1_User frame receiving method designation [2]	U0\G8226
	C24_1.stnCH1.wnSet_UserFrameReceiveFormat_D[3]	CH1_User frame receiving method designation [3]	U0\G8227
	C24_1.stnCH1.wnSet_UserFrameExclusiveFormat1ReceivedDataCount_D[1]	CH1_User frame method 1 dedicated receive end data quantity [1]	U0\G8229
	C24_1.stnCH1.wnSet_UserFrameExclusiveFormat1ReceivedDataCount_D[2]	CH1_User frame method 1 dedicated receive end data quantity [2]	U0\G8230
	C24_1.stnCH1.wnSet_UserFrameExclusiveFormat1ReceivedDataCount_D[3]	CH1_User frame method 1 dedicated receive end data quantity [3]	U0\G8231
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_Input	Reads the received data.	—

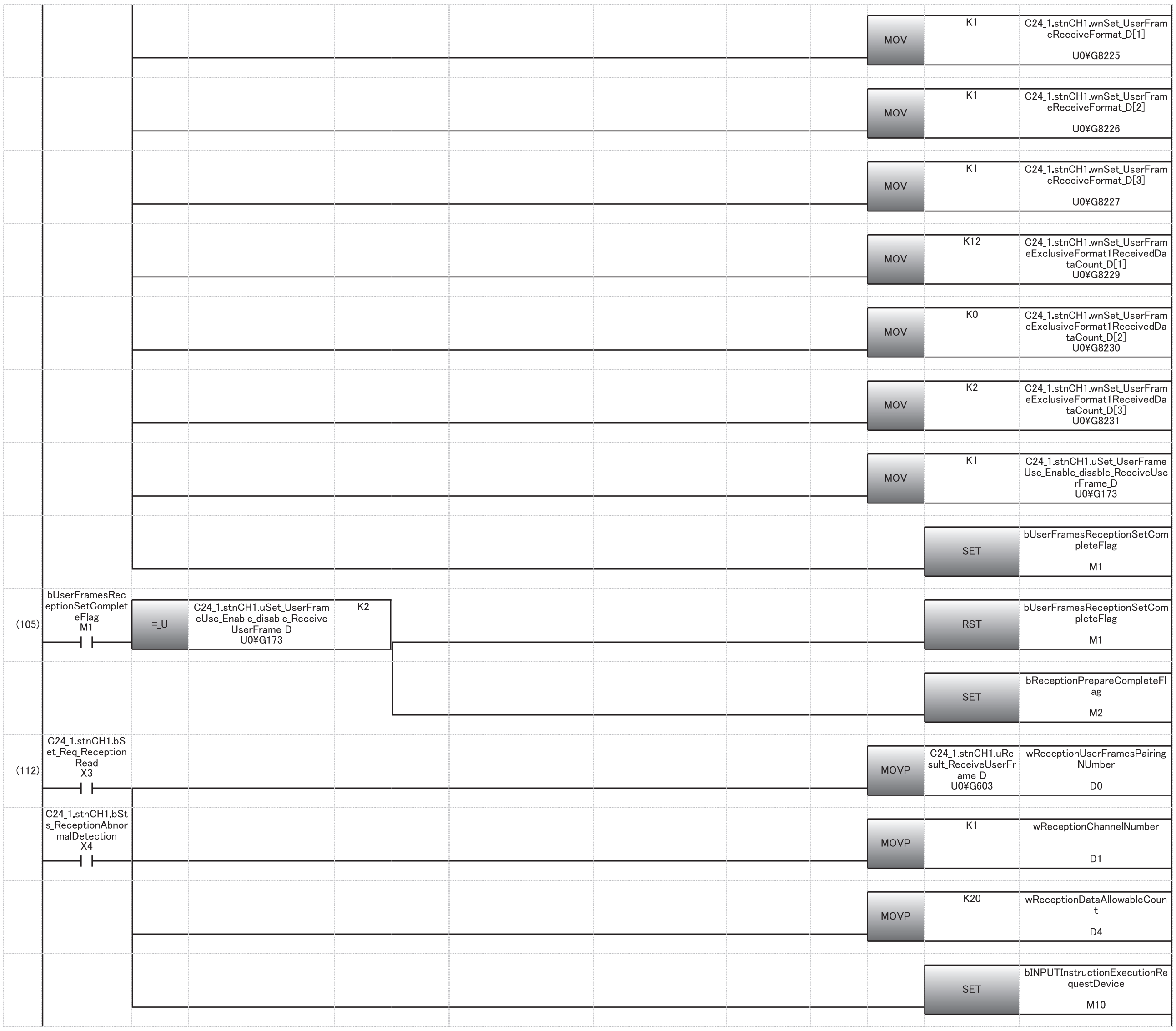
	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_UserFramesReceptionSet	Bit	...	VAR_GLOBAL	M0
2	bUserFramesReceptionSetCompleteFlag	Bit	...	VAR_GLOBAL	M1
3	bReceptionPrepareCompleteFlag	Bit	...	VAR_GLOBAL	M2
4	bINPUTInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
5	bINPUTInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101
6	wReceptionUserFramesPairingNUmber	Word [Signed]	...	VAR_GLOBAL	D0
7	bINPUTInstructionExecutionCompleteDevice	Bit	...	VAR_GLOBAL	M3
8	bINPUTInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M4
9	bINPUTInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M5
10	bINPUTInstructionExecutionRequestDevice	Bit	...	VAR_GLOBAL	M10
11	wReceptionChannelNumber	Word [Signed]	...	VAR_GLOBAL	D1
12	wReceptionResult	Word [Signed]	...	VAR_GLOBAL	D2
13	wReceptionDataCount	Word [Signed]	...	VAR_GLOBAL	D3
14	wReceptionDataAllowableCount	Word [Signed]	...	VAR_GLOBAL	D4
15	wnReceptionData	Word [Signed](0..15)	...	VAR_GLOBAL	D10

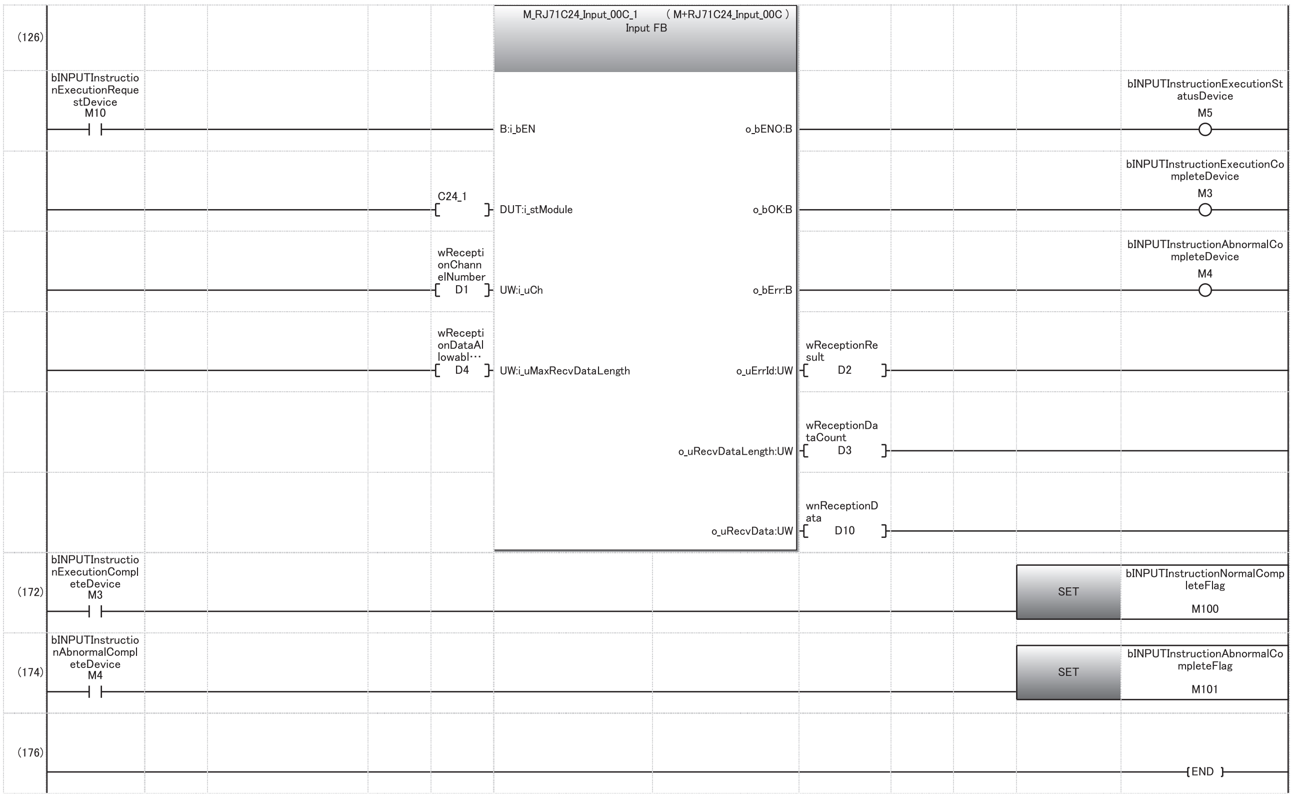
For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference



(0)
C24_1.bSts_UnitReady
X1E
PLS
bReq_UserFramesReceptionSet
M0
(3)
bReq_UserFramesReceptionSet
M0
MOV
K5
C24_1.wnForUserRegistrationNo8001H_D[0]
U0¥G6912
MOV
H5102
C24_1.wnForUserRegistrationNo8001H_D[1]
U0¥G6913
MOV
H0A20
C24_1.wnForUserRegistrationNo8001H_D[2]
U0¥G6914
MOV
H38
C24_1.wnForUserRegistrationNo8001H_D[3]
U0¥G6915
MOV
K3
C24_1.wnForUserRegistrationNo8002H_D[0]
U0¥G6953
MOV
H4102
C24_1.wnForUserRegistrationNo8002H_D[1]
U0¥G6954
MOV
H38
C24_1.wnForUserRegistrationNo8002H_D[2]
U0¥G6955
MOV
K3
C24_1.wnForUserRegistrationNo8005H_D[0]
U0¥G7076
MOV
H0FF03
C24_1.wnForUserRegistrationNo8005H_D[1]
U0¥G7077
MOV
H0F0
C24_1.wnForUserRegistrationNo8005H_D[2]
U0¥G7078
MOV
H8001
C24_1.stnCH1.wnSet_FirstFrameNum_1st_ReceiveUserFrame_D[0]
U0¥G174
MOV
H8002
C24_1.stnCH1.wnSet_FirstFrameNum_1st_ReceiveUserFrame_D[1]
U0¥G175
MOV
H6
C24_1.stnCH1.wnSet_FirstFrameNum_1st_ReceiveUserFrame_D[2]
U0¥G176
MOV
H15
C24_1.stnCH1.wnSet_FirstFrameNum_1st_ReceiveUserFrame_D[3]
U0¥G177
MOV
H8005
C24_1.stnCH1.wnSet_LastFrameNum_1st_ReceiveUserFrame_D[0]
U0¥G178
MOV
H0
C24_1.stnCH1.wnSet_LastFrameNum_1st_ReceiveUserFrame_D[1]
U0¥G179
MOV
H0
C24_1.stnCH1.wnSet_LastFrameNum_1st_ReceiveUserFrame_D[2]
U0¥G180
MOV
H0
C24_1.stnCH1.wnSet_LastFrameNum_1st_ReceiveUserFrame_D[3]
U0¥G181


MOV
K1
C24_1.stnCH1.wnSet_UserFrameReceiveFormat_D[1]
U0¥G8225
MOV
K1
C24_1.stnCH1.wnSet_UserFrameReceiveFormat_D[2]
U0¥G8226
MOV
K1
C24_1.stnCH1.wnSet_UserFrameReceiveFormat_D[3]
U0¥G8227
MOV
K12
C24_1.stnCH1.wnSet_UserFrameExclusiveFormat1ReceivedDataCount_D[1]
U0¥G8229
MOV
K0
C24_1.stnCH1.wnSet_UserFrameExclusiveFormat1ReceivedDataCount_D[2]
U0¥G8230
MOV
K2
C24_1.stnCH1.wnSet_UserFrameExclusiveFormat1ReceivedDataCount_D[3]
U0¥G8231
MOV
K1
C24_1.stnCH1.uSet_UserFrameUse_Enable_disable_ReceiveUserFrame_D
U0¥G173
SET
bUserFramesReceptionSetCompleteFlag
M1
(105)
bUserFramesReceptionSetCompleteFlag
M1
=_U
C24_1.stnCH1.uSet_UserFrameUse_Enable_disable_ReceiveUserFrame_D
U0¥G173
K2
RST
bUserFramesReceptionSetCompleteFlag
M1
SET
bReceptionPrepareCompleteFlag
M2
(112)
C24_1.stnCH1.bSet_Req_Reception Read
X3
MOVP
C24_1.stnCH1.uResult_ReceiveUserFrame_D
U0¥G603
wReceptionUserFramesPairingNUmber
D0
C24_1.stnCH1.bSts_ReceptionAbnormalDetection
X4
MOVP
K1
wReceptionChannelNumber
D1
MOVP
K20
wReceptionDataAllowableCount
D4
SET
bINPUTInstructionExecutionRequestDevice
M10


(126)
M_RJ71C24_Input_00C_1 (M+RJ71C24_Input_00C)
Input FB
bINPUTInstructionExecutionRequestDevice
M10
B:i_bEN
o_bENO:B
bINPUTInstructionExecutionStatusDevice
M5
C24_1
DUT:i_stModule
o_bOK:B
bINPUTInstructionExecutionCompleteDevice
M3
wReceptionChannelNumber
D1
UW:i_uCh
o_bErr:B
bINPUTInstructionAbnormalCompleteDevice
M4
wReceptionDataAllowabl…
D4
UW:i_uMaxRecvDataLength
o_uErrId:UW
wReceptionResult
D2
o_uRecvDataLength:UW
wReceptionDataCount
D3
o_uRecvData:UW
wnReceptionData
D10
(172)
bINPUTInstructionExecutionCompleteDevice
M3
SET
bINPUTInstructionNormalCompleteFlag
M100
(174)
bINPUTInstructionAbnormalCompleteDevice
M4
SET
bINPUTInstructionAbnormalCompleteFlag
M101
(176)
END

(3) Frame No. 8001H setting: registration 1 to 4
Frame No. 8002H setting: registration 1 to 4
Frame No. 8005H setting: registration 1 to 4
Start frame No. designation for receiving: 1 to 4
Final frame No. designation for receiving: 1 to 4
Receiving method designation: 2 to 3 - Method 1 designation
Method 1 dedicated receive end data quantity: 2 to 3
Set User frame use enable/disable designation to "Enable".

(105) Reception ready

(112) Read combined number.
Specify the reception channel.
Set the allowable number of data to be received.
Issue the INPUT instruction execution direction.

(172) Normal completion

(174) Abnormal completion

Application program for data reception using a combination that specifies the start frame

The following shows the program example when receiving data using a combination that specifies the start frame.

Reception program example

Data reception using a user frame set with the following conditions is shown below.

■C24 interface used for communication with target device

CH1 side RS-232 interface of C24

■Parameter settings

For other than the following items, use default values.

Item	Setting content	Buffer memory address to store registration value	Remarks
Word/byte units designation	Word/byte	150(96H)	Set to either one according to the program example.
Transparent code for receiving specification	Designated/No designation	288(120H)	When 'Designated', Additional code: 10H (DLE) Transparent code: 02H (STX)
ASCII-BIN conversion designation	Do not convert	289(121H)	—
Receive end data quantity	6	164 (A4H)	—
Receive end code	No designation	165 (A5H)	—
User frame use enable/disable designation	Use	173 (ADH)	Refer to the illustrations provided in this section.
Start frame No. designation Final frame No. designation[*1]	Designated	174 to 177 (AEH to B5H)	
User frame receiving method designation	Method 0/Method 1	8224 to 8227 (2020H to 2023H)	
Method 1 dedicated receive end data quantity designation	0H to FFFFH	8228 to 8231 (2024H to 2027H)	

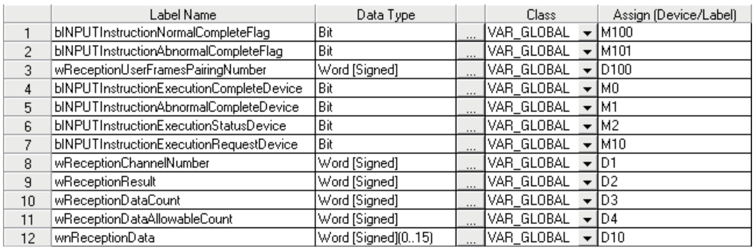
[*1] It indicates the registered contents of the user frame No. specified to the start and final frame No., respectively.

■Registration contents in user frame

Item		User frame No.	Registered code	Registered data content
Start frame No.	1st	3E8H	02H, 51H, 20H, 0AH, 3BH	STX, Q, (SP), Station No. of target device, ;
	2nd	3E9H	02H, 41H, 3BH	STX, A, ;
	3rd	6H	06H	ACK
	4th	15H	15H	NAK
Final frame No.	1st	41BH/8005H	03H, FFH, F0H	ETX, Sum check code
	2nd	0H (none)	—	—
	3rd	0H (none)	—	—
	4th	0H (none)	—	—

■Program example

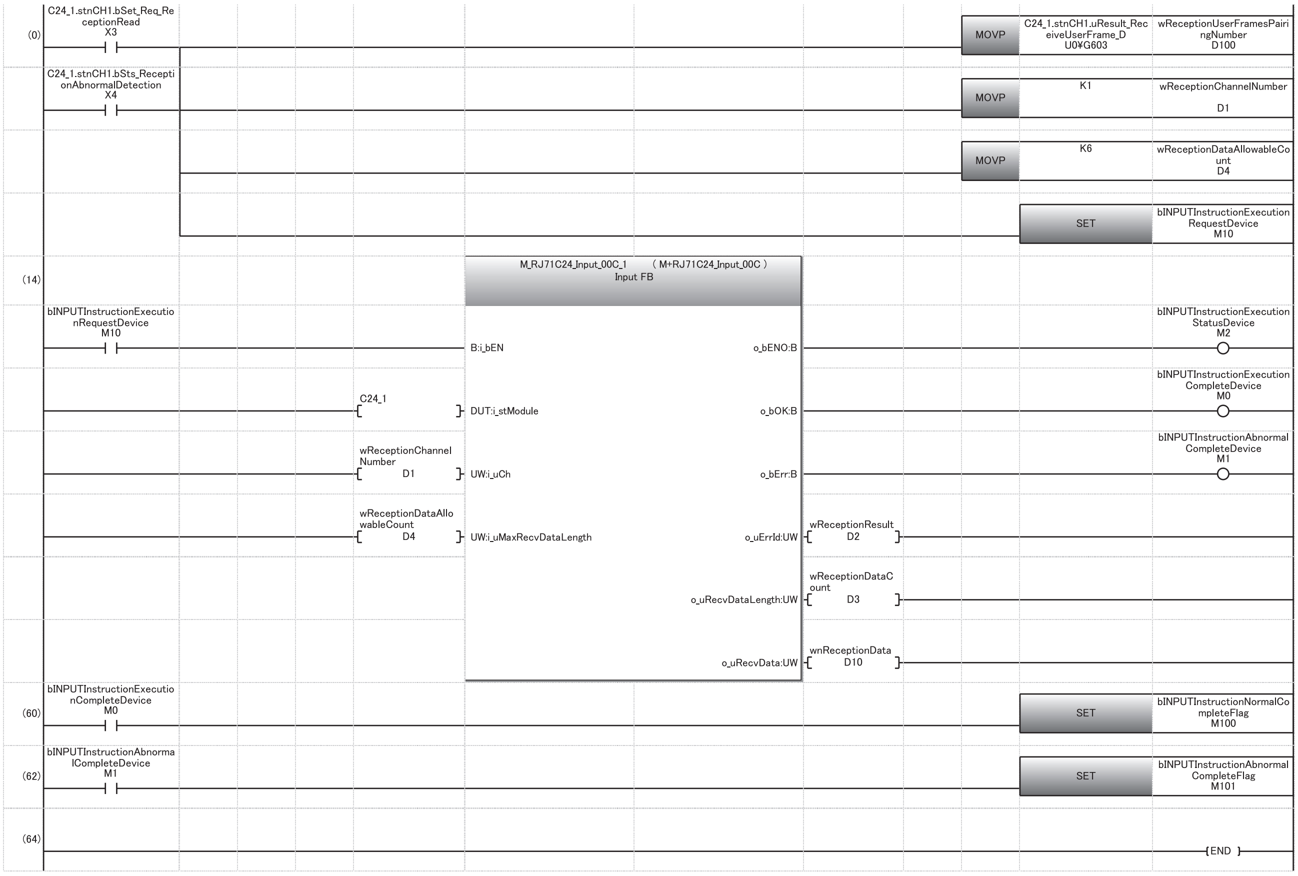
The following shows example of data reception with a user frame.

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection	CH1 Reception abnormal detection	X4
	C24_1.stnCH1.uResult_ReceiveUserFrame_D	CH1_Receive user frame (nth)	U0\G603
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_Input	Reads the received data.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bINPUTInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
2	bINPUTInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101
3	wReceptionUserFramesPairingNumber	Word [Signed]	...	VAR_GLOBAL	D100
4	bINPUTInstructionExecutionCompleteDevice	Bit	...	VAR_GLOBAL	M0
5	bINPUTInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
6	bINPUTInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M2
7	bINPUTInstructionExecutionRequestDevice	Bit	...	VAR_GLOBAL	M10
8	wReceptionChannelNumber	Word [Signed]	...	VAR_GLOBAL	D1
9	wReceptionResult	Word [Signed]	...	VAR_GLOBAL	D2
10	wReceptionDataCount	Word [Signed]	...	VAR_GLOBAL	D3
11	wReceptionDataAllowableCount	Word [Signed]	...	VAR_GLOBAL	D4
12	wnReceptionData	Word [Signed](0..15)	...	VAR_GLOBAL	D10

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
C24_1.stnCH1.bSet_Req_ReceptionRead X3
MOVP
C24_1.stnCH1.uResult_ReceiveUserFrame_D U0¥G603
wReceptionUserFramesPairingNumber D100
C24_1.stnCH1.bSts_ReceptionAbnormalDetection X4
MOVP
K1
wReceptionChannelNumber D1
MOVP
K6
wReceptionDataAllowableCount D4
SET
bINPUTInstructionExecutionRequestDevice M10
(14)
M_RJ71C24_Input_00C_1 (M+RJ71C24_Input_00C)
Input FB
bINPUTInstructionExecutionRequestDevice M10
B:i_bEN
o_bENO:B
bINPUTInstructionExecutionStatusDevice M2
C24_1
DUT:i_stModule
o_bOK:B
bINPUTInstructionExecutionCompleteDevice M0
wReceptionChannelNumber D1
UW:i_uCh
o_bErr:B
bINPUTInstructionAbnormalCompleteDevice M1
wReceptionDataAllowableCount D4
UW:i_uMaxRecvDataLength
o_uErrId:UW
wReceptionResult D2
o_uRecvDataLength:UW
wReceptionDataCount D3
o_uRecvData:UW
wnReceptionData D10
(60)
bINPUTInstructionExecutionCompleteDevice M0
SET
bINPUTInstructionNormalCompleteFlag M100
(62)
bINPUTInstructionAbnormalCompleteDevice M1
SET
bINPUTInstructionAbnormalCompleteFlag M101
(64)
END

(0) Read the combinations of user frames received.
Set the reception channel to CH1.
Set the allowable number of data to be received to 6.
Specify the execution instruction.

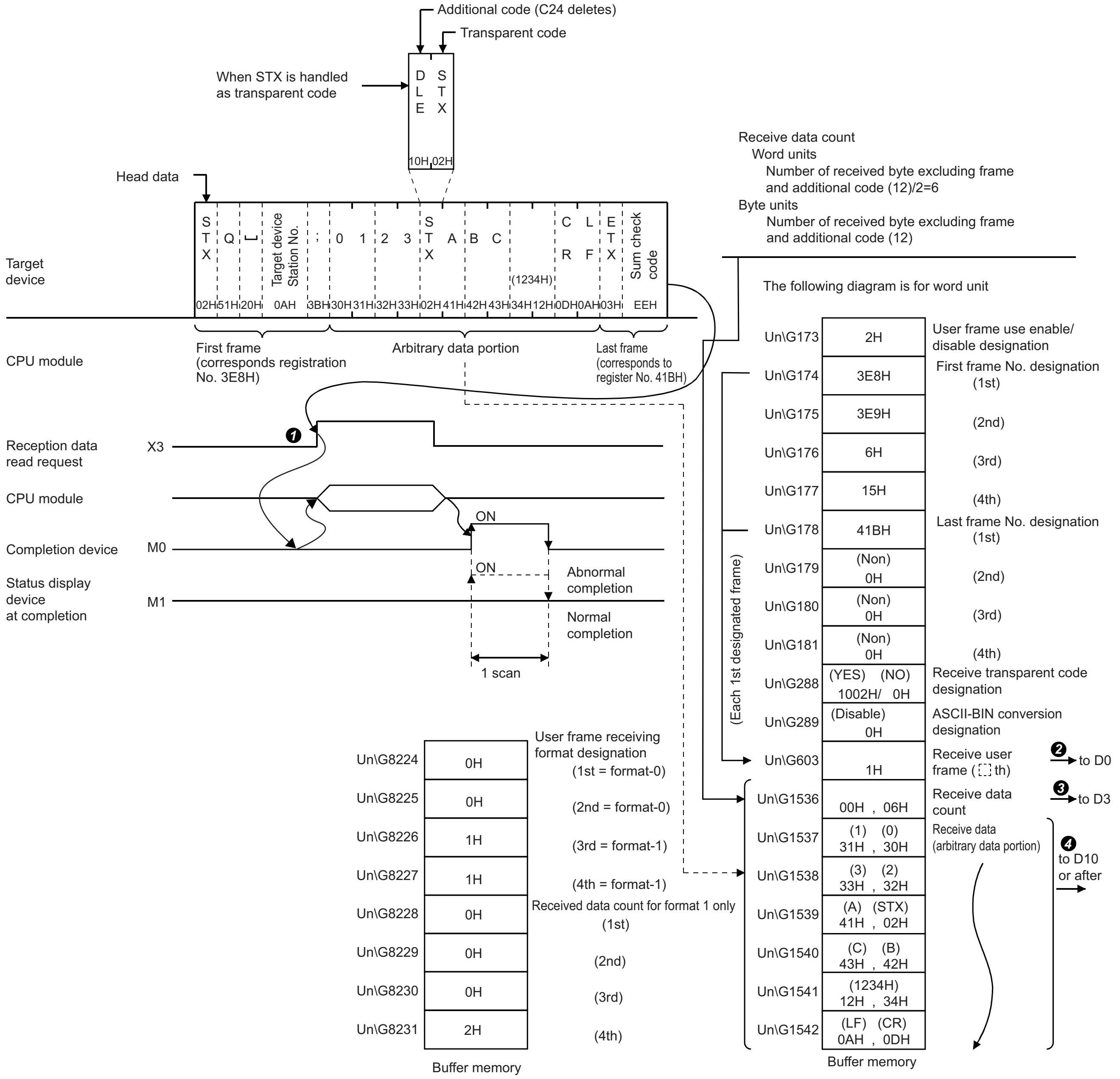
(14) Execute reception

(60) Set normal completion flag.

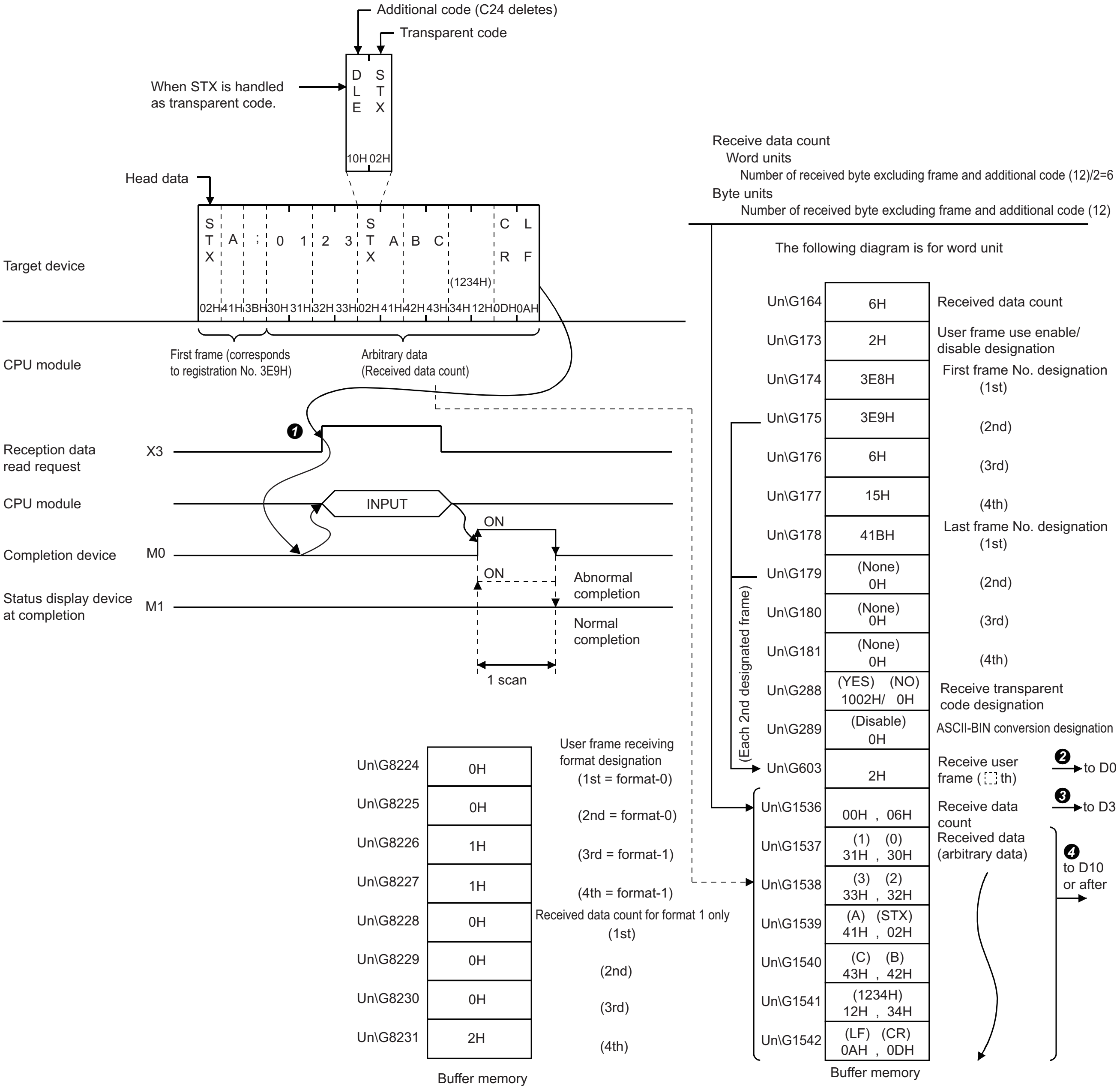
(62) Set abnormal completion flag.

Operation example of various combinations

■When receiving with a combination of the start frame, arbitrary data and final frame (reception using Method 0)


Additional code (C24 deletes)
Transparent code
When STX is handled as transparent code
D L E
S T X
10H 02H
Head data
Target device
S T X
Q
␣
Target device Station No.
;
0 1 2 3
S T X
A B C
(1234H)
C R
L F
E T X
Sum check code
02H 51H 20H 0AH 3BH 30H 31H 32H 33H 02H 41H 42H 43H 34H 12H 0DH 0AH 03H EEH
CPU module
First frame (corresponds registration No. 3E8H)
Arbitrary data portion
Last frame (corresponds to register No. 41BH)
Receive data count
Word units
Number of received byte excluding frame and additional code (12)/2=6
Byte units
Number of received byte excluding frame and additional code (12)
The following diagram is for word unit
Reception data read request
X3
❶
CPU module
Completion device
M0
ON
ON
Status display device at completion
M1
Abnormal completion
Normal completion
1 scan
Un\G173 2H User frame use enable/ disable designation
Un\G174 3E8H First frame No. designation (1st)
Un\G175 3E9H (2nd)
Un\G176 6H (3rd)
Un\G177 15H (4th)
Un\G178 41BH Last frame No. designation (1st)
Un\G179 (Non) 0H (2nd)
Un\G180 (Non) 0H (3rd)
Un\G181 (Non) 0H (4th)
Un\G288 (YES) (NO) 1002H/ 0H Receive transparent code designation
Un\G289 (Disable) 0H ASCII-BIN conversion designation
(Each 1st designated frame)
Un\G603 1H Receive user frame (□th)
❷ to D0
Un\G1536 00H , 06H Receive data count
❸ to D3
Un\G1537 (1) (0) 31H , 30H Receive data (arbitrary data portion)
Un\G1538 (3) (2) 33H , 32H
Un\G1539 (A) (STX) 41H , 02H
Un\G1540 (C) (B) 43H , 42H
Un\G1541 (1234H) 12H , 34H
Un\G1542 (LF) (CR) 0AH , 0DH
❹ to D10 or after
Buffer memory
User frame receiving format designation
Un\G8224 0H (1st = format-0)
Un\G8225 0H (2nd = format-0)
Un\G8226 1H (3rd = format-1)
Un\G8227 1H (4th = format-1)
Received data count for format 1 only
Un\G8228 0H (1st)
Un\G8229 0H (2nd)
Un\G8230 0H (3rd)
Un\G8231 2H (4th)
Buffer memory

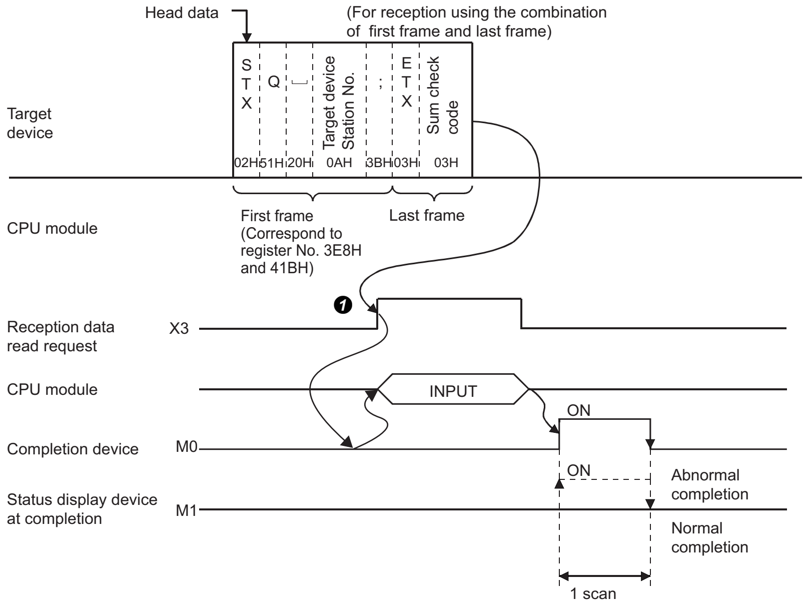
■When receiving with a combination of the start frame and arbitrary data (reception using Method 0)


Additional code (C24 deletes)
Transparent code
When STX is handled as transparent code.
D L E
S T X
10H 02H
Head data
Target device
S T X
A
;
0 1 2 3
S T X
A B C
(1234H)
C R
L F
02H 41H 3BH 30H 31H 32H 33H 02H 41H 42H 43H 34H 12H 0DH 0AH
CPU module
First frame (corresponds to registration No. 3E9H)
Arbitrary data (Received data count)
❶
Reception data read request
X3
CPU module
INPUT
ON
Completion device
M0
ON
Abnormal completion
Status display device at completion
M1
Normal completion
1 scan
Receive data count
Word units
Number of received byte excluding frame and additional code (12)/2=6
Byte units
Number of received byte excluding frame and additional code (12)
The following diagram is for word unit
(Each 2nd designated frame)
❷ to D0
❸ to D3
❹ to D10 or after
Buffer memory
Buffer memory

Un\G8224	0H	User frame receiving format designation (1st = format-0)
Un\G8225	0H	(2nd = format-0)
Un\G8226	1H	(3rd = format-1)
Un\G8227	1H	(4th = format-1)
Un\G8228	0H	Received data count for format 1 only (1st)
Un\G8229	0H	(2nd)
Un\G8230	0H	(3rd)
Un\G8231	2H	(4th)

Un\G164	6H	Received data count
Un\G173	2H	User frame use enable/disable designation
Un\G174	3E8H	First frame No. designation (1st)
Un\G175	3E9H	(2nd)
Un\G176	6H	(3rd)
Un\G177	15H	(4th)
Un\G178	41BH	Last frame No. designation (1st)
Un\G179	(None) 0H	(2nd)
Un\G180	(None) 0H	(3rd)
Un\G181	(None) 0H	(4th)
Un\G288	(YES) (NO) 1002H/ 0H	Receive transparent code designation
Un\G289	(Disable) 0H	ASCII-BIN conversion designation
Un\G603	2H	Receive user frame (⬚ th)
Un\G1536	00H , 06H	Receive data count
Un\G1537	(1) (0) 31H , 30H	Received data (arbitrary data)
Un\G1538	(3) (2) 33H , 32H	
Un\G1539	(A) (STX) 41H , 02H	
Un\G1540	(C) (B) 43H , 42H	
Un\G1541	(1234H) 12H , 34H	
Un\G1542	(LF) (CR) 0AH , 0DH	

■When receiving with user frame only (reception using Method 0)


Head data
(For reception using the combination of first frame and last frame)
Target device
S T X
Q
␣
Target device Station No.
;
E T X
Sum check code
02H
51H
20H
0AH
3BH
03H
03H
CPU module
First frame (Correspond to register No. 3E8H and 41BH)
Last frame
❶
Reception data read request
X3
CPU module
INPUT
ON
Completion device
M0
ON
Abnormal completion
Status display device at completion
M1
Normal completion
1 scan

Since there is no arbitrary data portion, the receive data count is [0].

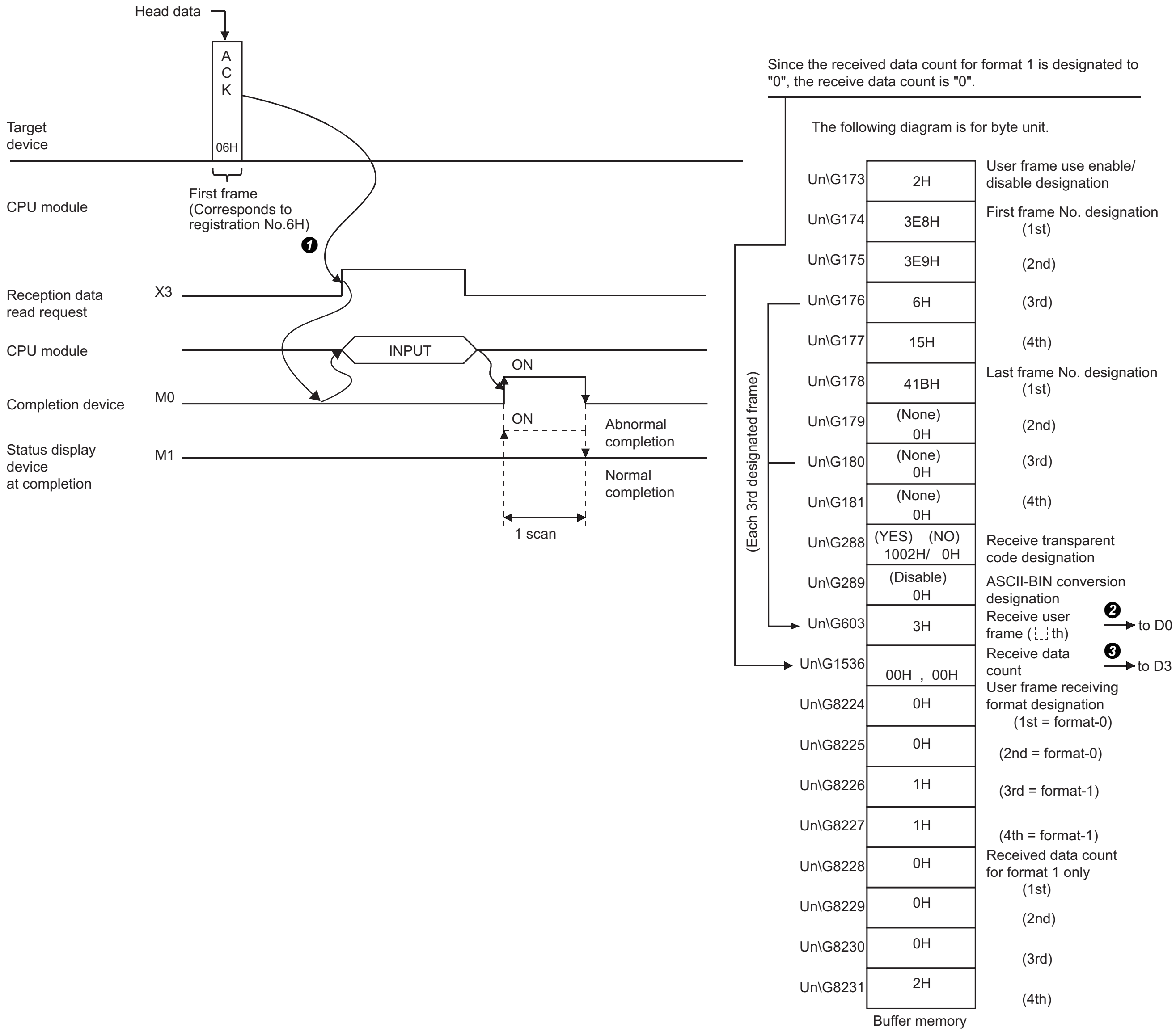
The following diagram is for reception using the combination of first frame and last frame.

(Each 1st designated frame)

	Buffer memory	
Un\G173	2H	User frame use enable/disable designation
Un\G174	3E8H	First frame No. designation (1st)
Un\G175	3E9H	(2nd)
Un\G176	6H	(3rd)
Un\G177	15H	(4th)
Un\G178	41BH	Last frame No. designation (1st)
Un\G179	(None) 0H	(2nd)
Un\G180	(None) 0H	(3rd)
Un\G181	(None) 0H	(4th)
Un\G288	(YES) (NO) 1002H/ 0H	Receive transparent code designation
Un\G289	(Disable) 0H	ASCII-BIN conversion designation
Un\G603	1H	Receive user frame (th) ❷ → to D0
Un\G1536	00H , 00H	Receive data count ❸ → to D3
Un\G8224	0H	User frame receive format designation (1st = format-0)
Un\G8225	0H	(2nd = format-0)
Un\G8226	1H	(3rd = format-1)
Un\G8227	1H	(4th = format-1)
Un\G8228	0H	Received data count for format 1 only (1st)
Un\G8229	0H	(2nd)
Un\G8230	0H	(3rd)
Un\G8231	2H	(4th)

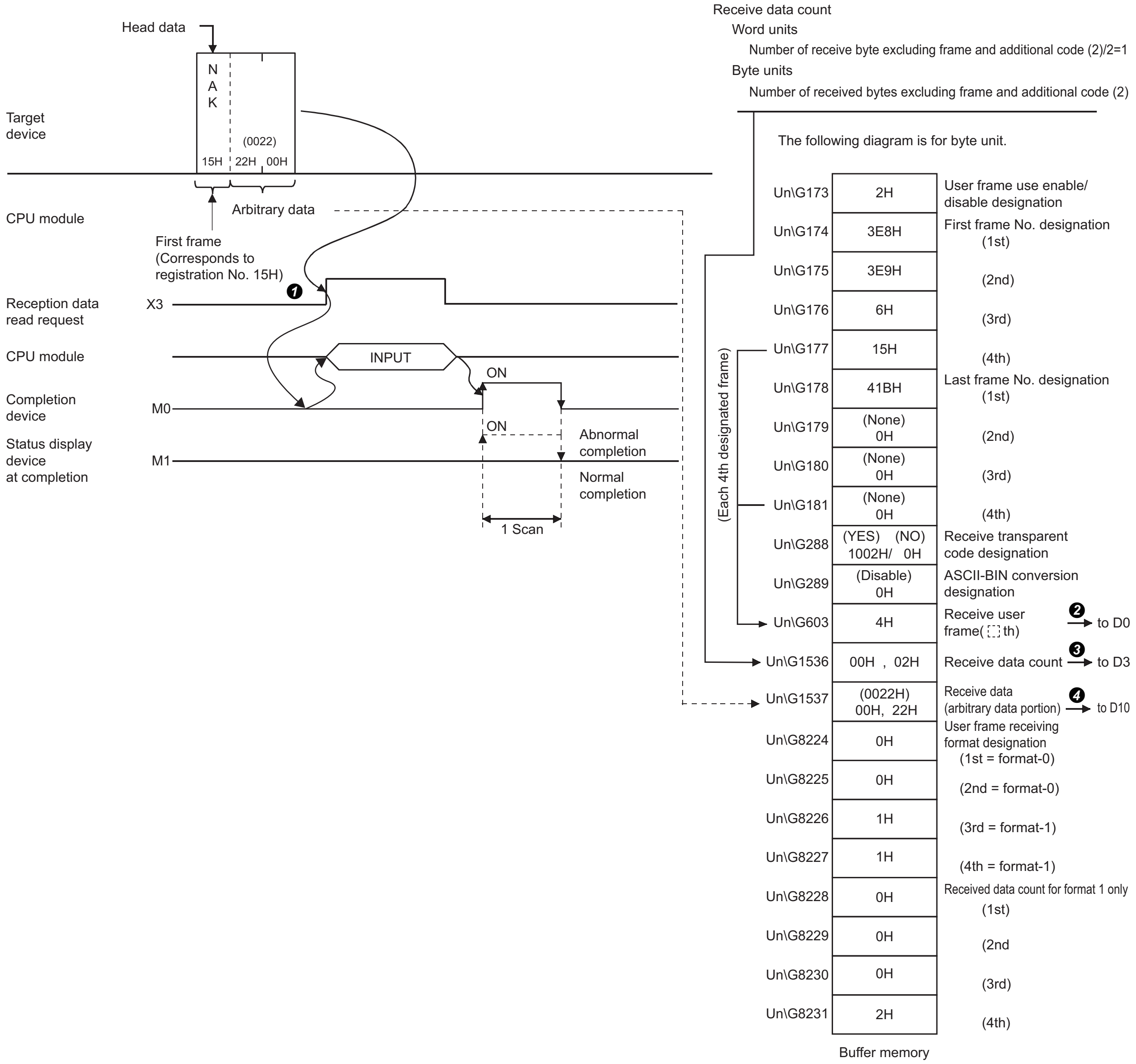
Buffer memory

■When receiving with first start only (reception using Method 1)


Head data
A
C
K
06H
Target device
CPU module
First frame (Corresponds to registration No.6H)
❶
Reception data read request
X3
CPU module
INPUT
ON
Completion device
M0
ON
Abnormal completion
Status display device at completion
M1
Normal completion
1 scan
Since the received data count for format 1 is designated to "0", the receive data count is "0".
The following diagram is for byte unit.
(Each 3rd designated frame)
❷ to D0
❸ to D3
Buffer memory

Address	Value	Description
Un\G173	2H	User frame use enable/disable designation
Un\G174	3E8H	First frame No. designation (1st)
Un\G175	3E9H	(2nd)
Un\G176	6H	(3rd)
Un\G177	15H	(4th)
Un\G178	41BH	Last frame No. designation (1st)
Un\G179	(None) 0H	(2nd)
Un\G180	(None) 0H	(3rd)
Un\G181	(None) 0H	(4th)
Un\G288	(YES) (NO) 1002H/ 0H	Receive transparent code designation
Un\G289	(Disable) 0H	ASCII-BIN conversion designation
Un\G603	3H	Receive user frame (⬚ th)
Un\G1536	00H , 00H	Receive data count
Un\G8224	0H	User frame receiving format designation (1st = format-0)
Un\G8225	0H	(2nd = format-0)
Un\G8226	1H	(3rd = format-1)
Un\G8227	1H	(4th = format-1)
Un\G8228	0H	Received data count for format 1 only (1st)
Un\G8229	0H	(2nd)
Un\G8230	0H	(3rd)
Un\G8231	2H	(4th)

■When receiving with a combination of the start frame and arbitrary data (method 1 dedicated receive end data quantity) (reception using Method 1)


Receive data count
Word units
Number of receive byte excluding frame and additional code (2)/2=1
Byte units
Number of received bytes excluding frame and additional code (2)
The following diagram is for byte unit.
Head data
N
A
K
(0022)
15H
22H
00H
Target device
CPU module
Arbitrary data
First frame
(Corresponds to registration No. 15H)
❶
Reception data read request
X3
CPU module
INPUT
ON
Completion device
M0
ON
Abnormal completion
Status display device at completion
M1
Normal completion
1 Scan
(Each 4th designated frame)

Un\G173	2H	User frame use enable/ disable designation
Un\G174	3E8H	First frame No. designation (1st)
Un\G175	3E9H	(2nd)
Un\G176	6H	(3rd)
Un\G177	15H	(4th)
Un\G178	41BH	Last frame No. designation (1st)
Un\G179	(None) 0H	(2nd)
Un\G180	(None) 0H	(3rd)
Un\G181	(None) 0H	(4th)
Un\G288	(YES) (NO) 1002H/ 0H	Receive transparent code designation
Un\G289	(Disable) 0H	ASCII-BIN conversion designation
Un\G603	4H	Receive user frame(th) ❷ → to D0
Un\G1536	00H , 02H	Receive data count ❸ → to D3
Un\G1537	(0022H) 00H, 22H	Receive data (arbitrary data portion) ❹ → to D10
Un\G8224	0H	User frame receiving format designation (1st = format-0)
Un\G8225	0H	(2nd = format-0)
Un\G8226	1H	(3rd = format-1)
Un\G8227	1H	(4th = format-1)
Un\G8228	0H	Received data count for format 1 only (1st)
Un\G8229	0H	(2nd
Un\G8230	0H	(3rd)
Un\G8231	2H	(4th)

Buffer memory

Application program for data reception using a combination that does not specify the start frame

The following shows the program example when receiving data using a combination that does not specify the start frame.

Reception program example

Data reception using a user frame set with the following conditions is shown below.

■C24 interface used for communication with target device

CH1 side RS-232 interface of C24

■Parameter settings

For other than the following items, use default values.

Item	Setting content	Buffer memory address to store registration value	Remarks
Word/byte units designation	Word/byte	150 (96H)	The unit is set to "Word" in the example.
Transparent code for receiving specification	Designated	288 (120H)	Specify as follows: Additional code: 10H (DLE),Transparent code: 02H (STX)
ASCII-BIN conversion designation	Do not convert	289 (121H)	—
Receive end data quantity	6	164 (A4H)	—
Receive end code	No designation	165 (A5H)	—
User frame use enable/disable designation	Use	173 (ADH)	
Start frame No. designation[*1]	No designation	174 to 177 (AEH to B1H)	
Final frame No. designation[*1]	Designated	178 to 181 (B2H to B5H)	
User frame receiving method designation	Method 0	8224 to 8227 (2020H to 2023H)	
Method 1 dedicated receive end data quantity designation	0H	8228 to 8231 (2024H to 2027H	

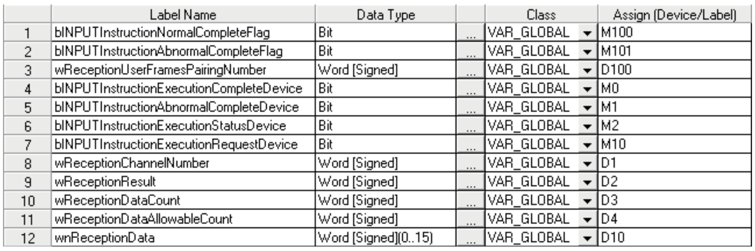
[*1] It indicates the registered contents of the user frame No. specified to the start and final frame No., respectively.

■Registration contents in user frame

Item		User frame No.	Registered code	Registered data content
Start frame No.	1st	0H (none)	—	—
	2nd	0H (none)	—	—
	3rd	0H (none)	—	—
	4th	0H (none)	—	—
Final frame No.	1st	419H	3BH, 04H	;, E0T
	2nd	0H (none)	—	—
	3rd	0H (none)	—	—
	4th	0H (none)	—	—

■Program example

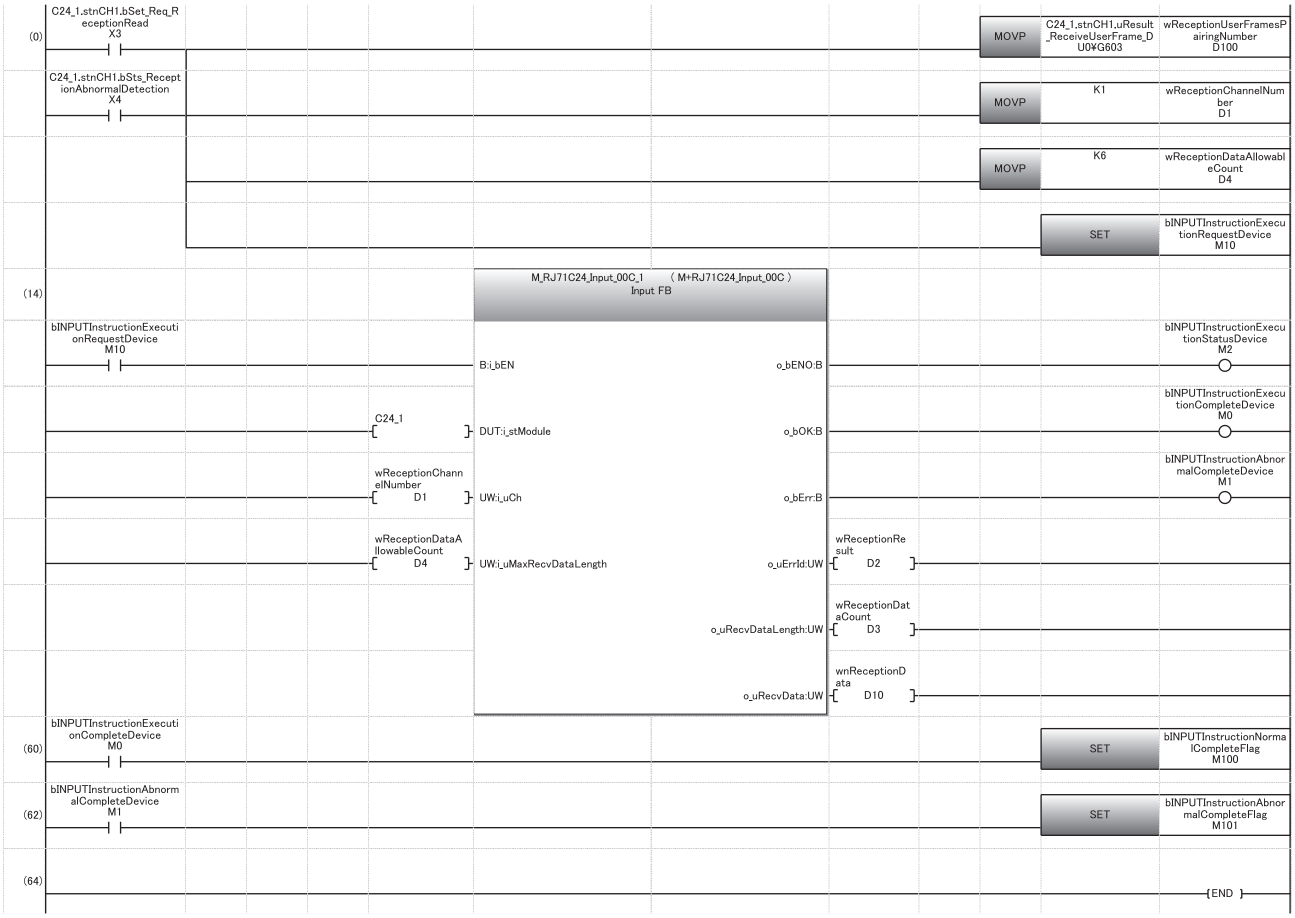
The following shows the example of data reception with a user frame.

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection	CH1 Reception abnormal detection	X4
	C24_1.stnCH1.uResult_ReceiveUserFrame_D	CH1_Receive user frame (nth)	U0\G603
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_Input	Reads the received data.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bINPUTInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
2	bINPUTInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101
3	wReceptionUserFramesPairingNumber	Word [Signed]	...	VAR_GLOBAL	D100
4	bINPUTInstructionExecutionCompleteDevice	Bit	...	VAR_GLOBAL	M0
5	bINPUTInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
6	bINPUTInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M2
7	bINPUTInstructionExecutionRequestDevice	Bit	...	VAR_GLOBAL	M10
8	wReceptionChannelNumber	Word [Signed]	...	VAR_GLOBAL	D1
9	wReceptionResult	Word [Signed]	...	VAR_GLOBAL	D2
10	wReceptionDataCount	Word [Signed]	...	VAR_GLOBAL	D3
11	wReceptionDataAllowableCount	Word [Signed]	...	VAR_GLOBAL	D4
12	wnReceptionData	Word [Signed](0..15)	...	VAR_GLOBAL	D10

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
C24_1.stnCH1.bSet_Req_ReceptionRead X3
MOVP
C24_1.stnCH1.uResult_ReceiveUserFrame_D U0¥G603
wReceptionUserFramesPairingNumber D100
C24_1.stnCH1.bSts_ReceptionAbnormalDetection X4
MOVP
K1
wReceptionChannelNumber D1
MOVP
K6
wReceptionDataAllowableCount D4
SET
bINPUTInstructionExecutionRequestDevice M10
(14)
M_RJ71C24_Input_00C_1 (M+RJ71C24_Input_00C)
Input FB
bINPUTInstructionExecutionRequestDevice M10
B:i_bEN
o_bENO:B
bINPUTInstructionExecutionStatusDevice M2
C24_1
DUT:i_stModule
o_bOK:B
bINPUTInstructionExecutionCompleteDevice M0
wReceptionChannelNumber D1
UW:i_uCh
o_bErr:B
bINPUTInstructionAbnormalCompleteDevice M1
wReceptionDataAllowableCount D4
UW:i_uMaxRecvDataLength
o_uErrId:UW
wReceptionResult D2
o_uRecvDataLength:UW
wReceptionDataCount D3
o_uRecvData:UW
wnReceptionData D10
(60)
bINPUTInstructionExecutionCompleteDevice M0
SET
bINPUTInstructionNormalCompleteFlag M100
(62)
bINPUTInstructionAbnormalCompleteDevice M1
SET
bINPUTInstructionAbnormalCompleteFlag M101
(64)
END

(0) Read the combinations of user frames received.
Set the reception channel to CH1.
Set the allowable number of data to be received to 6.
Specify the execution instruction.

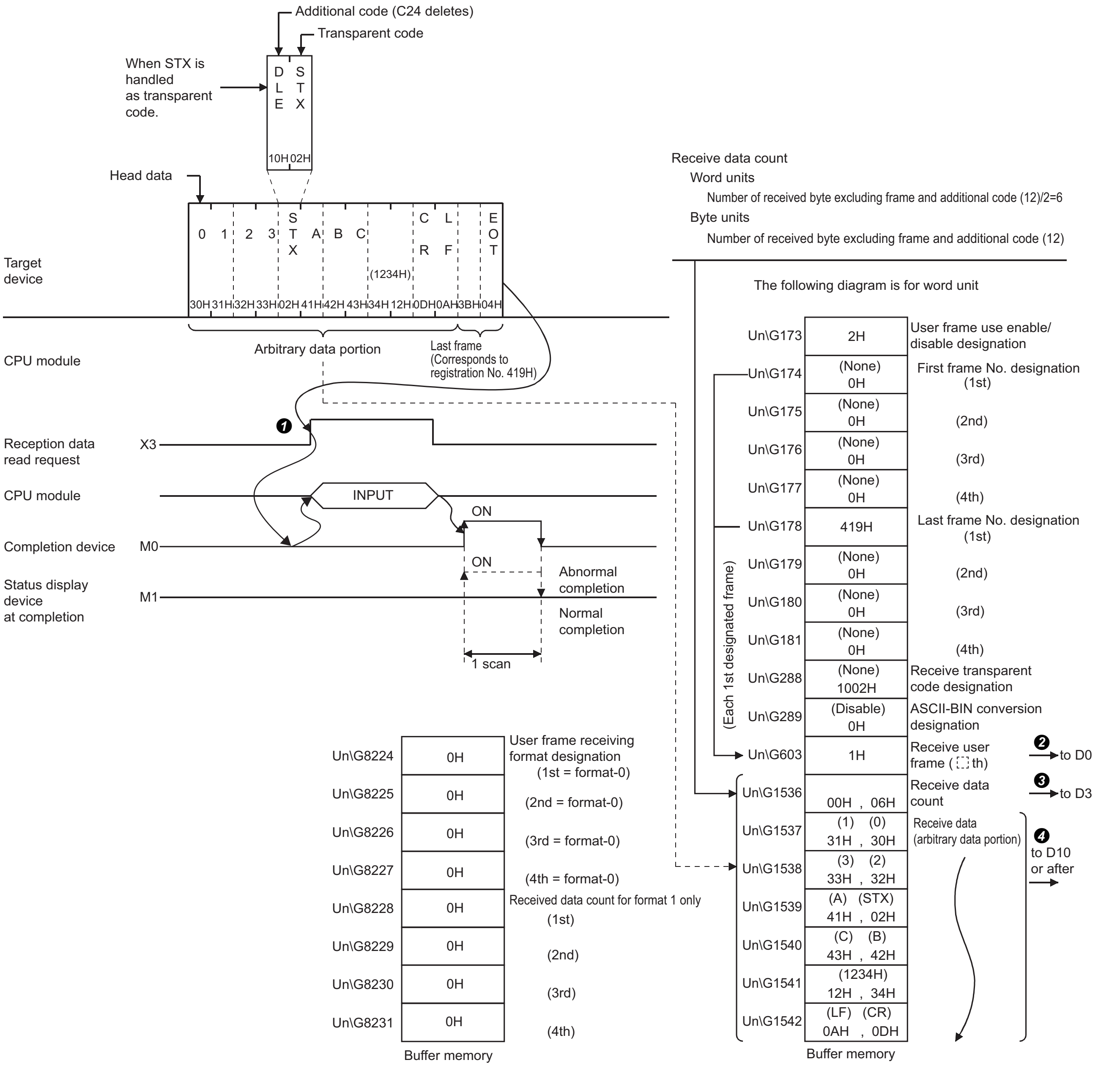
(14) Execute reception

(60) Set normal completion flag.

(62) Set abnormal completion flag.

Operation example

■When receiving data using a combination of arbitrary data and final frame (reception using Method 0)



Additional code (C24 deletes)
Transparent code
When STX is handled as transparent code.
D L E
S T X
10H 02H
Head data
Target device
0 1 2 3 STX A B C (1234H) CR LF EOT
30H 31H 32H 33H 02H 41H 42H 43H 34H 12H 0DH 0AH 3BH 04H
CPU module
Arbitrary data portion
Last frame (Corresponds to registration No. 419H)
Receive data count
Word units
Number of received byte excluding frame and additional code (12)/2=6
Byte units
Number of received byte excluding frame and additional code (12)
The following diagram is for word unit
1
Reception data read request
X3
CPU module
INPUT
ON
Completion device
M0
ON
Status display device at completion
M1
Abnormal completion
Normal completion
1 scan
Un\G173 2H User frame use enable/disable designation
Un\G174 (None) 0H First frame No. designation (1st)
Un\G175 (None) 0H (2nd)
Un\G176 (None) 0H (3rd)
Un\G177 (None) 0H (4th)
Un\G178 419H Last frame No. designation (1st)
Un\G179 (None) 0H (2nd)
Un\G180 (None) 0H (3rd)
Un\G181 (None) 0H (4th)
Un\G288 (None) 1002H Receive transparent code designation
Un\G289 (Disable) 0H ASCII-BIN conversion designation
(Each 1st designated frame)
Un\G603 1H Receive user frame (th)
2 to D0
Un\G1536 00H , 06H Receive data count
3 to D3
Un\G1537 (1) (0) 31H , 30H
Un\G1538 (3) (2) 33H , 32H
Un\G1539 (A) (STX) 41H , 02H
Un\G1540 (C) (B) 43H , 42H
Un\G1541 (1234H) 12H , 34H
Un\G1542 (LF) (CR) 0AH , 0DH
Receive data (arbitrary data portion)
4 to D10 or after
Buffer memory
Un\G8224 0H User frame receiving format designation (1st = format-0)
Un\G8225 0H (2nd = format-0)
Un\G8226 0H (3rd = format-0)
Un\G8227 0H (4th = format-0)
Un\G8228 0H Received data count for format 1 only (1st)
Un\G8229 0H (2nd)
Un\G8230 0H (3rd)
Un\G8231 0H (4th)
Buffer memory

17.4 Data Transmission

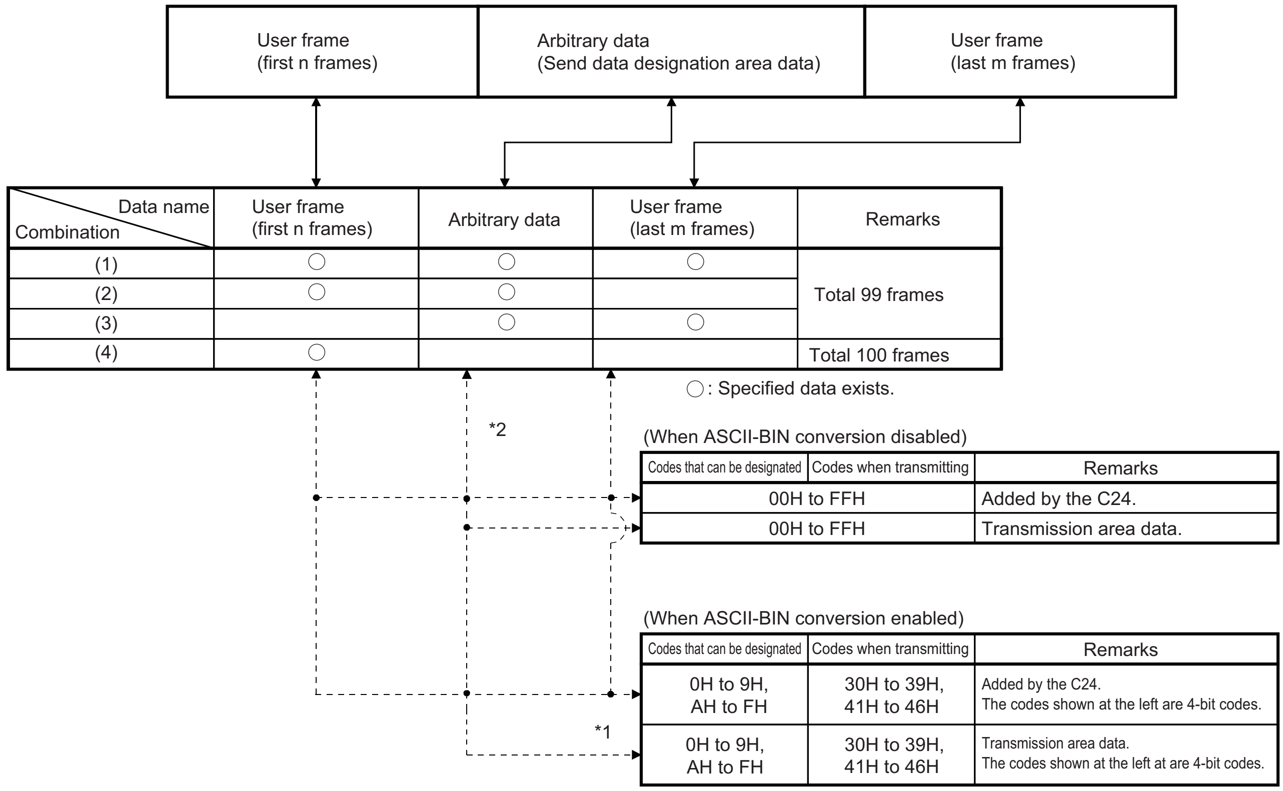
This section explains the arrangement of the data to be transmitted and transmission procedure when transmitting data using a user frame.

Send data

The order of data, codes, and handling of send data by C24 for data transmission using a user frame are shown below.

Send data arrangement

The combination shown below are only allowed as the order of data to be transmitted using a user frame.



Data name / Combination	User frame (first n frames)	Arbitrary data	User frame (last m frames)	Remarks
(1)	○	○	○	Total 99 frames
(2)	○	○		
(3)		○	○	
(4)	○			Total 100 frames

○: Specified data exists.

(When ASCII-BIN conversion disabled)

Codes that can be designated	Codes when transmitting	Remarks
00H to FFH		Added by the C24.
00H to FFH		Transmission area data.

(When ASCII-BIN conversion enabled)

Codes that can be designated	Codes when transmitting	Remarks
0H to 9H, AH to FH	30H to 39H, 41H to 46H	Added by the C24. The codes shown at the left are 4-bit codes.
0H to 9H, AH to FH	30H to 39H, 41H to 46H	Transmission area data. The codes shown at the left at are 4-bit codes.

*1 0H to FH data for four bits are converted to 30H to 39H and 41H to 46H ASCII code data and transmitted as the data codes of the data to be transmitted (including the transparent code data).

*2 Arbitrary data portion of send data
· When the send data quantity designated by the CPU module during transmission in byte units is an odd number of bytes, the data of the lower byte of the last transmitted data storage location in the send data storage area is transmitted.
· When ASCII-BIN conversion is enabled, the data to be transmitted is transmitted as 2 characters per byte.

Point

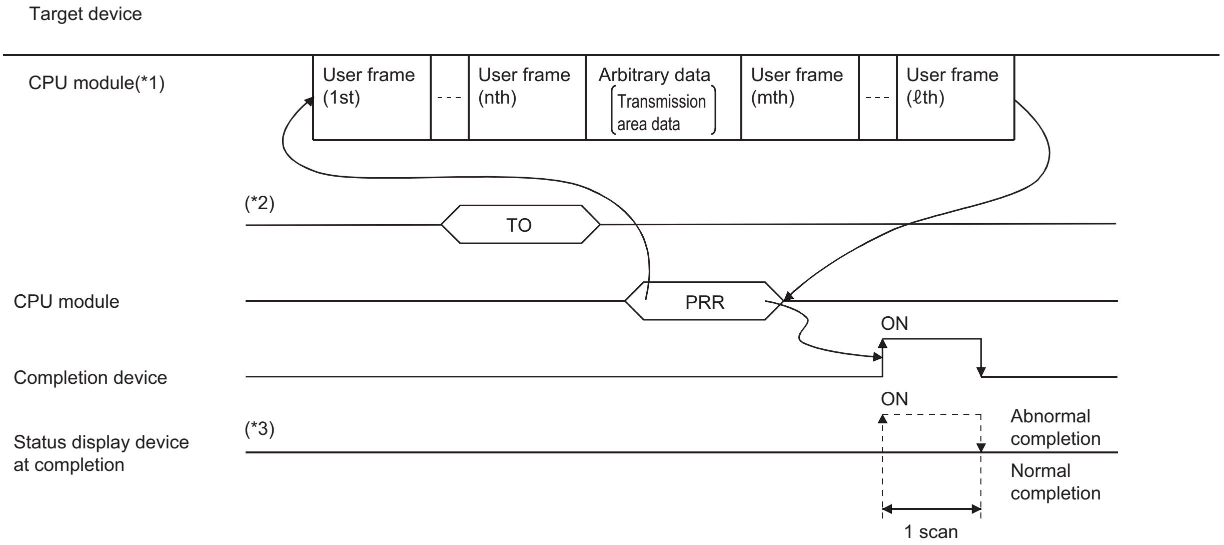
Handling of send data by C24

- The data of the user frame and the data in the send area designated from the CPU module is transmitted in accordance with the contents and order designated to the send user frame designation area in the buffer memory.
- For the user frame portion and arbitrary data portion, the data can be transmitted as ASCII code data using the ASCII-BIN conversion. (☞ Page 379 COMMUNICATING WITH ASCII CODE (ASCII-BIN CONVERSION))
- If a transparent code for sending is specified, the additional code data is added in front of the transparent code/additional code in the data of the designated area and transmitted.

Transmission procedure

The following describes the transmission procedure when a message containing user frames is transmitted to the target device.

Procedure



*1 Designate the user frame number for the data to be sent.

*2 CPU module processing

Before issuing a transmission request to C24, designate (write) the following data.

· Send user frame designation area : User frame number, transmission method and transmission order registered in C24

· Send area (Buffer memory): User data corresponding to arbitrary data portion in the transmission message

*3 When a send data quantity designation error or data transmission error occurred, C24 stores the error code to the transmission result storage area of control data for PRR instruction, and the status display device at completion turns ON.

Settings of send user frames

These settings are required for transmitting data to a target device using user frames by the nonprocedural protocol.
These settings are configured with Engineering tool or from the CPU module.

Settings with Engineering tool

- Configure the settings to transmit data in "Send user frame designation" of the parameter.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ Module Name ⇨ [Module Parameter] ⇨ "Application Settings" ⇨ "User frame specification"

Settings by the CPU module

■How to designate and write send data when transmitting with user frames

For arbitrary data portion when transmitting user frames and arbitrary data portion in combination, the send data quantity and send data are written in the send area as is the case in transmitting them in an arbitrary format.

User frames are registered using Engineering tool. Or, the user frame registration number to be sent is written from the CPU module to the send user frame designation area of the buffer memory as shown in the figure below.

After executing registration/write, C24 transmits the designated data in the designated order upon execution of the PRR instruction.

Ex.

Sending data in the following order

Transmission order	Send data type	User frame No.	Contents of transmitted/registered data
1	User frame	2H (2)	02H (STX)
2	User frame	3E8H (1000)	01H, 3BH (station No., ";")
3	Arbitrary data	8000H (-32768)	41H, 42H, 43H, 44H ("ABCD")
4	User frame	400H (1024)	03H, FFH, F6H, 0DH, 0AH (ETX, Sum check, CR, LF)

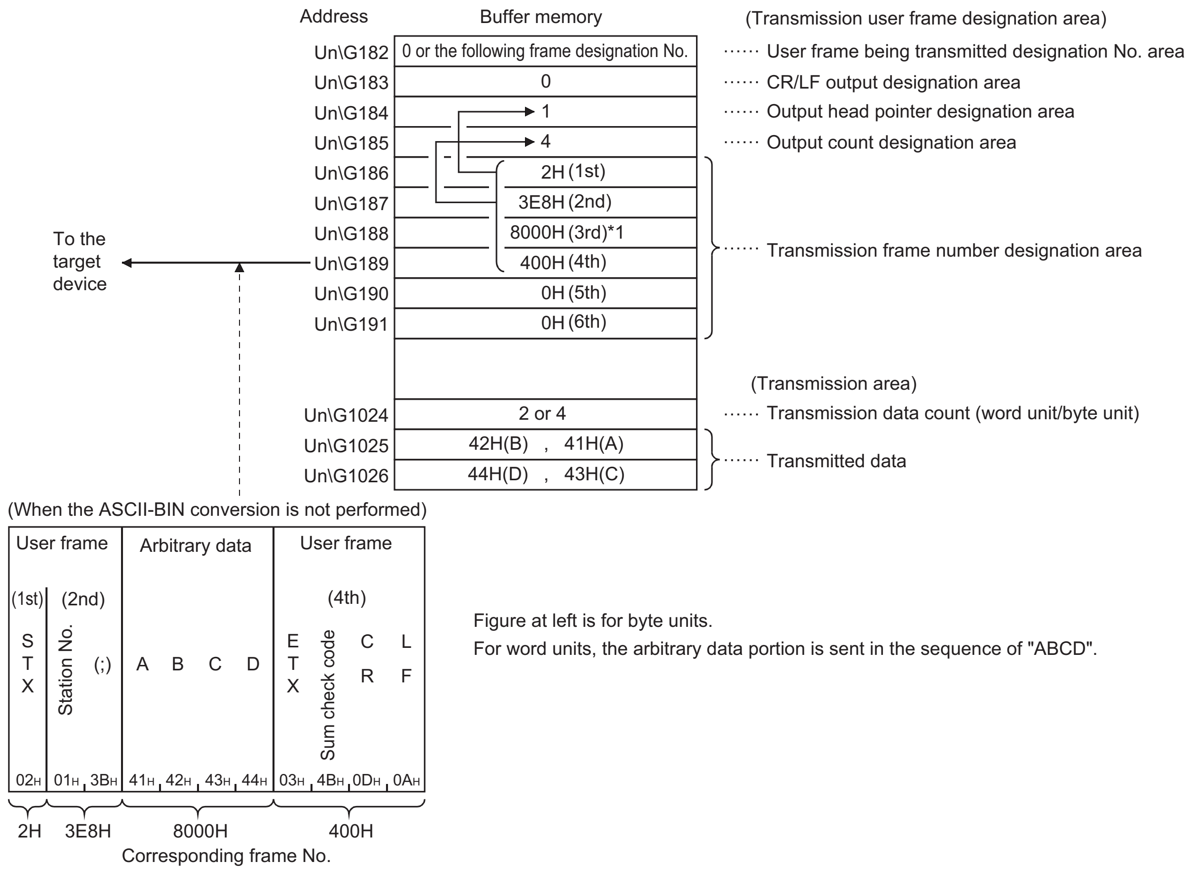


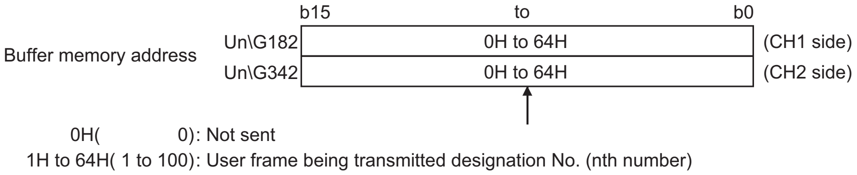
Figure at left is for byte units.
For word units, the arbitrary data portion is sent in the sequence of "ABCD".

*1 When transmitting data that is written in the send area, use 8000H as a temporary user frame No. (Page 348 Send user frame designation)

■Send user frame designation

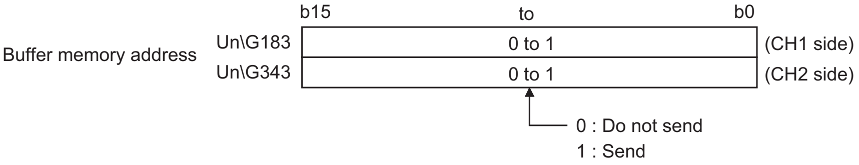
The following explains the purpose of the buffer memory to be used when transmitting data using user frames, along with the designated and stored values.

- 'Sending user frame' (Un\G182/342)
  In data communications using a user frame, what number of the send frame number designation area is being sent is stored.

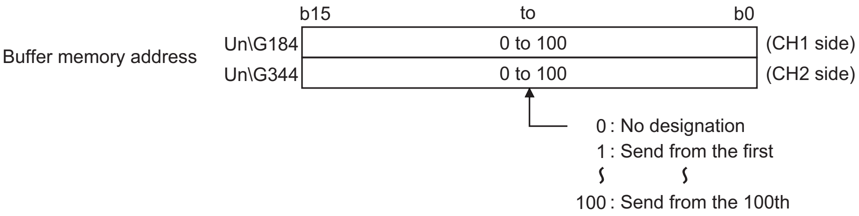


b15
to
b0
Buffer memory address
Un\G182
0H to 64H
(CH1 side)
Un\G342
0H to 64H
(CH2 side)
0H(0): Not sent
1H to 64H(1 to 100): User frame being transmitted designation No. (nth number)

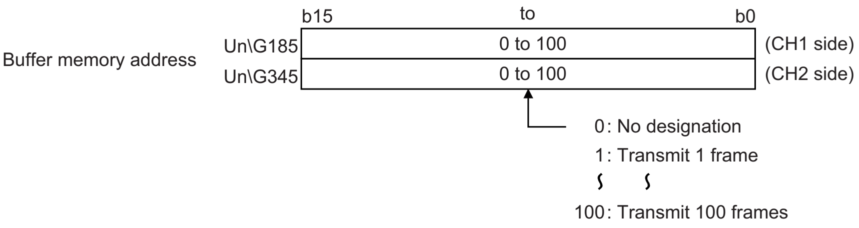
- 'CR/LF output designation' (Un\G183/343)
  When transmitting a user frame or arbitrary data that does not contain CR/LF, designate whether or not a CR+LF will be sent each time a user frame or arbitrary data is sent.



- 'Output start pointer designation' (Un\G184/344)
  Write the head position (□th) in the send frame number designation area for writing the registration number of the user frame to be sent.
  However, transmission using a user frame is not performed when the value of the output start pointer designation area is "0".



- 'Output quantity designation' (Un\G185/345)
  Write how many user frames are transmitted from the designated position in the output start pointer designation area.
  However, transmission is not performed and completed normally when the value of the output quantity designation area is "0".



- 'Send frame number designation' (Un\G186 to 285/346 to 445) Write the user frame No. to be sent in the order in which they are output from the position designated in the output start pointer designation area.
  When transmitting data that is written in the send area, use 8000H as a temporary user frame No.
  By designating user frame No. 8000H, C24 transmits the data in the send data specification area for the data quantity designated in the send data quantity designation area.

Buffer memory address	Un\G186	User frame No.(1st)	(CH1 side)
	to	to	
	Un\G285	User frame No.(100th)	
	Un\G346	User frame No.(1st)	(CH2 side)
	to	to	
	Un\G445	User frame No.(100th)	

Specifies the following user frame No. for the data to be sent.
Note that the number on the right side of the user frame No. below is the number used for transmission without ASCII-BIN conversion for only the data of any frame portion when transmission data is converted into ASCII-binary data and transmitted.

0H:	No transmission designation. (No additional transmission is allowed.)
1H to 3E7H/4001H to 43E7H:	Transmits the default frame having the designated number.
3E8H to 4AFH/43E8H to 44AFH:	Transmits the user frame having the designated number. (For flash ROM registration)
8000H/C000H:	Transmits data in the transmission area of the buffer memory.
8001H to 801FH/C001H to C01FH:	Transmits the user frame having the designated number. (For buffer memory registration)

Point

The following transmission can be performed by adding 4000H to the registered user frame number and specifying it.

- When ASCII-BIN conversion is enabled, a specified frame can be sent without the conversion. (☞ Page 380 ASCII-BIN Conversion for Data Communicated with Nonprocedural Protocol)
- A specified frame can be transmitted without adding the additional code by the transparent code for sending specification. (☞ Page 360 Handling Transparent and Additional Codes for Nonprocedural Protocol Data Communication)

17.5 Application Program for Data Transmission using a Combination that does not Specify the Start Frame

The following shows the program examples for data transmission using a user frame.

Transmission program example

Data transmission using user frames in the following conditions and settings is shown below:

■C24 interface used for communication with a target device

CH1 RS-232 interface of C24

■Parameter settings

Setting item				Setting value
Basic settings	Various control specification	Test mode setting		—
		Communication protocol setting		Nonprocedural protocol
		Communication speed setting		19200 bps
		Transmission setting	Data bit	8
			Parity bit	None
			Odd/even parity	Odd
			Stop bit	1
			Sum check code	None
			Online change	Disable
			Setting change	Disable
		Station number setting (CH1, 2 in common: 0 to 31)		1

■Buffer memory setting contents

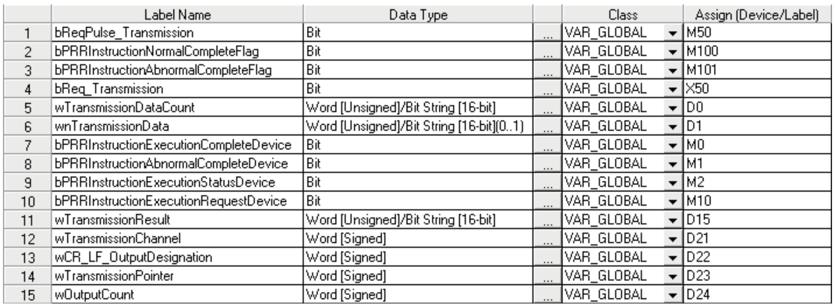
For other than the following items, use default values.

Item	Setting content	Buffer memory address to store registration value	Remarks
Word/byte units designation	Byte	150 (96H)	—
Transparent code for sending specification	No designation	287 (11FH)	—
ASCII-BIN conversion designation	Do not convert	289 (121H)	—
Send frame number designation (1st)	3F2H	186 (BAH)	—
Send frame number designation (2nd)	3F3H	187 (BBH)	—
Send frame number designation (3rd)	8001H	188 (BCH)	—
Send frame number designation (4th)	8000H	189(BDH)	—
Send frame number designation (5th)	41BH	190(BEH)	—

Program example when setting using Engineering tool

The following shows the method to send data by specifying the send frame number with Engineering tool.

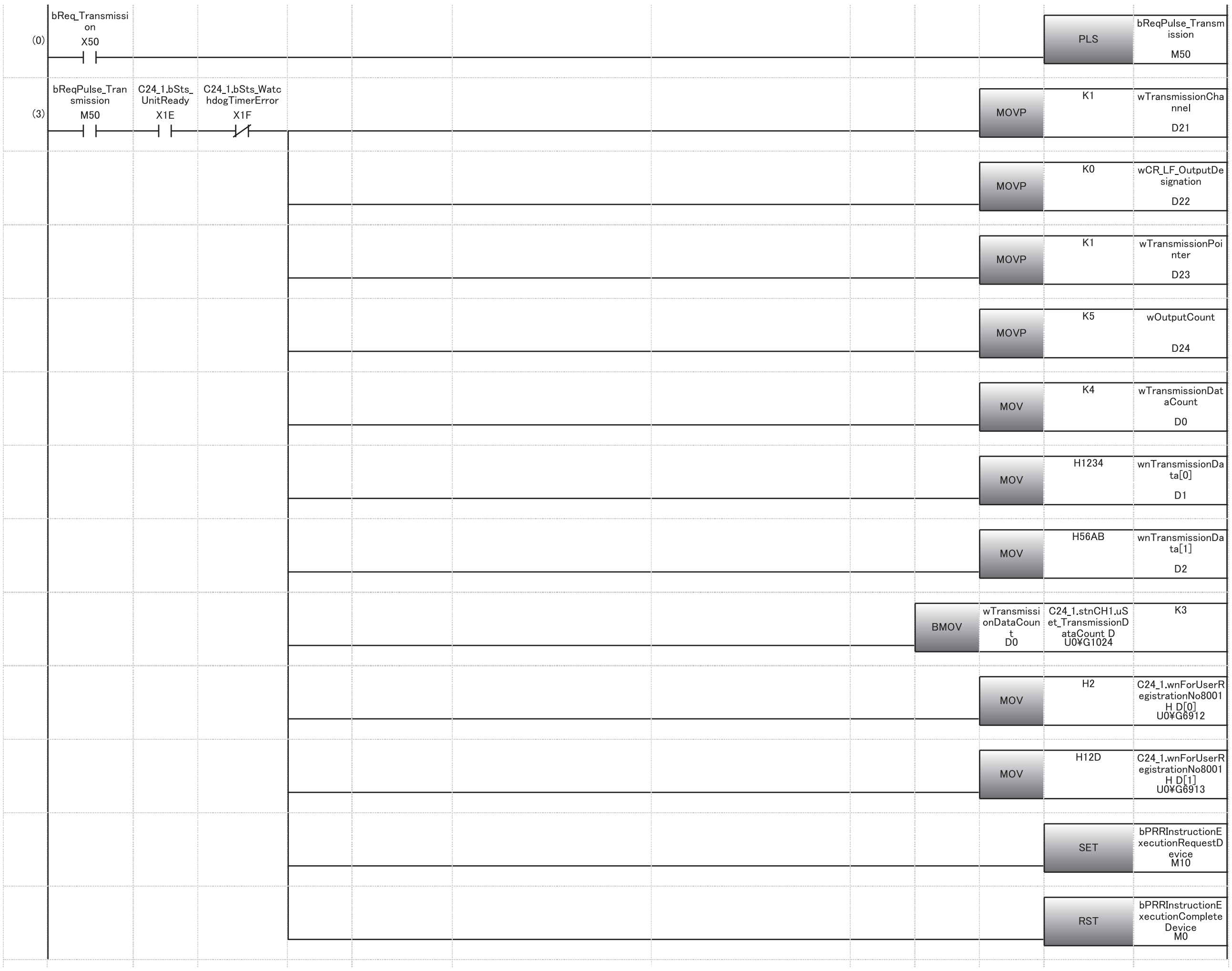
■Program example

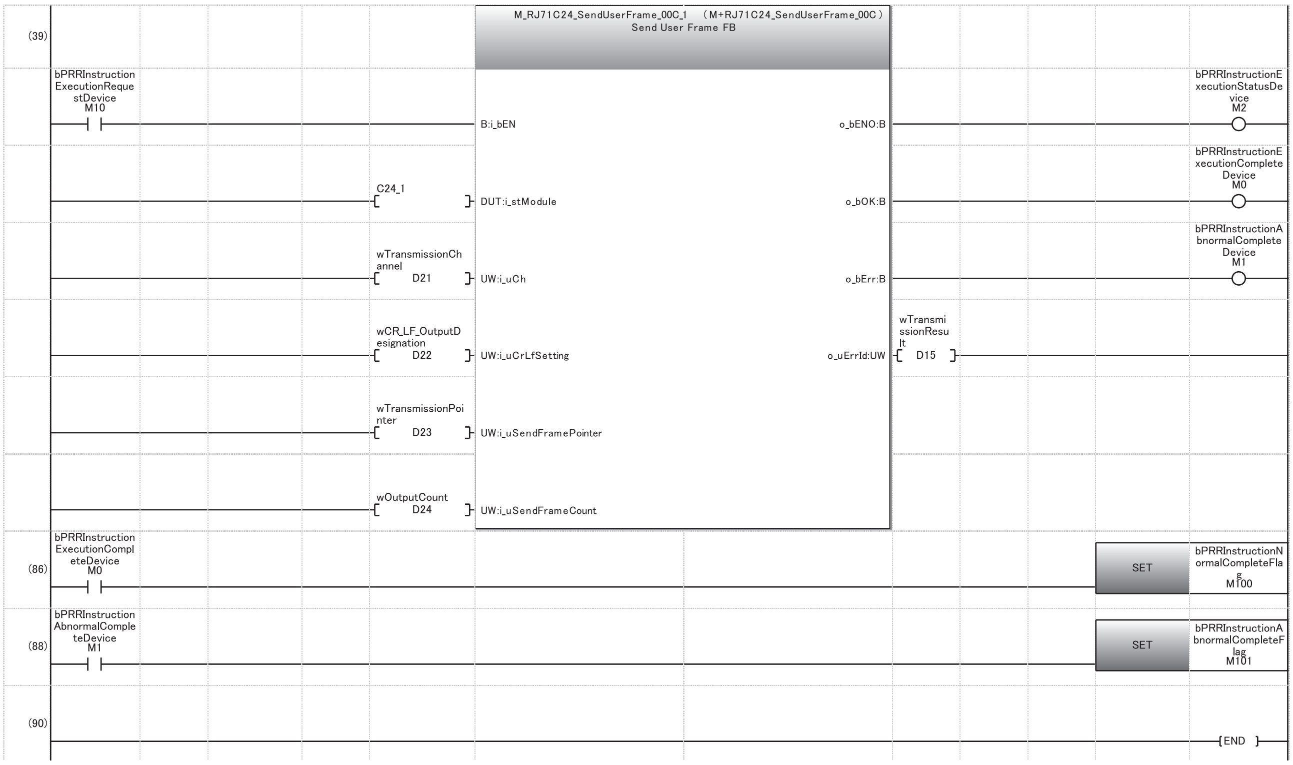
Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.bSts_UnitReady	Module READY	X1E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X1F
	C24_1.stnCH1.uSet_TransmissionDataCount_D	CH1_Send data quantity designation	U0\G1024
	C24_1.wnForUserRegistrationNo8001H_D[0]	For user registration number 8001H [0]	U0\G6912
	C24_1.wnForUserRegistrationNo8001H_D[1]	For user registration number 8001H [1]	U0\G6913
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_SendUserFrame	Sends data using the nonprocedural protocol communication and the user frame according to the setting of the user frame specification area for sending data.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReqPulse_Transmission	Bit	...	VAR_GLOBAL	M50
2	bPRRInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
3	bPRRInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101
4	bReq_Transmission	Bit	...	VAR_GLOBAL	X50
5	wTransmissionDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
6	wnTransmissionData	Word [Unsigned]/Bit String [16-bit](0..1)	...	VAR_GLOBAL	D1
7	bPRRInstructionExecutionCompleteDevice	Bit	...	VAR_GLOBAL	M0
8	bPRRInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
9	bPRRInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M2
10	bPRRInstructionExecutionRequestDevice	Bit	...	VAR_GLOBAL	M10
11	wTransmissionResult	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D15
12	wTransmissionChannel	Word [Signed]	...	VAR_GLOBAL	D21
13	wCR_LF_OutputDesignation	Word [Signed]	...	VAR_GLOBAL	D22
14	wTransmissionPointer	Word [Signed]	...	VAR_GLOBAL	D23
15	wOutputCount	Word [Signed]	...	VAR_GLOBAL	D24

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference

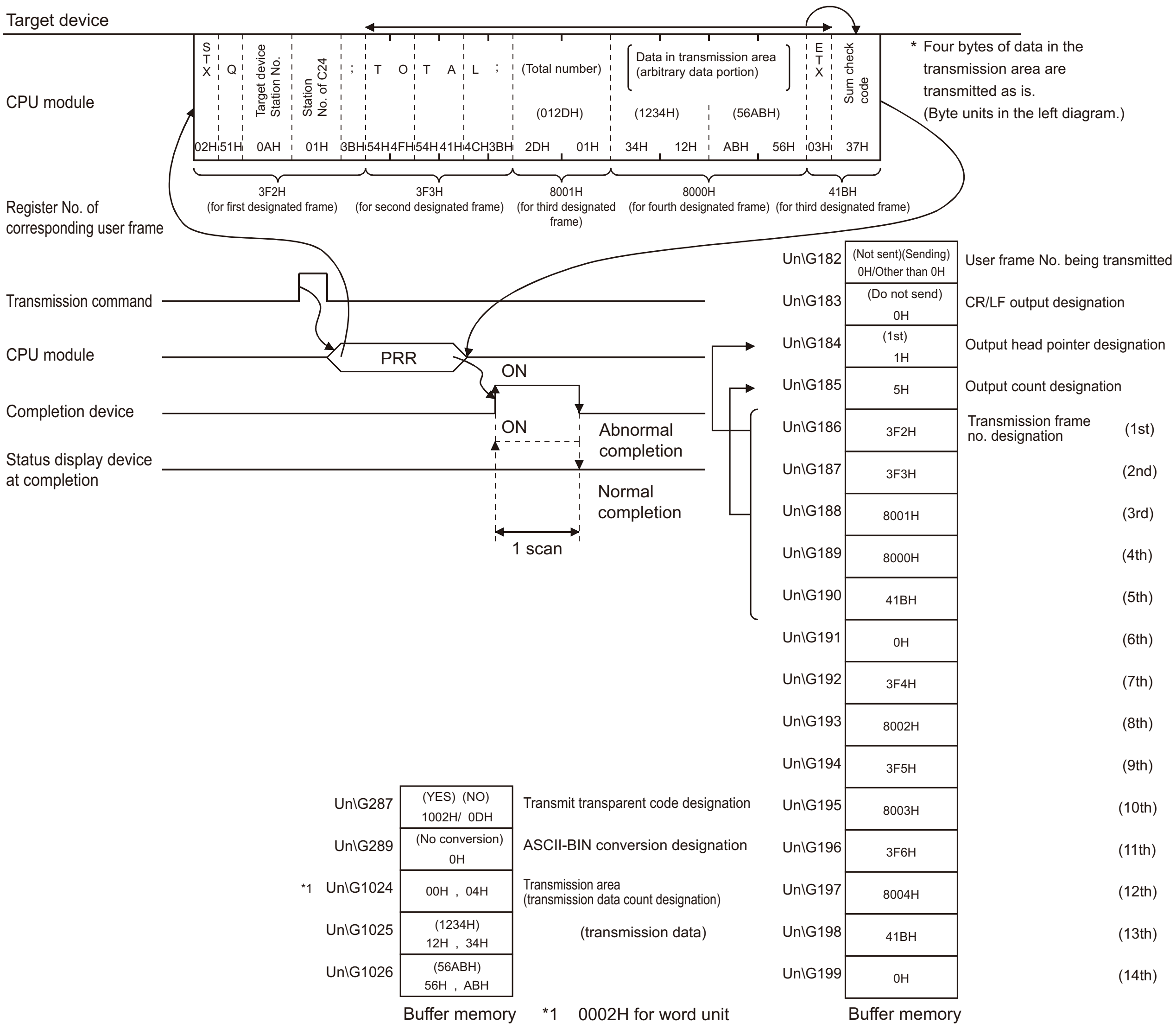

(0)
bReq_Transmission
X50
PLS
bReqPulse_Transmission
M50
(3)
bReqPulse_Transmission
M50
C24_1.bSts_UnitReady
X1E
C24_1.bSts_WatchdogTimerError
X1F
MOVP
K1
wTransmissionChannel
D21
MOVP
K0
wCR_LF_OutputDesignation
D22
MOVP
K1
wTransmissionPointer
D23
MOVP
K5
wOutputCount
D24
MOV
K4
wTransmissionDataCount
D0
MOV
H1234
wnTransmissionData[0]
D1
MOV
H56AB
wnTransmissionData[1]
D2
BMOV
wTransmissionDataCount
D0
C24_1.stnCH1.uSet_TransmissionDataCount D
U0¥G1024
K3
MOV
H2
C24_1.wnForUserRegistrationNo8001H D[0]
U0¥G6912
MOV
H12D
C24_1.wnForUserRegistrationNo8001H D[1]
U0¥G6913
SET
bPRRInstructionExecutionRequestDevice
M10
RST
bPRRInstructionExecutionCompleteDevice
M0


(39)
M_RJ71C24_SendUserFrame_00C_1 （M+RJ71C24_SendUserFrame_00C）
Send User Frame FB
bPRRInstructionExecutionRequestDevice
M10
B:i_bEN
o_bENO:B
bPRRInstructionExecutionStatusDevice
M2
C24_1
DUT:i_stModule
o_bOK:B
bPRRInstructionExecutionCompleteDevice
M0
wTransmissionChannel
D21
UW:i_uCh
o_bErr:B
bPRRInstructionAbnormalCompleteDevice
M1
wCR_LF_OutputDesignation
D22
UW:i_uCrLfSetting
o_uErrId:UW
wTransmissionResult
D15
wTransmissionPointer
D23
UW:i_uSendFramePointer
wOutputCount
D24
UW:i_uSendFrameCount
(86)
bPRRInstructionExecutionCompleteDevice
M0
SET
bPRRInstructionNormalCompleteFlag
M100
(88)
bPRRInstructionAbnormalCompleteDevice
M1
SET
bPRRInstructionAbnormalCompleteFlag
M101
(90)
END

(3) Set the transmission channel to 1.
Set CR/LF to "Do not send".
Set the send pointer to 1.
Set the output quantity to 5.
Set an arbitrary send data quantity.
Set an arbitrary send data.
Set an arbitrary send data in send area.
Set the transmission direction.

(86) Set normal completion flag.

(88) Set abnormal completion flag.


Target device
CPU module
S T X
Q
Target device Station No.
Station No. of C24
;
T
O
T
A
L
;
(Total number)
Data in transmission area (arbitrary data portion)
E T X
Sum check code
(012DH)
(1234H)
(56ABH)
02H
51H
0AH
01H
3BH
54H
4FH
54H
41H
4CH
3BH
2DH
01H
34H
12H
ABH
56H
03H
37H
3F2H (for first designated frame)
3F3H (for second designated frame)
8001H (for third designated frame)
8000H (for fourth designated frame)
41BH (for third designated frame)
* Four bytes of data in the transmission area are transmitted as is.
(Byte units in the left diagram.)
Register No. of corresponding user frame
Transmission command
CPU module
PRR
ON
Completion device
ON
Abnormal completion
Status display device at completion
Normal completion
1 scan

Un\G182	(Not sent)(Sending) 0H/Other than 0H	User frame No. being transmitted
Un\G183	(Do not send) 0H	CR/LF output designation
Un\G184	(1st) 1H	Output head pointer designation
Un\G185	5H	Output count designation
Un\G186	3F2H	Transmission frame no. designation (1st)
Un\G187	3F3H	(2nd)
Un\G188	8001H	(3rd)
Un\G189	8000H	(4th)
Un\G190	41BH	(5th)
Un\G191	0H	(6th)
Un\G192	3F4H	(7th)
Un\G193	8002H	(8th)
Un\G194	3F5H	(9th)
Un\G195	8003H	(10th)
Un\G196	3F6H	(11th)
Un\G197	8004H	(12th)
Un\G198	41BH	(13th)
Un\G199	0H	(14th)

Buffer memory

Un\G287	(YES) (NO) 1002H/ 0DH	Transmit transparent code designation
Un\G289	(No conversion) 0H	ASCII-BIN conversion designation
*1 Un\G1024	00H , 04H	Transmission area (transmission data count designation)
Un\G1025	(1234H) 12H , 34H	(transmission data)
Un\G1026	(56ABH) 56H , ABH	

Buffer memory

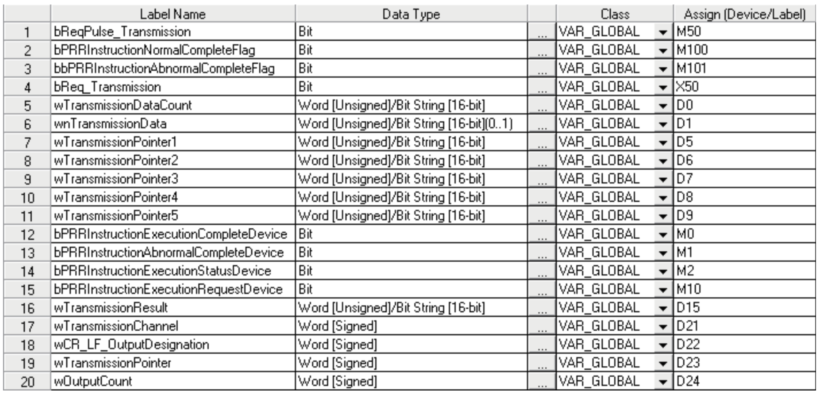
*1 0002H for word unit

Program example when not setting with Engineering tool

The following shows the methods to transmit data without specifying the send frame number with Engineering tool.

■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.bSts_UnitReady	Module READY	X1E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X1F
	C24_1.stnCH1.wnSet_OutputFrameNum_UserFrame_D	CH1_Send user frame_Send frame number designation	U0\G186
	C24_1.stnCH1.uSet_TransmissionDataCount_D	CH1_Send data quantity designation	U0\G1024
	C24_1.wnForUserRegistrationNo8001H_D[0]	For user registration number 8001H [0]	U0\G6912
	C24_1.wnForUserRegistrationNo8001H_D[1]	For user registration number 8001H [1]	U0\G6913
Label to be defined	Define the global label as shown below.		

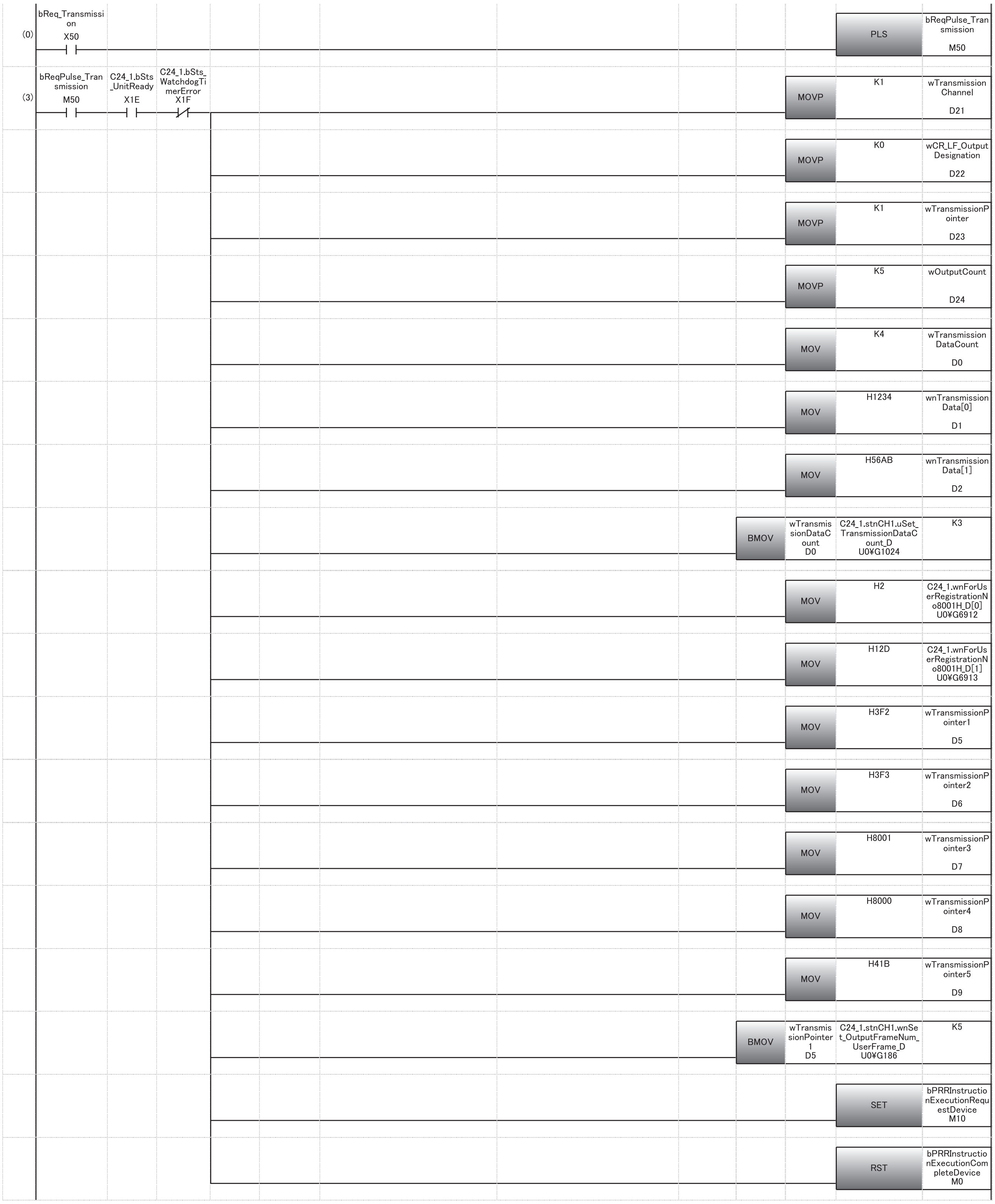


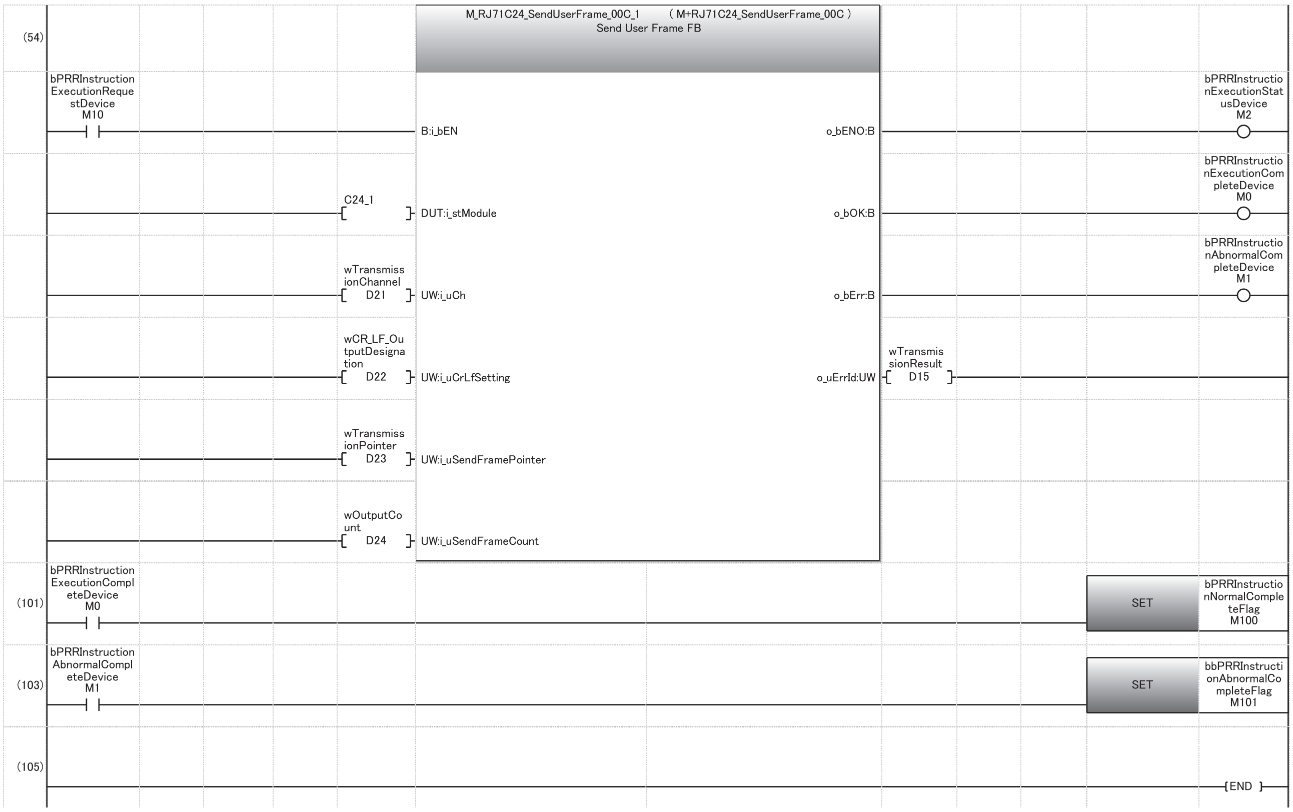
	Label Name	Data Type		Class	Assign (Device/Label)
1	bReqPulse_Transmission	Bit	...	VAR_GLOBAL	M50
2	bPRRInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
3	bbPRRInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101
4	bReq_Transmission	Bit	...	VAR_GLOBAL	X50
5	wTransmissionDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
6	wnTransmissionData	Word [Unsigned]/Bit String [16-bit](0..1)	...	VAR_GLOBAL	D1
7	wTransmissionPointer1	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D5
8	wTransmissionPointer2	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D6
9	wTransmissionPointer3	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D7
10	wTransmissionPointer4	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D8
11	wTransmissionPointer5	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D9
12	bPRRInstructionExecutionCompleteDevice	Bit	...	VAR_GLOBAL	M0
13	bPRRInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
14	bPRRInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M2
15	bPRRInstructionExecutionRequestDevice	Bit	...	VAR_GLOBAL	M10
16	wTransmissionResult	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D15
17	wTransmissionChannel	Word [Signed]	...	VAR_GLOBAL	D21
18	wCR_LF_OutputDesignation	Word [Signed]	...	VAR_GLOBAL	D22
19	wTransmissionPointer	Word [Signed]	...	VAR_GLOBAL	D23
20	wOutputCount	Word [Signed]	...	VAR_GLOBAL	D24

Category	Label name/FB name	Description	Device
FB	M+RJ71C24_SendUserFrame	Sends data using the nonprocedural protocol communication and the user frame according to the setting of the user frame specification area for sending data.	—

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
bReq_Transmission
X50
PLS
bReqPulse_Transmission
M50
(3)
bReqPulse_Transmission
M50
C24_1.bSts_UnitReady
X1E
C24_1.bSts_WatchdogTimerError
X1F
MOVP
K1
wTransmission Channel
D21
MOVP
K0
wCR_LF_Output Designation
D22
MOVP
K1
wTransmissionPointer
D23
MOVP
K5
wOutputCount
D24
MOV
K4
wTransmission DataCount
D0
MOV
H1234
wnTransmission Data[0]
D1
MOV
H56AB
wnTransmission Data[1]
D2
BMOV
wTransmissionDataCount
D0
C24_1.stnCH1.uSet_TransmissionDataCount_D
U0¥G1024
K3
MOV
H2
C24_1.wnForUserRegistrationNo8001H_D[0]
U0¥G6912
MOV
H12D
C24_1.wnForUserRegistrationNo8001H_D[1]
U0¥G6913
MOV
H3F2
wTransmissionPointer1
D5
MOV
H3F3
wTransmissionPointer2
D6
MOV
H8001
wTransmissionPointer3
D7
MOV
H8000
wTransmissionPointer4
D8
MOV
H41B
wTransmissionPointer5
D9
BMOV
wTransmissionPointer1
D5
C24_1.stnCH1.wnSet_OutputFrameNum_UserFrame_D
U0¥G186
K5
SET
bPRRInstructionExecutionRequestDevice
M10
RST
bPRRInstructionExecutionCompleteDevice
M0


(54)
M_RJ71C24_SendUserFrame_00C_1 (M+RJ71C24_SendUserFrame_00C)
Send User Frame FB
bPRRInstructionExecutionRequestDevice M10
B:i_bEN
o_bENO:B
bPRRInstructionExecutionStatusDevice M2
C24_1
DUT:i_stModule
o_bOK:B
bPRRInstructionExecutionCompleteDevice M0
wTransmissionChannel D21
UW:i_uCh
o_bErr:B
bPRRInstructionAbnormalCompleteDevice M1
wCR_LF_OutputDesignation D22
UW:i_uCrLfSetting
o_uErrId:UW
wTransmissionResult D15
wTransmissionPointer D23
UW:i_uSendFramePointer
wOutputCount D24
UW:i_uSendFrameCount
(101)
bPRRInstructionExecutionCompleteDevice M0
SET
bPRRInstructionNormalCompleteFlag M100
(103)
bPRRInstructionAbnormalCompleteDevice M1
SET
bbPRRInstructionAbnormalCompleteFlag M101
(105)
END

(3) Set CR/LF to "Do not send".
Set the send pointer to 1.
Set the output quantity to 5.
Set an arbitrary send data.
Set an arbitrary send data in send area.
Set the send user frame.

(101) Set normal completion flag.

(103) Set abnormal completion flag.

	Normal end		Abnormal end	
D0	Transmission data count	(0004H)	Transmission data count	(0004H)
D1	Transmission data	(3412H)	Transmission data	(3412H)
D2		(AB56H)		(AB56H)
D5	Transmission frame No.	(03F2H)	Transmission frame No.	(03F2H)
D6		(03F3H)		(03F3H)
D7		(8001H)		(8001H)
D8		(8000H)		(8000H)
D9		(041BH)		(041BH)
D10		(0000H)		(0000H)
D11	Interface No.	(0001H)	Interface No.	(0001H)
D12	Transmission result	(0000H)	Transmission result	(other than 0000H)
D13	CR/LF output	(0000H)	CR/LF output	(0000H)
D14	Output head pointer	(0001H)	Output head pointer	(0001H)
D15	Output counter	(0005H)	Output counter	(0005H)

18 TRANSPARENT CODES AND ADDITIONAL CODES

Transparent codes and additional codes are used during data communication with a target device to send/receive one-byte data for transmission control on the target device side as arbitrary data.

Transparent codes and additional codes are handled in data communication using the nonprocedural or bidirectional protocol.

Type	Description
Transparent code	One-byte data for transmission control
Additional code	During transmission, one-byte data added immediately before the transparent code and additional code data During reception, one-byte data to be deleted (The immediately following one-byte data is processed for reception.)

18.1 Handling Transparent Code and Additional Code Data

The following explains how C24 handles transparent codes and additional codes during data communication using the nonprocedural or bidirectional protocol.

For the range of additional code data that is added or deleted, refer to the following sections.

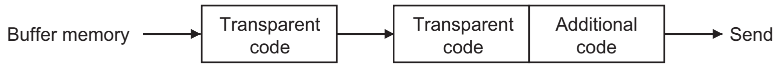
☞ Page 360 Handling Transparent and Additional Codes for Nonprocedural Protocol Data Communication

☞ Page 371 Handling Transparent and Additional Codes for Bidirectional Protocol Data Communication

At the time of data transmission

Additional code data is added immediately before the transparent code and additional code data set for transmission.

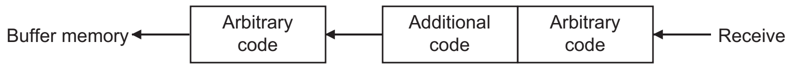
Ex.



At the time of data reception

When additional code data set for reception is detected, the additional code data is removed and the immediately succeeding 1-byte data is processed for reception.

Ex.



During data communication using the ASCII-BIN conversion

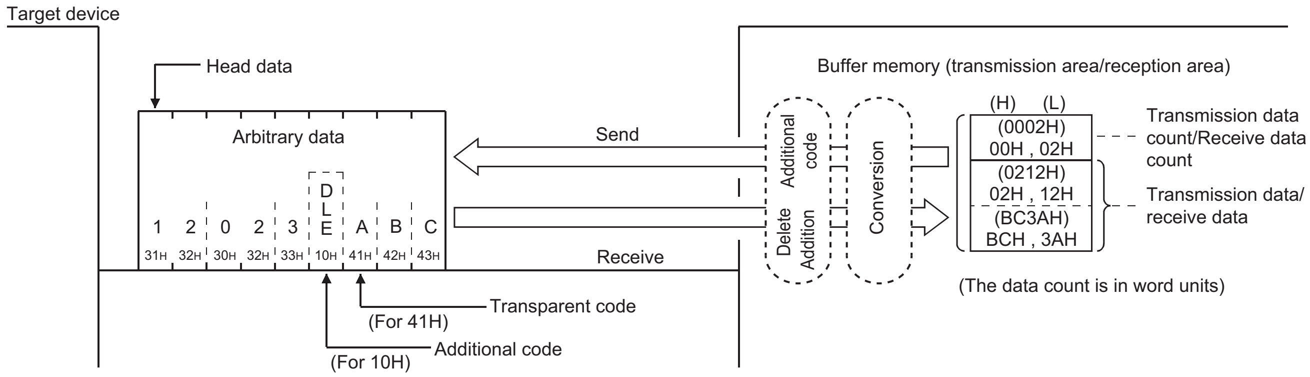
When transmitting data, the transparent code check is performed to the data after ASCII-BIN conversion. If a transparent code is detected, an additional code is added to the 1-byte immediately preceding it.

When receiving data, the additional code check is performed to the data before ASCII-BIN conversion. If an additional code is detected, the additional code are deleted.

Data which will be subject for the handling of a transparent code and an additional code is data after ASCII-BIN conversion for data transmission and data before ASCII-BIN conversion for data reception.

Ex.

When transmitting/receiving data using an arbitrary format of the nonprocedural protocol



18.2 Registering Transparent Codes and Additional Codes

To control transparent codes and additional codes for data to be sent/received with the nonprocedural or bidirectional protocol, configuring the settings to C24 prior to data communication is required.

Registration method

For each interface, 10 combinations and one combination of transparent codes and additional codes can be set for transmission and reception, respectively.

Transparent codes and additional codes are set in "Transparent code designation_Transparent code for sending specification (nth)", and "Transparent code designation_Transparent code for receiving specification" of the parameter.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

When specifying a transparent code for sending, set it by left justifying from the transparent code for sending specification (1st) in order.

Precautions

If additional code data is received during data reception, C24 will not treat the immediately succeeding one-byte data as the control data for the following frame:

- Data received as the start frame and final frame of the user frames

Therefore, do not configure the following settings. (The code for the data described above cannot be designated as additional codes for data reception.)

- A receive user frame setting that contains an additional code data for data reception
- The same receive same code setting as the additional code data for data reception

18.3 Handling Transparent and Additional Codes for Nonprocedural Protocol Data Communication

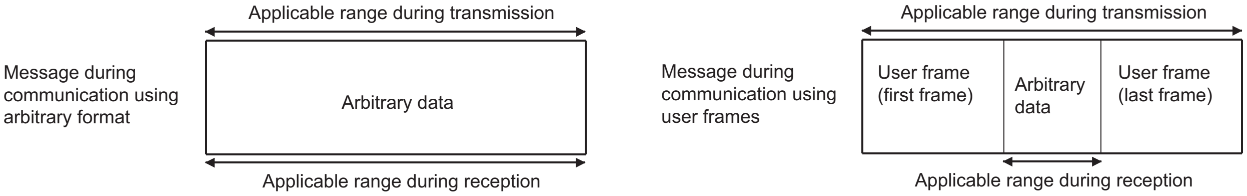
This section explains how to handle the transparent codes and additional codes during data communication using the nonprocedural protocol.

Adding/deleting data

The data designated by the additional code is added to/deleted from the data to be transmitted or received.

Transmitted/received data range for processing

The range of transmitted/received data to which the processing of transparent codes and additional codes is performed is as shown below.


Applicable range during transmission
Message during communication using arbitrary format
Arbitrary data
Applicable range during reception
Applicable range during transmission
Message during communication using user frames
User frame (first frame)
Arbitrary data
User frame (last frame)
Applicable range during reception

C24 performs the following processing during data transmission and reception.

■At the time of data reception

When an additional code set for reception is detected, C24 removes the additional code data and perform data reception processing by treating the immediately succeeding 1-byte data as a part of the received data.

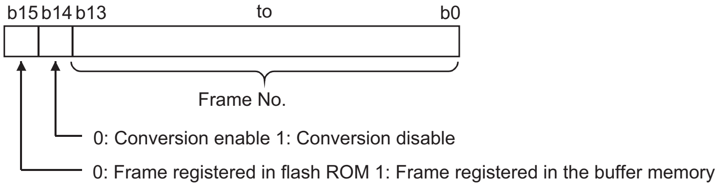
■At the time of data transmission

When transparent code/additional code data set for transmission is detected, C24 transmits data by adding the additional code designation data immediately before it.

During data transmission using user frames, even if a transparent code or additional code has been specified in the transparent code for sending specification area, it is possible to transmit data without adding the additional code data to the user frame portion or arbitrary data portion.

When sending data without adding the additional code specified by the transparent code for sending specification, specify the user frame number using the following method.

- Specify the number obtained by adding 4000H to the registered user frame No.


b15
b14
b13
to
b0
Frame No.
0: Conversion enable 1: Conversion disable
0: Frame registered in flash ROM 1: Frame registered in the buffer memory

When sending data in the portions designated in 4001H to 44AFH, C000H to C01FH, the data is transmitted without conversion even when the following designations are enabled.

- Designating "Enable" in the ASCII-BIN conversion designation area. (☞ Page 380 ASCII-BIN Conversion for Data Communicated with Nonprocedural Protocol)
- Designating "Auto delete enabled" in the NULL character automatic removal designation area. (☞ Page 441 NULL Character Removal to Send Data using Nonprocedural Protocol)

User frame No. to be sent	Specified No. when transmitting data without adding the additional code specified by the transparent code for sending specification (4000H added)
1H to 3E7H (1 to 999)	4001H to 43E7H (16385 to 17383)
3E8H to 4AFH (1000 to 1199)	43E8H to 44AFH (17384 to 17583)
8000H to 801FH (-32768 to -32737)	C000H to C01FH (-16384 to -16353)

Processing order for transmitting/receiving data

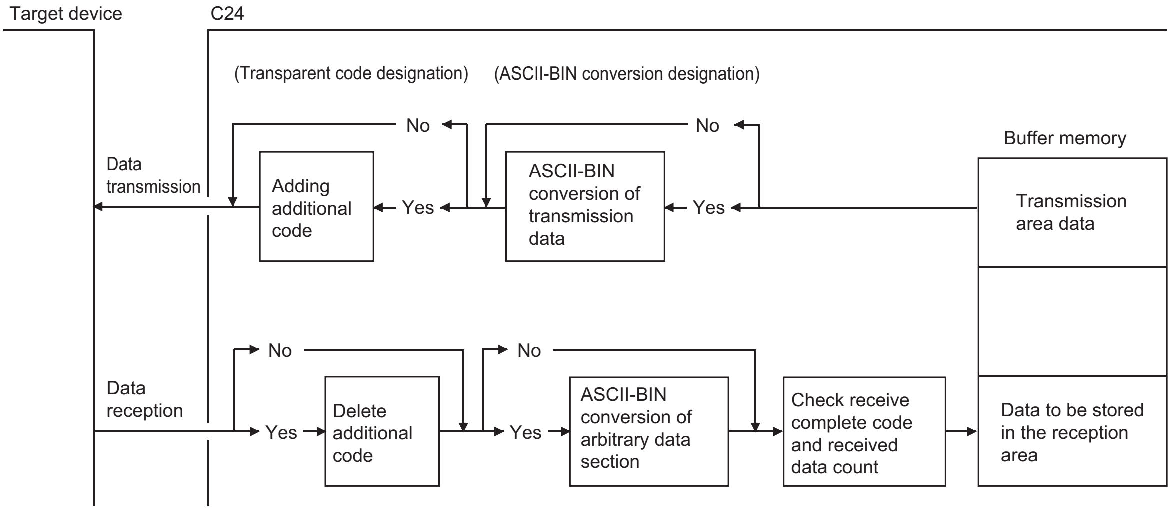
The processing steps taken by C24 to transmit/receive data by the transparent code designation and ASCII-BIN conversion enabled are as shown below.

■Data reception in arbitrary format

- C24 deletes (removes) the additional code designation data when a transparent code for receiving is specified.
- C24 stores the arbitrary data portion in the receive area on the buffer memory.
  When ASCII-BIN conversion is designated, C24 converts it to binary code data and stores in the buffer memory.
- C24 issues a reception data read request to the CPU module when the receive end code data is stored or data equivalent to the receive end data quantity is stored during reception of an arbitrary data portion.

■Data transmission in arbitrary format

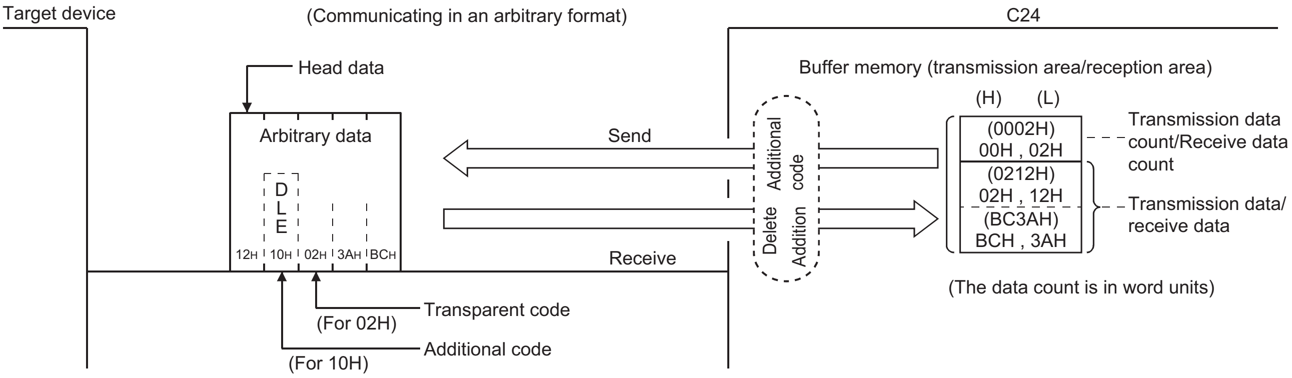
- C24 transmits the data to be sent (arbitrary data portion of a transmission message), which is designated by the CPU module.
  When ASCII-BIN conversion is designated, C24 converts it to ASCII code data and transmits it.
- C24 adds the additional code data immediately before the transparent code/additional code data, and transmits it when a transparent code for sending is specified.
- C24 notifies the CPU module of the transmission normal (abnormal) completion once transmission for the send data quantity has been completed.



Target device
C24
(Transparent code designation)
(ASCII-BIN conversion designation)
No
No
Buffer memory
Data transmission
Adding additional code
Yes
ASCII-BIN conversion of transmission data
Yes
Transmission area data
No
No
Data reception
Yes
Delete additional code
Yes
ASCII-BIN conversion of arbitrary data section
Check receive complete code and received data count
Data to be stored in the reception area

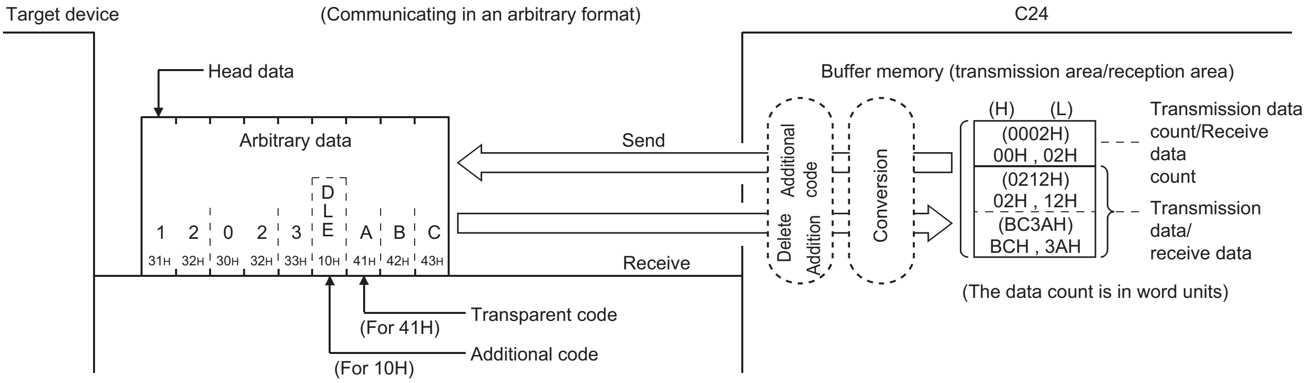
Ex.

When ASCII-BIN conversion is not performed



Ex.

When ASCII-BIN conversion is performed

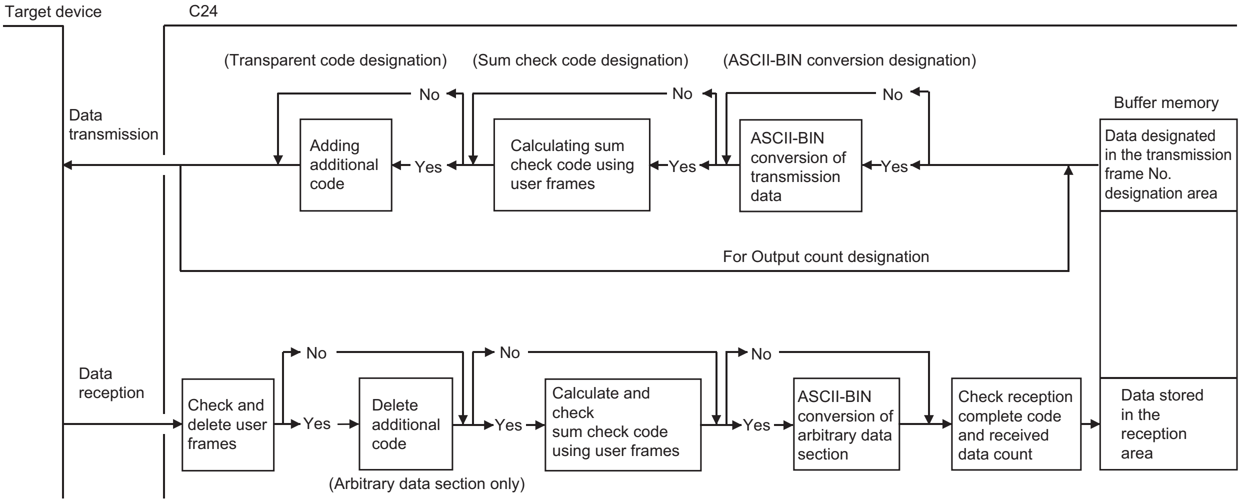

Target device
(Communicating in an arbitrary format)
C24
Head data
Buffer memory (transmission area/reception area)
Arbitrary data
Send
Receive
1 2 0 2 3 DLE A B C
31H 32H 30H 32H 33H 10H 41H 42H 43H
Delete Addition
Additional code
Conversion
(H) (L)
(0002H)
00H , 02H
(0212H)
02H , 12H
(BC3AH)
BCH , 3AH
Transmission data count/Receive data count
Transmission data/ receive data
(The data count is in word units)
Transparent code
(For 41H)
Additional code
(For 10H)

■Data reception using user frames

- C24 checks the reception of the user frame (start frame and final frame).
- C24 deletes (removes) the additional code designation data from an arbitrary data portion when a transparent code for receiving is specified.
- C24 calculates the sum check code when a sum check code is designated in a user frame (final frame).
- C24 stores the arbitrary data portion in the receive area on the buffer memory.
  When ASCII-BIN conversion is designated, C24 converts it to binary code data and stores in the buffer memory.
- C24 issues a reception data read request to the CPU module when the receive end code/data equivalent to the receive end data quantity is stored during reception of an arbitrary data portion, or when a user frame (final frame) is received.

■Transmitting data using user frames

- C24 transmits data to be sent (an arbitrary data portion of a transmission message), which is designated by the user frame or CPU module, in the order specified by the user.
- When ASCII-BIN conversion is designated, C24 converts the data in the applicable range to ASCII code data and transmits it.
  In addition, it adds the additional code data immediately before the transparent code/additional code data when transmitting the data in the applicable range, and transmits it when a transparent code for sending is specified.
- C24 notifies the CPU module of the transmission normal (abnormal) completion once transmission for the send data quantity has been completed.


Target device
C24
(Transparent code designation)
(Sum check code designation)
(ASCII-BIN conversion designation)
Data transmission
No
No
No
Buffer memory
Adding additional code
Yes
Calculating sum check code using user frames
Yes
ASCII-BIN conversion of transmission data
Yes
Data designated in the transmission frame No. designation area
For Output count designation
Data reception
No
No
No
Check and delete user frames
Yes
Delete additional code
Yes
Calculate and check sum check code using user frames
Yes
ASCII-BIN conversion of arbitrary data section
Check reception complete code and received data count
Data stored in the reception area
(Arbitrary data section only)

The figure above shows how C24 processes the data to be sent/received when the data sending/receiving function using user frame, the ASCII-BIN conversion function, and/or the data send/receive function by transparent code designation are enabled and disabled.

Use this information as a reference to select the method to send/receive data for data communication with a target device.

18.4 Data Communication Using Nonprocedural Protocol

This section shows examples of data communication using the nonprocedural protocol when the settings/registrations are made with Engineering tool.

Parameter settings

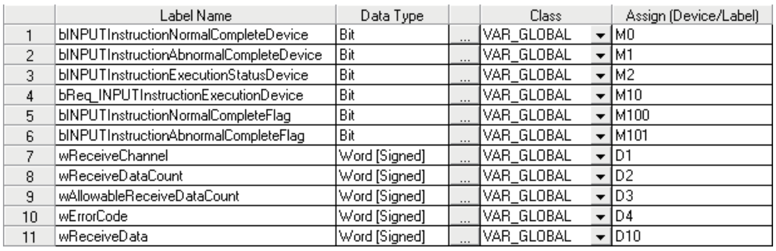
For details for each setting item, refer to "Parameter". (☞ Page 454 PARAMETER SETTING)

Setting item					Setting content	
					CH1	CH2
Basic settings	Various control specification	Test mode setting			—	CH2 is not used
		Communication speed setting			19200 bps	
		Communication protocol setting			Nonprocedural protocol	
		Transmission setting	Operation setting		Set according to the target device	
			Data bit			
			Parity bit			
			Odd/even parity			
			Stop bit			
			Sum check code			
			Online change		Enable	
			Setting change		Enable	
		Station number setting (CH1, 2 in common: 0 to 31)			—	
		Transparent code designation	Transparent code for sending specification (1st)	Transparent code	02H (STX)	
				Additional code	10H (DLE)	
			Transparent code for sending specification (2nd)	Transparent code	03H (ETX)	
				Additional code	10H (DLE)	
			Transparent code for receiving	Transparent code	02H (STX)	
				Additional code	10H (DLE)	
		Receiving end specification	Receive end data quantity designation		6	
			Receive end code designation		0D0A	

Example of data reception

Program example

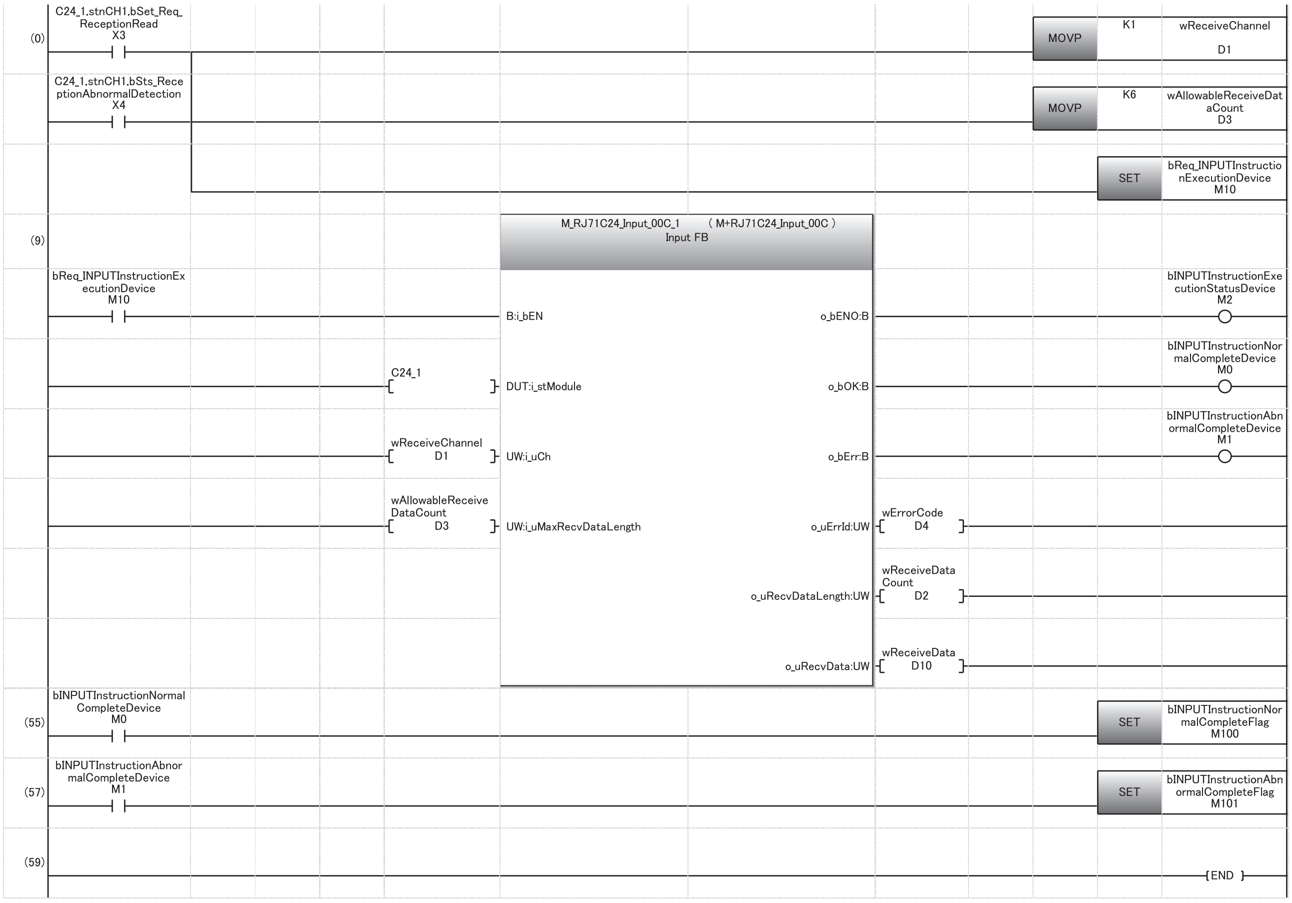
The following shows an example of storing received data in the data register.

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection	CH1 Reception abnormal detection	X4
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_Input	Reads the received data.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bINPUTInstructionNormalCompleteDevice	Bit	...	VAR_GLOBAL	M0
2	bINPUTInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
3	bINPUTInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M2
4	bReq_INPUTInstructionExecutionDevice	Bit	...	VAR_GLOBAL	M10
5	bINPUTInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
6	bINPUTInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101
7	wReceiveChannel	Word [Signed]	...	VAR_GLOBAL	D1
8	wReceiveDataCount	Word [Signed]	...	VAR_GLOBAL	D2
9	wAllowableReceiveDataCount	Word [Signed]	...	VAR_GLOBAL	D3
10	wErrorCode	Word [Signed]	...	VAR_GLOBAL	D4
11	wReceiveData	Word [Signed]	...	VAR_GLOBAL	D10

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
C24_1.stnCH1.bSet_Req_ReceptionRead
X3
MOVP
K1
wReceiveChannel
D1
C24_1.stnCH1.bSts_ReceptionAbnormalDetection
X4
MOVP
K6
wAllowableReceiveDataCount
D3
SET
bReq_INPUTInstructionExecutionDevice
M10
(9)
M_RJ71C24_Input_00C_1 (M+RJ71C24_Input_00C)
Input FB
bReq_INPUTInstructionExecutionDevice
M10
B:i_bEN
o_bENO:B
bINPUTInstructionExecutionStatusDevice
M2
C24_1
DUT:i_stModule
o_bOK:B
bINPUTInstructionNormalCompleteDevice
M0
wReceiveChannel
D1
UW:i_uCh
o_bErr:B
bINPUTInstructionAbnormalCompleteDevice
M1
wAllowableReceiveDataCount
D3
UW:i_uMaxRecvDataLength
o_uErrId:UW
wErrorCode
D4
o_uRecvDataLength:UW
wReceiveDataCount
D2
o_uRecvData:UW
wReceiveData
D10
(55)
bINPUTInstructionNormalCompleteDevice
M0
SET
bINPUTInstructionNormalCompleteFlag
M100
(57)
bINPUTInstructionAbnormalCompleteDevice
M1
SET
bINPUTInstructionAbnormalCompleteFlag
M101
(59)
END

(0) Set the reception channel to the CH1 side.
Set the allowable number of data to be received.
Set the reception direction.

(55) Normal completion

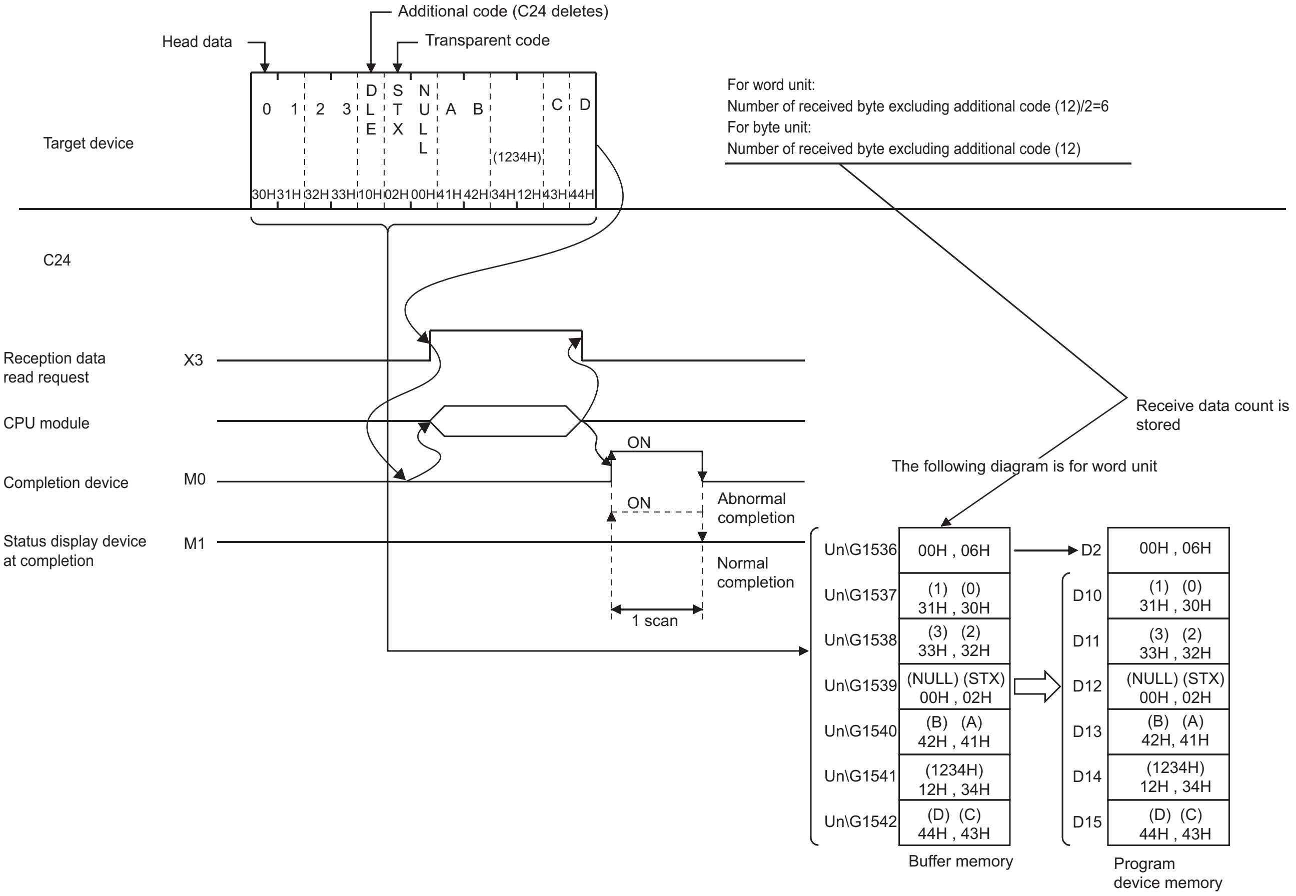
(57) Abnormal completion

Reception example based on the receive end data quantity

When C24 receives data from the target device when the transparent code designation is enabled

Ex.

Transparent code designation: Additional code: 10H (DLE), Transparent code: 02H (STX), Receive end data quantity designation: 6-word or 12-word


Additional code (C24 deletes)
Head data
Transparent code
Target device
0 1 2 3 DLE STX NULL A B (1234H) C D
30H 31H 32H 33H 10H 02H 00H 41H 42H 34H 12H 43H 44H
For word unit:
Number of received byte excluding additional code (12)/2=6
For byte unit:
Number of received byte excluding additional code (12)
C24
Reception data read request
X3
CPU module
Completion device
M0
Status display device at completion
M1
ON
ON
Abnormal completion
Normal completion
1 scan
Receive data count is stored
The following diagram is for word unit
Un\G1536 00H , 06H
Un\G1537 (1) (0) 31H , 30H
Un\G1538 (3) (2) 33H , 32H
Un\G1539 (NULL) (STX) 00H , 02H
Un\G1540 (B) (A) 42H , 41H
Un\G1541 (1234H) 12H , 34H
Un\G1542 (D) (C) 44H , 43H
Buffer memory
D2 00H , 06H
D10 (1) (0) 31H , 30H
D11 (3) (2) 33H , 32H
D12 (NULL) (STX) 00H , 02H
D13 (B) (A) 42H, 41H
D14 (1234H) 12H , 34H
D15 (D) (C) 44H , 43H
Program device memory

When the transparent code for receiving specification is enabled, the codes for receivable data and the codes for the received data which is stored is stored in the receive area on the buffer memory are as shown below.

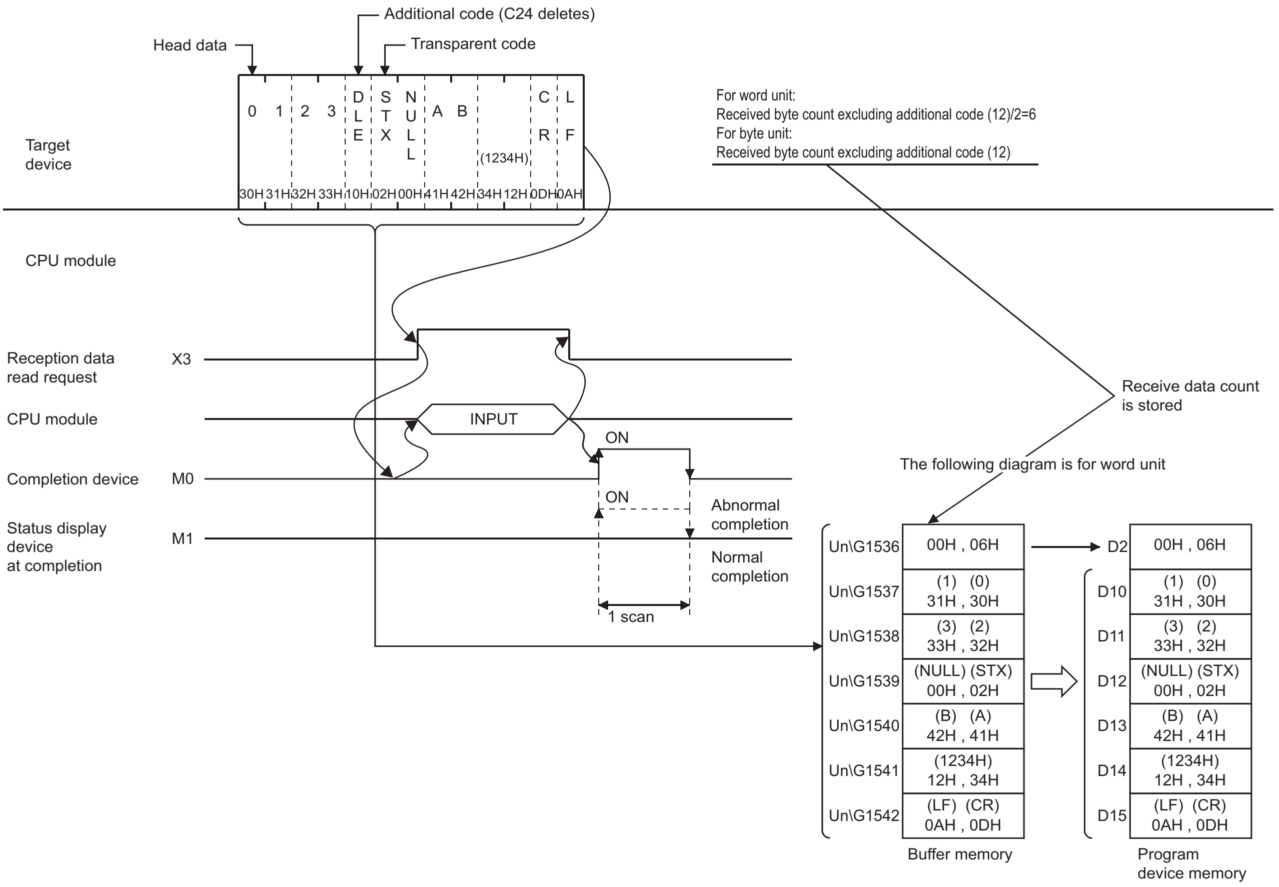
Type		Receivable codes	Codes stored in the receive area
Transparent code for receiving specification portion	Additional code	01H to FFH	(Delete)
	Transparent code	00H to FFH	00H to FFH
Arbitrary data portion (including end code)			

Operation example of reception by the receive end code

When C24 receives data from the target device when the transparent code designation is enabled

Ex.

Additional code: 10H (DLE), Transparent code: 02H (STX), End code: 0D0AH (CR+LF)



When the transparent code for receiving specification is enabled, the codes for receivable data and the codes for the received data which is stored in the receive area on the buffer memory are as shown below.

Type		Receivable codes	Codes stored in the receive area
Transparent code for receiving specification portion	Additional code	01H to FFH	(Delete)
	Transparent code	00H to FFH	00H to FFH
Arbitrary data portion (including end code)			

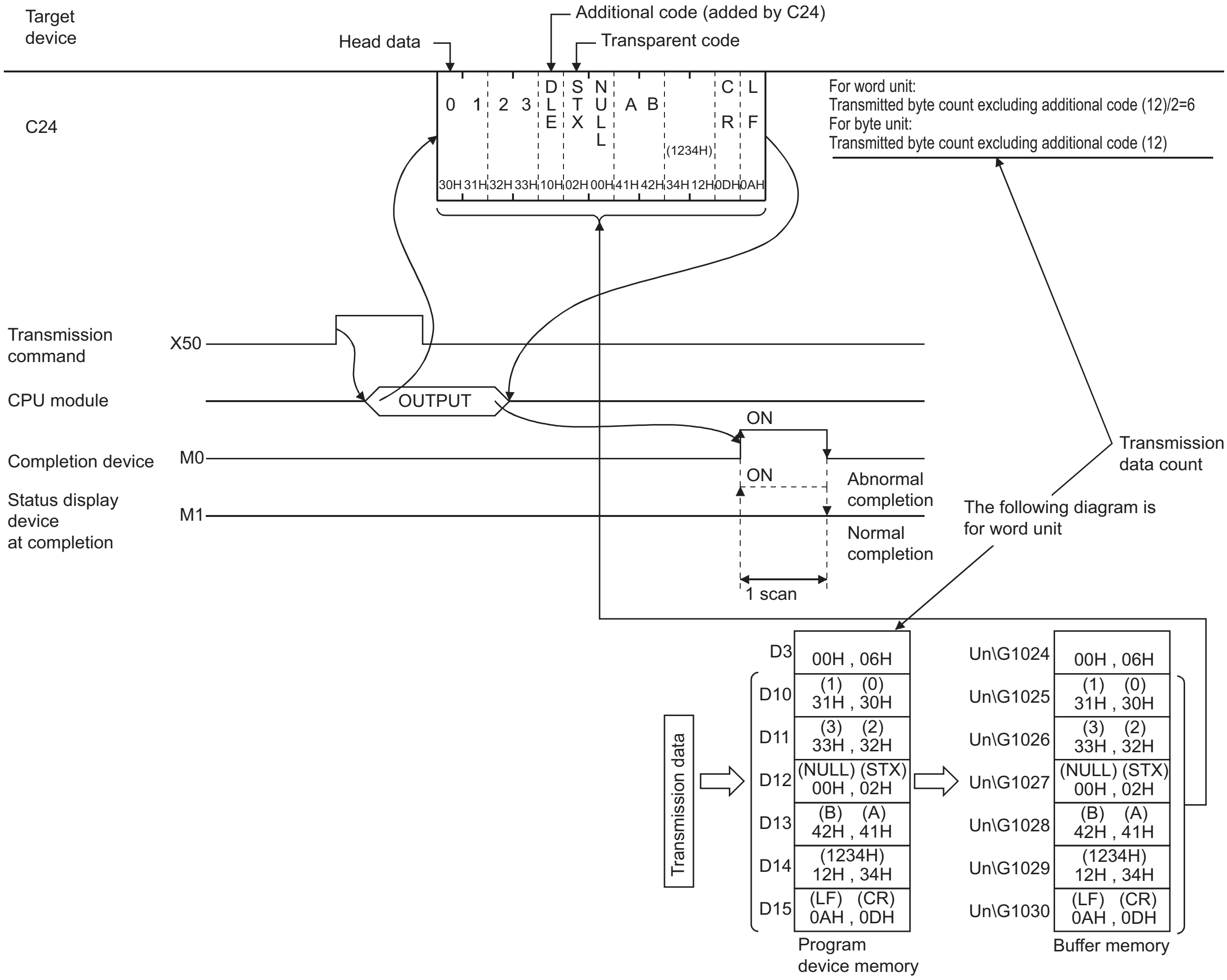
Example of data transmission

The following shows an example of data transmission.

Transmission by transparent code for sending specification, 1st

Ex.

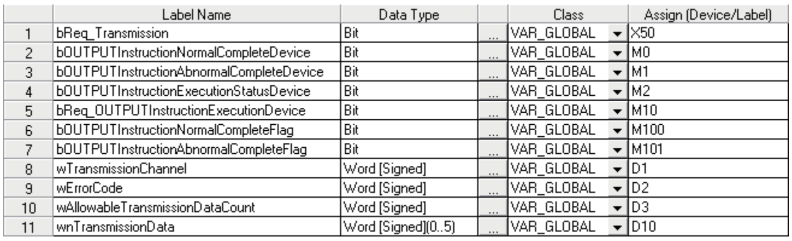
Transparent code designation: Additional code: 10H (DLE), Transparent code: 02H (STX)



	Program device memory		Buffer memory
D3	00H , 06H	Un\G1024	00H , 06H
D10	(1) (0) 31H , 30H	Un\G1025	(1) (0) 31H , 30H
D11	(3) (2) 33H , 32H	Un\G1026	(3) (2) 33H , 32H
D12	(NULL) (STX) 00H , 02H	Un\G1027	(NULL) (STX) 00H , 02H
D13	(B) (A) 42H , 41H	Un\G1028	(B) (A) 42H , 41H
D14	(1234H) 12H , 34H	Un\G1029	(1234H) 12H , 34H
D15	(LF) (CR) 0AH , 0DH	Un\G1030	(LF) (CR) 0AH , 0DH

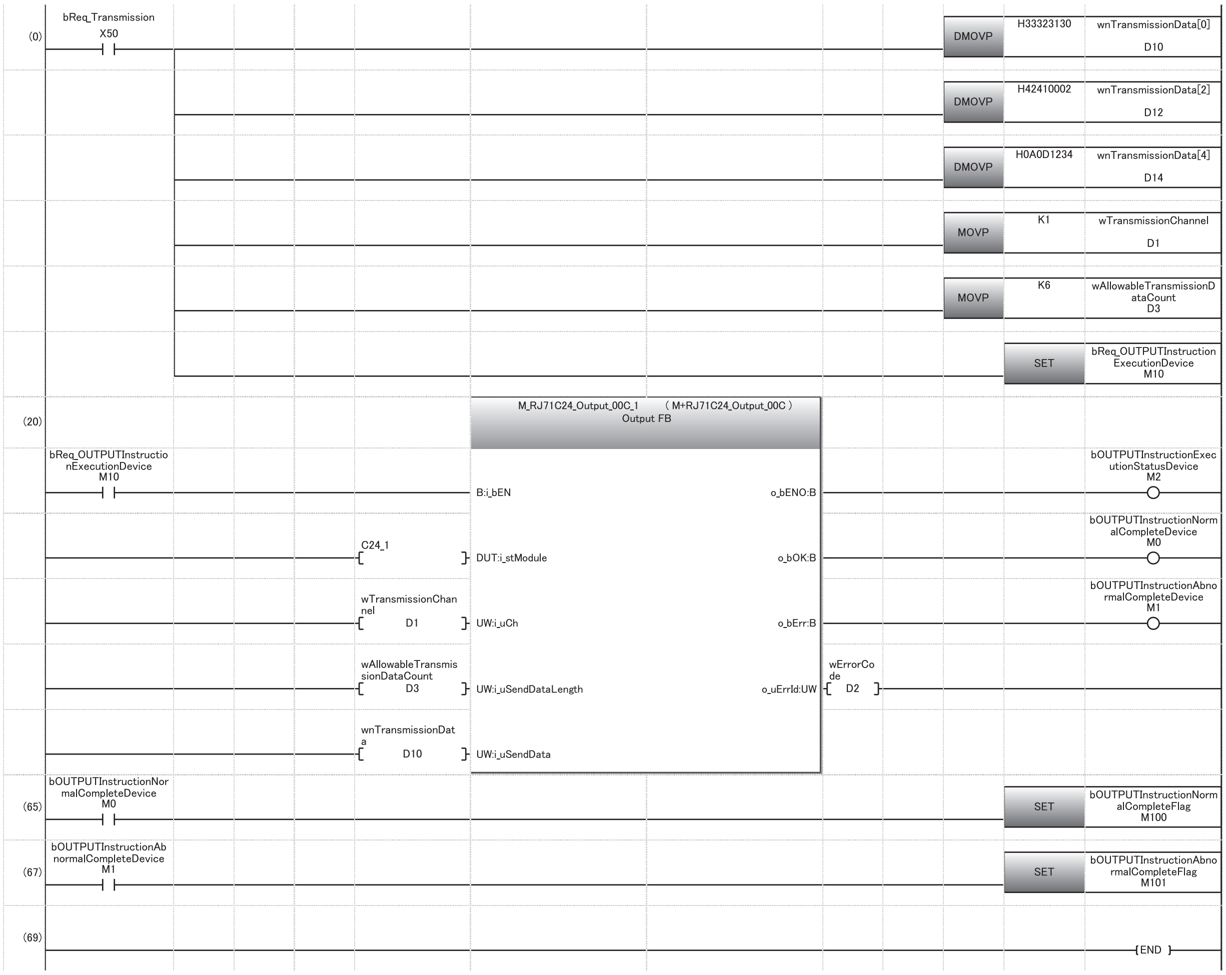
■Program example

When C24 transmits data to the target device by transparent code designation (1st combination of the transparent code for sending specification is set)

<table>
<tr><th>Category</th><th>Label name/FB name</th><th>Description</th><th>Device</th></tr>
<tr><td>Module label</td><td>C24_1</td><td>Target module</td><td>—</td></tr>
<tr><td>Label to be defined</td><td colspan="3">Define the global label as shown below.

<table>
<tr><th></th><th>Label Name</th><th>Data Type</th><th></th><th>Class</th><th>Assign (Device/Label)</th></tr>
<tr><td>1</td><td>bReq_Transmission</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>X50</td></tr>
<tr><td>2</td><td>bOUTPUTInstructionNormalCompleteDevice</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M0</td></tr>
<tr><td>3</td><td>bOUTPUTInstructionAbnormalCompleteDevice</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M1</td></tr>
<tr><td>4</td><td>bOUTPUTInstructionExecutionStatusDevice</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M2</td></tr>
<tr><td>5</td><td>bReq_OUTPUTInstructionExecutionDevice</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M10</td></tr>
<tr><td>6</td><td>bOUTPUTInstructionNormalCompleteFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M100</td></tr>
<tr><td>7</td><td>bOUTPUTInstructionAbnormalCompleteFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M101</td></tr>
<tr><td>8</td><td>wTransmissionChannel</td><td>Word [Signed]</td><td>...</td><td>VAR_GLOBAL</td><td>D1</td></tr>
<tr><td>9</td><td>wErrorCode</td><td>Word [Signed]</td><td>...</td><td>VAR_GLOBAL</td><td>D2</td></tr>
<tr><td>10</td><td>wAllowableTransmissionDataCount</td><td>Word [Signed]</td><td>...</td><td>VAR_GLOBAL</td><td>D3</td></tr>
<tr><td>11</td><td>wnTransmissionData</td><td>Word [Signed](0..5)</td><td>...</td><td>VAR_GLOBAL</td><td>D10</td></tr>
</table></td></tr>
<tr><td>FB</td><td>M+RJ71C24_Output</td><td>Sends data for specified data points.</td><td>—</td></tr>
</table>

For information on the module FB to be used, refer to the following manual.

MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
bReq_Transmission
X50
DMOVP
H33323130
wnTransmissionData[0]
D10
DMOVP
H42410002
wnTransmissionData[2]
D12
DMOVP
H0A0D1234
wnTransmissionData[4]
D14
MOVP
K1
wTransmissionChannel
D1
MOVP
K6
wAllowableTransmissionDataCount
D3
SET
bReq_OUTPUTInstructionExecutionDevice
M10
(20)
M_RJ71C24_Output_00C_1 (M+RJ71C24_Output_00C)
Output FB
bReq_OUTPUTInstructionExecutionDevice
M10
B:i_bEN
o_bENO:B
bOUTPUTInstructionExecutionStatusDevice
M2
C24_1
DUT:i_stModule
o_bOK:B
bOUTPUTInstructionNormalCompleteDevice
M0
wTransmissionChannel
D1
UW:i_uCh
o_bErr:B
bOUTPUTInstructionAbnormalCompleteDevice
M1
wAllowableTransmissionDataCount
D3
UW:i_uSendDataLength
o_uErrId:UW
wErrorCode
D2
wnTransmissionData
D10
UW:i_uSendData
(65)
bOUTPUTInstructionNormalCompleteDevice
M0
SET
bOUTPUTInstructionNormalCompleteFlag
M100
(67)
bOUTPUTInstructionAbnormalCompleteDevice
M1
SET
bOUTPUTInstructionAbnormalCompleteFlag
M101
(69)
END

(0) Set the data to be transmitted in D10 to D15.
Set the transmission channel to CH1.
Set the send data quantity to 6.
Execute transmission

(65) Normal completion

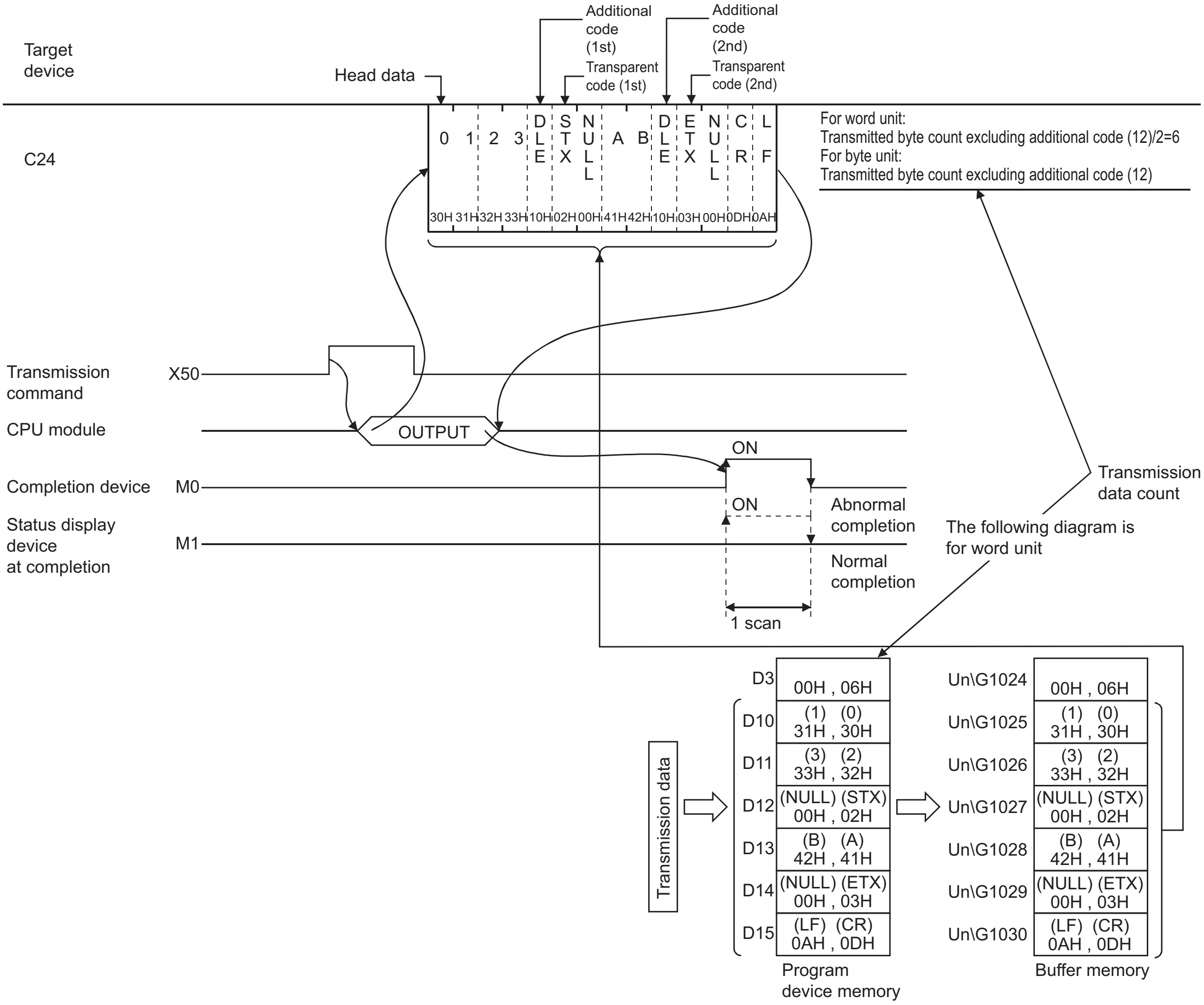
(67) Abnormal completion

Transmission by 1st/2nd combinations of transparent code for sending specification

Ex.

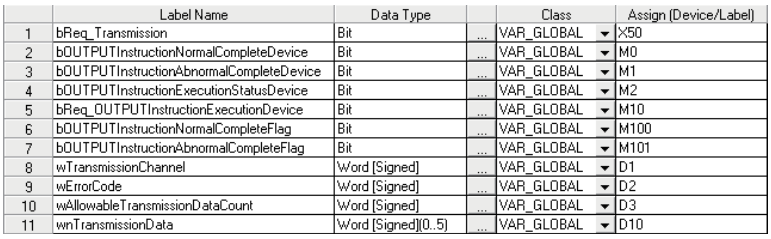
Transparent code designation (1st): Additional code: 10H (DLE),Transparent code: 02H (STX)

Transparent code designation (2nd): Additional code: 10H (DLE),Transparent code: 03H (ETX)



■Program example

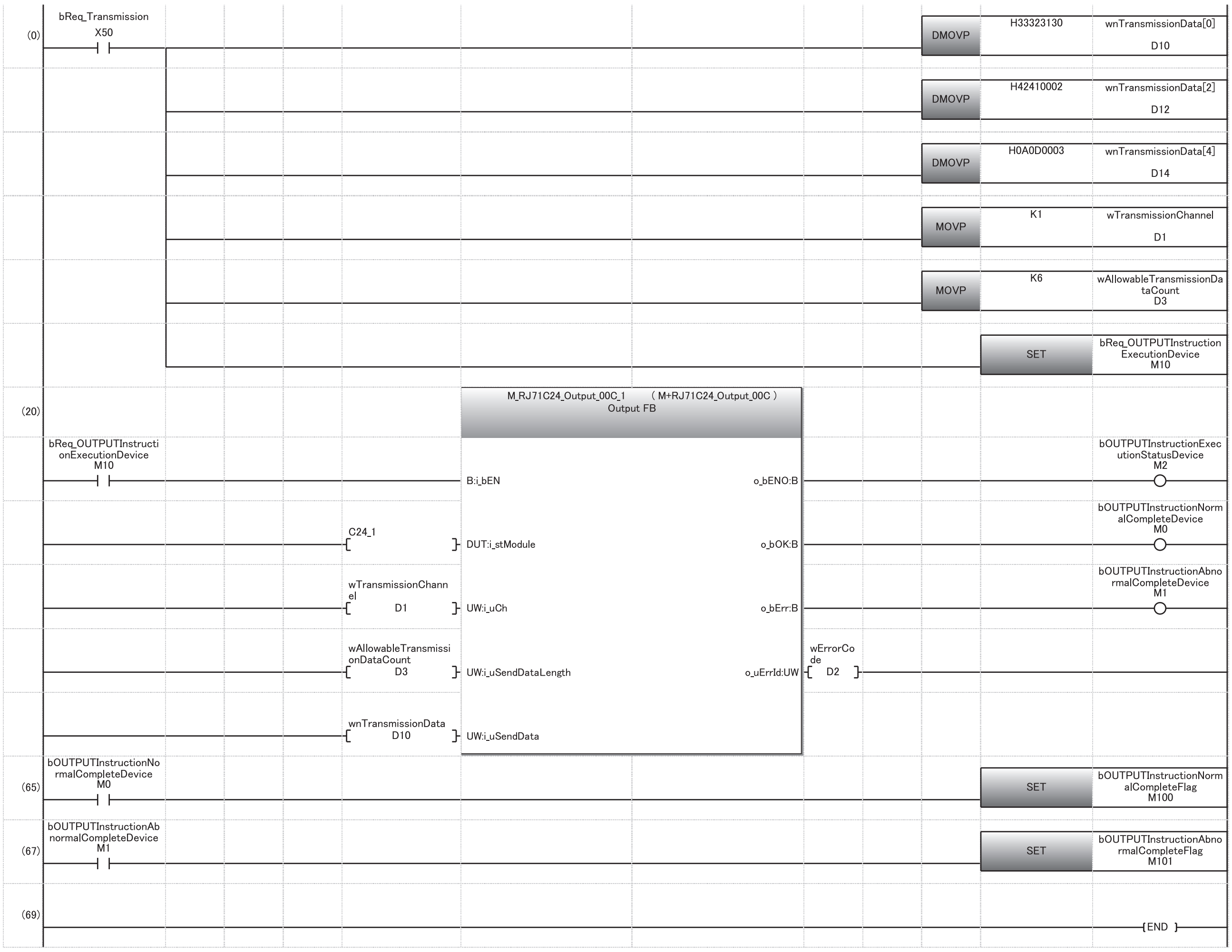
When C24 transmits data to the target device by transparent code designation (2nd combination of transparent code for sending specification is set.)

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_Output	Sends data for specified data points.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_Transmission	Bit	...	VAR_GLOBAL	X50
2	bOUTPUTInstructionNormalCompleteDevice	Bit	...	VAR_GLOBAL	M0
3	bOUTPUTInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
4	bOUTPUTInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M2
5	bReq_OUTPUTInstructionExecutionDevice	Bit	...	VAR_GLOBAL	M10
6	bOUTPUTInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
7	bOUTPUTInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101
8	wTransmissionChannel	Word [Signed]	...	VAR_GLOBAL	D1
9	wErrorCode	Word [Signed]	...	VAR_GLOBAL	D2
10	wAllowableTransmissionDataCount	Word [Signed]	...	VAR_GLOBAL	D3
11	wnTransmissionData	Word [Signed](0..5)	...	VAR_GLOBAL	D10

For information on the module FB to be used, refer to the following manual.

MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
bReq_Transmission
X50
DMOVP
H33323130
wnTransmissionData[0]
D10
DMOVP
H42410002
wnTransmissionData[2]
D12
DMOVP
H0A0D0003
wnTransmissionData[4]
D14
MOVP
K1
wTransmissionChannel
D1
MOVP
K6
wAllowableTransmissionDataCount
D3
SET
bReq_OUTPUTInstructionExecutionDevice
M10
(20)
M_RJ71C24_Output_00C_1 (M+RJ71C24_Output_00C)
Output FB
bReq_OUTPUTInstructionExecutionDevice
M10
B:i_bEN
o_bENO:B
bOUTPUTInstructionExecutionStatusDevice
M2
C24_1
DUT:i_stModule
o_bOK:B
bOUTPUTInstructionNormalCompleteDevice
M0
wTransmissionChannel
D1
UW:i_uCh
o_bErr:B
bOUTPUTInstructionAbnormalCompleteDevice
M1
wAllowableTransmissionDataCount
D3
UW:i_uSendDataLength
o_uErrId:UW
wErrorCode
D2
wnTransmissionData
D10
UW:i_uSendData
(65)
bOUTPUTInstructionNormalCompleteDevice
M0
SET
bOUTPUTInstructionNormalCompleteFlag
M100
(67)
bOUTPUTInstructionAbnormalCompleteDevice
M1
SET
bOUTPUTInstructionAbnormalCompleteFlag
M101
(69)
END

(0) Set the data to be transmitted in D10 to D15.
Set the transmission channel to the CH1 side.
Set the send data quantity to 6.
Set the transmission direction.

(65) Normal completion

(67) Abnormal completion

18.5 Handling Transparent and Additional Codes for Bidirectional Protocol Data Communication

This section explains how to handle the transparent codes and additional codes during data communication using the bidirectional protocol.

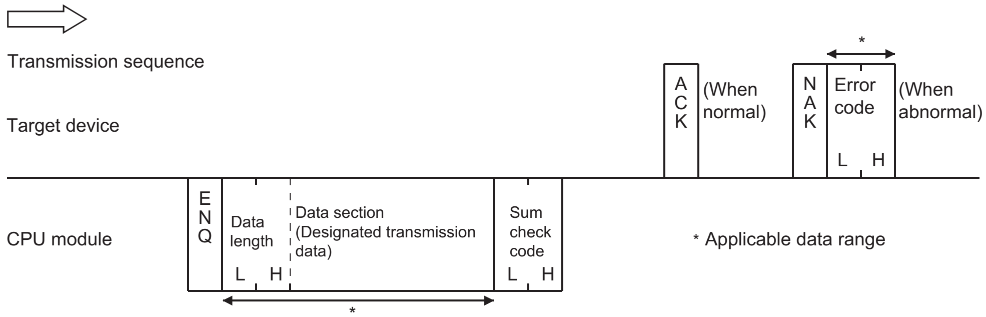
Adding/deleting additional code designation code

The data designated by the additional code is added to/deleted from the data to be transmitted or received.

Data range of transmit/receive processing

The range of transmitted/received data to which processing to transmit/receive transparent codes and additional codes is performed includes data length, data portion, and error code of a message.
Head code (ENQ, ACK, NAK) and sum check code of each message are not included.

Ex.



C24 performs the following processing during data transmission/reception.

■At the time of request data reception

When an additional code set for reception is detected, C24 removes the additional code data and perform data reception processing by treating the immediately succeeding 1-byte data as a part f the received data during the reception of a request data.

■At the time of response message transmission

When transparent code/additional code data set for transmission is detected, C24 transmits data by adding the additional code designation data immediately before it during the transmission of the response message to data reception.

■At the time of request data transmission

When transparent code/additional code data set for transmission is detected, C24 transmits data by adding the additional code designation data immediately before it during data transmission.

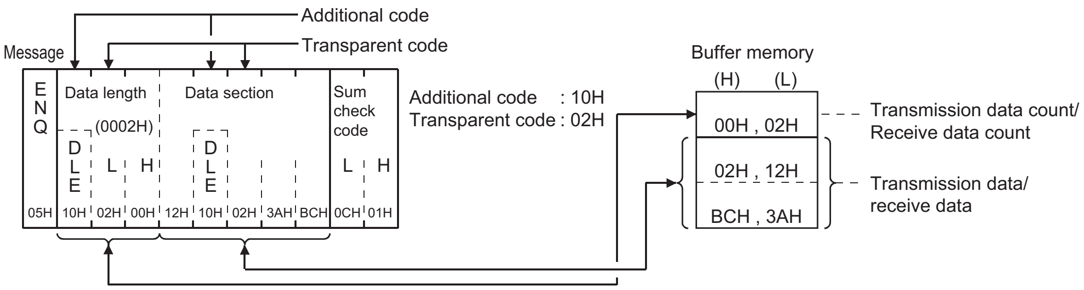
■At the time of response message reception

When an additional code set for reception is detected, C24 removes the additional code data and perform data reception processing by treating the immediately succeeding 1-byte data as a part of the received data during the reception of the response message to data transmission.

Precautions

The additional code data to be added/deleted is not included in the data length, nor will it be subject for sum check code.

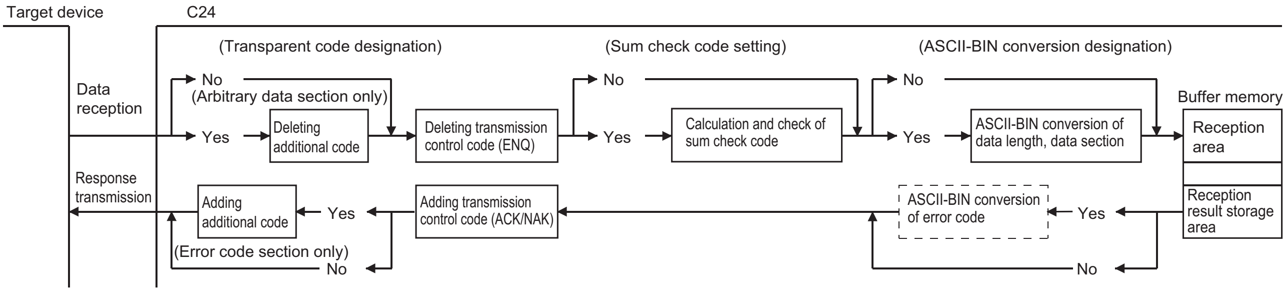
Example of request data transmission when ASCII-BIN conversion is disabled



Processing order for transmitting/receiving data

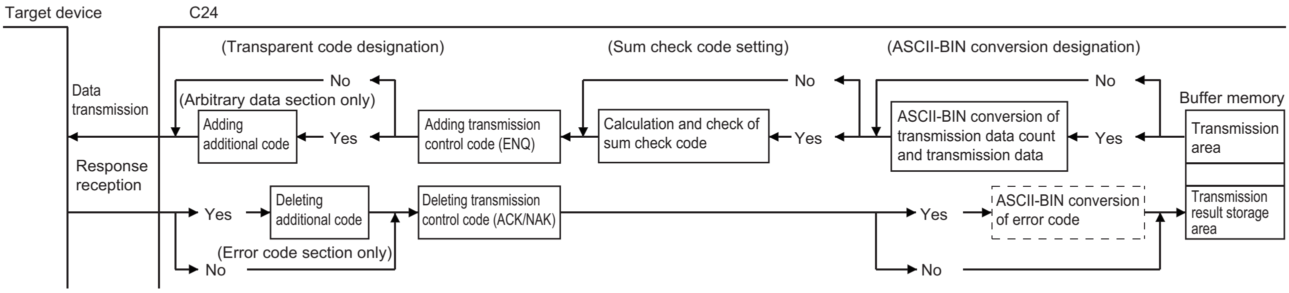
■At the time of reception

- C24 deletes (removes) the additional code designation data from an arbitrary data portion when a transparent code for receiving is specified.
- C24 stores the arbitrary data portion in the receive area on the buffer memory.
  When ASCII-BIN conversion is designated, C24 converts it to binary code data and stores in the buffer memory.
- When the data portion for the designated data length is received, reading the received data is requested of the CPU module.
  If the sum check code is set to 'Yes' in the transmission setting via an Engineering tool parameter settings, reading the received data is requested of the CPU module upon reception of the sum check code.


Target device
C24
(Transparent code designation)
(Sum check code setting)
(ASCII-BIN conversion designation)
Data reception
No
(Arbitrary data section only)
Yes
Deleting additional code
Deleting transmission control code (ENQ)
No
Yes
Calculation and check of sum check code
No
Yes
ASCII-BIN conversion of data length, data section
Buffer memory
Reception area
Response transmission
Adding additional code
Yes
Adding transmission control code (ACK/NAK)
ASCII-BIN conversion of error code
Yes
Reception result storage area
(Error code section only)
No
No

■At the time of transmission

- C24 adds the transmission control code to the data to be sent (arbitrary data portion of a transmission message), which is designated by the CPU module, and transmits it.
  When ASCII-BIN conversion is designated, C24 converts it to ASCII code data at first and transmits it.
- When the sum check code is set to be enabled, C24 calculates the sum check code from the transmission message and add it to the transmission message.
- C24 adds the additional code data immediately before the transparent code/additional code data in respect to arbitrary data portion, and transmits it when a transparent code for sending is specified.


Target device
C24
(Transparent code designation)
(Sum check code setting)
(ASCII-BIN conversion designation)
Data transmission
No
(Arbitrary data section only)
Adding additional code
Yes
Adding transmission control code (ENQ)
Calculation and check of sum check code
No
Yes
ASCII-BIN conversion of transmission data count and transmission data
No
Yes
Buffer memory
Transmission area
Response reception
Yes
Deleting additional code
Deleting transmission control code (ACK/NAK)
Yes
ASCII-BIN conversion of error code
Transmission result storage area
(Error code section only)
No
No

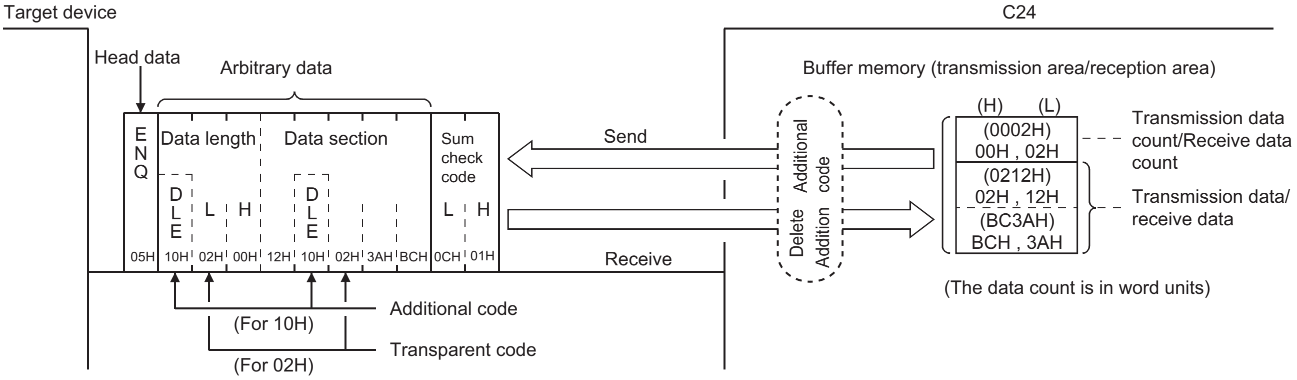
The figure above shows how C24 processes the data to be transmitted/received when the ASCII-BIN conversion function, and/or the data communication function by transparent code designation are enabled and disabled.

Use this information as a reference to select the method to send/receive data for data communication with a target device.

Ex.

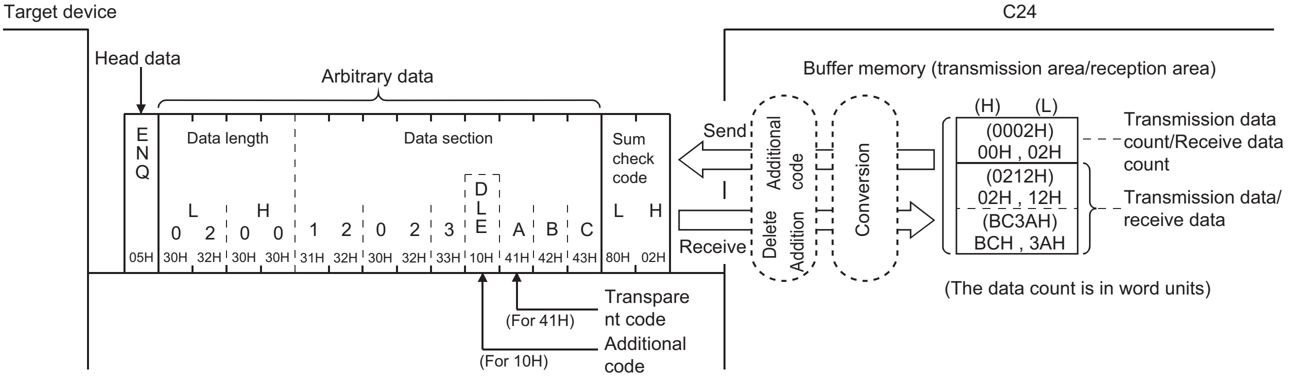
The following example shows the data array when data is transmitted and received. (The transmission/reception portion of the response message is omitted.)

When ASCII-BIN conversion is not performed



Ex.

When ASCII-BIN conversion is performed



18.6 Data Communication Using Bidirectional Protocol

This section shows examples of data communication using the bidirectional protocol when the settings/registrations are made with Engineering tool.

Parameter settings

For details for each setting item, refer to "Parameter". (☞ Page 454 PARAMETER SETTING)

Setting item				Setting content	
				CH1	CH2
Basic settings	Various control specification	Test mode setting		—	CH2 is not used
		Communication protocol setting		Bidirectional protocol	
		Communication speed setting		19200 bps	
		Transmission setting	Operation setting	Set according to the target device	
			Data bit		
			Parity bit		
			Odd/even parity		
			Stop bit		
			Sum check code		
			Online change	Enable	
			Setting change	Enable	
		Station number setting (CH1, 2 in common: 0 to 31)		—	
		Transparent code designation_Transparent code for sending specification (1st)	Transparent code	02H (STX)	
			Additional code	10H (DLE)	
		Transparent code designation_Transparent code for sending specification (2nd)	Transparent code	03H (ETX)	
			Additional code	10H (DLE)	
		Transparent code designation_Transparent code for receiving specification	Transparent code	02H (STX)	
			Additional code	10H (DLE)	

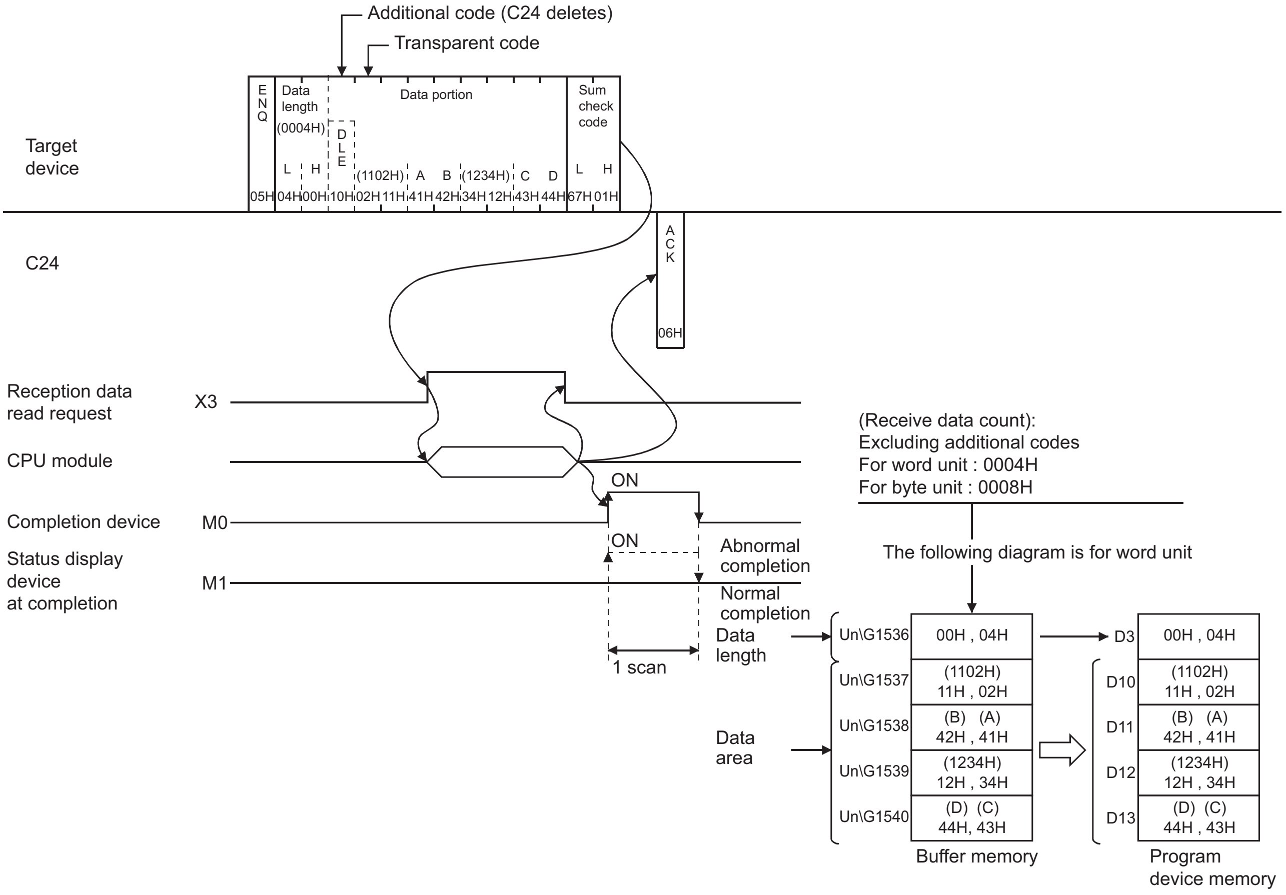
Example of data reception

The following example shows when receiving binary data including a transparent code from a target device.

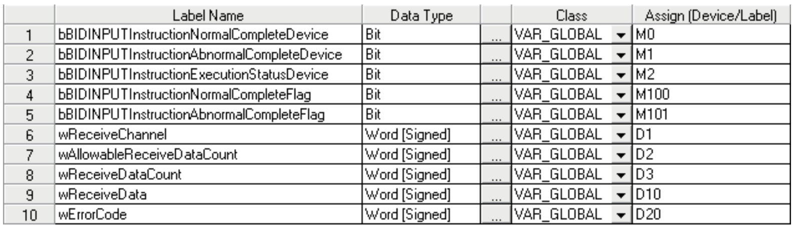
Transparent code for receiving specification is enabled

Ex.

Transparent code designation: Additional code: 10H (DLE), Transparent code: 02H (STX)



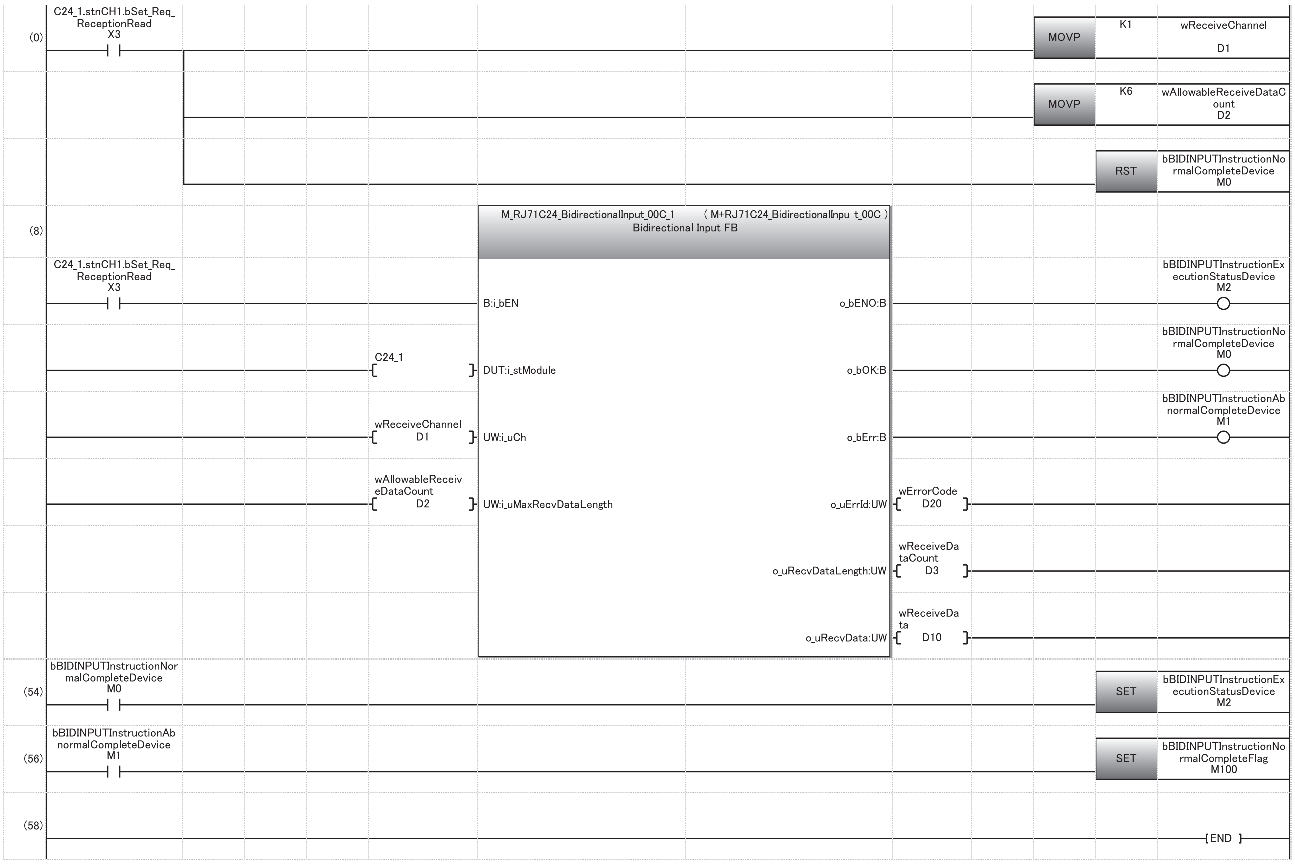
■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_BidirectionalInput	Reads the received data.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bBIDINPUTInstructionNormalCompleteDevice	Bit	...	VAR_GLOBAL	M0
2	bBIDINPUTInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
3	bBIDINPUTInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M2
4	bBIDINPUTInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
5	bBIDINPUTInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101
6	wReceiveChannel	Word [Signed]	...	VAR_GLOBAL	D1
7	wAllowableReceiveDataCount	Word [Signed]	...	VAR_GLOBAL	D2
8	wReceiveDataCount	Word [Signed]	...	VAR_GLOBAL	D3
9	wReceiveData	Word [Signed]	...	VAR_GLOBAL	D10
10	wErrorCode	Word [Signed]	...	VAR_GLOBAL	D20

For information on the module FB to be used, refer to the following manual.

MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
C24_1.stnCH1.bSet_Req_ReceptionRead X3
MOVP
K1
wReceiveChannel D1
MOVP
K6
wAllowableReceiveDataCount D2
RST
bBIDINPUTInstructionNormalCompleteDevice M0
(8)
M_RJ71C24_BidirectionalInput_00C_1 (M+RJ71C24_BidirectionalInpu t_00C)
Bidirectional Input FB
C24_1.stnCH1.bSet_Req_ReceptionRead X3
B:i_bEN
o_bENO:B
bBIDINPUTInstructionExecutionStatusDevice M2
C24_1
DUT:i_stModule
o_bOK:B
bBIDINPUTInstructionNormalCompleteDevice M0
wReceiveChannel D1
UW:i_uCh
o_bErr:B
bBIDINPUTInstructionAbnormalCompleteDevice M1
wAllowableReceiveDataCount D2
UW:i_uMaxRecvDataLength
o_uErrId:UW
wErrorCode D20
o_uRecvDataLength:UW
wReceiveDataCount D3
o_uRecvData:UW
wReceiveData D10
(54)
bBIDINPUTInstructionNormalCompleteDevice M0
SET
bBIDINPUTInstructionExecutionStatusDevice M2
(56)
bBIDINPUTInstructionAbnormalCompleteDevice M1
SET
bBIDINPUTInstructionNormalCompleteFlag M100
(58)
END

(0) Set the reception channel to CH1.
Set the allowable number of data to be received to 6.
Set the reception direction.

(8) Execute reception

(54) Normal completion

(56) Abnormal completion

When the transparent code for receiving specification is enabled, the codes for receivable arbitrary data portion and the codes for the received data which is stored in the receive area on the buffer memory are as shown below.

Type		Receivable codes	Codes stored in the receive area
Transparent code for receiving specification portion	Additional code	01H to FFH	(Delete)
	Transparent code	00H to FFH	00H to FFH
Data length, data portion			

When the data length unit is in bytes and the data length is an odd byte, 00H is stored in the upper byte of the last data storage position in the receive area.

Example of data transmission

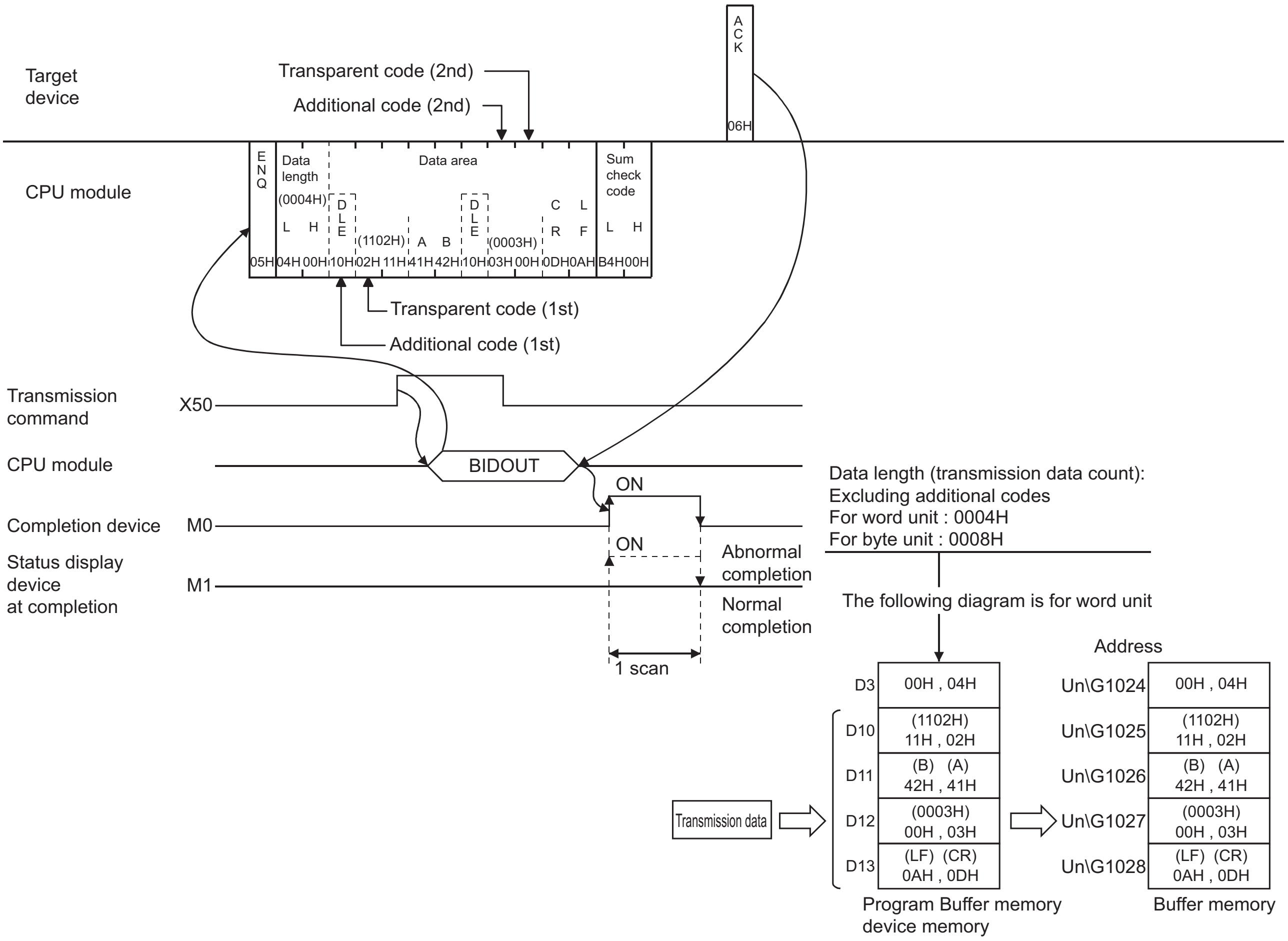
The following example shows when transmitting binary data including a transparent code to a target device.

Transparent code for sending specification is enabled

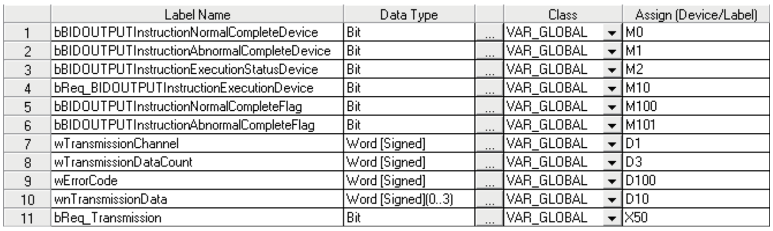
Ex.

Transparent code designation (1st): Additional code: 10H (DLE), Transparent code: 02H (STX)

Transparent code designation (2nd): Additional code: 10H (DLE), transparent code: 03H (ETX)



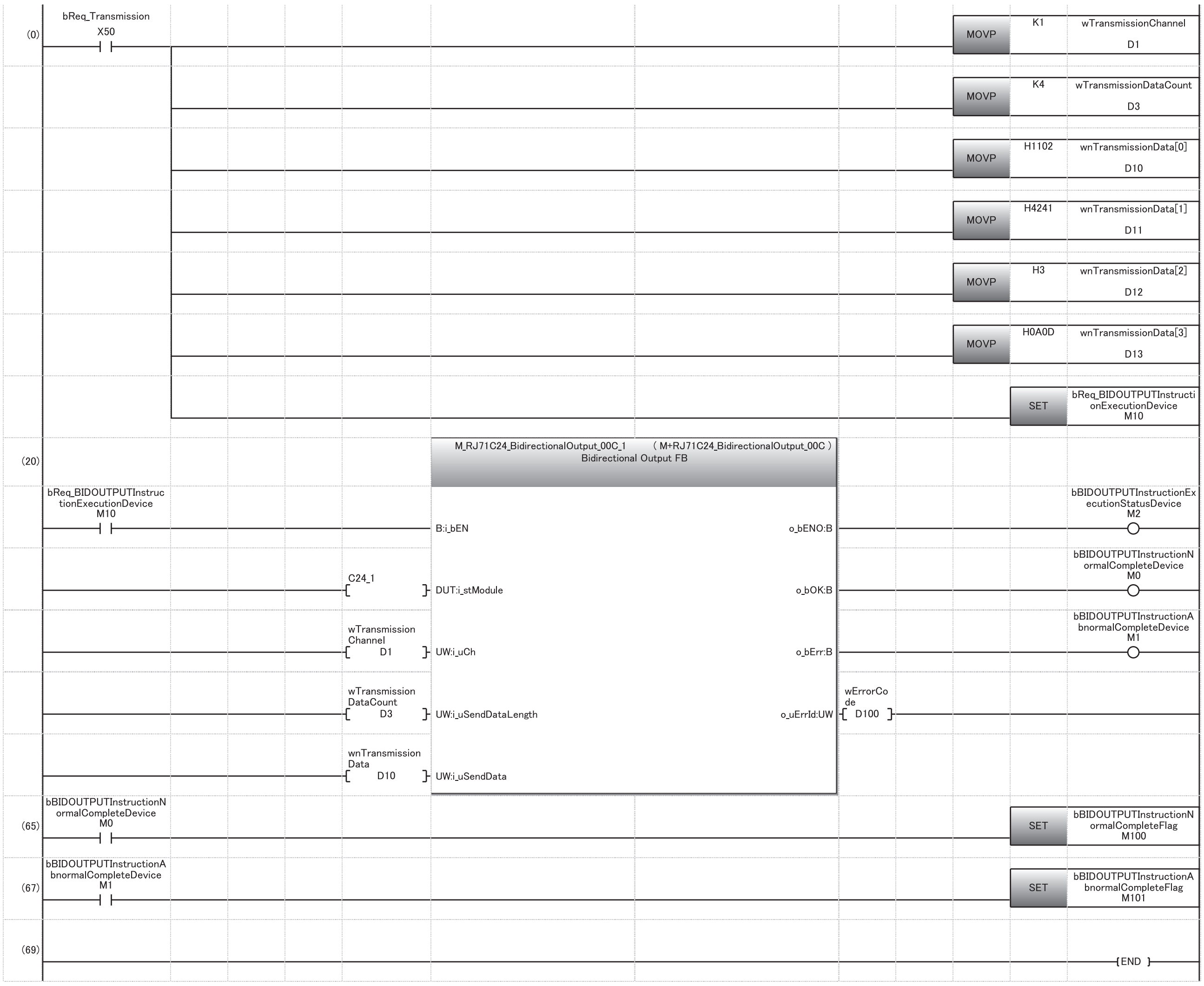
■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_BidirectionalOutput	Sends data for specified data points.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bBIDOUTPUTInstructionNormalCompleteDevice	Bit	...	VAR_GLOBAL	M0
2	bBIDOUTPUTInstructionAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
3	bBIDOUTPUTInstructionExecutionStatusDevice	Bit	...	VAR_GLOBAL	M2
4	bReq_BIDOUTPUTInstructionExecutionDevice	Bit	...	VAR_GLOBAL	M10
5	bBIDOUTPUTInstructionNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
6	bBIDOUTPUTInstructionAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101
7	wTransmissionChannel	Word [Signed]	...	VAR_GLOBAL	D1
8	wTransmissionDataCount	Word [Signed]	...	VAR_GLOBAL	D3
9	wErrorCode	Word [Signed]	...	VAR_GLOBAL	D100
10	wnTransmissionData	Word [Signed](0..3)	...	VAR_GLOBAL	D10
11	bReq_Transmission	Bit	...	VAR_GLOBAL	X50

For information on the module FB to be used, refer to the following manual.

MELSEC iQ-R Serial Communication Module Function Block Reference



(0) Set the transmission channel to CH1.
Set the send data quantity to 4.
Set the data to be transmitted in D10 to D13.
Set the transmission direction.

(20) Execute transmission.

(65) Normal completion

(67) Abnormal completion

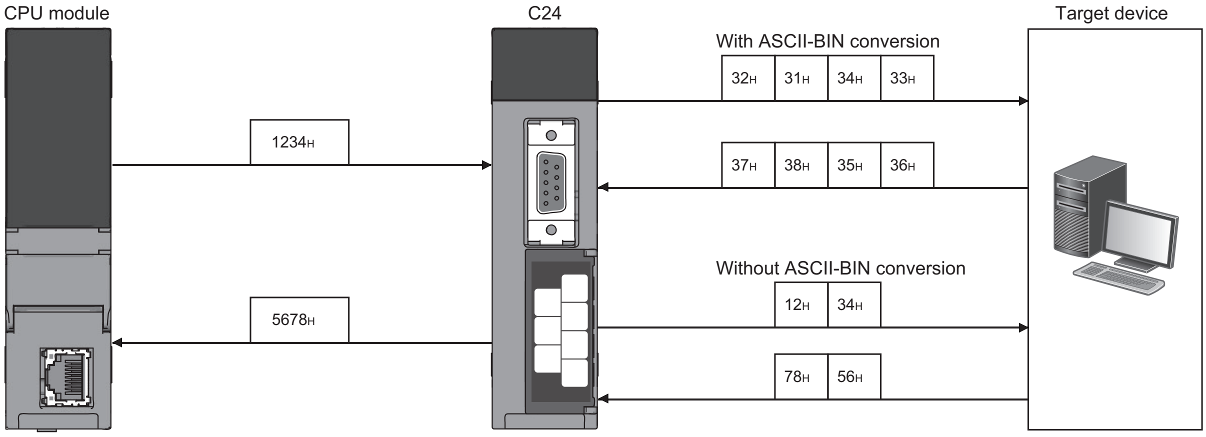
19 COMMUNICATING WITH ASCII CODE (ASCII-BIN CONVERSION)

This chapter explains the ASCII-binary conversion in order to transmit/receive data in ASCII code to/from a target device.

19.1 ASCII-Binary Conversion

ASCII-BIN conversion is the data conversion function that converts all data communicated between C24 and a target device to ASCII code data.

The ASCII-BIN conversion of data to be sent/received is performed by C24 according to the parameter settings.



19.2 Settings for ASCII-BIN Conversion

In order to convert data to be sent/received with the nonprocedural protocol and bidirectional protocol from binary code to ASCII code, it is necessary to configure the setting for C24 before performing data communication.

The settings for ASCII-BIN conversion are as shown below:

- The ASCII-BIN conversion settings can be configured for each interface.
- To enable ASCII-BIN conversion, set "ASCII-BIN conversion designation" to "ON".

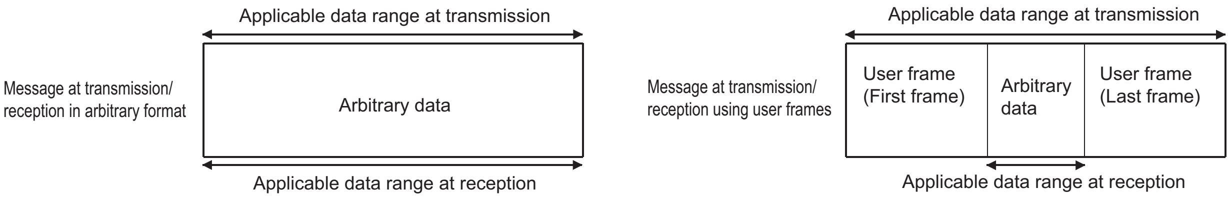
Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

19.3 ASCII-BIN Conversion for Data Communicated with Nonprocedural Protocol

This section explains the ASCII-BIN conversion of data to be communicated using the nonprocedural protocol.

Data range of transmit/receive processing

The range of data to be sent/received to which ASCII-BIN conversion is performed is as shown below:


Message at transmission/ reception in arbitrary format
Applicable data range at transmission
Arbitrary data
Applicable data range at reception
Message at transmission/ reception using user frames
Applicable data range at transmission
User frame (First frame)
Arbitrary data
User frame (Last frame)
Applicable data range at reception

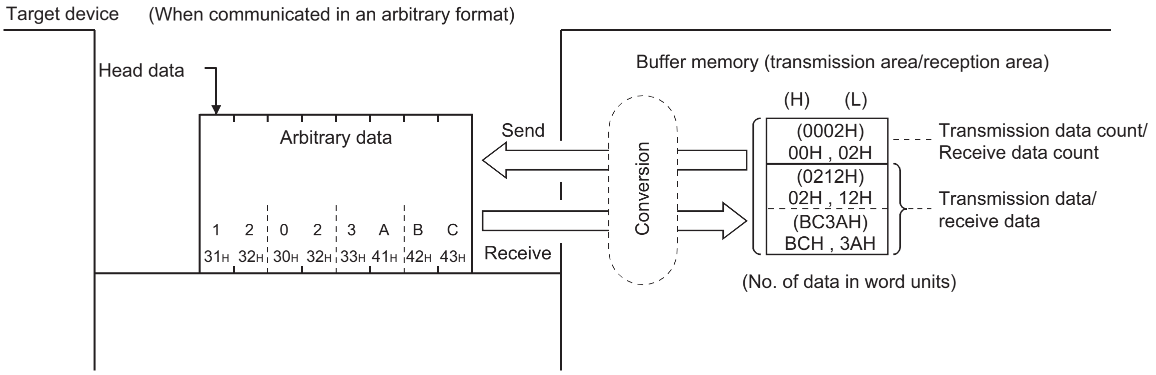
C24 performs the following processing when sending or receiving data.

■At the time of data reception

- C24 receives data of which data code is in the range of 30H to 39H ('0' to '9') and 41H to 46H ('A' to 'F') as the data within the above conversion applicable range.
- C24 regards given data portion as ASCII code, and converts it to binary code to store in the buffer memory.
- The user frame portion is received in the data format corresponding to the registration to C24.

■At the time of data transmission

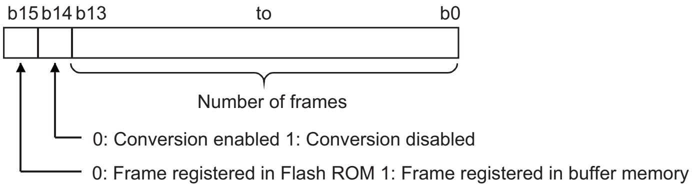
- C24 transmits data of which data code is in the range of 30H to 39H ('0' to '9') and 41H to 46H ('A' to 'F') as the data within the above conversion applicable range.
- C24 regards data designated by the CPU module (given data portion in a transmission message) and user frame portions as binary code data, and converts to ASCII code to transmit them.


Target device
(When communicated in an arbitrary format)
Head data
Arbitrary data
1 2 0 2 3 A B C
31H 32H 30H 32H 33H 41H 42H 43H
Send
Receive
Conversion
Buffer memory (transmission area/reception area)
(H) (L)
(0002H)
00H , 02H
(0212H)
02H , 12H
(BC3AH)
BCH , 3AH
Transmission data count/ Receive data count
Transmission data/ receive data
(No. of data in word units)

At the time of data transmission with the user frame, it is possible to transmit data without converting the data in any user frame portion or send area on the buffer memory even if "Convert" is designated in the ASCII-BIN conversion designation area.

When sending data without ASCII-BIN conversion, specify the user frame number by the method shown below.

- Specify the number obtained by adding 4000H to the registered user frame No.


b15
b14
b13
to
b0
Number of frames
0: Conversion enabled 1: Conversion disabled
0: Frame registered in Flash ROM 1: Frame registered in buffer memory

When sending data in the portions designated in 4001H to 44AFH, C000H to C01FH, the data is transmitted without conversion even when the following designations are enabled.

- Specifying an additional code by the transparent code for sending specification (☞ Page 358 TRANSPARENT CODES AND ADDITIONAL CODES)
- Designating "Auto delete enabled" in the NULL character automatic removal designation area (☞ Page 441 NULL Character Removal to Send Data using Nonprocedural Protocol)

User frame No. to be sent	Designated No. for the user frame when transmitting without ASCII-BIN conversion (4000H added)
1H to 3E7H (1 to 999)	4001H to 43E7H (16385 to 17383)
3E8H to 4AFH (1000 to 1199)	43E8H to 44AFH (17384 to 17583)
8000H to 801FH (-32768 to -32737)	C000H to C01FH (-16384 to -16353)

Processing order for data transmission/reception

The processing steps taken by C24 to transmit/receive data with ASCII-BIN conversion and transparent code designation enabled are explained in the section for transparent codes and additional codes. (☞ Page 360 Handling Transparent and Additional Codes for Nonprocedural Protocol Data Communication)

- Data transmission/reception in arbitrary formats
- Data transmission/reception using user frames

19.4 Data Communication Using Nonprocedural Protocol

This section shows examples of data communication using the nonprocedural protocol when the settings/registrations are made with Engineering tool.

Parameter settings

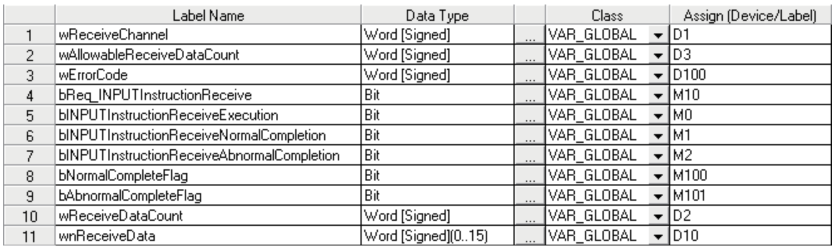
For details for each setting item, refer to the parameters. (☞ Page 454 PARAMETER SETTING)

Setting item				Setting content		Remarks
				CH1	CH2	
Basic settings	Various control specification	Test mode setting		—	CH2 is not used	—
		Communication protocol setting		Nonprocedural protocol		
		Communication speed setting		Set according to the target device		
		Transmission setting	Operation setting	Independent		
			Data bit	Set according to the target device		
			Parity bit			
			Odd/even parity			
			Stop bit			
			Sum check code			
			Online change			
			Setting change			
		Station number setting (CH1, 2 in common: 0 to 31)		—		
		Conversion designation	ASCII-BIN conversion designation	ON		
		Receiving end specification	Receive end data quantity designation	3		
			Receive end code designation	0009H		
Application settings	User frame specification	User frame specification for receiving	User frame use enable/disable designation	Use		Arbitrary registered data
			Start frame No. designation (1st)	03E8H		
			Start frame No. designation (2nd)	03E9H		
			Final frame No. designation (1st)	041BH		
			Final frame No. designation (2nd)	041BH		
		Send user frame designation	Output start pointer designation	0001H		1st
			Output quantity designation	0005H		5th
			Send frame No. designation (1st)	43F2H		Arbitrary registered data
			Send frame No. designation (2nd)	43F3H		
			Send frame No. designation (3rd)	C001H		
			Send frame No. designation (4th)	8000H		Buffer memory designation
			Send frame No. designation (5th)	441BH		Arbitrary registered data

Example of data reception

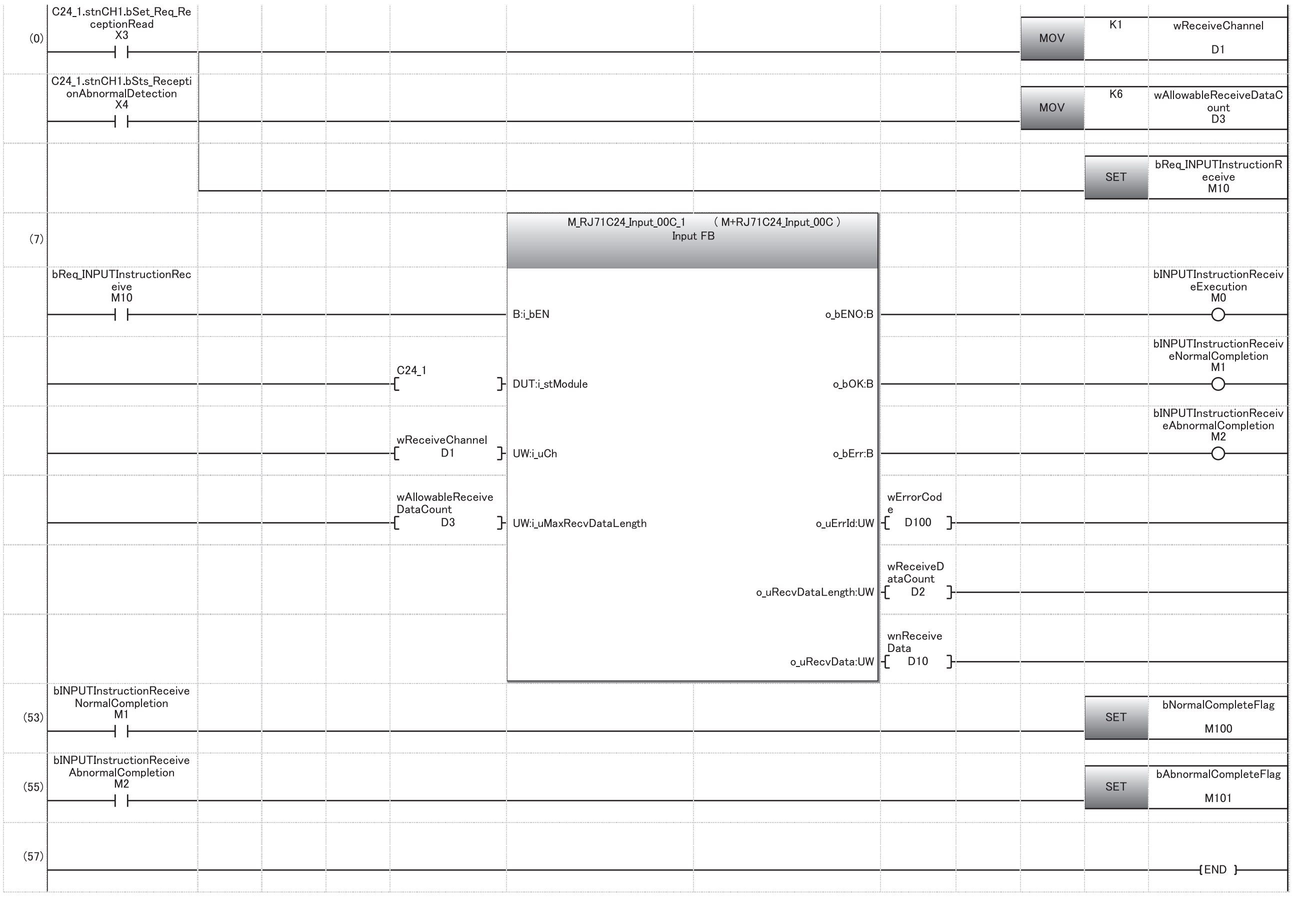
Program example

The following shows an example of data reception.

<table>
<tr><th>Category</th><th>Label name/FB name</th><th>Description</th><th>Device</th></tr>
<tr><td>Module label</td><td>C24_1</td><td>Target module</td><td>—</td></tr>
<tr><td></td><td>C24_1.stnCH1.bSet_Req_ReceptionRead</td><td>CH1 Reception data read request</td><td>X3</td></tr>
<tr><td></td><td>C24_1.stnCH1.bSts_ReceptionAbnormalDetection</td><td>CH1 Reception abnormal detection</td><td>X4</td></tr>
<tr><td>Label to be defined</td><td colspan="3">Define the global label as shown below.

<table>
<tr><th></th><th>Label Name</th><th>Data Type</th><th></th><th>Class</th><th>Assign (Device/Label)</th></tr>
<tr><td>1</td><td>wReceiveChannel</td><td>Word [Signed]</td><td>...</td><td>VAR_GLOBAL</td><td>D1</td></tr>
<tr><td>2</td><td>wAllowableReceiveDataCount</td><td>Word [Signed]</td><td>...</td><td>VAR_GLOBAL</td><td>D3</td></tr>
<tr><td>3</td><td>wErrorCode</td><td>Word [Signed]</td><td>...</td><td>VAR_GLOBAL</td><td>D100</td></tr>
<tr><td>4</td><td>bReq_INPUTInstructionReceive</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M10</td></tr>
<tr><td>5</td><td>bINPUTInstructionReceiveExecution</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M0</td></tr>
<tr><td>6</td><td>bINPUTInstructionReceiveNormalCompletion</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M1</td></tr>
<tr><td>7</td><td>bINPUTInstructionReceiveAbnormalCompletion</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M2</td></tr>
<tr><td>8</td><td>bNormalCompleteFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M100</td></tr>
<tr><td>9</td><td>bAbnormalCompleteFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M101</td></tr>
<tr><td>10</td><td>wReceiveDataCount</td><td>Word [Signed]</td><td>...</td><td>VAR_GLOBAL</td><td>D2</td></tr>
<tr><td>11</td><td>wnReceiveData</td><td>Word [Signed](0..15)</td><td>...</td><td>VAR_GLOBAL</td><td>D10</td></tr>
</table></td></tr>
<tr><td>FB</td><td>M+RJ71C24_Input</td><td>Reads the received data.</td><td>—</td></tr>
</table>

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
C24_1.stnCH1.bSet_Req_ReceptionRead X3
MOV K1 wReceiveChannel D1
C24_1.stnCH1.bSts_ReceptionAbnormalDetection X4
MOV K6 wAllowableReceiveDataCount D3
SET bReq_INPUTInstructionReceive M10
(7)
M_RJ71C24_Input_00C_1 (M+RJ71C24_Input_00C)
Input FB
bReq_INPUTInstructionReceive M10
B:i_bEN
o_bENO:B
bINPUTInstructionReceiveExecution M0
C24_1
DUT:i_stModule
o_bOK:B
bINPUTInstructionReceiveNormalCompletion M1
wReceiveChannel D1
UW:i_uCh
o_bErr:B
bINPUTInstructionReceiveAbnormalCompletion M2
wAllowableReceiveDataCount D3
UW:i_uMaxRecvDataLength
o_uErrId:UW
wErrorCode D100
o_uRecvDataLength:UW
wReceiveDataCount D2
o_uRecvData:UW
wnReceiveData D10
(53)
bINPUTInstructionReceiveNormalCompletion M1
SET bNormalCompleteFlag M100
(55)
bINPUTInstructionReceiveAbnormalCompletion M2
SET bAbnormalCompleteFlag M101
(57)
END

(0) Set the reception channel to CH1.
Set the allowable number of data to be received to 6.
Set the reception direction.

(7) Execute reception.

(53) Normal completion

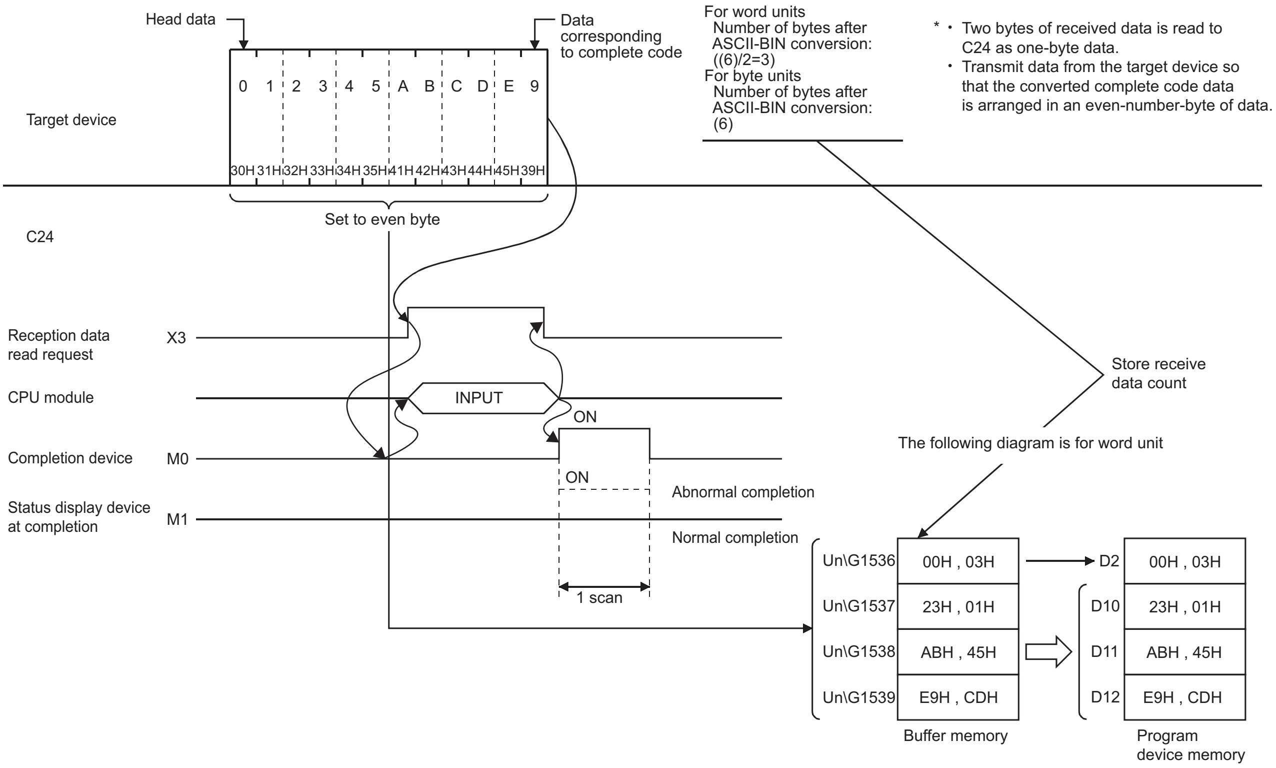
(55) Abnormal completion

Operation example of reception by the receive end code

When C24 receives data from the target device when ASCII-BIN conversion is enabled (Reception by the receive end code)

Ex.

End code: 9H (code after ASCII-BIN conversion)



For word units
Number of bytes after ASCII-BIN conversion: ((6)/2=3)
For byte units
Number of bytes after ASCII-BIN conversion: (6)

* • Two bytes of received data is read to C24 as one-byte data.
• Transmit data from the target device so that the converted complete code data is arranged in an even-number-byte of data.

When ASCII-BIN conversion is enabled, the codes for the receivable data and the codes for the received data which is stored in the receive area on the buffer memory are as shown below.

Type	Receivable codes	Codes stored in the receive area
Arbitrary data portion (including end code)	30H to 39H, 41H to 46H	0H to 9H, AH to FH

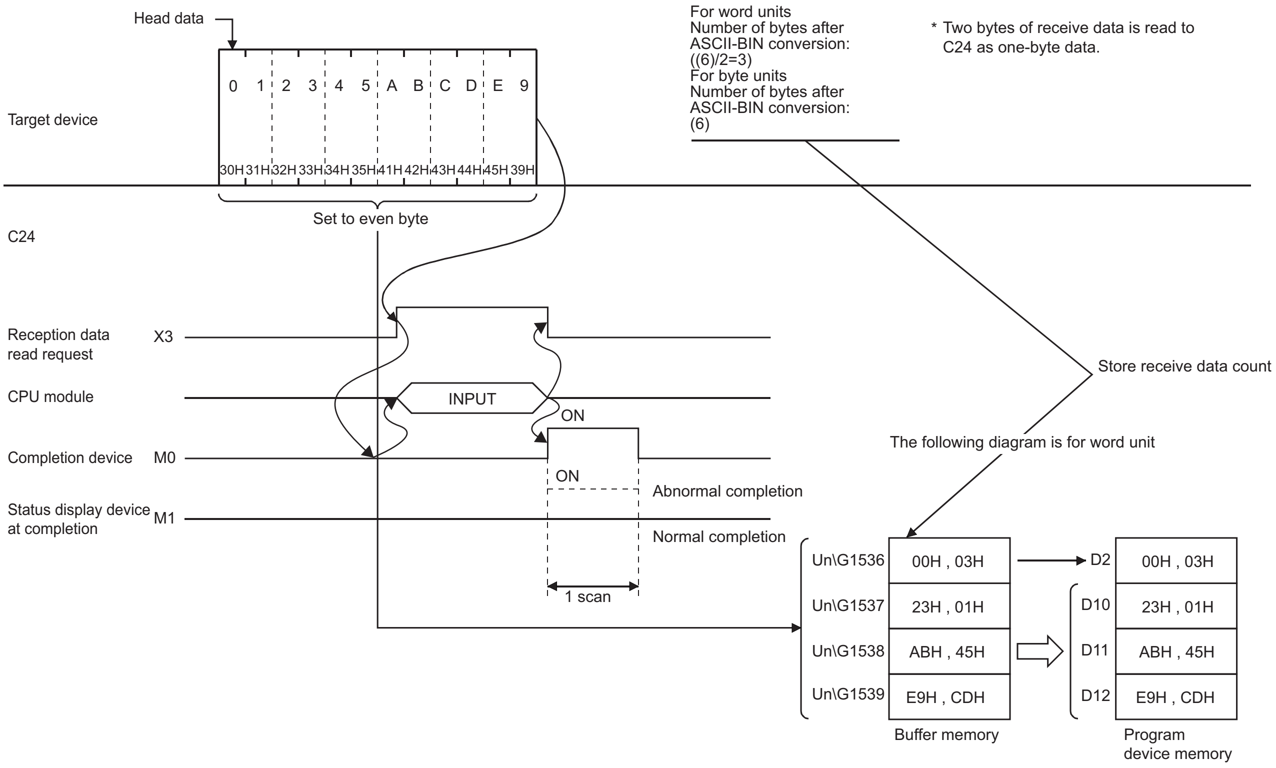
If data codes other than 30H to 39H and 41H to 46H are received for an arbitrary data portion, an error (7F20H) occurs after ASCII-BIN conversion processing by C24.

To perform ASCII-BIN conversion, ensure the end code is the code after conversion.

Operation example of reception based on the receive end data quantity

Ex.

When C24 receives data from the target device when ASCII-BIN conversion is enabled (Reception based on the receive end data quantity)



When the transparent code for receiving specification is disabled and ASCII-BIN conversion is enabled, the codes for the receivable data and the codes for the received data which is stored in the receive area on the buffer memory are as shown below.

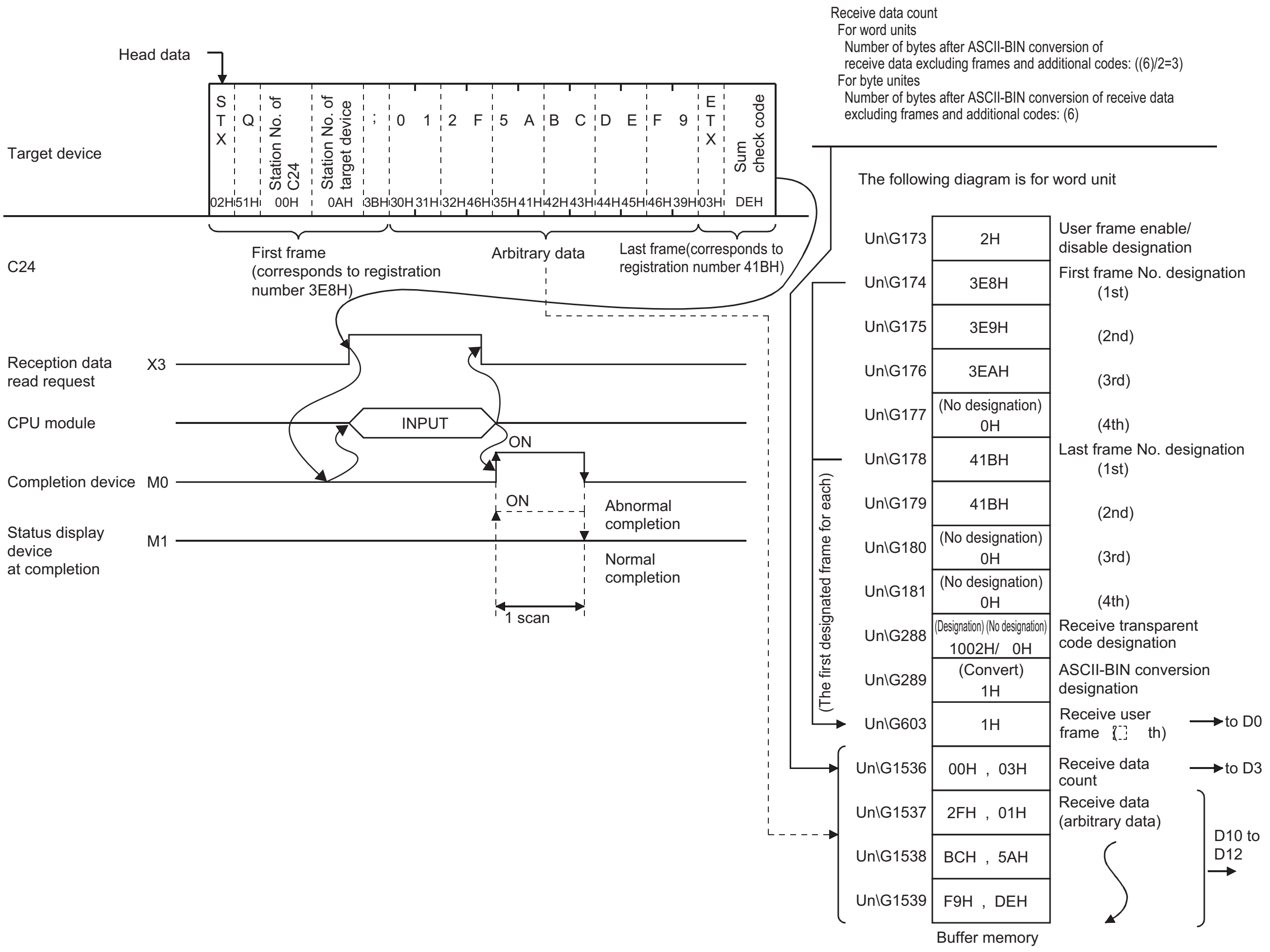
Type	Receivable codes	Codes stored in the receive area
Arbitrary data portion (including end code)	30H to 39H, 41H to 46H	0H to 9H, AH to FH

If data codes other than 30H to 39H and 41H to 46H are received for an arbitrary data portion, an error (7F20H) occurs after ASCII-BIN conversion by C24.

Operation example of reception using user frames

Ex.

When C24 receives data from the target device when ASCII-BIN conversion is enabled (Reception using user frames)


Head data
Target device
S T X
Q
Station No. of C24
Station No. of target device
;
0 1 2 F 5 A B C D E F 9
E T X
Sum check code
02H 51H 00H 0AH 3BH 30H 31H 32H 46H 35H 41H 42H 43H 44H 45H 46H 39H 03H DEH
C24
First frame (corresponds to registration number 3E8H)
Arbitrary data
Last frame(corresponds to registration number 41BH)
Receive data count
For word units
Number of bytes after ASCII-BIN conversion of receive data excluding frames and additional codes: ((6)/2=3)
For byte unites
Number of bytes after ASCII-BIN conversion of receive data excluding frames and additional codes: (6)
The following diagram is for word unit
Reception data read request
X3
CPU module
INPUT
Completion device
M0
ON
ON
Abnormal completion
Status display device at completion
M1
Normal completion
1 scan
(The first designated frame for each)
Un\G173
2H
User frame enable/ disable designation
Un\G174
3E8H
First frame No. designation
(1st)
Un\G175
3E9H
(2nd)
Un\G176
3EAH
(3rd)
Un\G177
(No designation) 0H
(4th)
Un\G178
41BH
Last frame No. designation
(1st)
Un\G179
41BH
(2nd)
Un\G180
(No designation) 0H
(3rd)
Un\G181
(No designation) 0H
(4th)
Un\G288
(Designation) (No designation) 1002H/ 0H
Receive transparent code designation
Un\G289
(Convert) 1H
ASCII-BIN conversion designation
Un\G603
1H
Receive user frame (th)
to D0
Un\G1536
00H , 03H
Receive data count
to D3
Un\G1537
2FH , 01H
Receive data (arbitrary data)
Un\G1538
BCH , 5AH
Un\G1539
F9H , DEH
D10 to D12
Buffer memory

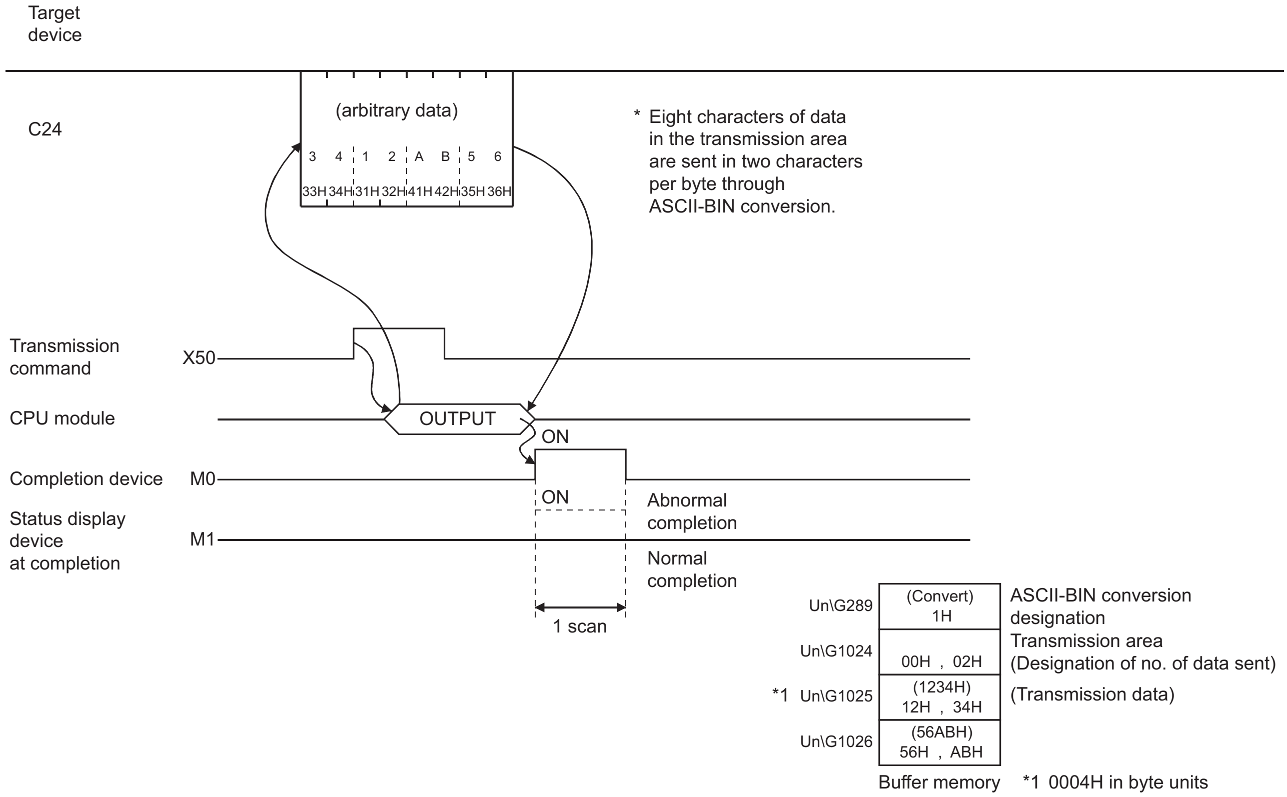
Example of data transmission

The following shows an example of data transmission.

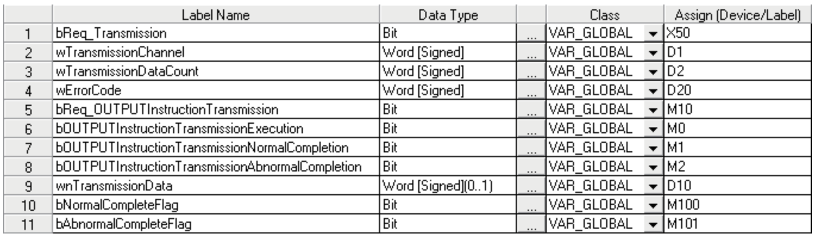
Transmission of arbitrary data

When C24 transmits data to the target device with ASCII-BIN conversion used (Transmitting given data with OUTPUT dedicated instruction)

Ex.



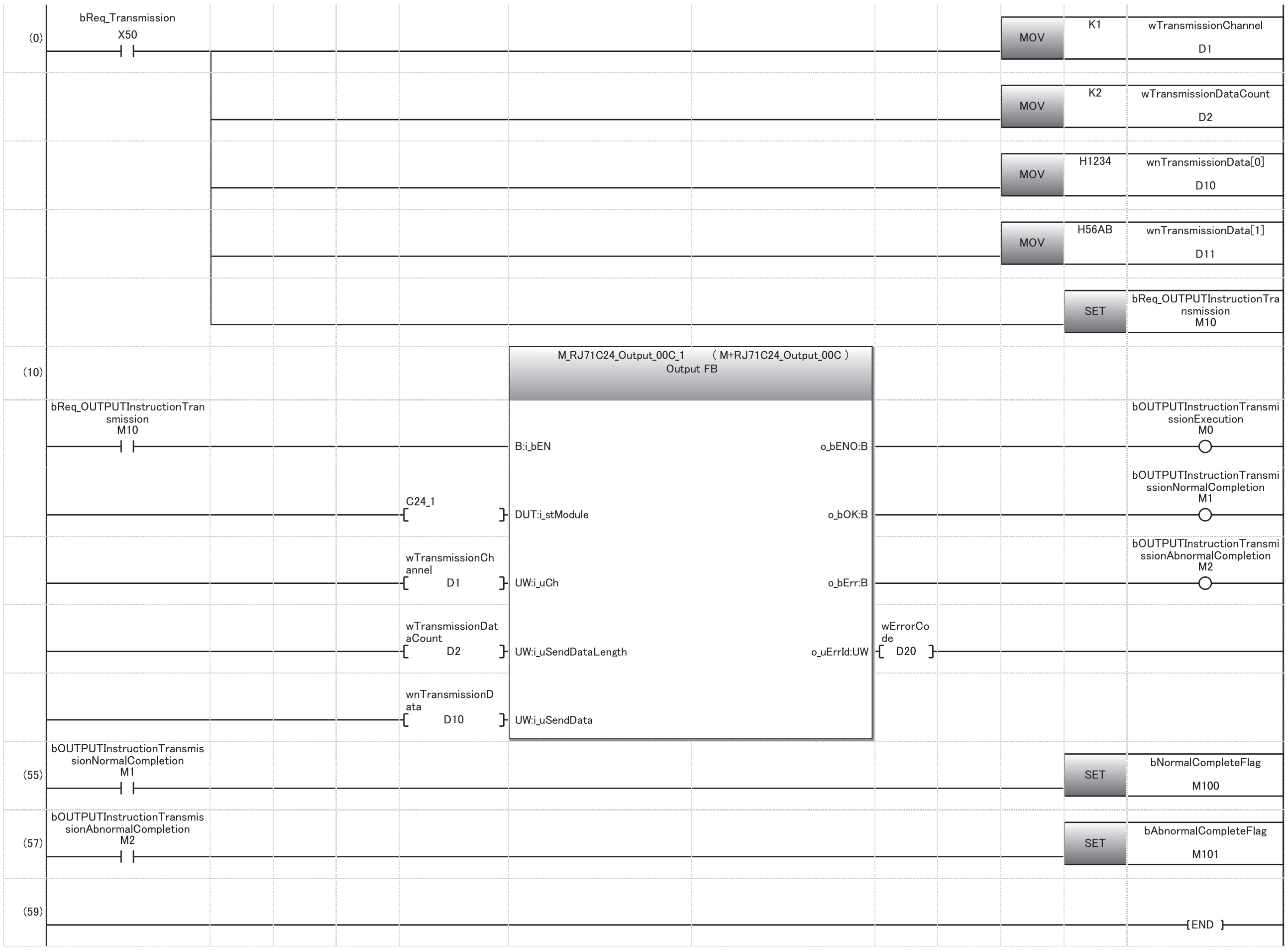
■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_Output	Sends data for specified data points.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_Transmission	Bit	...	VAR_GLOBAL	X50
2	wTransmissionChannel	Word [Signed]	...	VAR_GLOBAL	D1
3	wTransmissionDataCount	Word [Signed]	...	VAR_GLOBAL	D2
4	wErrorCode	Word [Signed]	...	VAR_GLOBAL	D20
5	bReq_OUTPUTInstructionTransmission	Bit	...	VAR_GLOBAL	M10
6	bOUTPUTInstructionTransmissionExecution	Bit	...	VAR_GLOBAL	M0
7	bOUTPUTInstructionTransmissionNormalCompletion	Bit	...	VAR_GLOBAL	M1
8	bOUTPUTInstructionTransmissionAbnormalCompletion	Bit	...	VAR_GLOBAL	M2
9	wnTransmissionData	Word [Signed](0..1)	...	VAR_GLOBAL	D10
10	bNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
11	bAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101

For information on the module FB to be used, refer to the following manual.

MELSEC iQ-R Serial Communication Module Function Block Reference



(0) Set the transmission channel to CH1.
Set the send data quantity to 2.
Set the data to be transmitted in D10 to D11.
Set the transmission direction.

(10) Execute transmission.

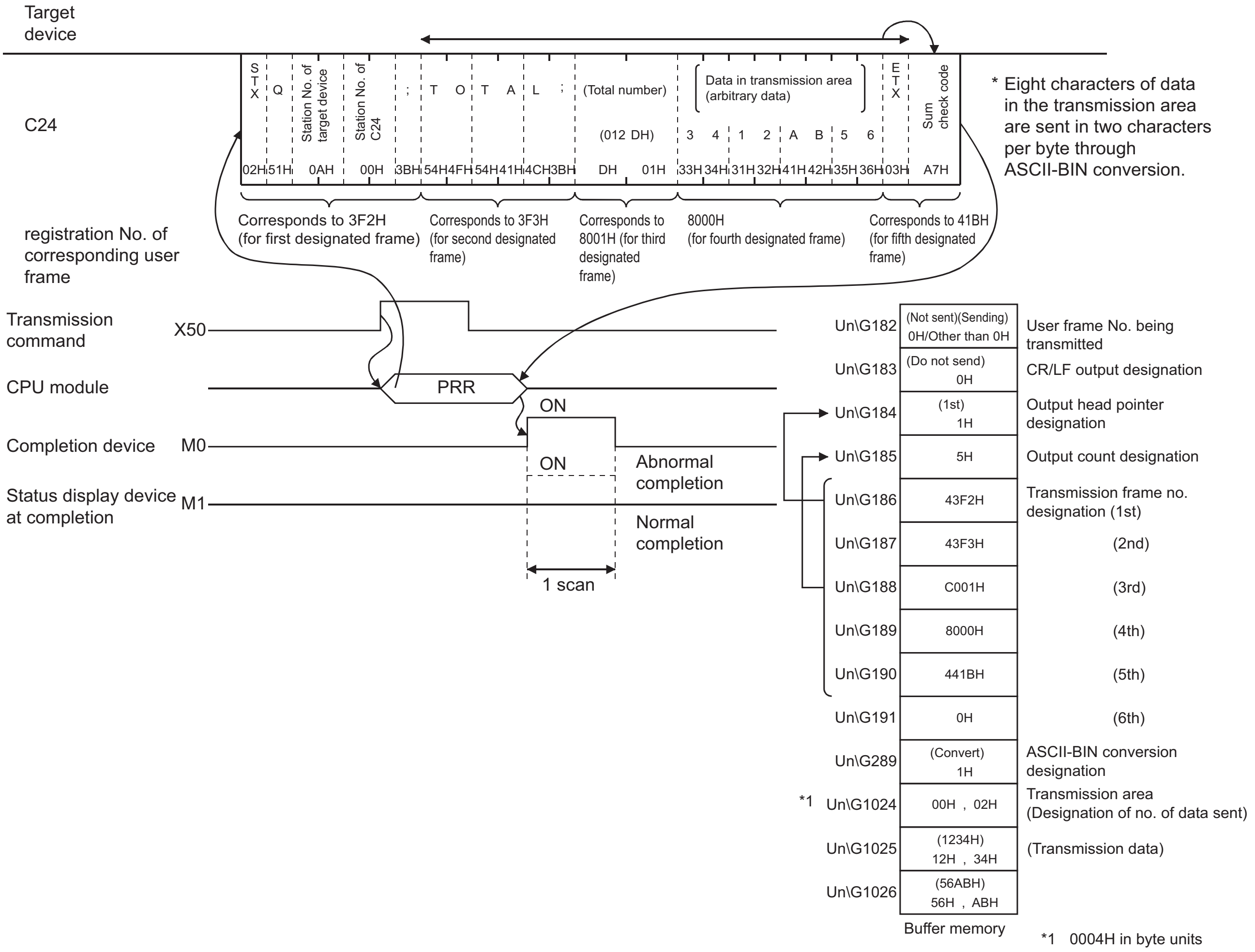
(55) Normal completion

(57) Abnormal completion

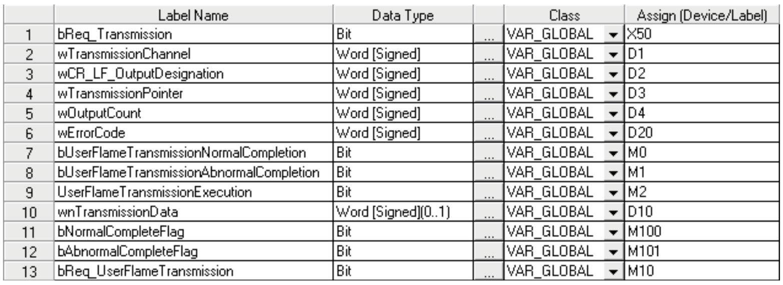
Transmission using user frame

When C24 transmits data to the target device with ASCII-BIN conversion used (Transmitting given data and user frame combined with PRR dedicated instruction)

Ex.


Target device
C24
S T X
Q
Station No. of target device
Station No. of C24
;
T O T A L ;
(Total number)
(012 DH)
Data in transmission area (arbitrary data)
3 4 1 2 A B 5 6
E T X
Sum check code
02H 51H 0AH 00H 3BH 54H 4FH 54H 41H 4CH 3BH DH 01H 33H 34H 31H 32H 41H 42H 35H 36H 03H A7H
* Eight characters of data in the transmission area are sent in two characters per byte through ASCII-BIN conversion.
registration No. of corresponding user frame
Corresponds to 3F2H (for first designated frame)
Corresponds to 3F3H (for second designated frame)
Corresponds to 8001H (for third designated frame)
8000H (for fourth designated frame)
Corresponds to 41BH (for fifth designated frame)
Transmission command
X50
CPU module
PRR
ON
Completion device
M0
ON
Abnormal completion
Status display device at completion
M1
Normal completion
1 scan
Un\G182 (Not sent)(Sending) 0H/Other than 0H User frame No. being transmitted
Un\G183 (Do not send) 0H CR/LF output designation
Un\G184 (1st) 1H Output head pointer designation
Un\G185 5H Output count designation
Un\G186 43F2H Transmission frame no. designation (1st)
Un\G187 43F3H (2nd)
Un\G188 C001H (3rd)
Un\G189 8000H (4th)
Un\G190 441BH (5th)
Un\G191 0H (6th)
Un\G289 (Convert) 1H ASCII-BIN conversion designation
*1 Un\G1024 00H , 02H Transmission area (Designation of no. of data sent)
Un\G1025 (1234H) 12H , 34H (Transmission data)
Un\G1026 (56ABH) 56H , ABH
Buffer memory
*1 0004H in byte units

■Program example

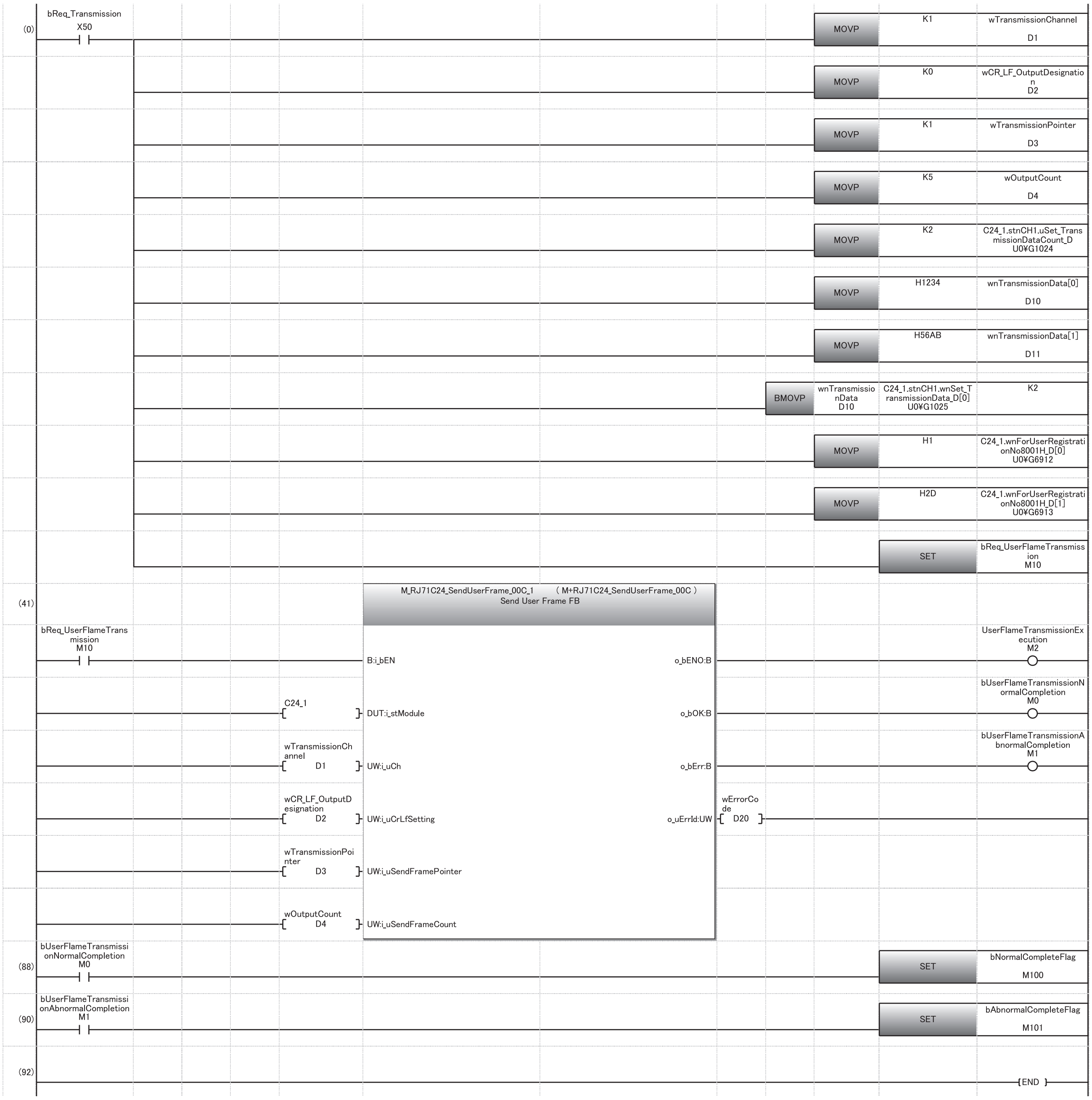
Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.wnSet_TransmissionData_D[0]	CH1_Send data specification	U0\G1025
	C24_1.wnForUserRegistrationNo8001H_D[0]	For user registration number 8001H [0]	U0\G6912
	C24_1.wnForUserRegistrationNo8001H_D[1]	For user registration number 8001H [1]	U0\G6913
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_Transmission	Bit	...	VAR_GLOBAL	X50
2	wTransmissionChannel	Word [Signed]	...	VAR_GLOBAL	D1
3	wCR_LF_OutputDesignation	Word [Signed]	...	VAR_GLOBAL	D2
4	wTransmissionPointer	Word [Signed]	...	VAR_GLOBAL	D3
5	wOutputCount	Word [Signed]	...	VAR_GLOBAL	D4
6	wErrorCode	Word [Signed]	...	VAR_GLOBAL	D20
7	bUserFlameTransmissionNormalCompletion	Bit	...	VAR_GLOBAL	M0
8	bUserFlameTransmissionAbnormalCompletion	Bit	...	VAR_GLOBAL	M1
9	UserFlameTransmissionExecution	Bit	...	VAR_GLOBAL	M2
10	wnTransmissionData	Word [Signed](0..1)	...	VAR_GLOBAL	D10
11	bNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
12	bAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101
13	bReq_UserFlameTransmission	Bit	...	VAR_GLOBAL	M10

Category	Label name/FB name	Description	Device
FB	M+RJ71C24_SendUserFrame	Sends data using the nonprocedural protocol communication and the user frame according to the setting of the user frame specification area for sending data.	—

For information on the module FB to be used, refer to the following manual.

MELSEC iQ-R Serial Communication Module Function Block Reference



(0)
bReq_Transmission
X50
MOVP K1 wTransmissionChannel D1
MOVP K0 wCR_LF_OutputDesignation D2
MOVP K1 wTransmissionPointer D3
MOVP K5 wOutputCount D4
MOVP K2 C24_1.stnCH1.uSet_TransmissionDataCount_D U0¥G1024
MOVP H1234 wnTransmissionData[0] D10
MOVP H56AB wnTransmissionData[1] D11
BMOVP wnTransmissionData D10 C24_1.stnCH1.wnSet_TransmissionData_D[0] U0¥G1025 K2
MOVP H1 C24_1.wnForUserRegistrationNo8001H_D[0] U0¥G6912
MOVP H2D C24_1.wnForUserRegistrationNo8001H_D[1] U0¥G6913
SET bReq_UserFlameTransmission M10
(41)
M_RJ71C24_SendUserFrame_00C_1 (M+RJ71C24_SendUserFrame_00C)
Send User Frame FB
bReq_UserFlameTransmission
M10
B:i_bEN
o_bENO:B
UserFlameTransmissionExecution
M2
C24_1
DUT:i_stModule
o_bOK:B
bUserFlameTransmissionNormalCompletion
M0
wTransmissionChannel
D1
UW:i_uCh
o_bErr:B
bUserFlameTransmissionAbnormalCompletion
M1
wCR_LF_OutputDesignation
D2
UW:i_uCrLfSetting
o_uErrId:UW
wErrorCode
D20
wTransmissionPointer
D3
UW:i_uSendFramePointer
wOutputCount
D4
UW:i_uSendFrameCount
(88)
bUserFlameTransmissionNormalCompletion
M0
SET bNormalCompleteFlag M100
(90)
bUserFlameTransmissionAbnormalCompletion
M1
SET bAbnormalCompleteFlag M101
(92)
END

(0) Set the transmission channel to CH1.
Set CR/LF to "Do not send".
Set the send pointer to 1.
Set the output quantity to 5.
Set the send data length to 2WORD.
Set an arbitrary send data.
Set an arbitrary transmitted data in send buffer.
Set the data to be transmitted to user frame 8001H.
Set the transmission direction.

(41) Execute user frame transmission.

(88) Normal completion

(90) Abnormal completion

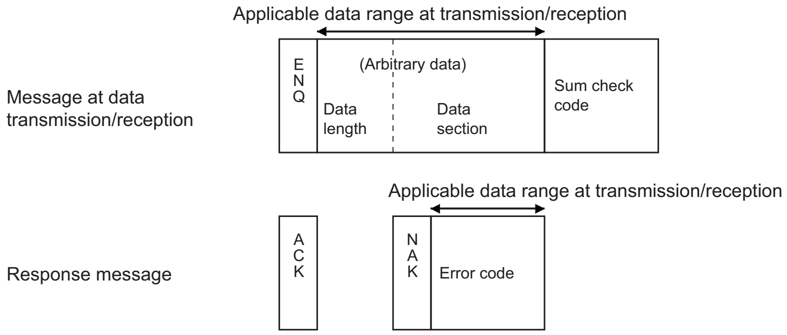
19.5 ASCII-BIN Conversion for Data Communicated with Bidirectional Protocol

This section explains the ASCII-BIN conversion of data to be communicated using the bidirectional protocol.

Transmitted/received data range for ASCII-BIN conversion

The range of data to be sent/received to which ASCII-BIN conversion is performed is as shown below:

Ex.


Applicable data range at transmission/reception
Message at data transmission/reception
ENQ
(Arbitrary data)
Data length
Data section
Sum check code
Applicable data range at transmission/reception
Response message
ACK
NAK
Error code

C24 performs the following processing when sending or receiving data.

■At the time of data reception

Processing flow when a message is received during data reception

- C24 converts a received data length (4-digit ASCII code data (hexadecimal)) to 2-byte binary code data and stores it in the receive data quantity storage area.
- C24 converts each two characters of a received data portion (2-digit ASCII code data (hexadecimal)) to 1-byte binary code data and stores it in the receive data storage area.
- C24 calculates a sum check code using the received data length and data portion prior to ASCII-BIN conversion, then checks the head of the received sum check code using the code as a lower byte.

Processing flow when a response message is received

- C24 converts the first 2 digits of a received error code (4-digit ASCII code data (hexadecimal)) to 2-byte binary code data as the lower byte, and stores it in the transmission result storage area. (When '1234' is received, it is converted to 3412H and stored.)

■At the time of data transmission

Processing flow when a message is transmitted during data transmission

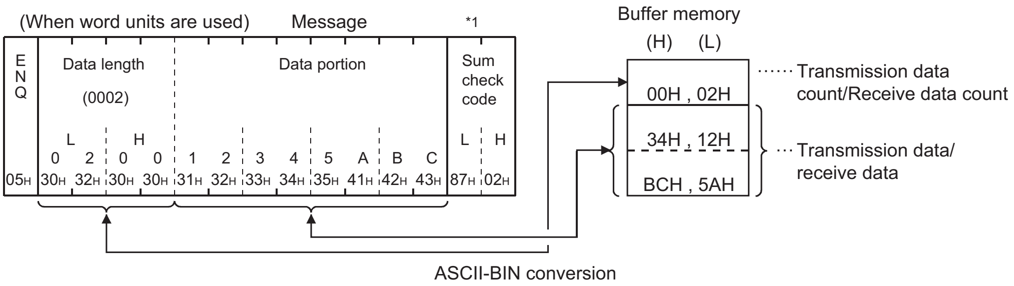
- C24 converts a data length (send data quantity) to 4-digit ASCII code data (hexadecimal), and sends it beginning with the lowest byte (L).
- C24 converts a data portion (send data) for one address to 4-digit ASCII code data (hexadecimal), and sends it beginning with the lowest byte (L).
- C24 calculates a sum check code using the data length and data portion after ASCII-BIN conversion, and adds it to the transmission message.

Processing flow when a response message is transmitted

- C24 converts an error code for a detected error to 4-digit ASCII code data (hexadecimal), then sends it beginning with the lowest byte (L).
  (For 3412H, it is converted to '3412' and sent beginning with '12'.)

Ex.

Calculation of sum check code


(When word units are used)
Message
*1
ENQ
Data length
(0002)
Data portion
Sum check code
L H
0 2 0 0 1 2 3 4 5 A B C
05H 30H 32H 30H 30H 31H 32H 33H 34H 35H 41H 42H 43H 87H 02H
L H
Buffer memory
(H) (L)
00H , 02H
34H , 12H
BCH , 5AH
Transmission data count/Receive data count
Transmission data/ receive data
ASCII-BIN conversion

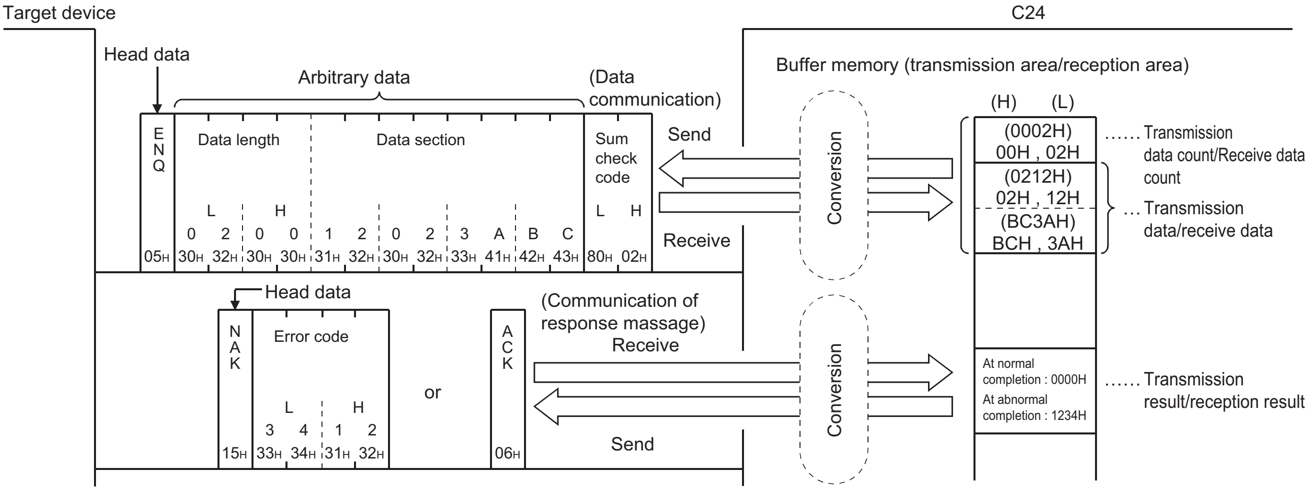
*1 Calculation of sum check code

30H+32H+30H+30H+31H+32H+33H+34H+35H+41H+42H+43H=0287H

(H)(L)

Processing order for data transmission/reception

The processing steps taken by C24 to transmit/receive data with ASCII-BIN conversion and transparent code designation enabled are explained in the section for transparent codes and additional codes. (☞ Page 371 Handling Transparent and Additional Codes for Bidirectional Protocol Data Communication)


Target device
C24
Head data
Arbitrary data
(Data communication)
ENQ
Data length
Data section
Sum check code
L H
0 2 0 0 1 2 0 2 3 A B C
05H 30H 32H 30H 30H 31H 32H 30H 32H 33H 41H 42H 43H 80H 02H
L H
Send
Receive
Buffer memory (transmission area/reception area)
(H) (L)
(0002H)
00H , 02H
(0212H)
02H , 12H
(BC3AH)
BCH , 3AH
Transmission data count/Receive data count
Transmission data/receive data
Conversion
Head data
NAK
Error code
L H
3 4 1 2
15H 33H 34H 31H 32H
or
ACK
06H
(Communication of response massage)
Receive
Send
Conversion
At normal completion : 0000H
At abnormal completion : 1234H
Transmission result/reception result

19.6 Data Communication Using Bidirectional Protocol

This section shows examples of data communication using the bidirectional protocol when the settings/registrations are made with Engineering tool.

Parameter settings

For details for each setting item, refer to the parameter. (☞ Page 454 PARAMETER SETTING)

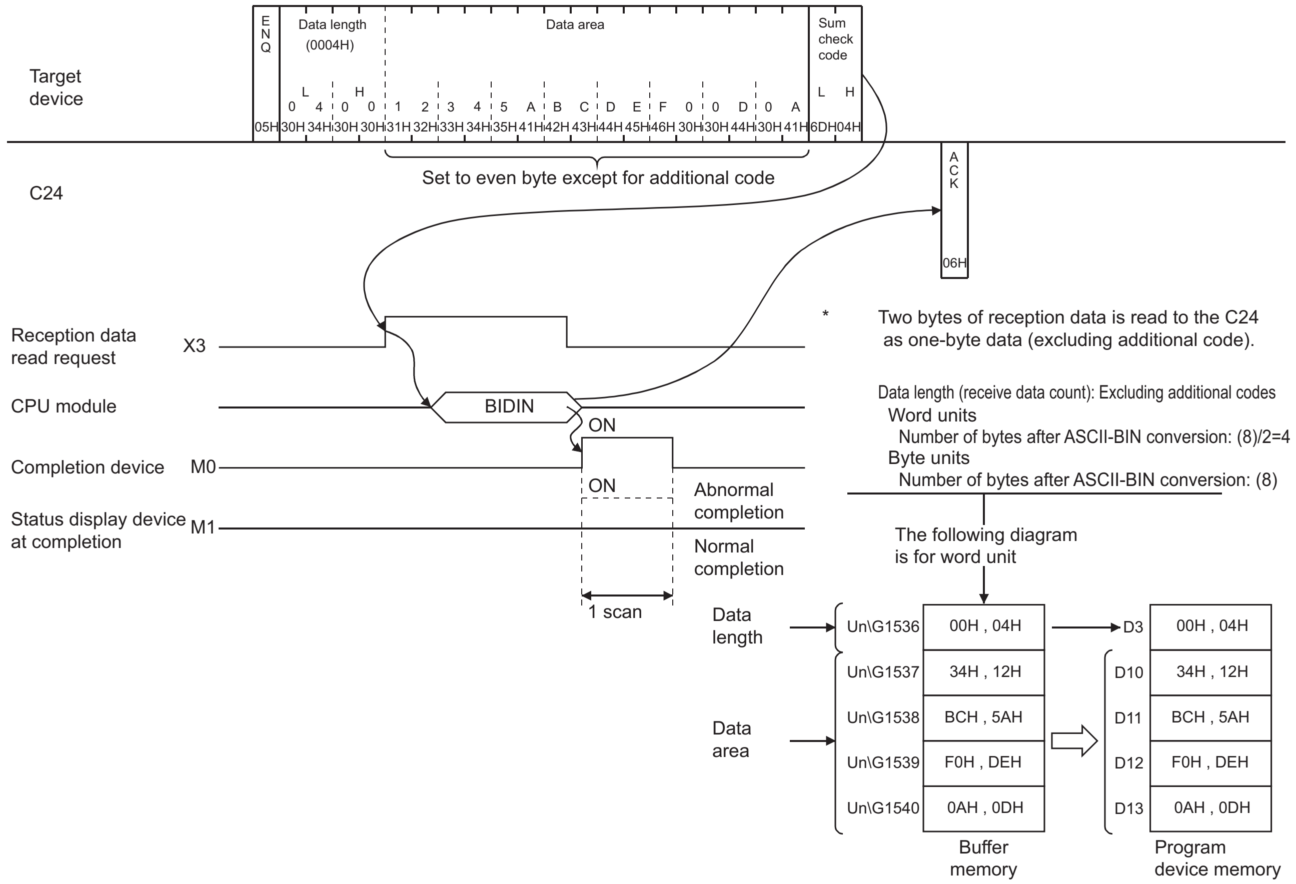
Setting item				Setting content	
				CH1	CH2
Basic settings	Various control specification	Test mode setting		—	CH2 is not used
		Communication protocol setting		Bidirectional protocol	
		Communication speed setting		Set according to the target device.	
		Transmission setting	Operation setting	Independent	
			Data bit	Set according to the target device.	
			Parity bit		
			Odd/even parity		
			Stop bit		
			Sum check code		
			Online change		
			Setting change		
		Station number setting (CH1, 2 in common: 0 to 31)		—	
		Conversion designation	ASCII-BIN conversion designation	ON	

Example of data reception

Data reception example (ASCII-BIN conversion is enabled)

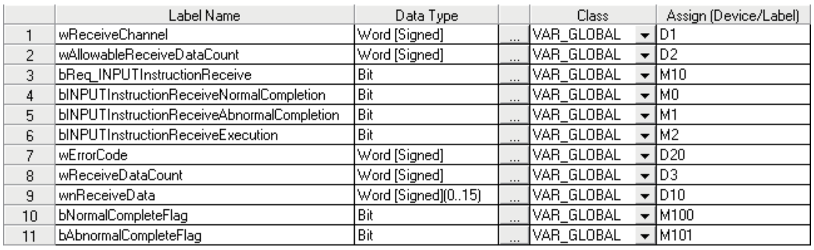
The following shows an example of data reception when ASCII-BIN conversion is enabled.

Ex.



Target device
E N Q
Data length (0004H)
Data area
Sum check code
L H
0 4 0 0
1 2 3 4 5 A B C D E F 0 0 D 0 A
L H
05H 30H 34H 30H 30H 31H 32H 33H 34H 35H 41H 42H 43H 44H 45H 46H 30H 30H 44H 30H 41H 6DH 04H
C24
Set to even byte except for additional code
A C K
06H
Reception data read request
X3
CPU module
BIDIN
ON
Completion device
M0
ON
Abnormal completion
Status display device at completion
M1
Normal completion
1 scan
*
Two bytes of reception data is read to the C24 as one-byte data (excluding additional code).
Data length (receive data count): Excluding additional codes
Word units
Number of bytes after ASCII-BIN conversion: (8)/2=4
Byte units
Number of bytes after ASCII-BIN conversion: (8)
The following diagram is for word unit
Data length
Data area
Un\G1536
00H , 04H
Un\G1537
34H , 12H
Un\G1538
BCH , 5AH
Un\G1539
F0H , DEH
Un\G1540
0AH , 0DH
Buffer memory
D3
00H , 04H
D10
34H , 12H
D11
BCH , 5AH
D12
F0H , DEH
D13
0AH , 0DH
Program device memory

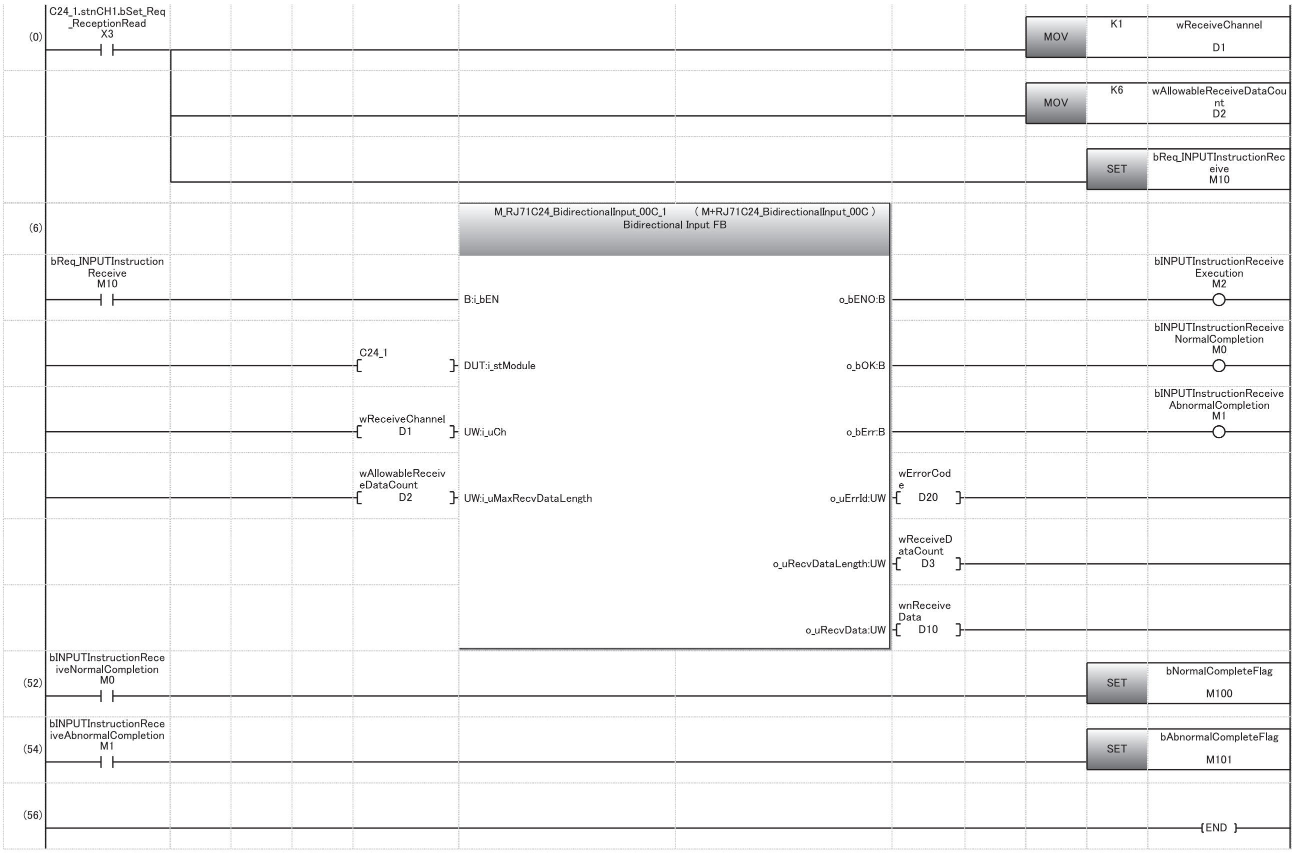
■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_BidirectionalInput	Reads the received data.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	wReceiveChannel	Word [Signed]	...	VAR_GLOBAL	D1
2	wAllowableReceiveDataCount	Word [Signed]	...	VAR_GLOBAL	D2
3	bReq_INPUTInstructionReceive	Bit	...	VAR_GLOBAL	M10
4	bINPUTInstructionReceiveNormalCompletion	Bit	...	VAR_GLOBAL	M0
5	bINPUTInstructionReceiveAbnormalCompletion	Bit	...	VAR_GLOBAL	M1
6	bINPUTInstructionReceiveExecution	Bit	...	VAR_GLOBAL	M2
7	wErrorCode	Word [Signed]	...	VAR_GLOBAL	D20
8	wReceiveDataCount	Word [Signed]	...	VAR_GLOBAL	D3
9	wnReceiveData	Word [Signed](0..15)	...	VAR_GLOBAL	D10
10	bNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
11	bAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101

For information on the module FB to be used, refer to the following manual.

MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
C24_1.stnCH1.bSet_Req_ReceptionRead
X3
MOV
K1
wReceiveChannel
D1
MOV
K6
wAllowableReceiveDataCount
D2
SET
bReq_INPUTInstructionReceive
M10
(6)
M_RJ71C24_BidirectionalInput_00C_1 (M+RJ71C24_BidirectionalInput_00C)
Bidirectional Input FB
bReq_INPUTInstructionReceive
M10
B:i_bEN
o_bENO:B
bINPUTInstructionReceiveExecution
M2
C24_1
DUT:i_stModule
o_bOK:B
bINPUTInstructionReceiveNormalCompletion
M0
wReceiveChannel
D1
UW:i_uCh
o_bErr:B
bINPUTInstructionReceiveAbnormalCompletion
M1
wAllowableReceiveDataCount
D2
UW:i_uMaxRecvDataLength
o_uErrId:UW
wErrorCode
D20
o_uRecvDataLength:UW
wReceiveDataCount
D3
o_uRecvData:UW
wnReceiveData
D10
(52)
bINPUTInstructionReceiveNormalCompletion
M0
SET
bNormalCompleteFlag
M100
(54)
bINPUTInstructionReceiveAbnormalCompletion
M1
SET
bAbnormalCompleteFlag
M101
(56)
END

(0) Set the reception channel to CH1.
Set the allowable number of data to be received to 6.
Set the reception direction.

(6) Execute reception.

(52) Normal completion

(54) Abnormal completion

When ASCII-BIN conversion is enabled, the codes for receivable arbitrary data portion and the codes for the received data that is stored in the receive area on the buffer memory are as follows:

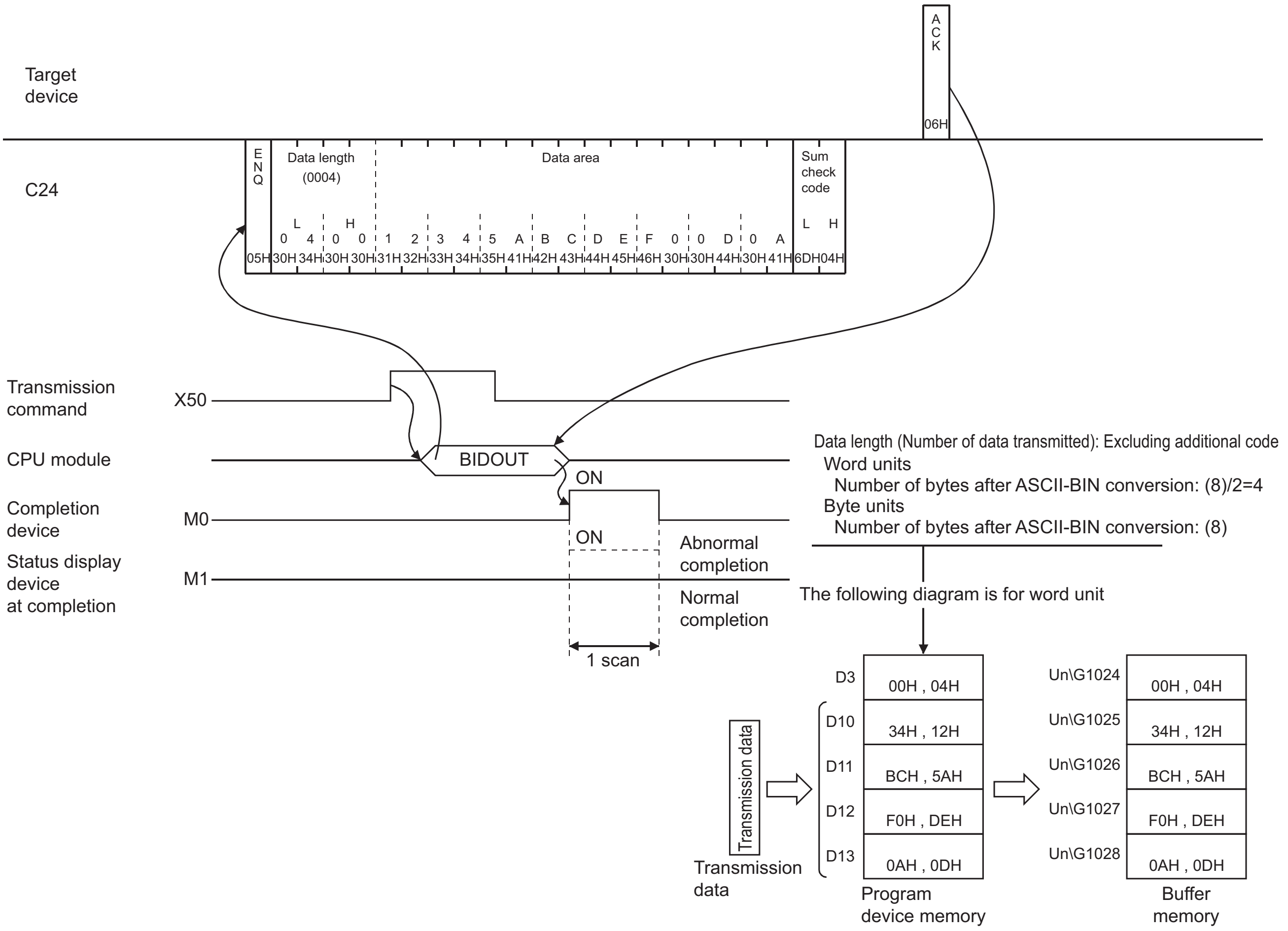
Type	Receivable codes	Codes stored in the receive area
Data length, data portion	30H to 39H, 41H to 46H	0H to 9H, AH to FH

Example of data transmission

Data transmission example (ASCII-BIN conversion is enabled)

The following shows an example of data transmission with ASCII-BIN conversion used.

Ex.

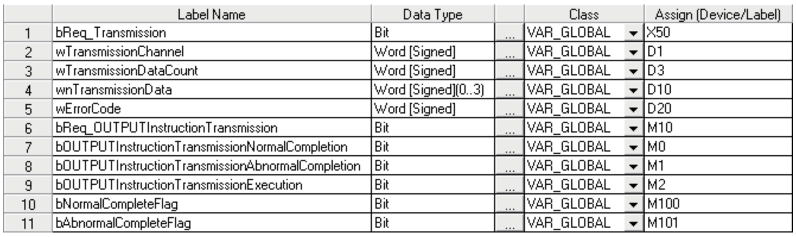


Data length (Number of data transmitted): Excluding additional code
Word units
Number of bytes after ASCII-BIN conversion: (8)/2=4
Byte units
Number of bytes after ASCII-BIN conversion: (8)

The following diagram is for word unit

For data designated by the CPU module, data for one word is converted to 4-byte data ('0' to '9' and 'A' to 'F') and transmitted.

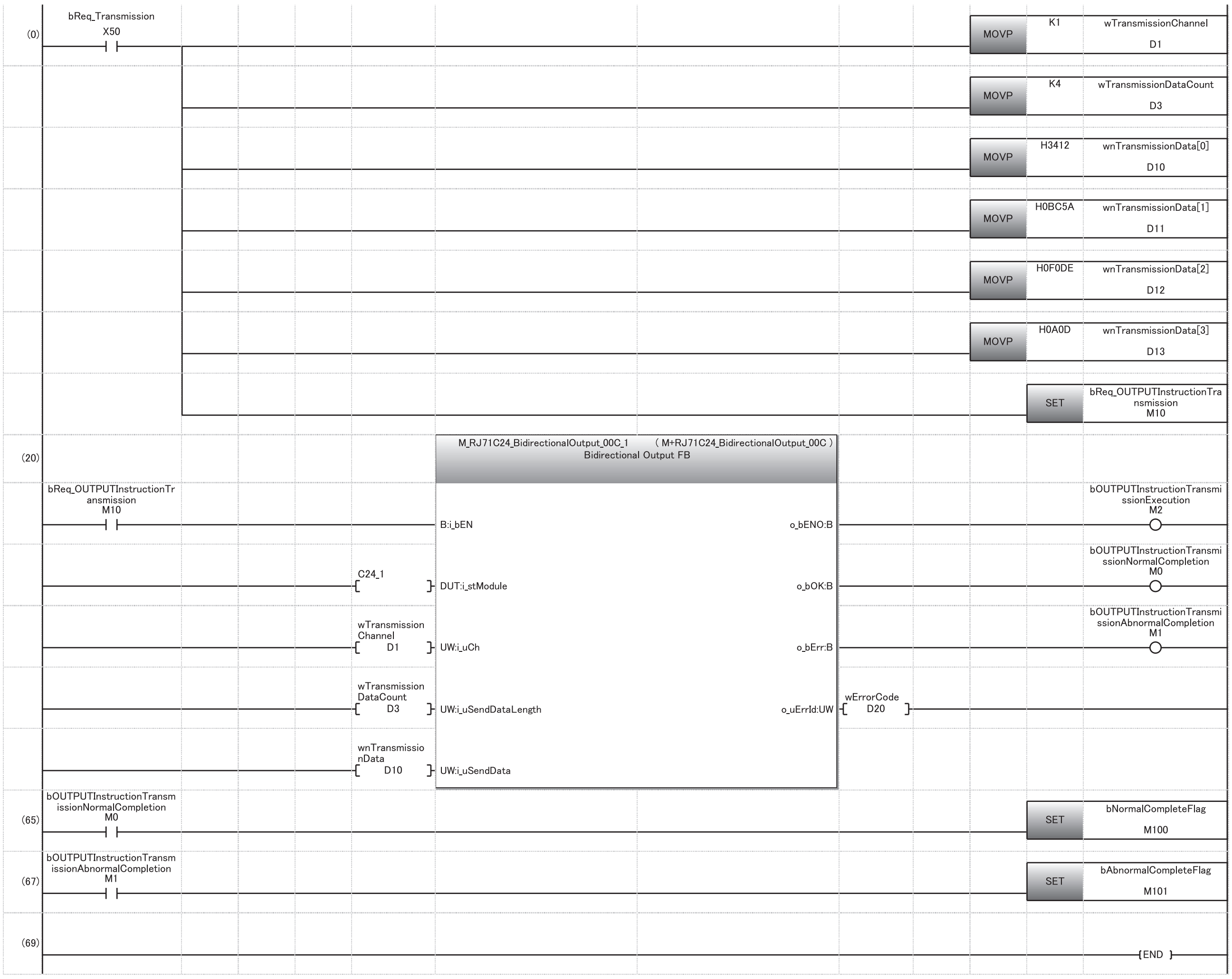
■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_BidirectionalOutput	Sends data for specified data points.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_Transmission	Bit	...	VAR_GLOBAL ▼	X50
2	wTransmissionChannel	Word [Signed]	...	VAR_GLOBAL ▼	D1
3	wTransmissionDataCount	Word [Signed]	...	VAR_GLOBAL ▼	D3
4	wnTransmissionData	Word [Signed](0..3)	...	VAR_GLOBAL ▼	D10
5	wErrorCode	Word [Signed]	...	VAR_GLOBAL ▼	D20
6	bReq_OUTPUTInstructionTransmission	Bit	...	VAR_GLOBAL ▼	M10
7	bOUTPUTInstructionTransmissionNormalCompletion	Bit	...	VAR_GLOBAL ▼	M0
8	bOUTPUTInstructionTransmissionAbnormalCompletion	Bit	...	VAR_GLOBAL ▼	M1
9	bOUTPUTInstructionTransmissionExecution	Bit	...	VAR_GLOBAL ▼	M2
10	bNormalCompleteFlag	Bit	...	VAR_GLOBAL ▼	M100
11	bAbnormalCompleteFlag	Bit	...	VAR_GLOBAL ▼	M101

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference



(0) Set the transmission channel to CH1.
Set the send data quantity to 4.
Set arbitrary data to be transmitted in D10 to D13.
Set the instruction execution.

(20) Execute transmission.

(65) Normal completion

(67) Abnormal completion

20 DATA COMMUNICATIONS BETWEEN TARGET DEVICE AND CPU MODULE IN M:N BASIS

Point

Always read this chapter when communicating data by the multidrop connection in the system configuration that the target device and CPU module are connected on an m:n basis.

This section explains data communications between the target devices and the CPU module by the multidrop connection of multiple target devices (m stations) and C24s (n stations). (The total number of 'm' and 'n' is up to 32 stations.)

With this m:n multidrop communication, only MC protocol data communications by command transmission from the target devices can be performed.

20.1 Considerations on Data Communications

Concurrent data communication with multiple target devices

When communicating data in m:n system configuration, more than one target devices cannot communicate data with the CPU module at the same time.

Configure interlock among the target devices so that the target device can communicate data with the CPU module in a 1:1 basis.

For details on arrangement and method for interlock among target devices, refer to the following sections.

☞ Page 400 Arrangement for Interlock among Target Devices

☞ Page 403 Examples of Procedure for Data Communications with CPU Module

Data communication method

Data communication between target devices and the CPU module can be performed by the following methods only:

- Data communication with the full-duplex communication method (The m:n data communications is impossible with half-duplex communications)
- Data communications by command transmission from target device using the MC protocol excluding the format 3 and format 5 control procedure (Data communications using the format 3 and format 5 control procedure and data transmission from sequence program using the on-demand function cannot be performed.)

Data transmitted from any of target devices

Data transmitted by one target device is received by all the other target devices, including the target device that transmitted the data. The sent data from the CPU module is also received by all the target devices.

Therefore, devices that received data not addressed to their own stations (by judging by the station number in a message) are required to ignore the received data.

At the CPU module side, C24 also ignores the received data not addressed to it.

Numbers designated for "Station No." and "Self-station No."

Designate the numbers below for "Station No." and "Self-station No." in the command message transmitted from the target device for data communications in the m:n system configuration.

■When accessing to C24

Item	Communications with 2C/3C/4C frame	Communications with 1C frame
Station No.	Station number of C24 to be passed through	
Self-station No.	Station number of access source target device[*1]	No designation necessary (No "Self-station No." item)

■When accessing to other device (communications for interlock)

Item	Communications with 2C/3C/4C frame	Communications with 1C frame
Station No.	Station number of access destination device[*1]	
Self-station No.	Station number of access source device[*1]	No designation necessary (No "Self-station No." item)

[*1] The number which has not been set to the respective C24s on the CPU module side is used within the range of "0" to "31" (00H to 1FH) as the number for the target device side in "Station No." and "Self-station No." in the message.
Select and designate the number of each target device.
The designation method is as shown in 'Contents of the data designation items' of each frame.
(📖MELSEC Communication Protocol Reference Manual)
· Station No.: Designate the number of the transmit destination device.
· Self-station No.: Designate the number of the transmit source device. (No designation necessary when 1C frame is used.)

20.2 Arrangement for Interlock among Target Devices

When using the multidrop connection to communicate data between target devices and the CPU module in a m:n basis, the interlocks need to be configured among the target devices to prevent more than one target devices from communicating data with the CPU module at the same time.

The arrangements necessary for providing the interlocks among the target devices in order for one of the target devices to communicate data with the CPU module are as shown below.

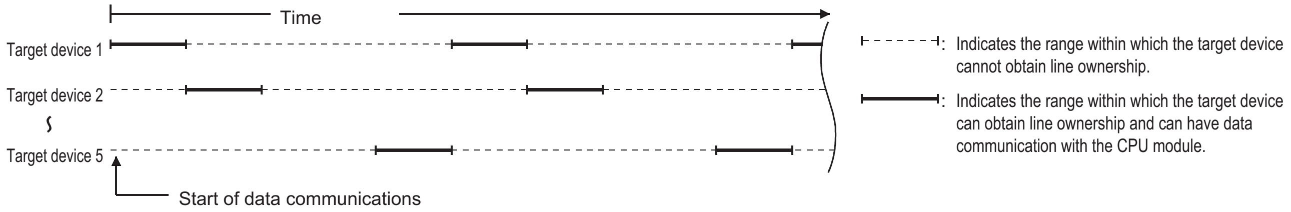
(Arrangements for priority use of the line (obtaining line ownership) from the start to the end of data communications with the CPU module.)

Maximum communication time per target device station

Determine the maximum time for each target device to communicate data with the CPU module once it obtained the line ownership.

This prevents loss of data communications between other devices and the CPU module in such cases as the target device, which obtained the line ownership, went down.

Ex.



Time
Target device 1
Target device 2
Target device 5
Start of data communications
Indicates the range within which the target device cannot obtain line ownership.
Indicates the range within which the target device can obtain line ownership and can have data communication with the CPU module.

Restriction

- The maximum data communication time per target device station must be the maximum time required by the respective devices to communicate data with the CPU module.
- The device which obtained the line ownership must complete data communication with the CPU module within the maximum data communication time after the system started. (If data communications cannot be completed within the time, initialize the C24 transmission sequence by transmitting the EOT/CL code to the target CPU module within the maximum communication time. (📖 MELSEC Communication Protocol Reference Manual))
- While a target device and the CPU module are communicating data, have the other devices check for timeout so that they do not transmit data during this time.

Message structure when communicating data among target devices

Determine the message structure to communicate data among target devices by any of the following.

This is determined to provide interlocks among the other target devices so that each of them can communicate data with CPU module on a 1:1 basis.

Same message structure as control procedure format frame

■Use an unset number for "Station No." and "Self-station No." in a message.

The number which has not been set to the respective C24s on the CPU module side is used within the range of "0" to "31" (00H to 1FH) as the number for the target device side in "Station No." and "Self-station No." in the message.

■Select and designate the target device numbers.

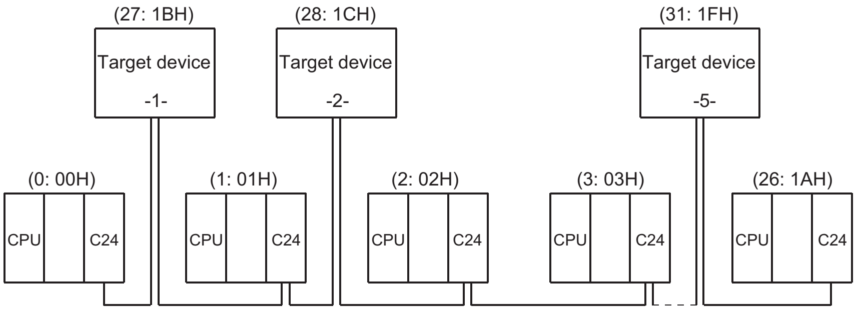
The designation method is as shown in 'Contents of the data designation items' of each frame. (📖 MELSEC Communication Protocol Reference Manual)

- Station No. : Designate the number of the transmission destination device.
- Self-station No.: Designate the number of the transmission source device. (No designation necessary when 1C frame is used.)

Ex.

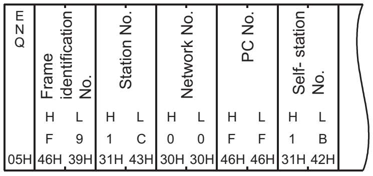
When the m:n configuration is 5:27

The values in ( ) indicate the respective station number of the target device and C24. (decimal: hexadecimal)



The above C24 includes Q series C24, L series C24, QC24(N), and UC24.

(3C frame format 1)



ENQ	Frame identification No.		Station No.		Network No.		PC No.		Self- station No.	
	H	L	H	L	H	L	H	L	H	L
	F	9	1	C	0	0	F	F	1	B
05H	46H	39H	31H	43H	30H	30H	46H	46H	31H	42H

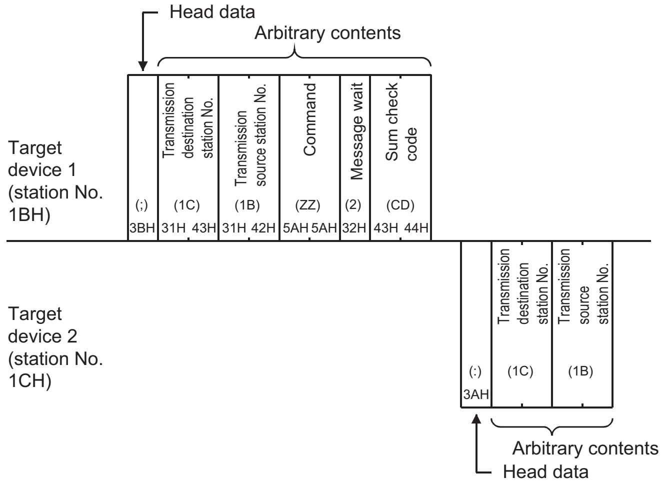
Different message structure from control procedure format frame

■Change the head data of each message to other arbitrary data.

- When format 1, format 2, or format 4 is selected for communication using ASCII code, change ENQ (05H).

■Use arbitrary data array after the head data of each message according to the user specifications.

Ex.


Head data
Arbitrary contents
Target device 1 (station No. 1BH)
Transmission destination station No.
Transmission source station No.
Command
Message wait
Sum check code
(;)
(1C)
(1B)
(ZZ)
(2)
(CD)
3BH
31H 43H
31H 42H
5AH 5AH
32H
43H 44H
Target device 2 (station No. 1CH)
Transmission destination station No.
Transmission source station No.
(:)
(1C)
(1B)
3AH
Arbitrary contents
Head data

Point

For the message structure for general report to all of the other devices except the CPU module station, use an unused station number or a massage structure different from the C24 control procedure format.

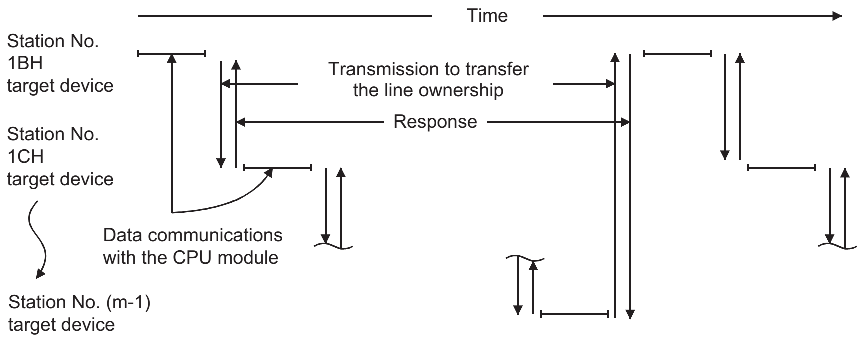
20.3 Examples of Procedure for Data Communications with CPU Module

This section shows examples of procedures to communicate data with the CPU module by configuring interlocks among the target devices.

Sequential data communications between target devices and the CPU module

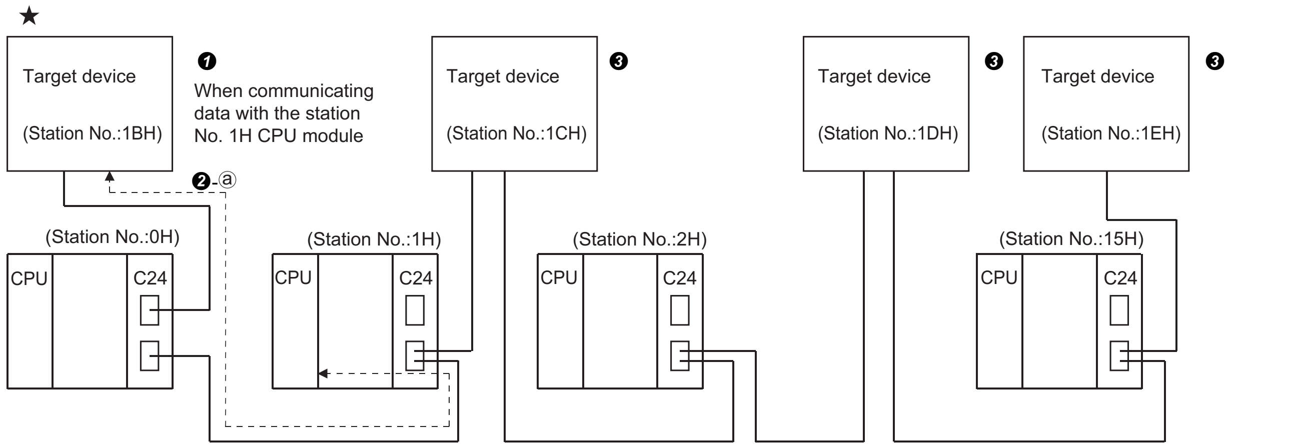
Each target device obtains the line ownership in turns according to the station number to communicate with the CPU module.

Ex.


Time
Station No. 1BH target device
Station No. 1CH target device
Station No. (m-1) target device
Transmission to transfer the line ownership
Response
Data communications with the CPU module

Ex.

The example below shows the procedures for each target device to communicate data with the CPU module.


★
Target device (Station No.:1BH)
❶ When communicating data with the station No. 1H CPU module
❷-ⓐ
Target device (Station No.:1CH)
❸
Target device (Station No.:1DH)
❸
Target device (Station No.:1EH)
❸
(Station No.:0H)
CPU
C24
(Station No.:1H)
CPU
C24
(Station No.:2H)
CPU
C24
(Station No.:15H)
CPU
C24

★: Target device that obtained the line ownership

The above C24 includes Q series C24, L series C24, QC24(N), and UC24.

❶ When the system starts, the target device with the lowest station number (1BH) obtains the line ownership.

- When format 1, format 2, or format 4 is selected for communication using ASCII code, change ENQ (05H).

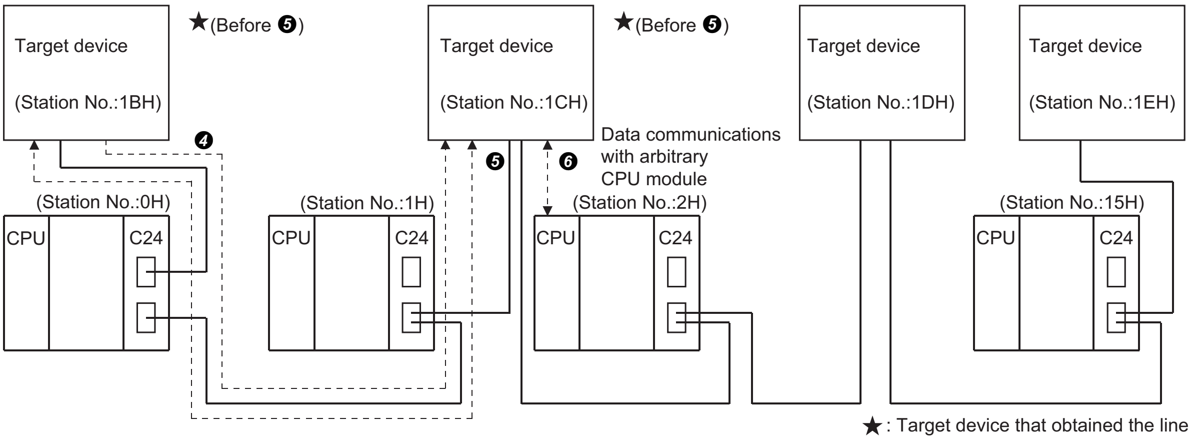
❷ The target device that obtained the line ownership,

- Begins the processing from ❹ after communicating data with the CPU module within the maximum data communication time arranged among the target devices, when communicating data with the CPU module.
- Immediately begins the processing from ❹, when not communicating data with the CPU module.

❸ The target devices that have not obtained the line ownership check the line usage time of the target device that obtained the line ownership and ignore the received data not addressed to their own station.

When the line usage time exceeds the maximum data communication time, each target device performs the processing of ❼.

Ex.



★: Target device that obtained the line ownership

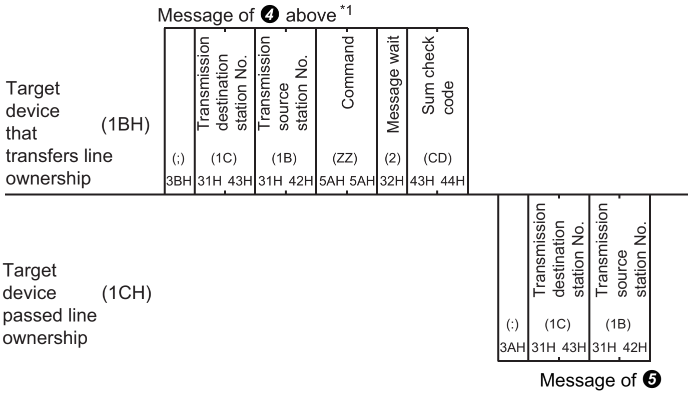
The above C24 includes Q series C24, L series C24, QC24(N), and UC24.

❹ The target device that has communicated data with the CPU module, or a target device that does not need to communicate data with the CPU module transmits the data for transferring the line ownership to the target device with the next station number. A message structure example is shown in ❺.

When a response message (refer to ❺ in this section) is not received from the next target device to which the line ownership to be passed, data transmission for transferring the line ownership to the target device with the next station number is repeated until the line ownership is transferred.

❺ The target device that receives the line ownership transmits a response message to the target device that passes it the line ownership.

Ex.


Message of ❹ above *1
Target device that transfers line ownership (1BH)
Transmission destination station No.
Transmission source station No.
Command
Message wait
Sum check code
(;) 3BH
(1C) 31H 43H
(1B) 31H 42H
(ZZ) 5AH 5AH
(2) 32H
(CD) 43H 44H
Target device passed line ownership (1CH)
Transmission destination station No.
Transmission source station No.
(:) 3AH
(1C) 31H 43H
(1B) 31H 42H
Message of ❺

*1 Command "ZZ" is a command for description purpose. Determine the command to transfer/obtain the line ownership at the user side.

❻ The target device that accepted the line ownership by transmitting the response message performs the processing from ❷.

❼ When the line usage time of the target device that currently has the line ownership exceeds the maximum data communications time,

- The target device, which has the next station number, transmits all of the other devices excluding the CPU module station the general report data and obtains the line ownership and performs the processing❷.

Ex.

Target device that obtains the line ownership (1CH)

	*1 Transmission destination station No.	Transmission source station No.	*2 Command	Message wait	Sum check code
(;)	(1F)	(1C)	(ZZ)	(0)	(0A)
3BH	31H 46H	31H 43H	5AH 5AH	30H	30H 41H

*1 It is a station number for general report to all of the other devices.

*2 Command "ZZ" is a command for description purpose. Determine the command to transfer/obtain the line ownership at the user side.

- The other target devices check if the general report data was received.

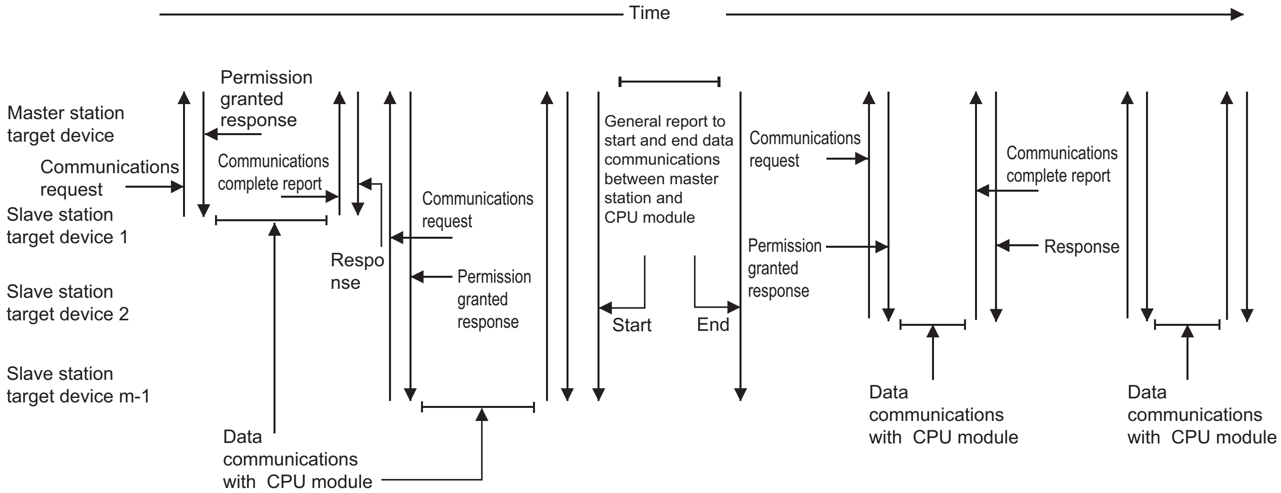
  If the data was received, the target device performs the processing ❸.

  If the data was not received, the next target device transmits general report data and obtains the line ownership and performs the processing ❷ . The other target devices checks if the general report data has been received.

Data communications between target devices and CPU module by designating a master station and slave stations

One of the target devices becomes the master station and the other target devices communicate data with the CPU module after obtaining permission from the master station.

Ex.



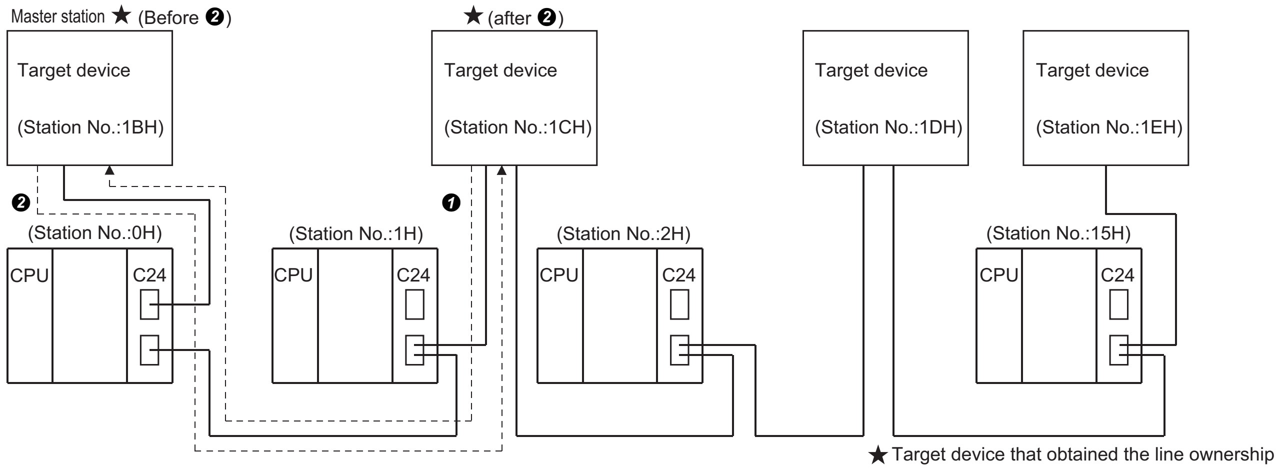
Time
Master station target device
Communications request
Slave station target device 1
Slave station target device 2
Slave station target device m-1
Permission granted response
Communications complete report
Response
Communications request
Permission granted response
Data communications with CPU module
General report to start and end data communications between master station and CPU module
Start
End
Communications request
Permission granted response
Data communications with CPU module
Communications complete report
Response
Data communications with CPU module

The example below shows the procedures for each target device to communicate data with the CPU module.

In this example, after the start of data communications between a target device and the CPU module, each device checks for timeout of the maximum data communication time. In addition, devices on the slave station, which are not communicating data with the CPU module, check if the target device that completed data communications with the CPU module transmitted a communication complete report.

In the following descriptions, the target device with the lowest station number(1BH) is the master station and the other target devices are slave stations.

Ex.



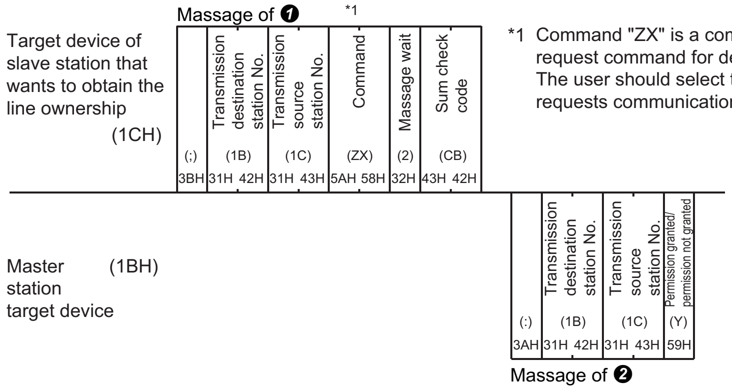
Master station ★ (Before ❷)
★ (after ❷)
Target device (Station No.:1BH)
Target device (Station No.:1CH)
Target device (Station No.:1DH)
Target device (Station No.:1EH)
❷
❶
(Station No.:0H)
(Station No.:1H)
(Station No.:2H)
(Station No.:15H)
CPU
C24
★Target device that obtained the line ownership

The above C24 includes Q series C24, L series C24, QC24(N), and UC24.

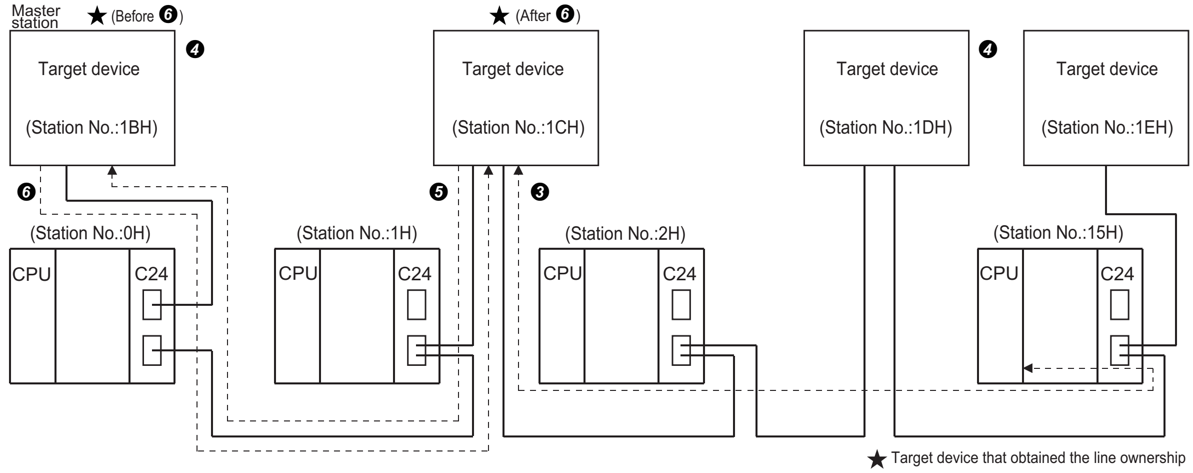
❶ A slave station that wants to communicate data with the CPU module sends a communications request to the master station to obtain the line ownership. A message structure example is shown in ❷.

❷ The master station transmits a permission response to the slave station that issued the communications request.

Ex.


Massage of ❶
*1
Target device of slave station that wants to obtain the line ownership
(1CH)
Transmission destination station No.
Transmission source station No.
Command
Massage wait
Sum check code
(;) (1B) (1C) (ZX) (2) (CB)
3BH 31H 42H 31H 43H 5AH 58H 32H 43H 42H
Master station target device
(1BH)
Transmission destination station No.
Transmission source station No.
Permission granted/ permission not granted
(:) (1B) (1C) (Y)
3AH 31H 42H 31H 43H 59H
Massage of ❷

*1 Command "ZX" is a communications request command for description purpose. The user should select the command that requests communications.



The above C24 includes Q series C24, L series C24, QC24(N), and UC24.

❸ The slave station that received the permission response performs the processing from ❺ after communicating data with the CPU module within the maximum data communication time arranged among the target devices.

❹ The master station that transmitted the permission response and the slave stations that does not obtain line ownership check the line usage time of the slave station that obtained the line ownership, and ignore the received data not addressed to their own station.

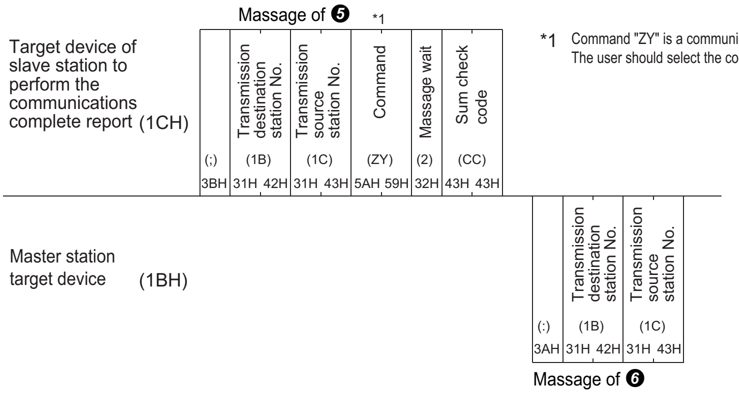
If the line usage time exceeds the maximum communication time, each target device performs the processing ❼.

❺ The slave station that has communicated data with the CPU module transmits a communication complete report to the master station after data communications completed. A message structure example is shown in ❻.

Slave stations that do not communicate data with the CPU module check if a communications complete report was transmitted, and do not communicate data with the master station during that time.

❻ The master station that received the communication complete report transmits a response to the slave station that transmitted the communication complete report.

Ex.


Massage of ❺
*1
Target device of slave station to perform the communications complete report (1CH)
Transmission destination station No.
Transmission source station No.
Command
Massage wait
Sum check code
(;) (1B) (1C) (ZY) (2) (CC)
3BH 31H 42H 31H 43H 5AH 59H 32H 43H 43H
Master station target device (1BH)
Transmission destination station No.
Transmission source station No.
(:) (1B) (1C)
3AH 31H 42H 31H 43H
Massage of ❻

*1 Command "ZY" is a communications complete report command for description purpose. The user should select the communications complete report command.

❼ After the completion of ❻ above, or when the line usage time of the slave station that obtained the line ownership exceeds the maximum data communication time,

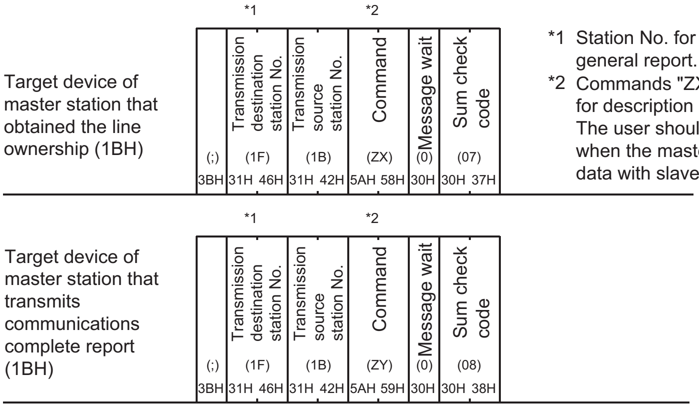
- The master station waits for a communication request from a slave station.
  When the master station receives a communication request, it performs processing from ❷.
- The slave stations do not communicate data with the master station until data communication with the CPU module is required.
  When data communications with the CPU module is required, they perform the processing from ❶.

❽ When the master station itself communicates data with the CPU module, it transmits all devices excluding the CPU module station a general report data and obtains the line ownership before communicating data with the CPU module while slave stations do not have the line ownership.

After data communications with the CPU module were completed, the master station transmits general report data to inform the slave stations that data communications with the CPU module are completed.

During this time, the slave stations do not communicate data with the master station until master station data communications have been completed.

Ex.


*1 *2
Target device of master station that obtained the line ownership (1BH)
Transmission destination station No.
Transmission source station No.
Command
Message wait
Sum check code
(;) (1F) (1B) (ZX) (0) (07)
3BH 31H 46H 31H 42H 5AH 58H 30H 30H 37H
*1 *2
Target device of master station that transmits communications complete report (1BH)
Transmission destination station No.
Transmission source station No.
Command
Message wait
Sum check code
(;) (1F) (1B) (ZY) (0) (08)
3BH 31H 46H 31H 42H 5AH 59H 30H 30H 38H

*1 Station No. for all target devices general report.

*2 Commands "ZX" and "ZY" are commands for description purposes. The user should select the command when the master station communicates data with slave stations.

21 SWITCHING THE MODE AFTER STARTING

This function forcefully switches the current communication protocol and transmission specifications of a designated interface from a target device and by program after C24 starts up.

C24 starts operating with the setting values in "Module Parameter" of Engineering tool at the time of the startup.

Point

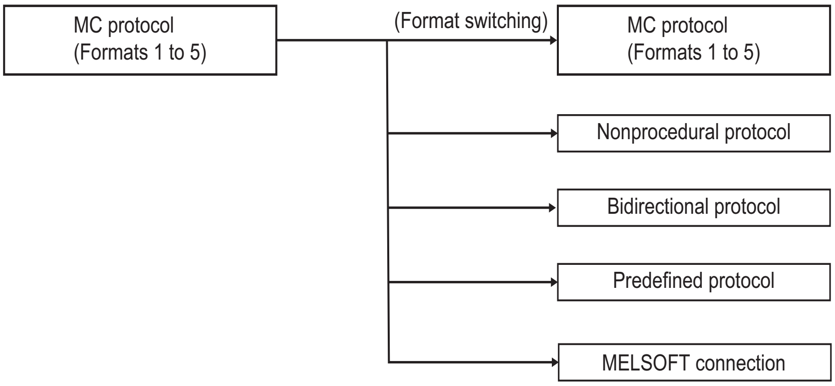
This function is used to change the communication protocol and transmission specification of the designated interface and continue data communications without restarting the CPU module. For mode switching by using the dedicated instruction (UINI instruction) or the module FB (InitialUnit instruction), refer to the following section.

☞ Page 580 Dedicated Instructions

📖MELSEC iQ-R Serial Communication Module Function Block Reference

Mode switching from target device

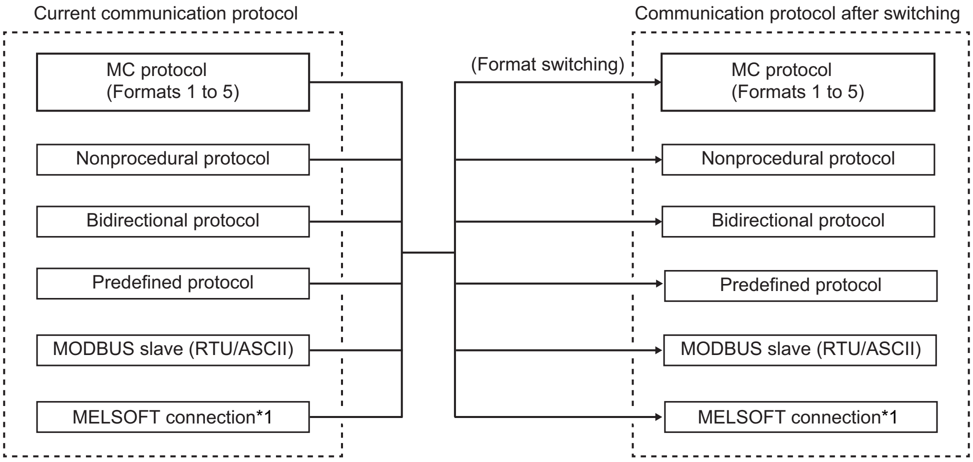
- Mode switching can be performed when the communication protocol of an interface connected to the target device is the MC protocol.
- The communication protocol can be changed from the MC protocol as shown below.



- The transmission specifications set with the parameter setting of Engineering tool can be changed.
- When switching the mode to MODBUS (slave function), use a C24 the firmware version of which is '13' or later.
- The mode cannot be switched to the simple CPU communication.

Mode switching by program

- Regardless of the current communication protocol of an interface connected to the target device, mode switching can be performed by program.
- The communication protocol can be changed by program as shown below.



*1 Change from MELSOFT connection to MELSOFT connection is not available.

- The transmission specifications set with the parameter setting of Engineering tool can be changed.
- The mode cannot be switched from/to the simple CPU communication.

Point

When using the mode switching function, set "Setting change" of the Engineering tool parameter settings (by selecting [Basic settings] ⇨ [Various control specification]⇨[Transmission Setting]) to "Enable".

21.1 Mode Switching Operation and Contents that can be Changed

This section describes the settings that can be changed with the mode switching function and the operation of C24 after mode switching.

Settings that can be changed with mode switching

The settings that can be changed with mode switching are as shown below.

Switching the communication protocol

- The communication protocol setting of each interface can be switched.
- The communication protocol after mode switching is specified with the 'Switching mode No. designation' (Un\G144/304) in the area for designation of mode switching on the buffer memory.

Switching the transmission specifications

- The transmission setting of each interface can be switched.
- The transmission setting after mode switching is specified with the 'Transmission specifications after switching designation' (Un\G145/305) in the area for designation of mode switching on the buffer memory.

Point

With the mode switching function, the settings corresponding to "Communication protocol setting" and "Transmission setting" in the parameter settings of Engineering tool can be changed.

Operation for mode switching

The following describes C24 operation for mode switching.

Processing currently in execution

- When a mode switching request is issued, the mode switching process starts immediately.
- If any of the following processing is performed when a mode switching request is issued, the processing will be terminated.

Processing in execution	Operation for mode switching
Data communication using the MC protocol	• Receiving command messages and sending response messages or on-demand data are all terminated. • The transmission completion signal for a transmission request of on-demand data does not turn ON.
Data communication using the nonprocedural protocol, bidirectional protocol, and predefined protocol	• Transmission processing and reception processing are all terminated. • All input signals from a CPU module related to transmission processing and reception processing turn OFF. • While received data from the target device was being stored to C24, the received data up to that point is ignored and the data is processed with the current receive data quantity treated as "0".
Data communication using the MODBUS slave function	Receiving command messages and sending response messages are all terminated.

Modification of buffer memory stored value

- Special applications area (Un\G594 to 595, Un\G610 to 611)
  The communication protocol status and transmission specifications after mode switching has been completed are stored.
  The values stored to areas other than the above are not changed. The contents before mode switching are preserved.
- User setting area (Un\G1024 to 6911, Un\G9728 to 16383)
  The stored values are not changed. The contents before mode switching are preserved.
- User setting area for MODBUS (Un\G49152 to 53247)
  The stored values are not changed. The contents before mode switching are preserved.

21.2 Considerations for Mode Switching

Modification of buffer memory stored value

Determine the following items between the target device and the CPU module to prevent mode switching during data communications.

- How to perform mode switching - from the target device or by program
- Timing for each mode switching pattern (☞ Page 411 Operation for mode switching)
- Interlocking of all connected devices for mode switching

  Method to configure interlock when informing all of the connected stations the mode switching execution

  Method and message structure when informing all of the connected stations the mode switching completion

  Device number and meaning of the value when a CPU module word device is used

Transmission specifications for each communication protocol

The transmission specifications must be changed depending on the communication protocol when switching the mode.

For the transmission specifications for each communication protocol, refer to the following section.

☞ Page 456 Various control specification

Mode switching from target device

- Once the mode has been switched to a mode other than MC protocol (Formats 1 to 5), it cannot be changed from the target device.

  In this case, switch the mode by program.
- Mode switching from the target device can be performed to only C24 connected to a target device (including connected stations by multidrop connection).

  It is not available for C24s on the other stations via network system. (📖 MELSEC Communication Protocol Reference Manual)

Point

It is recommended to perform mode switching by program.

Mode switching of the MODBUS slave function

When switching the mode to the MODBUS slave, set values for the following buffer memories in advance.

- MODBUS device assignment parameter (Un\G30976 to 31359)
- Setting area for MODBUS (Un\G28680 to 28689)

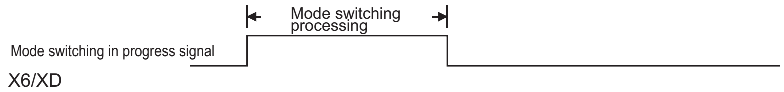
When using a C24 the firmware version of which is '12' or earlier, mode switching to/from the MODBUS slave can be performed only by using I/O signals for handshake with a CPU module.

■Switching by using the dedicated instruction or MC protocol

When switching the mode by using the dedicated instruction (UINI) or MC protocol, use a C24 the firmware version of which is '13' or later.

Data communications after mode switching

While the mode switching is being executed, data communications between C24 and the target device cannot be performed, including the processing time to store the setting value to the special applications area on the buffer memory.



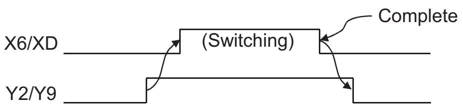
Mode switching related to interlink operation

Do not switch the mode when the two interfaces of C24 are in the interlink operation. Also, do not switch the mode for the interlink operation.

21.3 I/O Signals for Handshake with CPU Module and Buffer Memory

This section explains the I/O signals for handshake and the buffer memories used when mode switching is performed.

I/O signals for handshake with CPU module

Type	I/O signal		Signal name	Device turned ON/ OFF	Timing
	CH1	CH2			
Mode switching	X6	XD	Switching mode	C24	
	Y2	Y9	Mode switching request	CPU	

Point

The following signals can also be used as I/O signals, in addition to the above.

- Module READY signal (X1E): Turns ON when C24 can be accessed from the CPU module.
- Watchdog timer error signal (X(n+1)F): Turns ON when C24 does not operate normally.
- 'CH1 Error occurrence' (XE): Turns ON when error occurred on the CH1 side.
- 'CH2 Error occurrence' (XF): Turns ON when error occurred on the CH2 side.

For details on the CPU module I/O signals, refer to the following section.

(☞ Page 545 Input/Output Signal List)

Buffer memory

Address (DEC (HEX))		Name		Setting value/Stored value
CH1	CH2			
144 (90H)	304 (130H)	For designation of mode switching	Switching mode No. designation (☞ Page 413 Switching mode No. designation (Un\G144/304))	0001H: MC protocol (Format 1) to 00FFH: MELSOFT connection
145 (91H)	305 (131H)		Transmission specifications after switching designation(☞ Page 414 Transmission specifications after switching designation (Un\G145/305))	0000H : Match the setting in MELSOFT. 8000H to 8FFFH: Match the setting to that of this area
515 (203H)		For confirmation of parameter setting and mode switching	Parameter setting error and mode switching error status	0: No error Other than 0: Parameter setting error, mode switching error (☞ Page 519 Error Code List)

■Switching mode No. designation (Un\G144/304)

Designate a mode number after mode switching.

- 0000H (default): Match the setting in MELSOFT.
- 0001H: MC protocol (format 1)
- 0002H: MC protocol (format 2)
- 0003H: MC protocol (format 3)
- 0004H: MC protocol (format 4)
- 0005H: MC protocol (format 5)
- 0006H: Nonprocedural protocol
- 0007H: Bidirectional protocol
- 0009H: Predefined protocol
- 000AH: MODBUS slave (RTU)
- 000BH: MODBUS slave (ASCII)
- 00FFH: MELSOFT connection[*1]

[*1] Specify "00H" for the communication protocol setting when specifying the MELSOFT connection mode with the parameter settings of Engineering tool.

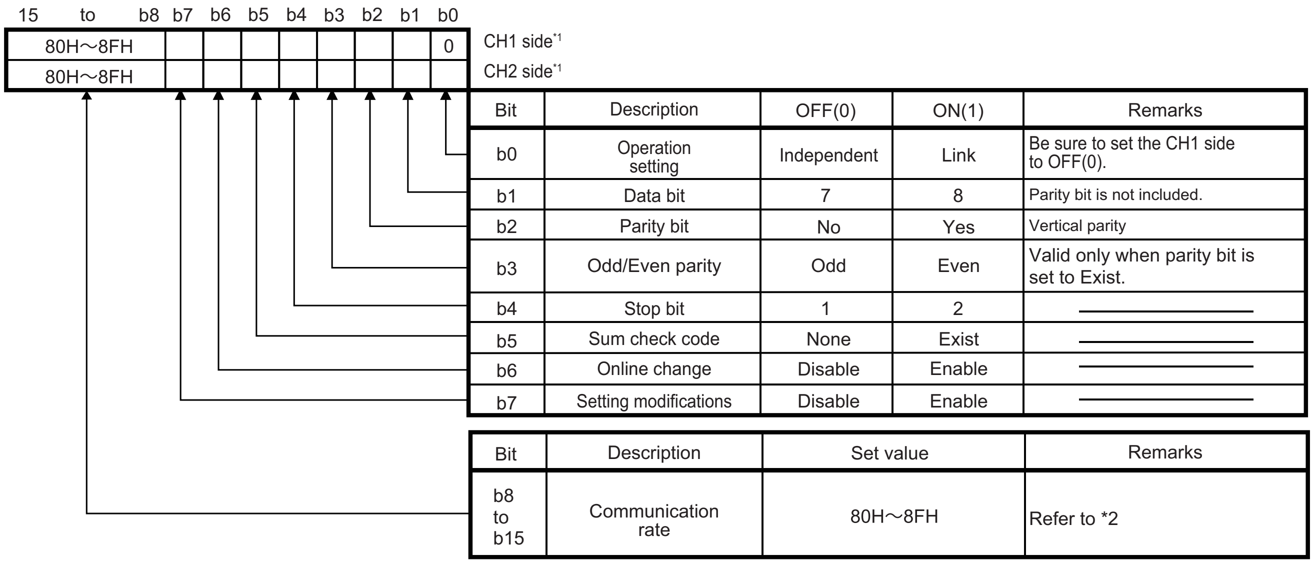
■Transmission specifications after switching designation (Un\G145/305)

Specify the transmission specifications after mode switching.

Write "0000H" when returning to the settings configured with Engineering tool.

When setting arbitrary transmission specifications (other than the transmission specification set with Engineering tool), write the value corresponding to ON/OFF of the relevant bit. The relevant bit 1 (ON)/0 (OFF) is specified as the same as the transmission setting of Engineering tool.

- For C24



Bit	Description	OFF(0)	ON(1)	Remarks
b0	Operation setting	Independent	Link	Be sure to set the CH1 side to OFF(0).
b1	Data bit	7	8	Parity bit is not included.
b2	Parity bit	No	Yes	Vertical parity
b3	Odd/Even parity	Odd	Even	Valid only when parity bit is set to Exist.
b4	Stop bit	1	2	――――
b5	Sum check code	None	Exist	――――
b6	Online change	Disable	Enable	――――
b7	Setting modifications	Disable	Enable	――――

Bit	Description	Set value	Remarks
b8 to b15	Communication rate	80H～8FH	Refer to *2

*1 On the side of interface to which "MELSOFT connection" is set for the switching mode No. designation, set all items to the OFF side. C24 operates with the setting values set with Engineering tool.

Transmission setting	Engineering tool side setting
Operation setting	Independent
Data bit	8
Parity bit	Yes
Odd/even parity	Odd
Stop bit	1
Sum check code	Yes
Online change	Enable
Setting change	Enable

*2 It indicates the setting value for the communication speed. Set "80H" to the interface to which "MELSOFT connection" is set for the switching mode No. designation. The communication speed set on the Engineering tool side is applied.

Communication speed (units: bps)	Bit position	Communication speed (units: bps)	Bit position	Remarks
	b15 to b8		b15 to b8	
1200	82H	38400	89H	Decrease the communication speed when data cannot be communicated normally due to overrun error, framing error, etc. in communication with the target device.
2400	83H	57600	8AH	
4800	84H	115200	8BH	
9600	85H	230400	8CH	
14400	86H	-	-	

Precautions

The following I/O signals for handshake and buffer memories need to be considered on the CPU module side at the time of mode switching from the target device.

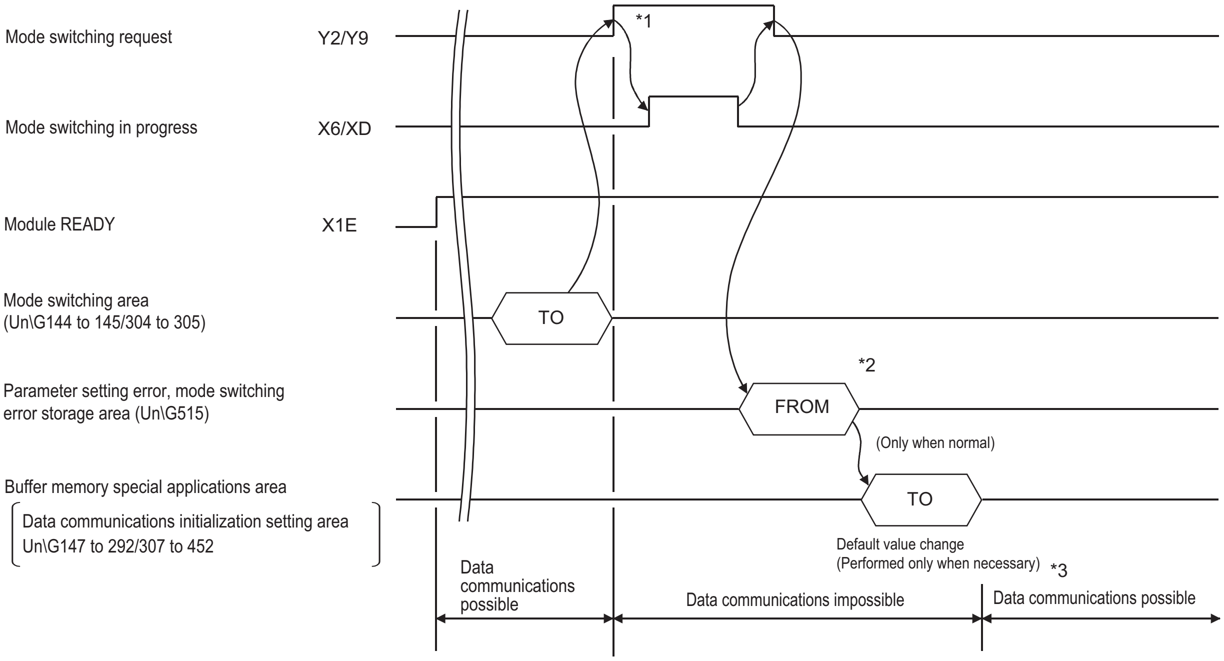
- 'CH□ switching mode' (X6/XD)
- 'Parameter setting error, mode switching error status' (Un\G515)

21.4 Mode Switching Method by Program

This section shows how to switch the C24 mode by program.

Mode switching procedure

The procedures for switching C24 mode by program are as shown below.


Mode switching request
Y2/Y9
*1
Mode switching in progress
X6/XD
Module READY
X1E
Mode switching area
(Un\G144 to 145/304 to 305)
TO
Parameter setting error, mode switching
error storage area (Un\G515)
*2
FROM
(Only when normal)
Buffer memory special applications area
Data communications initialization setting area
Un\G147 to 292/307 to 452
TO
Default value change
(Performed only when necessary)
*3
Data communications possible
Data communications impossible
Data communications possible

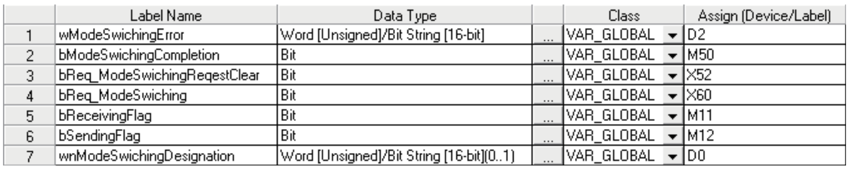
*1 Inform in advance all the connected device that data communications cannot be performed due to mode switching.

*2 When an error occurs after checking 'Parameter setting error, mode switching error status' (Un\G515), take the corresponding actions. (☞ Page 519 Error Code List)
· Checking of mode switching designation contents for buffer memory mode switching area and writing of mode switching designation contents within the range that can be designated
· Re-execution of mode switching

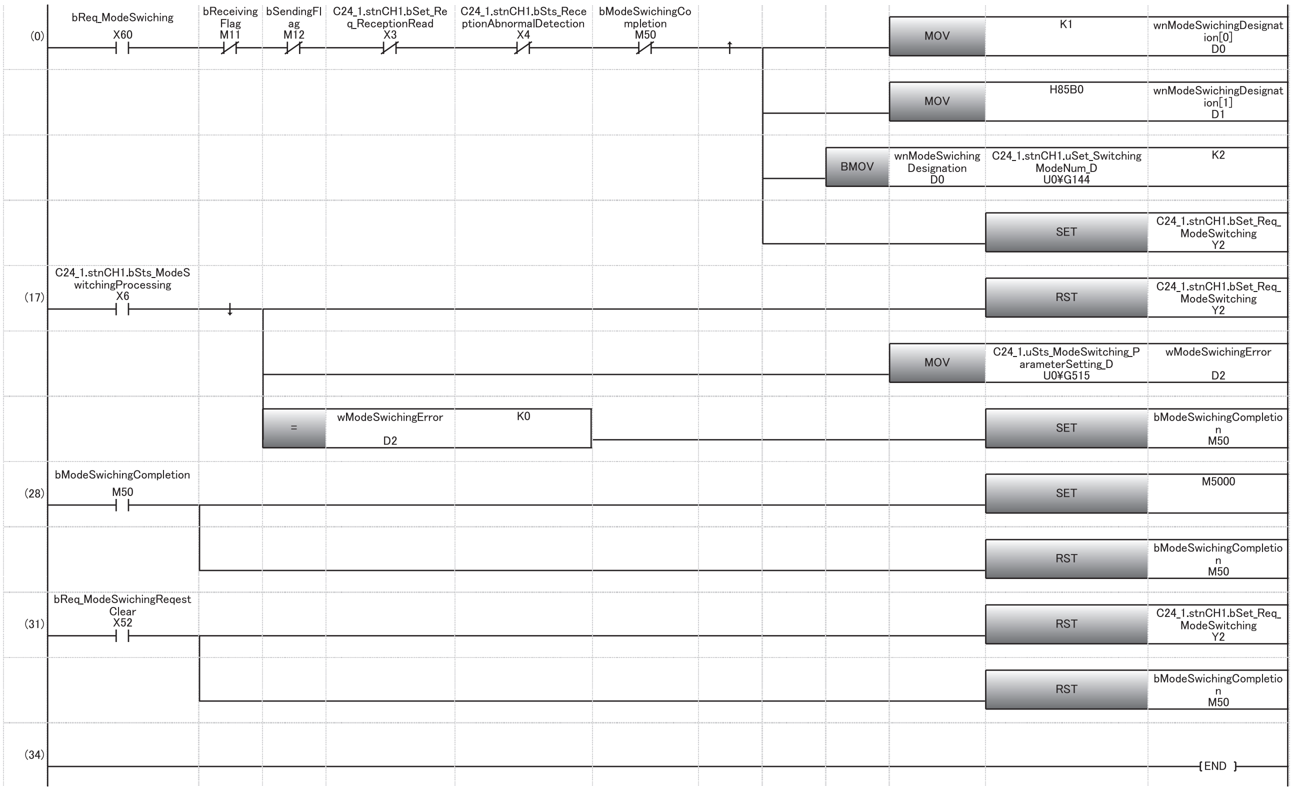
*3 After confirming that mode switching was completed normally, inform all the connected devices that data communications are possible, and restart data communications.
To check C24 mode (communication protocol, transmission specifications) after mode switching, read the buffer memory (Un\G594 to 595/610 to 611). (☞ Page 489 Current operation status reading method)

■Program example

The following shows an example of program for mode switching of the CH1 side interface.

Category	Label name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X3
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection	CH1 Reception abnormal detection	X4
	C24_1.stnCH1.bSts_ModeSwitchingProcessing	CH1 Switching mode	X6
	C24_1.stnCH1.bSet_Req_ModeSwitching	CH1 Mode switching request	Y2
	C24_1.stnCH1.uSet_SwitchingModeNum	CH1_Switching mode No. designation	U0\G144
	C24_1.uSts_ModeSwitching_ParameterSetting_D	Parameter setting error, mode switching error status	U0\G515
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	wModeSwichingError	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2
2	bModeSwichingCompletion	Bit	...	VAR_GLOBAL	M50
3	bReq_ModeSwichingReqestClear	Bit	...	VAR_GLOBAL	X52
4	bReq_ModeSwiching	Bit	...	VAR_GLOBAL	X60
5	bReceivingFlag	Bit	...	VAR_GLOBAL	M11
6	bSendingFlag	Bit	...	VAR_GLOBAL	M12
7	wnModeSwichingDesignation	Word [Unsigned]/Bit String [16-bit](0..1)	...	VAR_GLOBAL	D0


(0)
bReq_ModeSwiching X60
bReceiving Flag M11
bSendingFlag M12
C24_1.stnCH1.bSet_Req_ReceptionRead X3
C24_1.stnCH1.bSts_ReceptionAbnormalDetection X4
bModeSwichingCompletion M50
MOV K1 wnModeSwichingDesignation[0] D0
MOV H85B0 wnModeSwichingDesignation[1] D1
BMOV wnModeSwiching Designation D0 C24_1.stnCH1.uSet_SwitchingModeNum_D U0¥G144 K2
SET C24_1.stnCH1.bSet_Req_ModeSwitching Y2
(17)
C24_1.stnCH1.bSts_ModeSwitchingProcessing X6
RST C24_1.stnCH1.bSet_Req_ModeSwitching Y2
MOV C24_1.uSts_ModeSwitching_ParameterSetting_D U0¥G515 wModeSwichingError D2
= wModeSwichingError D2 K0
SET bModeSwichingCompletion M50
(28)
bModeSwichingCompletion M50
SET M5000
RST bModeSwichingCompletion M50
(31)
bReq_ModeSwichingReqestClear X52
RST C24_1.stnCH1.bSet_Req_ModeSwitching Y2
RST bModeSwichingCompletion M50
(34)
END

(0) Change the operation mode to MC protocol format 1.
Set the transmission specifications.
Write the mode switching designation contents to the buffer memory.
Set the mode switching request.

(17) Reset the mode switching request signal.
Read the mode switching request contents.
Mode switching completes.

(28) Change the setting value of the communication protocol according to the communication specification.
Mode switching complete reset

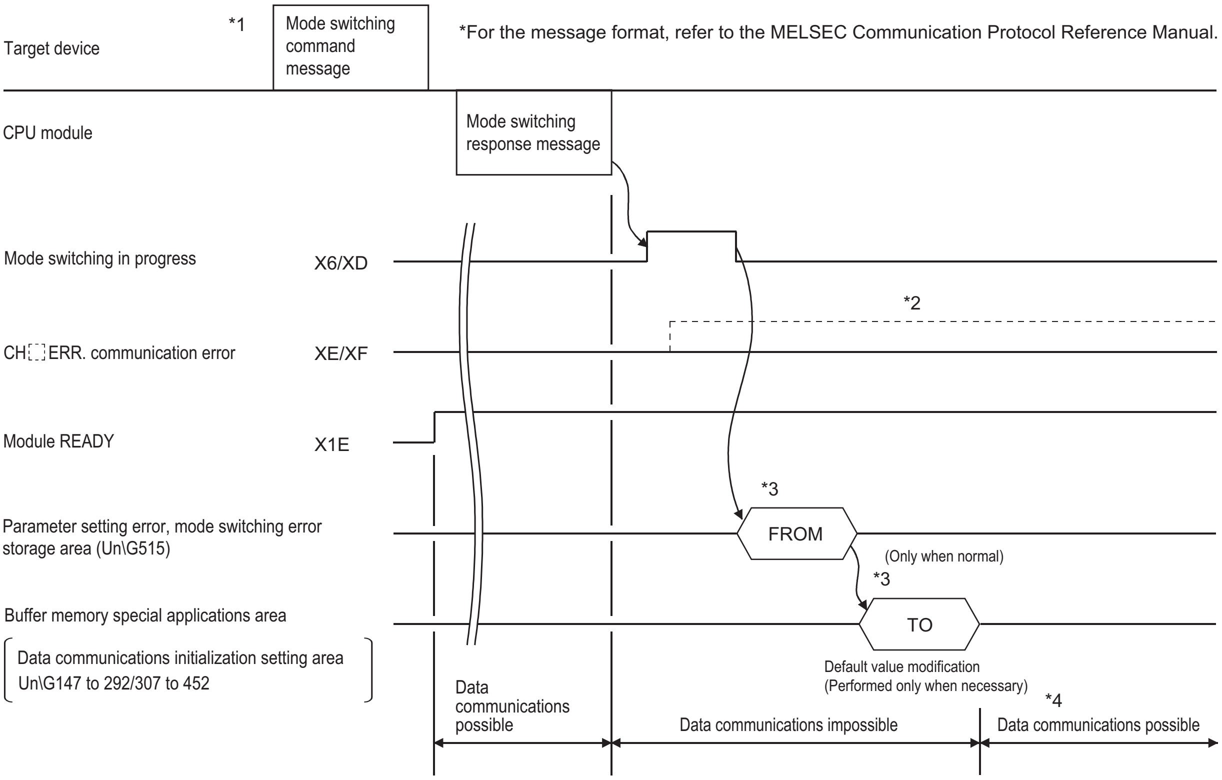
(31) Clear the mode switching request with the clear direction.

21.5 How to Switch Mode from Target Device

This section shows how to switch the C24 mode from the target device.

Mode switching procedure

The procedures for switching C24 mode from the target device.



Target device
*1
Mode switching command message
*For the message format, refer to the MELSEC Communication Protocol Reference Manual.
CPU module
Mode switching response message
Mode switching in progress
X6/XD
*2
CH□ERR. communication error
XE/XF
Module READY
X1E
*3
Parameter setting error, mode switching error storage area (Un\G515)
FROM
(Only when normal)
*3
Buffer memory special applications area
Data communications initialization setting area Un\G147 to 292/307 to 452
TO
Default value modification
(Performed only when necessary)
*4
Data communications possible
Data communications impossible
Data communications possible

*1 Inform in advance all the connected device that data communications cannot be performed due to mode switching.

*2 When 'CH□ Error occurrence' (XE, XF) turns ON, check the error contents and take the corresponding actions. (☞ Page 519 Error Code List)

· Checking of mode switching designation contents for buffer memory mode switching area and writing of mode switching designation contents within the range that can be designated

· Re-execution of mode switching

*3 When mode switching is performed from the target device, perform the read/write operation from/to the special applications area on the buffer memory shown below from the CPU module after mode switching is completed.

· 'Parameter setting error, mode switching error status' (Un\G515)

· Initial setting areas for data communication (Un\G147 to 289/307 to 449) (☞ Page 550 Buffer memory list)

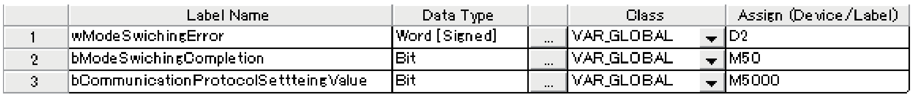
*4 After confirming that mode switching was completed normally, inform all the connected devices that data communications are possible, and restart data communications.

Point

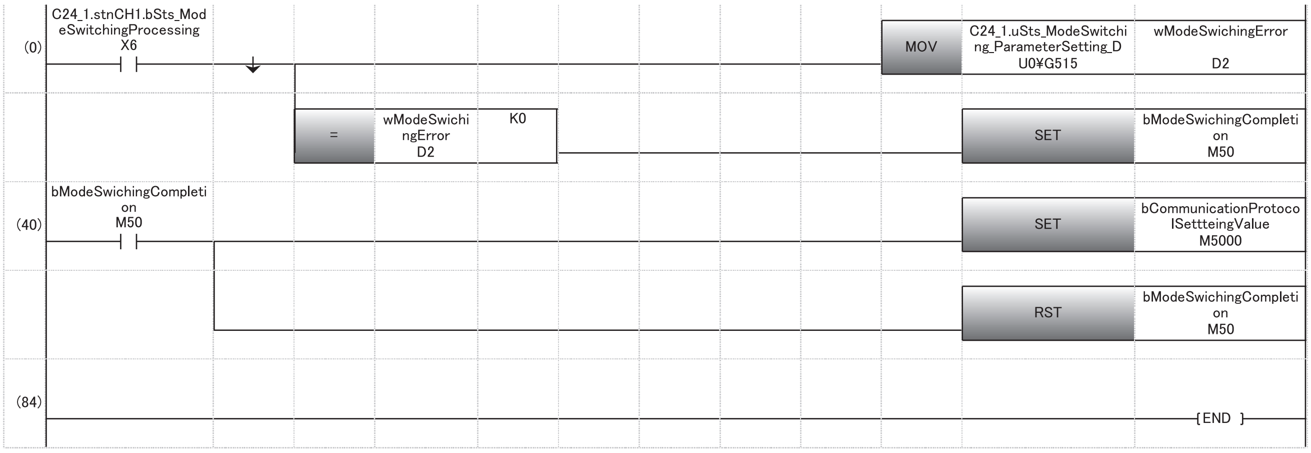
To check C24 mode (communication protocol, transmission specifications) after mode switching, read the buffer memory (Un\G594 to 595/610 to 611). (Read the buffer memory with the MC protocol buffer memory read function when checking by the target device.) (📖 MELSEC Communication Protocol Reference Manual)

■Program example

The following shows an example of program on the CPU module side for mode switching of the CH1 side interface from the target device.

Category	Label name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.bSts_ModeSwitchingProcessing	CH1 Switching mode	X6
	C24_1.uSts_ModeSwitching_ParameterSetting_D	Parameter setting error, mode switching error status	U0/G515
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	wModeSwichingError	Word [Signed]	...	VAR_GLOBAL	D2
2	bModeSwichingCompletion	Bit	...	VAR_GLOBAL	M50
3	bCommunicationProtocolSettteingValue	Bit	...	VAR_GLOBAL	M5000



(0) Read the mode switching request contents.
Mode switching is completed.

(10) Change the setting value of the communication protocol according to the communication specification.
Mode switching complete reset

22 USING SEND/RECEIVE DATA MONITORING FUNCTION

The following describes C24 send/receive data monitoring function.

22.1 Send/receive Data Monitoring Function

Overview

The send/receive data monitoring function is a function to monitor send/receive data between the C24 and a target device on communication lines.

The system startup phase can be shortened by analyzing the send/receive data on the communication lines when debugging the program.

The two methods below are available for the send/receive data monitoring method.

This chapter explains the send/receive data monitoring by the program.

Send/receive data monitoring by program

Monitoring is performed by writing the monitor start instruction to the send/receive data monitoring designation area of the buffer memory.

Send/receive data monitoring by Engineering tool

Monitoring (circuit trace) is performed by using the circuit trace function of Engineering tool (☞ Page 142 Circuit Trace).

Send/receive data monitoring operation

The following describes the send/receive data monitoring operation.

Point

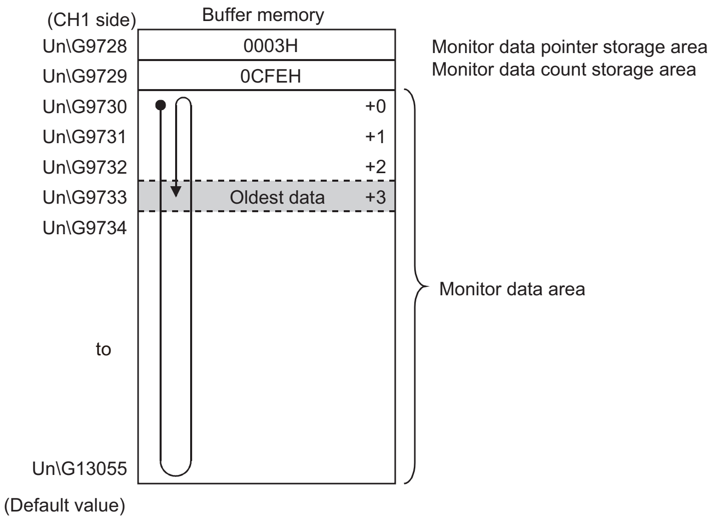
- When using the send/receive data monitoring function, the total transmission speed of the two interfaces should not exceed 230400 bps.

Start monitoring

- Write the monitor start instruction "0001H" to the send/receive data monitoring designation (Un\G8216/8472). (Set by user)
- By the monitor start instruction, the monitor data pointer/monitor data size area are cleared to '0' and "0002H" is written in the send/receive data monitoring designation area to start monitoring. (Set by C24)
  Monitor data area (Un\G9730 to 13055/13058 to 16383 (default)) is not cleared to '0'.
- When a monitor setting error occurs, "100FH" is written to the send/receive data monitoring designation area. (Set by C24)
  Review the monitor buffer start address and the monitor buffer size setting range and perform monitoring again.

During monitoring

- Once monitoring starts, data of reception, transmission, data reception error, and signal change are stored from the start address of the monitor data area in the order of occurrence.
- When the monitored data exceeds the monitor data area size, monitoring is continued by writing new data over the oldest data.
  However, monitoring may automatically stop according to the settings of the 'Monitor option specification' (Un\G8217/8473). (☞ Page 423 Monitor option specification (Un\G8217/8473))



- Each data is stored in the monitor data area at the timing shown below.

Data classification		Data storage timing
Receive data		When data is received
Send data		When data is transmitted
Reception error data		When a reception error is detected
RS-232 line	RS (RTS), ER (DTR) signals	When the RS (RTS), ER (DTR) signals change
	CS (CTS), DR (DSR), CD (DCD) signals	When signal changes are detected during the periodic processing (operation interval: 1 to 19 ms) and at the each timing described above (Monitoring is not performed to changes during the periodic processing)

Stop monitoring

Monitoring stops when any of the following conditions is satisfied.

- For monitor stop by user instructions
  Monitoring stops when "0000H" is written in the send/receive data monitoring designation area of the buffer memory. (Set by user)
- For monitor stop by the system when the monitor data area is full.
  The stop at buffer full specification (bit 0) of the monitor option specification area of the buffer memory is set to "ON".
  When the monitor data area is full (monitor buffer size), monitoring is automatically stopped and "1002H" is written in the send/receive data monitoring designation area. (Set by C24)
- For monitor stop by the system when the timer 0 error occurs
  The 0 timer error outbreak stop designation (bit 2) of the monitor option specification area of the buffer memory is set to "ON".
  When the timer 0 error (error code 7F40H) occurs, monitoring is automatically stopped and "1002H" is written in the send/receive data monitoring designation area. (Set by C24)
  Part of the data immediately after the occurrence of a timer 0 error may be entered in the monitor data area.

22.2 Send/receive Data Monitoring Function Settings

This section explains how to configure the settings for the send/receive data monitoring function.
The settings are configured with Engineering tool or by program.

Setting with Engineering tool

The setting items for the send/receive data monitoring function are listed below.

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

■Parameter settings

Parameter setting item				Description
Basic settings	Various control specification	Send/receive data monitoring function specification	Send/receive data monitoring function specification	Select "monitor start instruction" for constant monitoring.
			Stop at buffer full specification	Select "ON" to stop monitoring when the monitor data area is full.
			0 timer error outbreak stop designation	Select "ON" to stop monitoring when a timer 0 error occurs.
			Monitor buffer start address designation	Designate the value to change the monitor buffer address. (Default: 2600H/3300H)
			Monitor buffer size designation	Designate the value to change the monitor buffer size (Default: 0D00H)

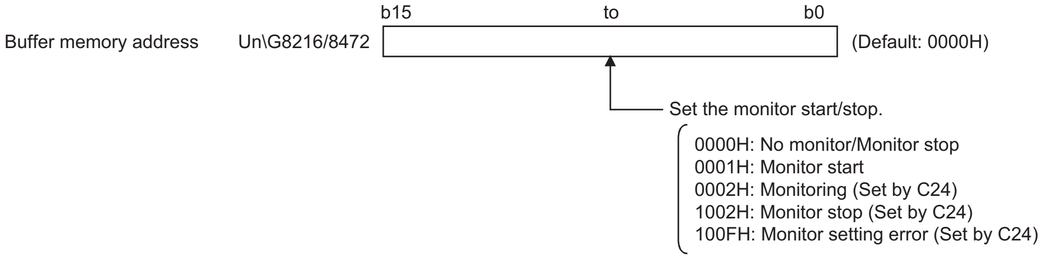
Buffer memory for send/receive data monitoring function

Send/receive data monitoring designation (Un\G8216/8472)

Designate the start/stop of the send/receive data monitoring function.

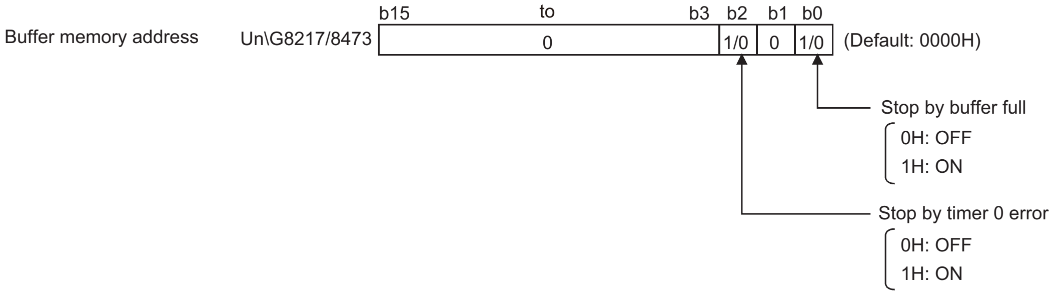
For details on the send/receive data monitoring designation operation, refer to the following section.

☞ Page 420 Send/receive data monitoring operation


b15
to
b0
Buffer memory address
Un\G8216/8472
(Default: 0000H)
Set the monitor start/stop.
0000H: No monitor/Monitor stop
0001H: Monitor start
0002H: Monitoring (Set by C24)
1002H: Monitor stop (Set by C24)
100FH: Monitor setting error (Set by C24)

Monitor option specification (Un\G8217/8473)

Designate the options of the send/receive data monitoring function.



■Stop at buffer full specification (bit 0)

- When the stop at buffer full specification is OFF
  When the monitored data exceeds the monitor data area size, monitoring is continued by writing new data over the oldest data.
- When stop at buffer full specification is ON
  When the monitor data area is full (monitor buffer size), monitoring is automatically stopped and "1002H" is written in the send/receive data monitoring designation area. (Set by C24)

■0 timer error outbreak stop designation (bit 2)

- When the o timer error outbreak stop designation is OFF
  Monitoring continues even if the timer 0 error (error code 7F40H) occurs.
- When the 0 timer error outbreak stop designation is ON
  When the timer 0 error (error code 7F40H) occurs, monitoring is automatically stopped and "1002H" is written in the send/receive data monitoring designation area. (Set by C24)

Monitor buffer start address designation (Un\G8218/8474)

Designate the start address of the storage area (monitor data area) for the monitored data within the range of the user setting area of the buffer memory.

Setting range: 0400H to 1AFDH, 2600H to 3FFDH (Default: 9728/13056 (2600H/3300H))

Monitor buffer size designation (Un\G8219/8475)

Designate the size of the monitor data area.

Setting range: 0003H to 1A00H (Default: 0D00H)

Monitor data pointer storage area (Un\G9728/13056 (default))

■The oldest data position of the monitor data area is stored using the offset value from the start address of the monitor data area.

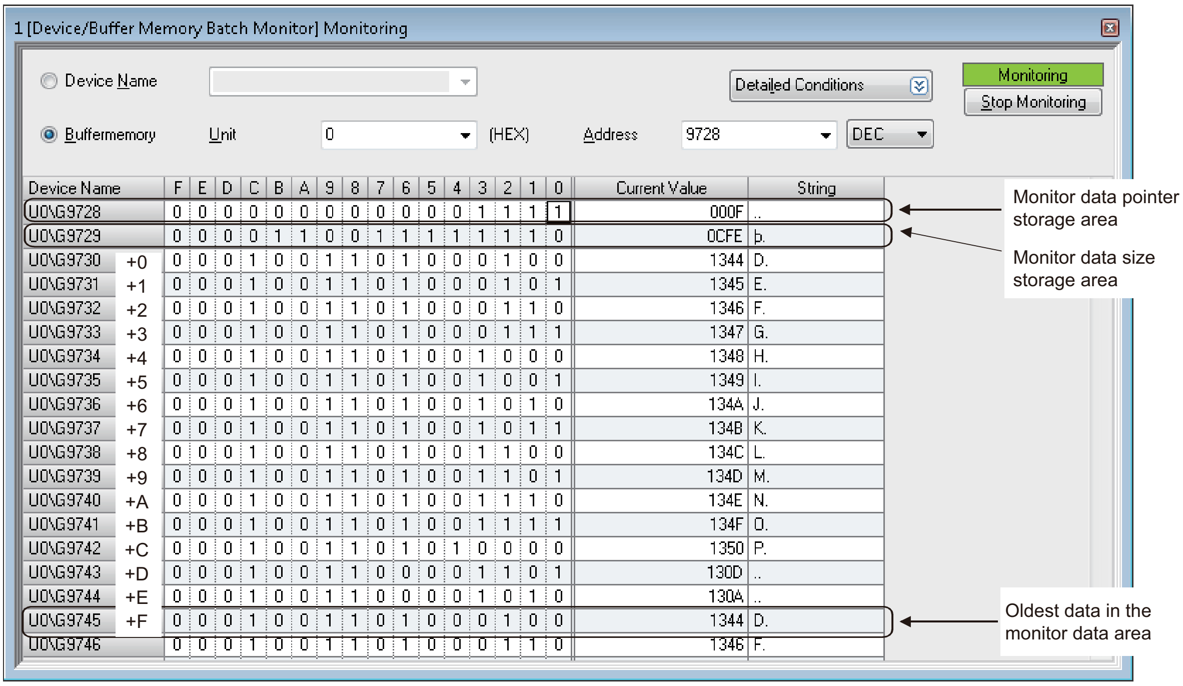
0 to (monitor buffer size designation (default: 0D00H) - 3): the oldest data position

■When the monitored data exceeds the monitor buffer size, the new data is written over the oldest data. (Ring buffer type)

The oldest data position of the monitor data area can be confirmed from this area.

Ex.

"Un\G9745" becomes the oldest data when "000FH" is the monitor data pointer using "Un\G9730" as the start address of the monitor data area.



1 [Device/Buffer Memory Batch Monitor] Monitoring

○ Device Name

● Buffermemory Unit 0 (HEX) Address 9728 DEC

Device Name		F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	Current Value	String
U0\G9728		0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	000F	..
U0\G9729		0	0	0	0	1	1	0	0	1	1	1	1	1	1	1	0	0CFE	þ.
U0\G9730	+0	0	0	0	1	0	0	1	1	0	1	0	0	0	1	0	0	1344	D.
U0\G9731	+1	0	0	0	1	0	0	1	1	0	1	0	0	0	1	0	1	1345	E.
U0\G9732	+2	0	0	0	1	0	0	1	1	0	1	0	0	0	1	1	0	1346	F.
U0\G9733	+3	0	0	0	1	0	0	1	1	0	1	0	0	0	1	1	1	1347	G.
U0\G9734	+4	0	0	0	1	0	0	1	1	0	1	0	0	1	0	0	0	1348	H.
U0\G9735	+5	0	0	0	1	0	0	1	1	0	1	0	0	1	0	0	1	1349	I.
U0\G9736	+6	0	0	0	1	0	0	1	1	0	1	0	0	1	0	1	0	134A	J.
U0\G9737	+7	0	0	0	1	0	0	1	1	0	1	0	0	1	0	1	1	134B	K.
U0\G9738	+8	0	0	0	1	0	0	1	1	0	1	0	0	1	1	0	0	134C	L.
U0\G9739	+9	0	0	0	1	0	0	1	1	0	1	0	0	1	1	0	1	134D	M.
U0\G9740	+A	0	0	0	1	0	0	1	1	0	1	0	0	1	1	1	0	134E	N.
U0\G9741	+B	0	0	0	1	0	0	1	1	0	1	0	0	1	1	1	1	134F	O.
U0\G9742	+C	0	0	0	1	0	0	1	1	0	1	0	1	0	0	0	0	1350	P.
U0\G9743	+D	0	0	0	1	0	0	1	1	0	0	0	0	1	1	0	1	130D	..
U0\G9744	+E	0	0	0	1	0	0	1	1	0	0	0	0	1	0	1	0	130A	..
U0\G9745	+F	0	0	0	1	0	0	1	1	0	1	0	0	0	1	0	0	1344	D.
U0\G9746		0	0	0	1	0	0	1	1	0	1	0	0	0	1	1	0	1346	F.

Monitor data pointer storage area

Monitor data size storage area

Oldest data in the monitor data area

Monitor data quantity storage area (Un\G9729/13057(default))

The number of monitor data stored in the monitor data area is stored.

0 to (monitor buffer size designation (default: 0D00H) - 2): number of monitor data

Monitor data area (Un\G9730 to 13055/13058 to 16383(default))

The monitor data is stored in one-word units in the configuration below.

For an example of the send/receive data monitoring, refer to the following section.

☞ Page 426 Send/receive Data Monitoring Example

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	
0	0	0	0	✻	CD (DCD)	CS (CTS)	DR (DSR)	Received data								When receiving data
0	0	0	1	✻	CD (DCD)	CS (CTS)	DR (DSR)	Transmitted data								When transmitting data
0	0	1	0	✻	CD (DCD)	CS (CTS)	DR (DSR)	0						RS (RTS)	ER (DTR)	When the RS(RTS)/ER(DTR) signal is changed
0	0	1	1	✻	CD (DCD)	CS (CTS)	DR (DSR)	0					FE	OVR	PE	When a reception error occurs
													(*1)			
0	1	0	0	✻	CD (DCD)	CS (CTS)	DR (DSR)	0								When detecting a CD(DCD) to DR(DSR) signal change

Data classification (b12 to b15)
0: When receiving data
1: When transmitting data
2: When the RS(RTS)/ER(DTR)signal is changed
3: When a reception error occurs
4: When detecting a CD(DCD) to DR(DSR)signal change

Signal condition (b8 to b11)
* System area

[1] FE : Framing error
OVR : Overrun error
PE : Parity error

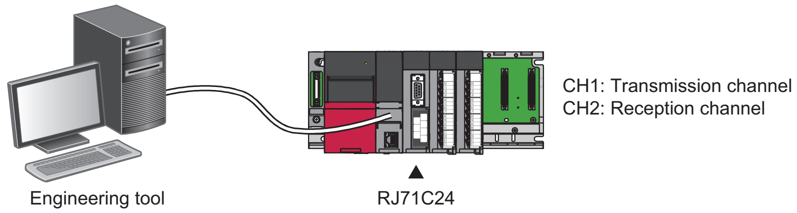
22.3 Send/receive Data Monitoring Example

The following shows an example of the send/receive data monitoring for communication using the nonprocedural protocol.

System configuration

The system configuration for the program example is as shown below.

The CH used for operation of this sample program is CH1 only.

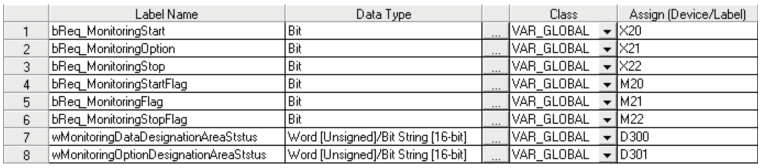


Program example for send/receive data monitoring

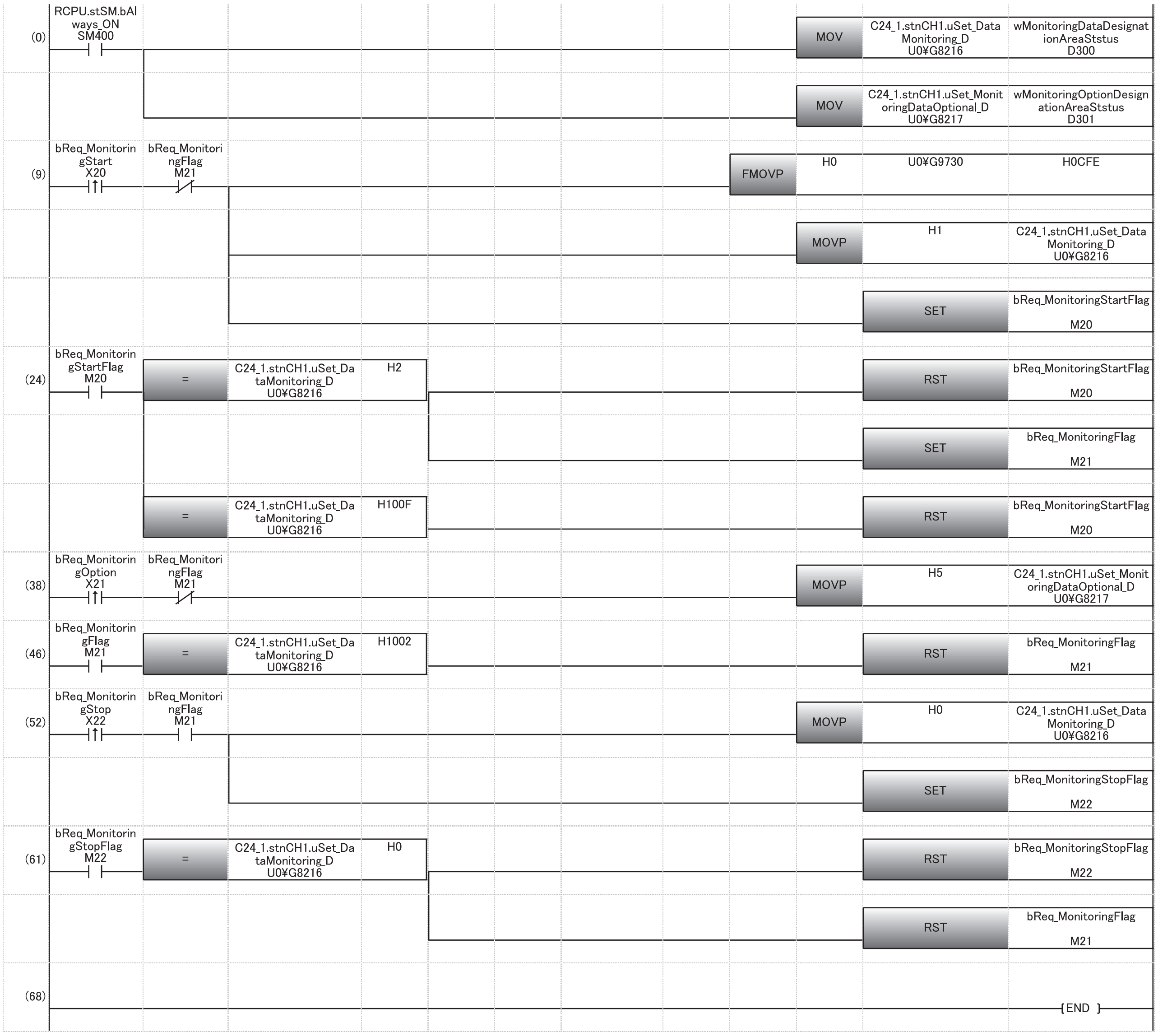
■Parameter settings

Setting item				Setting content	
				CH1	CH2
Basic settings	Various control specification	Test mode setting		—	CH2: Not used
		Communication protocol setting		Nonprocedural protocol	
		Communication speed setting		19200 bps	
		Transmission setting	Operation setting	Independent	
			Data bit	8	
			Parity bit	None	
			Odd/even parity	Odd	
			Stop bit	1	
			Sum check code	None	
			Online change	Enable	
			Setting change	Enable	
		Station number setting (CH1, 2 in common: 0 to 31)		0	

■Program example

Category	Label name	Description	Device
Module label	C24_1	Target module	—
	C24_1.stnCH1.uSet_DataMonitoring_D	CH1_Send/receive data monitoring designation	U0\G8216
	C24_1.stnCH1.uSet_MonitoringDataOptional_D	CH1_Monitor option specification	U0\G8217
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_MonitoringStart	Bit	...	VAR_GLOBAL	X20
2	bReq_MonitoringOption	Bit	...	VAR_GLOBAL	X21
3	bReq_MonitoringStop	Bit	...	VAR_GLOBAL	X22
4	bReq_MonitoringStartFlag	Bit	...	VAR_GLOBAL	M20
5	bReq_MonitoringFlag	Bit	...	VAR_GLOBAL	M21
6	bReq_MonitoringStopFlag	Bit	...	VAR_GLOBAL	M22
7	wMonitoringDataDesignationAreaStstus	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D300
8	wMonitoringOptionDesignationAreaStstus	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D301


(0)
RCPU.stSM.bAlways_ON SM400
MOV C24_1.stnCH1.uSet_DataMonitoring_D U0¥G8216 wMonitoringDataDesignationAreaStstus D300
MOV C24_1.stnCH1.uSet_MonitoringDataOptional_D U0¥G8217 wMonitoringOptionDesignationAreaStstus D301
(9)
bReq_MonitoringStart X20
bReq_MonitoringFlag M21
FMOVP H0 U0¥G9730 H0CFE
MOVP H1 C24_1.stnCH1.uSet_DataMonitoring_D U0¥G8216
SET bReq_MonitoringStartFlag M20
(24)
bReq_MonitoringStartFlag M20
= C24_1.stnCH1.uSet_DataMonitoring_D U0¥G8216 H2
RST bReq_MonitoringStartFlag M20
SET bReq_MonitoringFlag M21
= C24_1.stnCH1.uSet_DataMonitoring_D U0¥G8216 H100F
RST bReq_MonitoringStartFlag M20
(38)
bReq_MonitoringOption X21
bReq_MonitoringFlag M21
MOVP H5 C24_1.stnCH1.uSet_MonitoringDataOptional_D U0¥G8217
(46)
bReq_MonitoringFlag M21
= C24_1.stnCH1.uSet_DataMonitoring_D U0¥G8216 H1002
RST bReq_MonitoringFlag M21
(52)
bReq_MonitoringStop X22
bReq_MonitoringFlag M21
MOVP H0 C24_1.stnCH1.uSet_DataMonitoring_D U0¥G8216
SET bReq_MonitoringStopFlag M22
(61)
bReq_MonitoringStopFlag M22
= C24_1.stnCH1.uSet_DataMonitoring_D U0¥G8216 H0
RST bReq_MonitoringStopFlag M22
RST bReq_MonitoringFlag M21
(68)
END

(9) Clear the monitor data area to '0'.
During monitor start instruction

(24) During monitoring (002H)
Monitor setting error (100FH)

(46) During monitor stop (1002H)

(52) During monitor stop instruction

(61) Monitor stopped (0000H)

Confirming monitor data with Engineering tool

Monitor the monitor data area on the CH1 side (Un\G9730 to 13055) with the Device/Buffer Memory Batch Monitor function of Engineering tool.

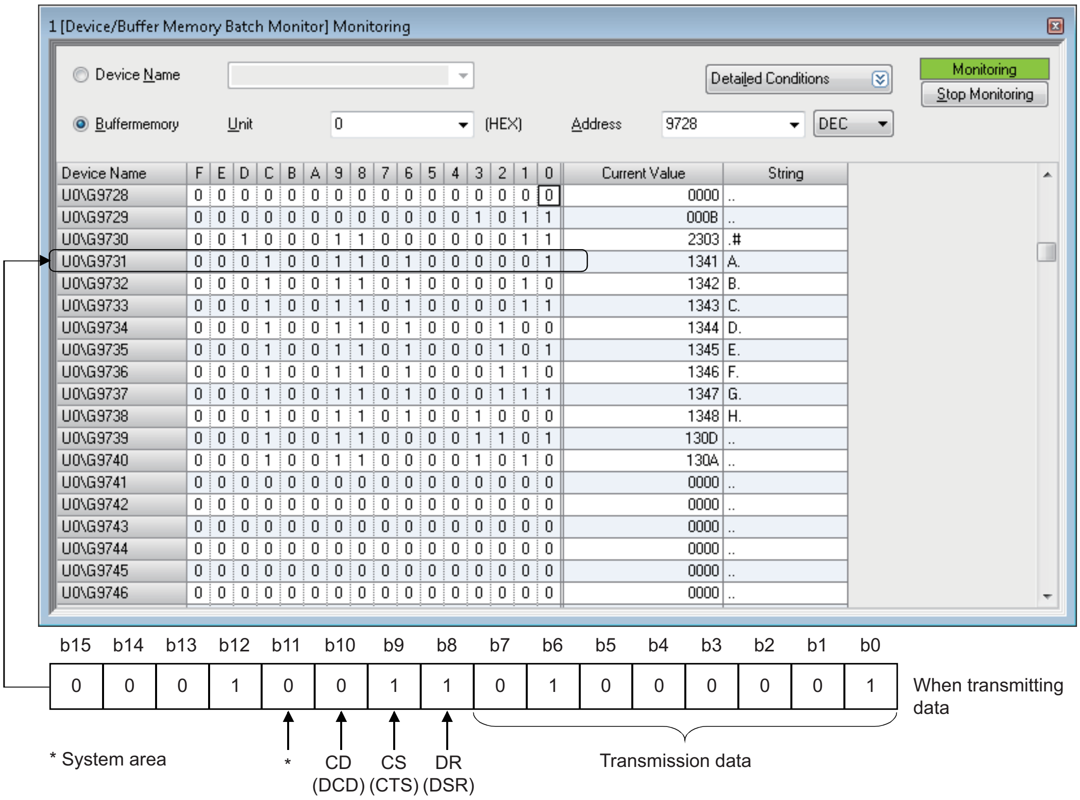
For the data configuration of the monitor data area, refer to the following section.

☞ Page 425 Monitor data area (Un\G9730 to 13055/13058 to 16383(default))

Ex.

Monitoring example when the following data is transmitted from CH1 for data communication using the nonprocedural protocol.

Send data; "ABCDEFGH" + 0D0AH (5-word)



1 [Device/Buffer Memory Batch Monitor] Monitoring

○ Device Name

● Buffermemory Unit 0 (HEX) Address 9728 DEC

Device Name	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	Current Value	String
U0\G9728	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	..
U0\G9729	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	000B	..
U0\G9730	0	0	1	0	0	0	1	1	0	0	0	0	0	0	1	1	2303	.#
U0\G9731	0	0	0	1	0	0	1	1	0	1	0	0	0	0	0	1	1341	A.
U0\G9732	0	0	0	1	0	0	1	1	0	1	0	0	0	0	1	0	1342	B.
U0\G9733	0	0	0	1	0	0	1	1	0	1	0	0	0	0	1	1	1343	C.
U0\G9734	0	0	0	1	0	0	1	1	0	1	0	0	0	1	0	0	1344	D.
U0\G9735	0	0	0	1	0	0	1	1	0	1	0	0	0	1	0	1	1345	E.
U0\G9736	0	0	0	1	0	0	1	1	0	1	0	0	0	1	1	0	1346	F.
U0\G9737	0	0	0	1	0	0	1	1	0	1	0	0	0	1	1	1	1347	G.
U0\G9738	0	0	0	1	0	0	1	1	0	1	0	0	1	0	0	0	1348	H.
U0\G9739	0	0	0	1	0	0	1	1	0	0	0	0	1	1	0	1	130D	..
U0\G9740	0	0	0	1	0	0	1	1	0	0	0	0	1	0	1	0	130A	..
U0\G9741	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	..
U0\G9742	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	..
U0\G9743	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	..
U0\G9744	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	..
U0\G9745	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	..
U0\G9746	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	..

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	1	0	0	1	1	0	1	0	0	0	0	0	1

* System area

23 INDEPENDENT/LINKED OPERATION OF INTERFACE

This chapter explains the independent/linked operation of C24.

23.1 Setting and Data Flow of Independent Operation

Setting

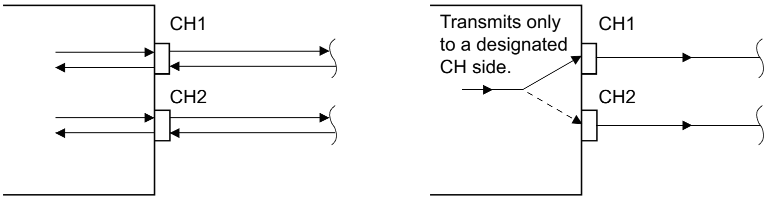
For the independent operation of two interfaces on C24, configure the parameter setting as shown below.

Setting item				Setting content	
				CH1	CH2
Basic settings	Various control specification	Test mode setting		—	
		Communication protocol setting		Set according to the target device.	Set according to the target device.
		Communication speed setting			
		Transmission setting	Operation setting	Independent	Independent
			Data bit	Set according to the target device.	Set according to the target device.
			Parity bit		
			Odd/even parity		
			Stop bit		
			Sum check code		
			Online change		
			Setting change		
		Station number setting (CH1, 2 in common: 0 to 31)		Set arbitrary value for data communication using the MC protocol or for multidrop connection when MELSOFT connection is selected.	

Data flow in independent operation

Two interfaces operate independently according to the settings in the parameter settings "Basic Settings".

C24 (data flow in independent operation)



23.2 Setting and Data Flow of Interlink Operation

During the interlink operation, all data received from either one of the two interfaces is transmitted to the other interface.

Point

When two interfaces are in the interlink operation, the transmission time for one character becomes equal to the H/W gate OFF time.

Setting

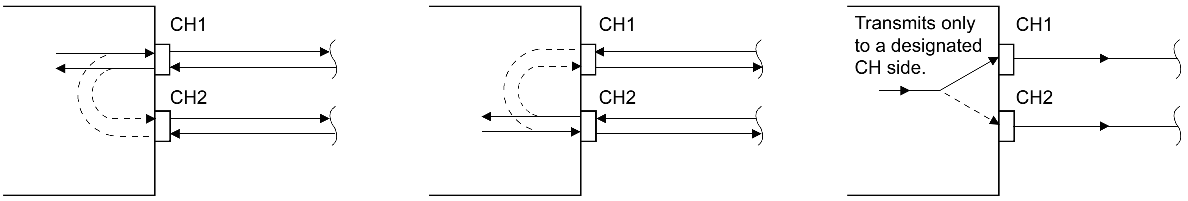
For the interlink operation of two interfaces on C24, configure the parameter setting as shown below.

Setting item				Setting content	
				CH1	CH2
Basic settings	Various control specification	Test mode setting		—	
		Communication protocol setting		Set according to the target device.	The setting is configured automatically. (No changing necessary)
		Communication speed setting			Set according to the target device.
		Transmission setting	Operation setting	Independent	Interlink
			Data bit	Set CH1 and CH2 to the same specifications.	Set CH1 and CH2 to the same specifications.
			Parity bit		
			Odd/even parity		
			Stop bit		
			Sum check code		
			Online change		
			Setting change		
		Station number setting (CH1, 2 in common: 0 to 31)		Set arbitrary value for data communication using the MC protocol or for multidrop connection when MELSOFT connection is selected.	
		MODBUS station number setting		1 to 247	The setting is configured automatically. (Same value as CH1)

Data flow in interlink operation

The two interfaces operate interconnectedly using the functions configured in "Communication protocol setting" from "Basic Settings" ⇨ "CH2 side" in the module parameter (MC protocol (same format), MODBUS slave (RTU), MODBUS slave (ASCII), or nonprocedural protocol) within the transmission specifications set in the respective transmission settings.

C24 (data flow in linked operation)



Precautions

■When setting to nonprocedural protocol

Exclusive control of the received data is necessary when data is communicated using the nonprocedural protocol since all of the connected stations receive the data.

■When setting to MC protocol

When data communication is performed using the MC protocol, only the C24, of which station number is specified in the massage, performs the process designated by the command.

Furthermore, when data communication is performed with 2C/3C/4C frame of the MC protocol, the header information for interlink operation is added to messages directed to other stations linked by multidrop connection.

● Processing performed by C24 connected to a target device

- C24 adds the header information to the command massages directed to other stations, which was received from the target device, and transmits them to the other stations through the other interface.
- C24 deletes the header information from the response messages received from the other stations, and transmits them to the target device through the other interface. (The header information is also sent during the multidrop connection using the interlink operation setting.)

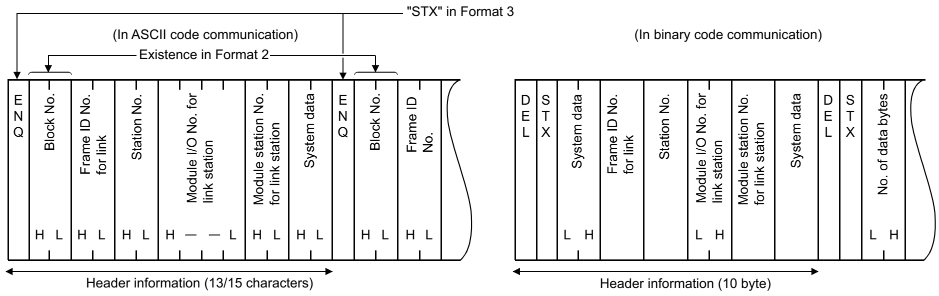
● Operation of the accessed station

- The accessed station processes the request contained in a command message, adds the header information to a response message, and sends it using the interface that received the command message.

Point

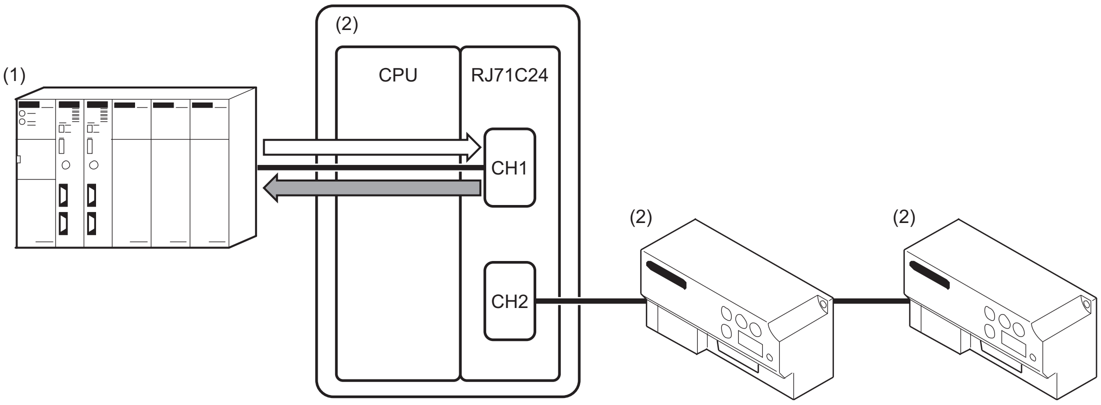
The header information for the interlink operation, which is added by C24 to a message is as shown below.

- When communicating with ASCII code (Formats 1 to 4)
  The following 13- or 15-character header information is added immediately before the control code (ENQ/STX/ACK/NAK) at the head of each message (13 characters for Formats 1, 3 and 4; 15 characters for Format 2).
- When communicating in binary mode (Format 5)
  The following 10-byte header information is added immediately before the control code (DLE + STX) at the head of each message.


"STX" in Format 3
(In ASCII code communication)
Existence in Format 2
ENQ
Block No.
Frame ID No. for link
Station No.
Module I/O No. for link station
Module station No. for link station
System data
ENQ
Block No.
Frame ID No.
H L
H — — L
Header information (13/15 characters)
(In binary code communication)
DEL
STX
System data
Frame ID No. for link
Station No.
Module I/O No. for link station
Module station No. for link station
System data
DEL
STX
No. of data bytes
L H
Header information (10 byte)

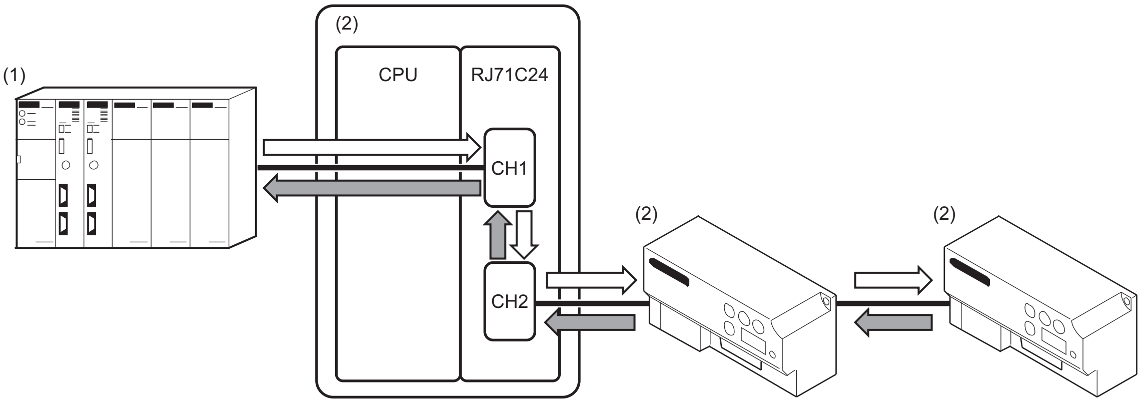
■When performing the interlink operation on a MODBUS slave

- For the interlink operation of a MODBUS slave, use a C24 the firmware version of which is '13' or later.
- A MODBUS master device cannot be connected to CH2 of a C24 when performing the interlink operation. Connect it to CH1.
- When a request message to a C24 is received from a MODBUS master, the C24 does not relay the request message to CH2 and sends a response message as a MODBUS slave.



⇨ Request message
➡ Response message
(1) MODBUS master
(2) MODBUS slave

- When a request message to a slave station other than a C24 is received from a MODBUS master, the C24 sends and receives the message via CH1 and CH2.



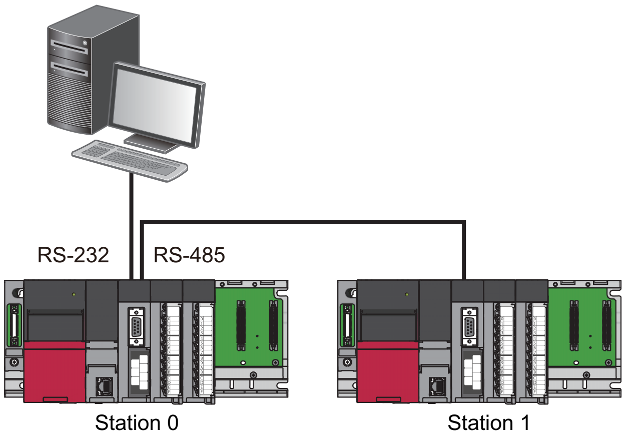
⇨ Request message
➡ Response message
(1) MODBUS master
(2) MODBUS slave

Restriction

Do not use the above settings for the following cases because the interlink operation cannot be performed.
- When using RJ71C24-R2.
- When a target device is not connected to either of the interfaces.
- When communicating data using the predefined or bidirectional protocol.
- When communicating data using the modem function.

Setting example for MELSOFT connection via C24

Examples of the parameter settings for MELSOFT connection via C24 are as shown below.



- Setting at station No.0

Setting item				Setting content	
				CH1	CH2
Basic settings	Transmission setting	Test mode setting		—	
		Communication protocol setting		MELSOFT Connection	—
		Communication speed setting		115200 bps	115200 bps
		Transmission setting	Operation setting	Independent	Interlink
			Data bit	8	8
			Parity bit	Yes	Yes
			Odd/even parity	Odd	Odd
			Stop bit	1	1
			Sum check code	Yes	Yes
			Online change	Enable	Enable
			Setting change	Enable	Enable
		Station number setting (CH1, 2 in common: 0 to 31)		0	

- Setting at station No.1

Setting item				Setting content	
				CH1	CH2
Basic settings	Transmission setting	Test mode setting		—	
		Communication protocol setting		Set depending on the use	MC protocol (format 5)
		Communication speed setting			115200 bps
		Transmission setting	Operation setting		Independent
			Data bit		8
			Parity bit		Yes
			Odd/even parity		Odd
			Stop bit		1
			Sum check code		Yes
			Online change		Enable
			Setting change		Enable
		Station number setting (CH1, 2 in common: 0 to 31)		1	

24 PARAMETER REGISTRATION TO CPU MODULE

C24 enables the parameter settings to be saved in the CPU module.

It reduces man-hours for any recovery works since the parameter resetting is not necessary at the time of replacement of C24.

The parameters controlled by the CPU module are as shown below.

Parameters name	Storage location of parameters	File name*1
Module parameter	CPU built-in memory (data memory)	UNIT.PRM
Module extended parameter	Select from the following for each file.*2 • CPU built-in memory (data memory) • SD memory card installed in the CPU • Built-in memory of Intelligent function module (in the flash ROM of C24)	UEXmmmnn.PRM*3,*4 UEXmmm00.PPR*3

*1 It indicates a file name to store the parameters in CPU built-in memory or SD memory card.

*2 To mount C24 on the redundant extension base unit in the redundant system, store module extended parameters to the built-in memory of intelligent function module (in the flash ROM of C24).

*3 The value indicating the head input/output number of the module ÷ 10H (3 digits in hexadecimal) is entered in 'mmm'.

*4 The sequential serial number of each module, 00 is entered in 'nn'.

For more details on parameters controlled by CPU modules, refer to the following manual.

📖MELSEC iQ-R CPU Module User's Manual (Application)

The parameters controlled by C24 parameter corresponding to CPU module are as shown below.

Corresponding parameter controlled by CPU module	Parameter of C24		Remarks
Module parameter	Basic settings	Various control specification	—
	Application settings	User frame specification	"User frame contents" is stored in the module extended parameter.
		Modem function setting	"Modem initialization data" and "Modem connection data" are stored in the module extended parameters.
		Programmable controller CPU monitoring function designation	—
	Interrupt settings		—
	MODBUS slave setting		—
	Refresh setting		—
Module extended parameter	User frame contents		File name: UEXmmmnn.PRM
	Modem initialization data		
	Modem connection data		
	Simple CPU communication setting		
	Protocol setting data		File name: UEXmmm00.PPR This is set with the Predefined protocol support function of Engineering tool.

Each file of module extended parameters held in the CPU module is given priority over the one in the flash ROM of C24. The contents in flash ROM of C24 is replaced with the contents of module extended parameters registered in the CPU module by powering OFF → ON or switching the CPU module STOP → RUN after the module extended parameters were registered in the CPU built-in memory or SD memory card.

Deleting module extended parameters

The following explains how to delete each file of module extended parameters.

■Deleting module extended parameter "UEXmmmnn.PRM"

It can be deleted by the following procedure.

1. Place the CPU module in the STOP state.
2. On the "Module extended parameter setting " screen for RJ71C24(-R2/4) of Engineering tool, restore the default setting value of module extended parameter.
3. Select [Online]⇨[Write to PLC].
4. On the "Online Data Operation" screen, select the checkbox of the corresponding Intelligent function module (C24), and perform the write operation. At this point, the content in the flash ROM of C24 is deleted.
5. On the "Online Data Operation" screen, select the [Delete] tab. When there are module extended parameters on the CPU module, "Module extended parameter: mmmm[*1]" is displayed on the [Parameter] in the tree. Select the checkbox and perform the delete operation. By performing this operation, the module extended parameter on the CPU module is deleted.
6. Return the CPU module to the RUN state as necessary.

[*1] The 'mmmm' indicates the head input/output number of the module.

■Deleting module extended parameter "UEXmmm00.PPR"

It can be deleted by the following procedure.

1. Place the CPU module in the STOP state.
2. Start the Predefined protocol support function from the [Tool] menu of Engineering tool.
3. Select [Write to Module] from the [Online] menu without adding protocol.
4. On the "Module Write" screen, select "Intelligent Function Module" as the target memory, and perform the write operation. At this point, the content in the flash ROM of C24 is deleted.
5. After the step 4 above, select [Online] → [User Data] → [Delete] of Engineering tool. The "User Data Operation" screen is displayed.
6. When there are module extended parameters on the CPU module, user data "UEXmmm00.PPR[*1]" is displayed. Select the checkbox and perform the delete operation. By performing this operation, the module extended parameter on the CPU module is deleted.
7. Return the CPU module to the RUN state as necessary.

[*1] The value indicating the head input/output number of the module ÷ 10H (3 digits in hexadecimal) is entered in 'mmm'.

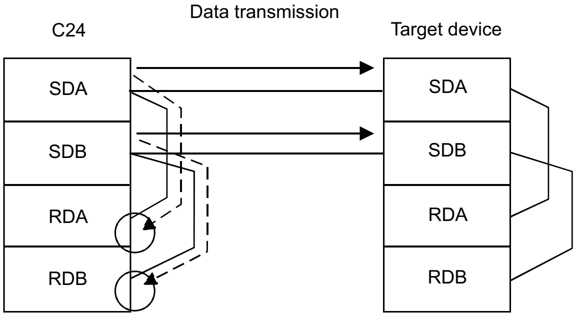
25 RS-422/485 INTERFACE ECHO BACK ENABLE/ PROHIBIT SPECIFICATION

This chapter explains the echo back enable/prohibit specification.

This function can be used for the both interfaces on RJ71C24(-R4).

The echo back function is a function that transmitted data is also sent to RDA or RDB on the host station when data communication is performed through RS-458 (2-wire type) from RS-422/485 interface.

Whether to receive the returned data or not (discard) can be specified.



Receives or not receive (discards) the data.

Application

- Disabling the echo back will eliminate the processing to ignore unnecessary data, which has been programmed for nonprocedural protocol communication.
- When the same message format is used for a request (command) and a response of the target device protocol during data communication using the predefined protocol, communication with the target device can be performed by specifying "Echo back enable."

Setting method

- Setting by engineering tool

  Set "Echo back enable/prohibit specification" to "Echo back enable" or "Echo back prohibit".

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

- Setting by a program

  In the 'RS-422/485 interface echo back enable/prohibit specification' (Un\G290/450), store either of the following values. (Default: 0)

Setting value	Description
0: Echo back enable	When C24 transmits data, it also receives the transmitted data with its own RDA or RDB.
1: Echo back prohibit	When C24 transmits data, it does not receive the transmitted data with its own RDA or RDB. (The data is discarded.)

Precautions

The setting for the echo back enable/prohibit specification must be configured before sending data. Any setting changed during transmission does not take effect.

MEMO

26 ERRONEOUS NOISE SIGNAL RECEPTION PREVENTION FUNCTION

This chapter explains the erroneous noise signal reception prevention function of C24.

26.1 Erroneous Noise Signal Reception Prevention Function

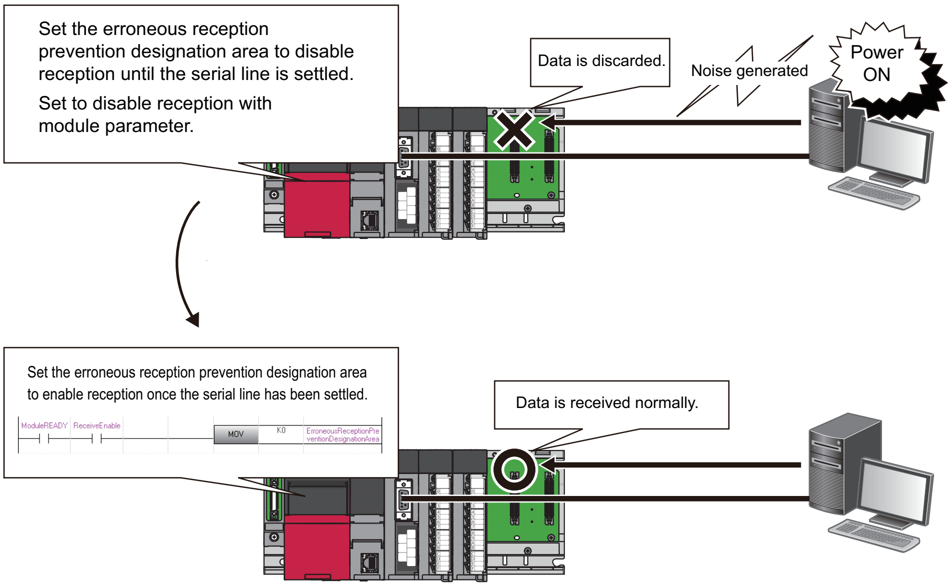
Depending on the device to be connected to C24, noise signals may be added on the line when turning ON the power or connecting cables of the device.

When noise signals are added, C24 falsely recognizes it has received improper data, and discards the data resulting in abnormal detection.

Using this function avoids erroneous noise signal reception.

This function is set in the module parameter and the 'Communication data reception enable/disable designation' (Un\G292/452).

Configure the setting to disable the reception with the parameter settings of Engineering tool, and set the buffer memory to 'Receive enable' (0) at a certain time before starting data reception after powering ON.



Set the erroneous reception prevention designation area to disable reception until the serial line is settled.

Set to disable reception with module parameter.

Set the erroneous reception prevention designation area to enable reception once the serial line has been settled.

ModuleREADY
ReceiveEnable
MOV
K0
ErroneousReceptionPreventionDesignationArea
Data is received normally.

26.2 Setting for the Erroneous Noise Signal Reception Prevention Function

The setting procedure to use the erroneous noise signal reception prevention function is shown below.

1. Setting to disable the reception with Engineering tool

Set "Communication data reception enable/disable designation" to "Receive disable".

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

2. Setting to enable the reception by a program

Set the following buffer memory to 'Receive enable' (0) with a program at the timing when the serial line is settled.

Channel	Address	Setting value
CH1	Un\G292	0: Receive enable 1: Receive disable
CH2	Un\G452	

Precautions

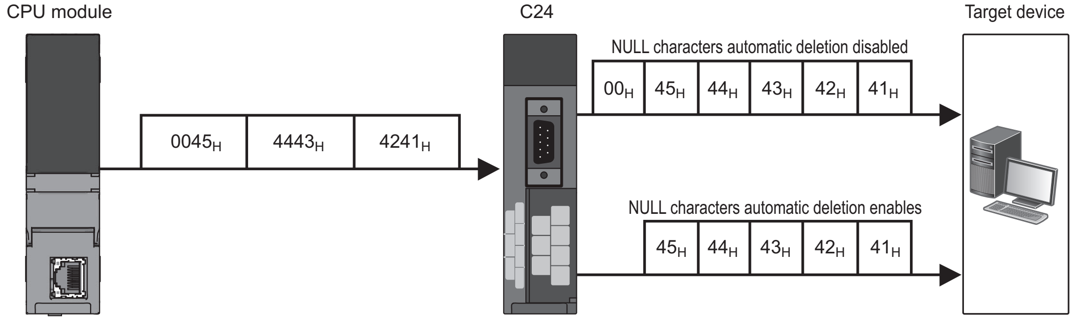
- Once the erroneous reception prevention designation area of noise signal is set to 'Receive disable' (1), normal data is not received as well.
  Set the erroneous reception prevention designation area of noise signal to 'Receive enable' (0) promptly after the serial line has been settled.
- Once it is set to 'Receive enable', it cannot be set to 'Receive disable' unless powering ON again or resetting the CPU module.

27 AUTOMATIC REMOVAL OF NULL CHARACTERS INCLUDED IN SEND DATA

This chapter explains the automatic removal of NULL characters in data transmission of C24.

27.1 Automatic Removal of NULL Characters during Data Transmission

The automatic removal of NULL characters during data transmission is a function to automatically delete NULL characters included in data transmitted to a target device from C24.


CPU module
C24
Target device
NULL characters automatic deletion disabled
00_H 45_H 44_H 43_H 42_H 41_H
0045_H 4443_H 4241_H
NULL characters automatic deletion enables
45_H 44_H 43_H 42_H 41_H

27.2 Setting for NULL Character Automatic Removal Function

The setting to C24 is required before data transmission in order to enable the NULL character automatic removal function for data to be transmitted using the nonprocedural protocol and the bidirectional protocol.

The settings to delete NULL characters automatically are shown below.

- The settings for automatic removal of NULL characters can be configured for each interface.
- To perform the NULL characters automatic removal function, set "NULL character automatic removal designation" to "Auto delete enabled".

Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ "Module Name" ⇨ [Module Parameter] ⇨ "Basic Settings" ⇨ "Various control specification"

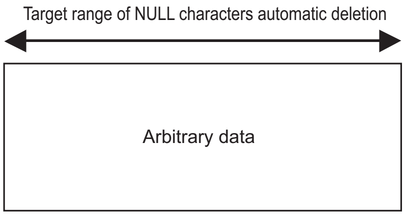
27.3 NULL Character Removal to Send Data using Nonprocedural Protocol

This section explains automatic removal of NULL characters to data to be transmitted using the nonprocedural protocol.

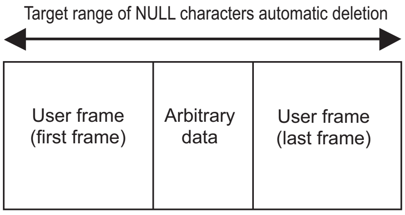
Target range of automatic removal of NULL characters

The send data range to which automatic removal of NULL characters is performed is as shown below:

- Transmitting data in arbitrary format



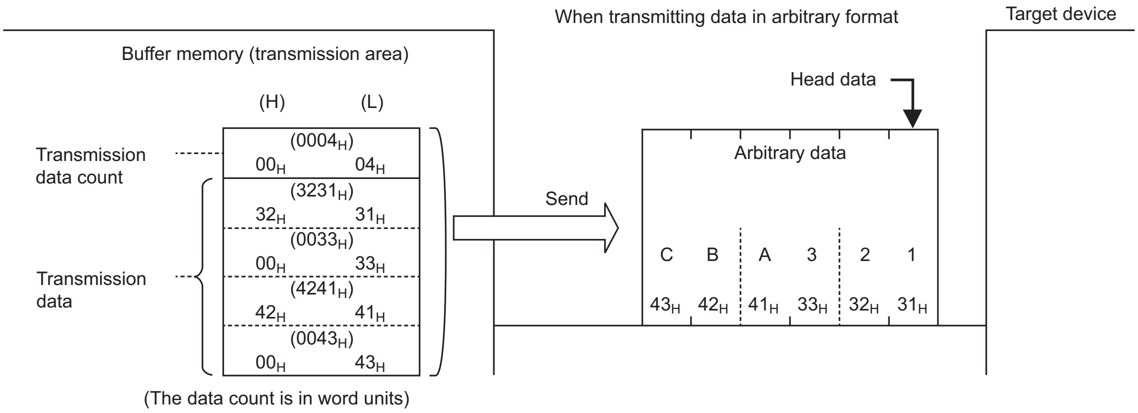
- Transmitting data with user frame



When data is transmitted, C24 performs the processing as follows:

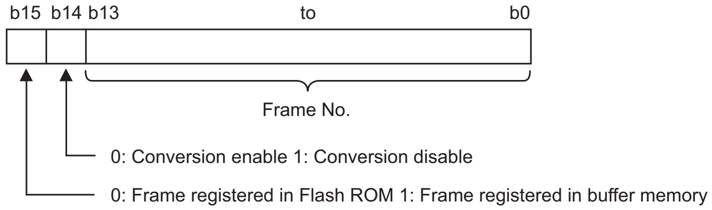
C24 deletes NULL characters (data code: 00H) included in the data specified by the CPU module (arbitrary data portion in a transmission message) or a user frame portion, and transmits data.

NULL characters (00H) included in the send data in the buffer memory (send area) are deleted from the actual data to be transmitted.



To transmit data without removing NULL characters included in arbitrary user frame portion or the send area on the buffer memory when 0001H (Auto delete enabled) is set for NULL character automatic removal designation for transmitting data for data transmission with a user frame, designate the user frame number as shown below.

- Specify the number obtained by adding 4000H to the registered user frame No.


b15 b14 b13
to
b0
Frame No.
0: Conversion enable 1: Conversion disable
0: Frame registered in Flash ROM 1: Frame registered in buffer memory

When sending data in the portions designated in 4001H to 44AFH, C000H to C01FH, the data is transmitted without conversion even when the following designations are enabled.

- Specifying an additional code by the transparent code for sending specification (☞ Page 358 TRANSPARENT CODES AND ADDITIONAL CODES)
- Designating "Enable" in the ASCII-BIN conversion designation area (☞ Page 379 COMMUNICATING WITH ASCII CODE (ASCII-BIN CONVERSION))

User frame number to be transmitted	User frame designation number to transmit data without removing NULL characters (4000H added)
1H to 3E7H (1 to 999)	4001H to 43E7H (16385 to 17383)
3E8H to 4AFH (1000 to 1199)	43E8H to 44AFH (17384 to 17583)
8000H to 801FH (-32768 to -32737)	C000H to C01FH (-16384 to -16353)

27.4 Program Example

This examples of data communication using the nonprocedural protocol are shown below.

Program example of data communication using the nonprocedural protocol

This section shows program examples of data communication using the nonprocedural protocol when the settings/registrations are made with Engineering tool.

Parameter setting

For details for each setting item, refer to the parameters. (☞ Page 454 PARAMETER SETTING)

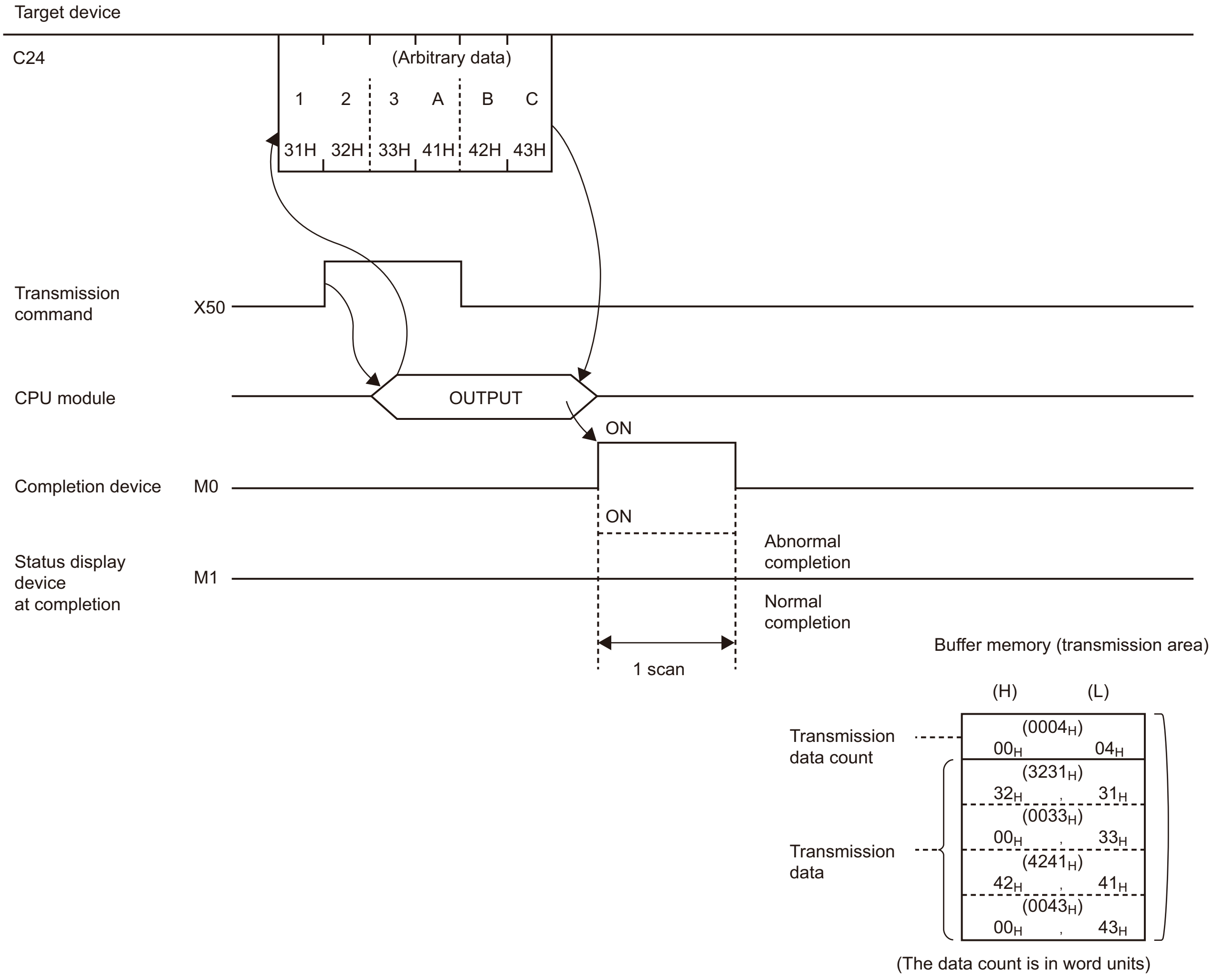
Setting item				Setting content		Remarks
				CH1	CH2	
Basic settings	Various control specification	Test mode setting		—	CH2 is not used	—
		Communication protocol setting		Nonprocedural protocol		
		Communication speed setting		Set according to the target device		
		Transmission setting	Operation setting	Independent		
			Data bit	Set according to the target device		
			Parity bit			
			Odd/even parity			
			Stop bit			
			Sum check code			
			Online change			
			Setting change			
		Station number setting (CH1, 2 in common: 0 to 31)		—		
		Communication control specification	NULL character automatic removal designation	Auto delete enabled		
Application settings	User frame specification	Send user frame designation	Send frame No. designation (1st)	43F2H		Arbitrary registered data
			Send frame No. designation (2nd)	43F3H		
			Send frame No. designation (3rd)	C001H		
			Send frame No. designation (4th)	8000H		Buffer memory designation
			Send frame No. designation (5th)	441BH		Arbitrary registered data

Example of arbitrary data transmission

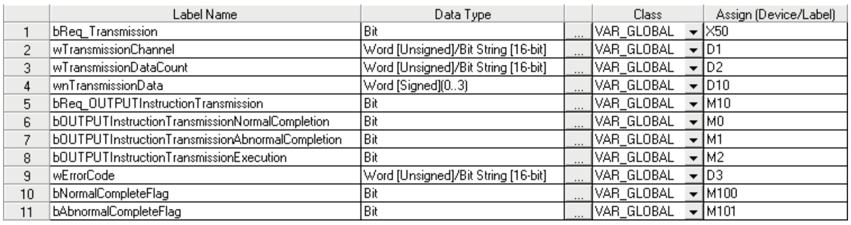
C24 transmits a string data using the NULL character automatic removal function to the target device.

Ex.

Transmitting given data with OUTPUT dedicated instruction



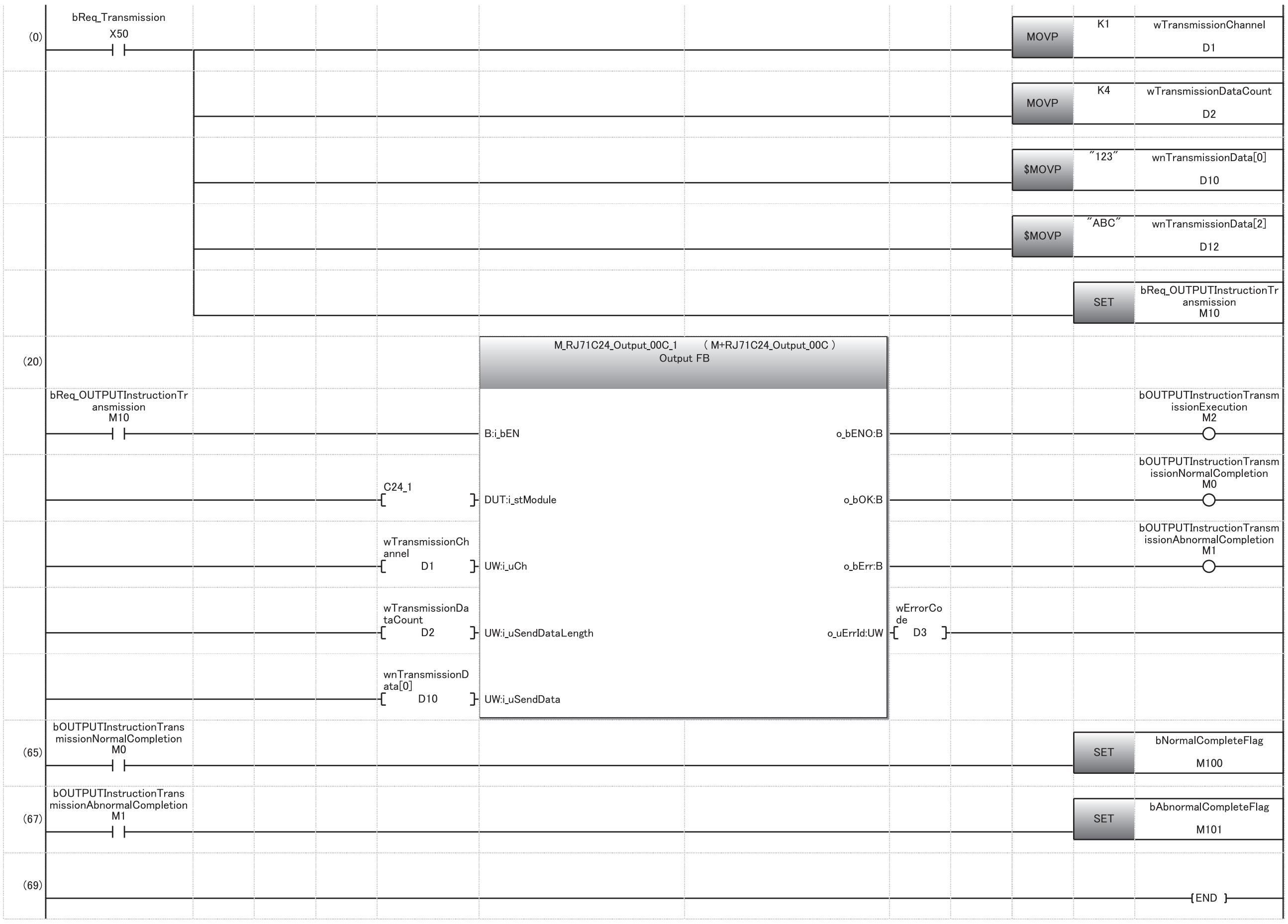
■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_Output	Sends data for specified data points.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_Transmission	Bit	...	VAR_GLOBAL	X50
2	wTransmissionChannel	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
3	wTransmissionDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2
4	wnTransmissionData	Word [Signed](0..3)	...	VAR_GLOBAL	D10
5	bReq_OUTPUTInstructionTransmission	Bit	...	VAR_GLOBAL	M10
6	bOUTPUTInstructionTransmissionNormalCompletion	Bit	...	VAR_GLOBAL	M0
7	bOUTPUTInstructionTransmissionAbnormalCompletion	Bit	...	VAR_GLOBAL	M1
8	bOUTPUTInstructionTransmissionExecution	Bit	...	VAR_GLOBAL	M2
9	wErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D3
10	bNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
11	bAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101

For information on the module FB to be used, refer to the following manual.

MELSEC iQ-R Serial Communication Module Function Block Reference



(0) Set the transmission channel to CH1.
Set the send data quantity to 4.
Set the data to be transmitted.
Set the transmission direction.

(65) Normal completion

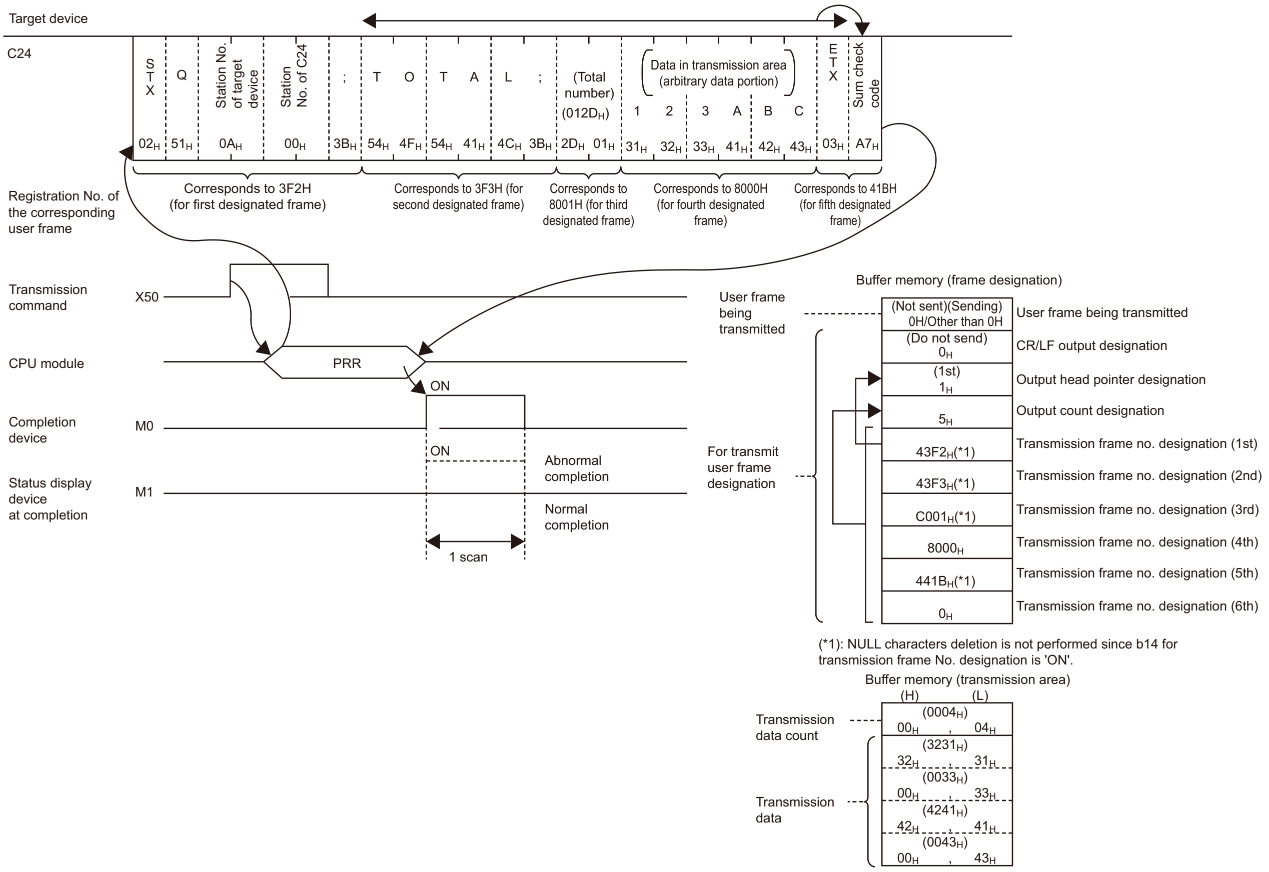
(67) Abnormal completion

Example of transmission using user frame

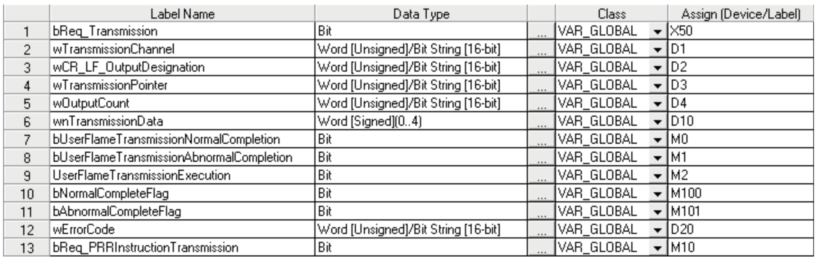
C24 transmits a string data using the NULL character automatic removal function to the target device.

Ex.

Transmitting given data in combination with a user frame with PRR dedicated instruction

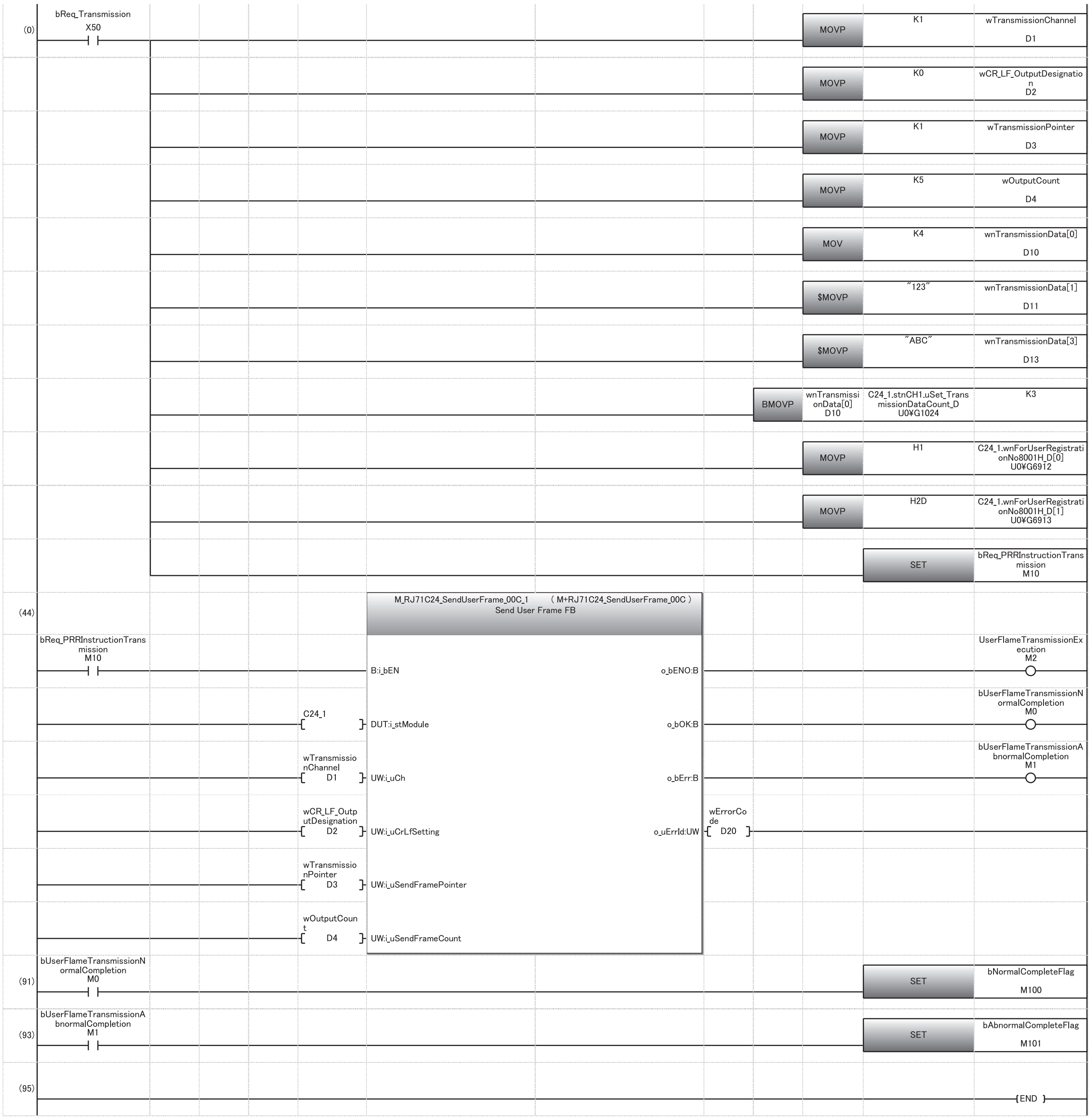

Target device
C24
STX
Q
Station No. of target device
Station No. of C24
;
T O T A L ;
(Total number)
(012DH)
Data in transmission area (arbitrary data portion)
1 2 3 A B C
ETX
Sum check code
02H 51H 0AH 00H 3BH 54H 4FH 54H 41H 4CH 3BH 2DH 01H 31H 32H 33H 41H 42H 43H 03H A7H
Corresponds to 3F2H (for first designated frame)
Corresponds to 3F3H (for second designated frame)
Corresponds to 8001H (for third designated frame)
Corresponds to 8000H (for fourth designated frame)
Corresponds to 41BH (for fifth designated frame)
Registration No. of the corresponding user frame
Transmission command
X50
CPU module
PRR
ON
Completion device
M0
ON
Abnormal completion
Status display device at completion
M1
Normal completion
1 scan
User frame being transmitted
For transmit user frame designation
Buffer memory (frame designation)
(Not sent)(Sending) 0H/Other than 0H
User frame being transmitted
(Do not send) 0H
CR/LF output designation
(1st) 1H
Output head pointer designation
5H
Output count designation
43F2H(*1)
Transmission frame no. designation (1st)
43F3H(*1)
Transmission frame no. designation (2nd)
C001H(*1)
Transmission frame no. designation (3rd)
8000H
Transmission frame no. designation (4th)
441BH(*1)
Transmission frame no. designation (5th)
0H
Transmission frame no. designation (6th)
(*1): NULL characters deletion is not performed since b14 for transmission frame No. designation is 'ON'.
Buffer memory (transmission area)
(H) (L)
Transmission data count
(0004H) 00H , 04H
Transmission data
(3231H) 32H , 31H
(0033H) 00H , 33H
(4241H) 42H , 41H
(0043H) 00H , 43H

■Program example

<table>
<tr><th>Category</th><th>Label name/FB name</th><th>Description</th><th>Device</th></tr>
<tr><td>Module label</td><td>C24_1</td><td>Target module</td><td>—</td></tr>
<tr><td>Label to be defined</td><td colspan="3">Define the global label as shown below.

<table>
<tr><th></th><th>Label Name</th><th>Data Type</th><th></th><th>Class</th><th>Assign (Device/Label)</th></tr>
<tr><td>1</td><td>bReq_Transmission</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>X50</td></tr>
<tr><td>2</td><td>wTransmissionChannel</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D1</td></tr>
<tr><td>3</td><td>wCR_LF_OutputDesignation</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D2</td></tr>
<tr><td>4</td><td>wTransmissionPointer</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D3</td></tr>
<tr><td>5</td><td>wOutputCount</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D4</td></tr>
<tr><td>6</td><td>wnTransmissionData</td><td>Word [Signed](0..4)</td><td>...</td><td>VAR_GLOBAL</td><td>D10</td></tr>
<tr><td>7</td><td>bUserFlameTransmissionNormalCompletion</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M0</td></tr>
<tr><td>8</td><td>bUserFlameTransmissionAbnormalCompletion</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M1</td></tr>
<tr><td>9</td><td>UserFlameTransmissionExecution</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M2</td></tr>
<tr><td>10</td><td>bNormalCompleteFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M100</td></tr>
<tr><td>11</td><td>bAbnormalCompleteFlag</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M101</td></tr>
<tr><td>12</td><td>wErrorCode</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D20</td></tr>
<tr><td>13</td><td>bReq_PRRInstructionTransmission</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M10</td></tr>
</table></td></tr>
<tr><td>FB</td><td>M+RJ71C24_SendUserFrame</td><td>Sends data using the nonprocedural protocol communication and the user frame according to the setting of the user frame specification area for sending data.</td><td>—</td></tr>
</table>

For information on the module FB to be used, refer to the following manual.

MELSEC iQ-R Serial Communication Module Function Block Reference



(0) Set the transmission channel to CH1.
Set CR/LF to "Do not send".
Set the send pointer to 1.
Set the output quantity to 5.
Set the send data quantity to 4.
Set the data to be transmitted from D11.
Set an arbitrary send data in send buffer.
Set the transmission direction.

(91) Normal completion

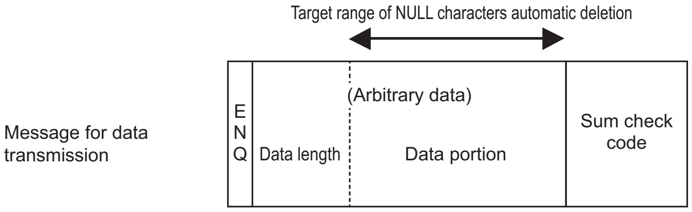
(93) Abnormal completion

27.5 NULL Characters Removal for Send Data using Bidirectional Protocol

This section explains automatic removal of NULL characters to data to be transmitted using the bidirectional protocol.

Target range of automatic removal of NULL characters

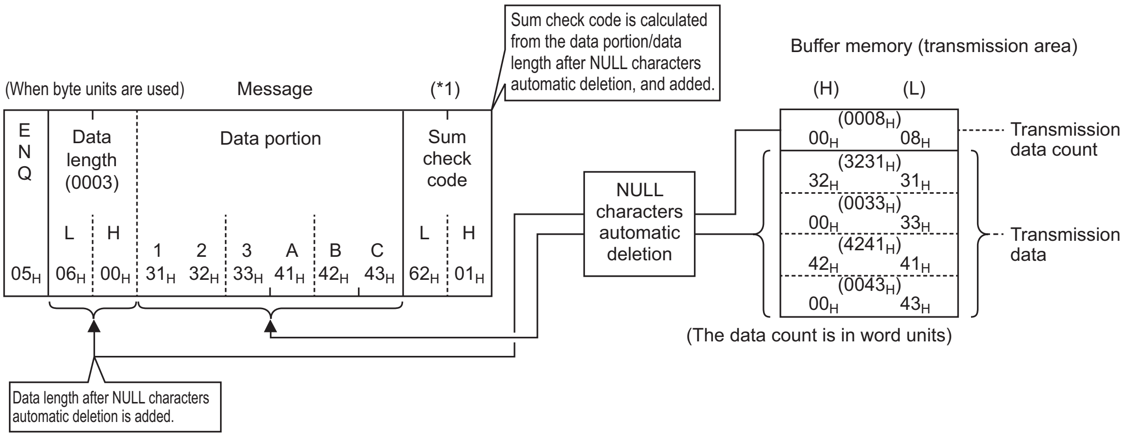
The send data range to which automatic removal of NULL characters is performed is as shown below. However, the automatic removal of NULL characters is not performed when a response message is transmitted.


Target range of NULL characters automatic deletion
Message for data transmission
ENQ
(Arbitrary data)
Data length
Data portion
Sum check code

*NULL characters automatic deletion is disabled when transmitting a response message.

When data is transmitted, C24 performs the processing as follows:

- Range of automatic removal of NULL characters
  C24 performs the automatic removal of NULL characters to an arbitrary data portion of a message (data length is not included).
- Automatic removal of NULL characters in data portion
  C24 deletes NULL characters included in send data, and then transmits data.
- Handling of sum check code
  The results of adding the data portion after automatic removal of NULL characters as binary code data are added to the transmission message as sum check code.


Sum check code is calculated from the data portion/data length after NULL characters automatic deletion, and added.
Buffer memory (transmission area)
(When byte units are used)
Message
(*1)
(H)
(L)
ENQ
Data length (0003)
Data portion
Sum check code
L H
1 2 3 A B C
L H
05H 06H 00H 31H 32H 33H 41H 42H 43H 62H 01H
NULL characters automatic deletion
(0008H) 00H 08H
(3231H) 32H 31H
(0033H) 00H 33H
(4241H) 42H 41H
(0043H) 00H 43H
Transmission data count
Transmission data
(The data count is in word units)
Data length after NULL characters automatic deletion is added.

*1 Calculation of sum check code is shown below:

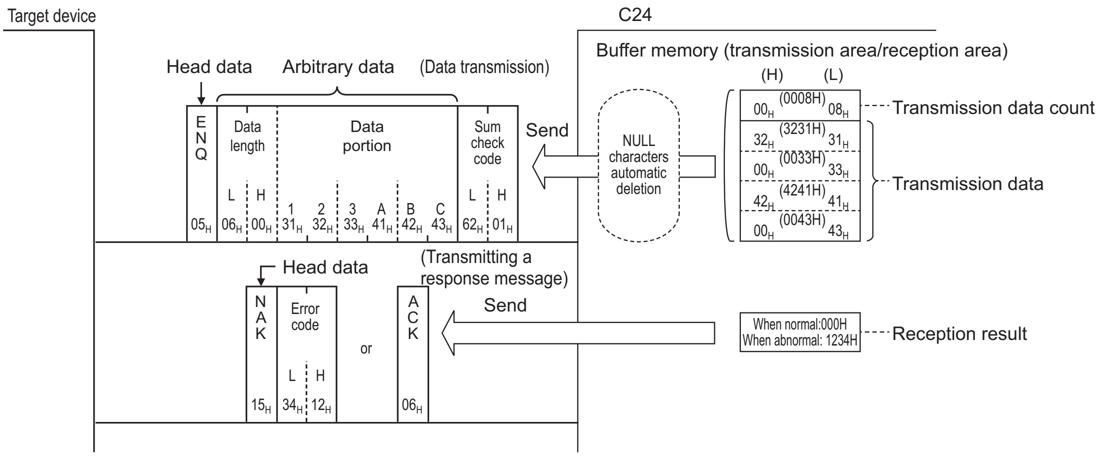
$$06H+00H+31H+32H+33H+41H+42H+43H=0\underbrace{1}_{(H)}\underbrace{62}_{(L)}H$$

Precautions

When the automatic removal of NULL characters is performed to data transmitted using the bidirectional protocol, always specify the "Word/byte units designation" to "Byte units".

Ex.

Example of data transmission after automatic removal of NULL characters using the bidirectional protocol



27.6 Program Example

The examples of data communication using the bidirectional protocol are shown below.

Program example of data communication using the bidirectional protocol

This section shows program examples data communication using the bidirectional protocol when the settings/registrations are made with Engineering tool.

parameter setting

For details for each setting item, refer to the parameters. (☞ Page 454 PARAMETER SETTING)

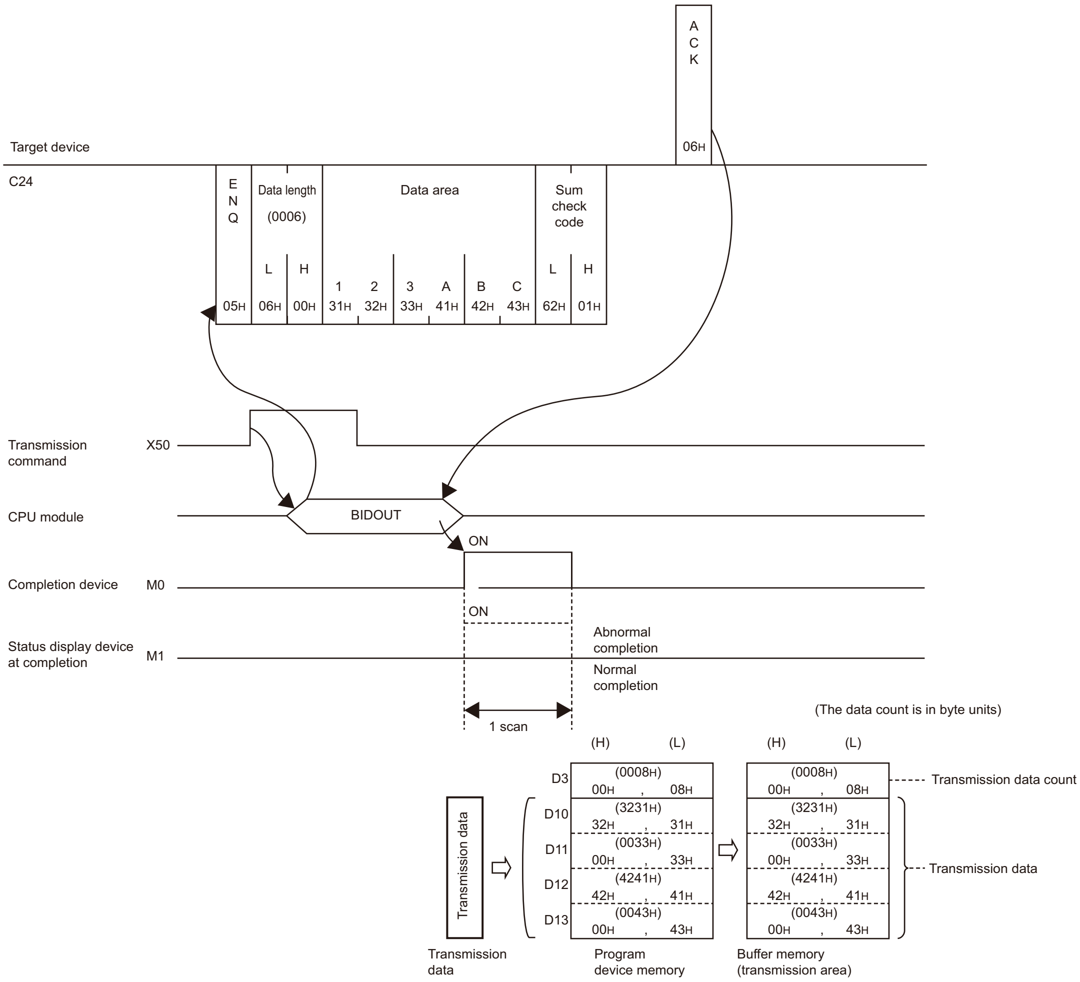
Setting item				Setting content	
				CH1	CH2
Basic settings	Various control specification	Test mode setting		—	CH2 is not used
		Communication protocol setting		Bidirectional protocol	
		Communication speed setting		Set according to the target device	
		Transmission setting	Operation setting	Independent	
			Data bit	Set according to the target device	
			Parity bit		
			Odd/even parity		
			Stop bit		
			Sum check code		
			Online change		
			Setting change		
		Station number setting (CH1, 2 in common: 0 to 31)		—	
		Communication control specification	NULL character automatic removal designation	Auto delete enabled	
			Word/byte units designation	Byte unit	

Example of data transmission

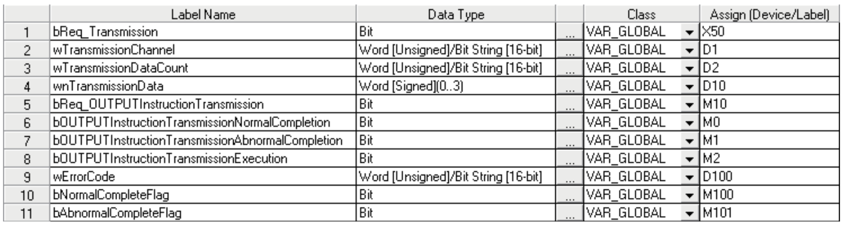
C24 transmits a string data using the NULL character automatic removal function to the target device.

Ex.

Transmitting given data with BIDOUT dedicated instruction



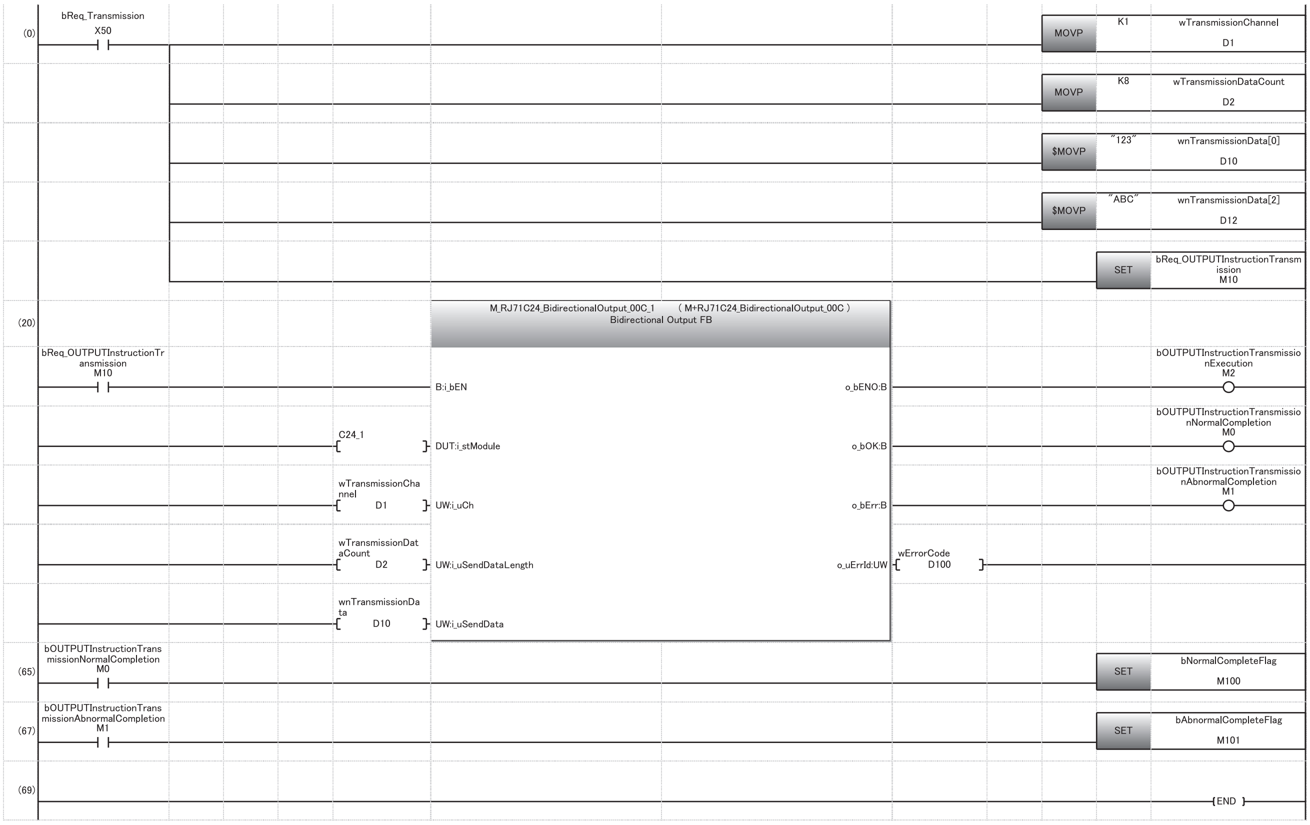
■Program example

Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_BidirectionalOutput	Sends data for specified data points.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_Transmission	Bit	...	VAR_GLOBAL	X50
2	wTransmissionChannel	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
3	wTransmissionDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2
4	wnTransmissionData	Word [Signed](0..3)	...	VAR_GLOBAL	D10
5	bReq_OUTPUTInstructionTransmission	Bit	...	VAR_GLOBAL	M10
6	bOUTPUTInstructionTransmissionNormalCompletion	Bit	...	VAR_GLOBAL	M0
7	bOUTPUTInstructionTransmissionAbnormalCompletion	Bit	...	VAR_GLOBAL	M1
8	bOUTPUTInstructionTransmissionExecution	Bit	...	VAR_GLOBAL	M2
9	wErrorCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D100
10	bNormalCompleteFlag	Bit	...	VAR_GLOBAL	M100
11	bAbnormalCompleteFlag	Bit	...	VAR_GLOBAL	M101

For information on the module FB to be used, refer to the following manual.

MELSEC iQ-R Serial Communication Module Function Block Reference



(0) Set the transmission channel to CH1.
Set the send data quantity to 8.
Set the data to be transmitted.
Set the transmission direction.

(65) Normal completion

(67) Abnormal completion

PART 3 PARAMETER SETTINGS AND TROUBLESHOOTING

This part explains parameters and troubleshooting of C24.

28 PARAMETER SETTING

This section explains parameter settings necessary for data communication between C24 and a target device.

28.1 Parameter Setting Procedure

This section explains the parameter setting procedure to use RJ71C24.

1. Add C24 to Engineering tool.

🖱 [Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ right-click ⇨ [Add New Module]

2. Module parameters and module extended parameters are included in parameters. There are four types of module parameters: basic settings, application settings, interrupt settings, and refresh setting, and three types of module extended parameters: user frame contents, modem initialization data, and modem connection data, which are selected from the tree in the screen below.

🖱 Navigation window ⇨ [Parameter] ⇨ [Module Information] ⇨ RJ71C24 ⇨ [Module Parameter]/[Module Extended Parameter]

3. Write the settings to the CPU module with Engineering tool after the parameter setting is completed.

🖱 [Online] ⇨ [Write to PLC]

4. The settings are reflected by resetting the CPU module or turning the power OFF → ON.

28.2 Module Parameters

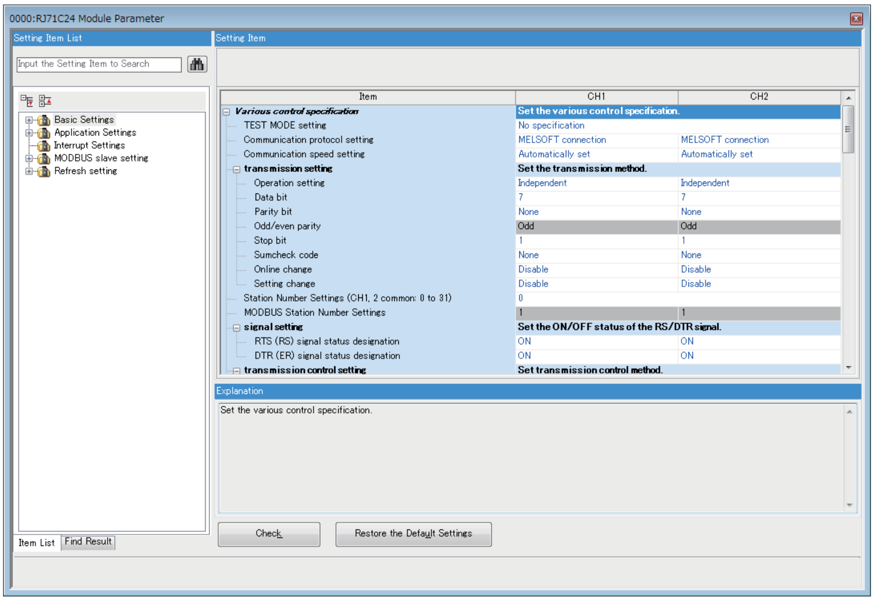
Set the module parameters. The module parameter includes the following four types: basic settings, application settings, interrupt settings, and refresh setting, which is selected from the tree in the screen below.

Navigation window ⇨ [Parameters] ⇨ [Module Information] ⇨ Module name ⇨ [Module Parameter]

Basic settings

Configure the settings to operate the basic functions of C24.

Window



Item	CH1	CH2
Various control specification	**Set the various control specification.**	
TEST MODE setting	No specification	
Communication protocol setting	MELSOFT connection	MELSOFT connection
Communication speed setting	Automatically set	Automatically set
transmission setting	**Set the transmission method.**	
Operation setting	Independent	Independent
Data bit	7	7
Parity bit	None	None
Odd/even parity	Odd	Odd
Stop bit	1	1
Sumcheck code	None	None
Online change	Disable	Disable
Setting change	Disable	Disable
Station Number Settings (CH1, 2 common: 0 to 31)	0	
MODBUS Station Number Settings	1	1
signal setting	**Set the ON/OFF status of the RS/DTR signal.**	
RTS (RS) signal status designation	ON	ON
DTR (ER) signal status designation	ON	ON
transmission control setting	**Set transmission control method.**	

Set the various control specification.

Various control specification

Specify the various controls.

Item		Description	Setting range
Test mode setting[*1]		Sets the test mode. When a problem occurs during data communication with a target device, check whether or not there is any problem in the operation of the C24.	• No specification (other than test mode) • Hardware test • Self-loopback test (Default: No specification (other than test mode))
Communication protocol setting		Sets a protocol for communicating with a target device.	• MELSOFT Connection • MC Protocol (Format 1) • MC Protocol (Format 2) • MC Protocol (Format 3) • MC Protocol (Format 4) • MC Protocol (Format 5) • Nonprocedural protocol • Bidirectional protocol • Predefined protocol • MODBUS Slave (RTU) • MODBUS Slave (ASCII) • Simple PLC Communication (Default: MELSOFT Connection)
Communication speed setting		Sets a communication rate.	• Automatically set[*2] • 1200 bps • 2400 bps • 4800 bps • 9600 bps • 14400 bps • 19200 bps • 28800 bps • 38400 bps • 57600 bps • 115200 bps • 230400 bps (Default: Automatically set)
Transmission setting	Operation setting[*3]	Configures an operation settings of two interfaces.	• Independent • Interlink (Default: Independent)
	Data bit	Sets a bit length for one character in data to be transmitted/received to/from a target device.	• 7 • 8[*4] (Default: 7)
	Parity bit	Set whether or not a parity bit (vertical parity) is added for one character of data to be transmitted/received.	• None • Yes (Default: None)
	Odd/even parity	Sets which parity bit is used, odd parity or even parity, when a parity bit is added. (This will be enabled only when the parity bit is set to "Yes".)	• Odd • Even (Default: Odd)
	Stop bit	Sets a stop bit length for one character in data to be transmitted/received to/from a target device.	• 1 • 2 (Default: 1)
	Sum check code	Sets whether or not a sum check code is added to transmission message and reception message.	• None • Yes (Default: None)
	Online change	Sets whether or not data is written to a CPU module during RUN when writing data from a target device to the CPU module.	• Disable • Enable (Default: Disable)
	Setting change	Sets whether or not the following processing is allowed after the startup. • Data communication function of each interface, changing of transmission specification, changing of communication protocol • Writing data to flash ROM	• Disable • Enable (Default: Disable)
Station number setting (CH1, 2 in common: 0 to 31)		Sets the host station number during multidrop communication.	0 to 31 (Default: 0)
MODBUS station number setting[*5]		Sets the station number of a MODBUS slave station.	1 to 247 (Default: 1)

Item		Description	Setting range
Signal specification	RTS (RS) signal status designation	Sets the ON/OFF state of RTS(RS) signal.	• OFF • ON (Default: ON)
	DTR (ER) signal status designation	Sets the ON/OFF state of DTR (ER) signal.	• OFF • ON (Default: ON)
Transmission control specification	Transmission control	Sets which controls data communication with a target device, transmission control signal or DC code.	• DTR/DSR control • DC code control (Default: DTR/DSR control)
	DC1/DC3 control	Sets whether or not DC1/DC3 control is enabled.	• Control disabled • Control enabled (Default: Control disabled)
	DC2/DC4 control	Sets whether or not DC2/DC4 control is enabled.	• Control disabled • Control enabled (Default: Control disabled)
	DC1 code	Sets the control data to inform a target device that C24 can receive data.	00H to FFH (Default: 11H)
	DC3 code	Sets the control data to inform a target device that C24 cannot receive data.	00H to FFH (Default: 12H)
	DC2 code	Sets the control data to inform a target device the start of a valid data.	00H to FFH (Default: 13H)
	DC4 code	Sets the control data to inform a target device the end of a valid data.	00H to FFH (Default: 14H)
	Transmission control start free space designation	Designates a free capacity in the OS area that informs data cannot be received.	64 to 4095 (Default: 64)
	Transmission control end free space designation	Designates a free capacity in the OS area that informs data can be received.	263 to 4096 (Default: 263)
	Nonprocedural no-reception monitoring time method designation	Sets the format to use the no-reception monitoring time (timer 0).	• Method 0 • Method 1 (Default: Method 0)
Communication control specification	Word/byte units designation	Sets the unit of data length (data count).	• Word unit • Byte unit (Default: Word unit)
	CD terminal check designation	Checks the state of the CD (DCD) signal.	• Check • Do not check (Default: Do not check)
	Communication method designation	Sets the communication method.	• Full-duplex communication • Half-duplex communication (Default: Full-duplex communication)
	Echo back enable/prohibit specification	Sets whether or not the returned data is received.	• Echo back enable • Echo back prohibit (Echo back prohibit)
	NULL character automatic removal designation	Sets whether or not NULL character is automatically deleted.	• Auto delete disabled • Auto delete enabled (Default: Auto delete disabled)
	Communication data reception enable/disable designation	Sets this if noise signals are added on the line when turning ON the power of the target device or connecting the communication cable. Setting to "Receive enable", improper data due to noise signal addition can be discarded. Set this designation to "Receive enable" by a program after the serial line has been settled. While this designation is set to "Receive disable", the normal data also is discarded.	• Receive enable • Receive disable (Default: Receive enable)
	Receive data enable/disable	Sets whether data received simultaneously is valid or invalid.	• Enable • Disable (Default: Enable)
	Send data enable/disable	Sets whether data transmitted simultaneously is valid or invalid.	• Enable • Disable (Default: Enable)

Item		Description	Setting range
Half-duplex communication control specification	Simultaneous transmission priority/non-priority designation	Sets whether the transmission from C24 is continued or discontinued when C24 and a target device stat data transmission at the same time. 0.0: Priority 0.1 to 25.5: Non-priority (Transmission wait time : unit: s)	0.0 to 25.5 (Default: 0.0: Priority)
	Retransmission time transmission method designation	Sets whether the discontinued message is transmitted from the beginning or it is transmitted from the part disconnected when C24 restarts transmission following the discontinuation at the time of simultaneous transmission.	• Do not resend • Resend (Default: Do not resend)
Data communication time monitoring specification	No-reception monitoring time (timer 0) designation	Sets a monitoring time for the no-reception monitoring time (timer 0). 0: Unlimited wait 40 to 4000: Monitoring designation (Unit: byte)	0, 40 to 4000 (Default: 0: Unlimited wait)
	Response monitoring time (timer 1) designation	Set a monitoring time for the response monitoring time (timer 1). 0: Unlimited wait 0.1 to 300.0: Monitoring designation (Unit: s)	0, 0.1 to 300.0 (Default: 0.5)
	Send monitoring time (timer 2) designation	Set the monitoring time for transmission monitoring time (timer 2). 0: Unlimited wait 0.1 to 300.0: Monitoring designation (Unit: s)	0, 0.1 to 300.0 (Default: 180.0)
Send wait time specification	Transmission wait time designation	Set a wait time to transmit a response message to a message received from a target device. 0: No wait time 10 to 150: Monitoring designation (multiples of ten) (Unit: ms)	0, 10 to 150 (multiples of ten) (Default: 0: No wait time)
Send/receive data monitoring function specification	Send/receive data monitoring designation	Set the start/stop communication data monitoring.	• No monitor/stop instruction • Monitor start instruction (Default: No monitor/stop instruction)
	Stop at buffer full specification	Sets whether the monitor is stopped when the monitor data area is full (buffer full).	• OFF • ON (Default: OFF)
	0 timer error outbreak stop designation	Set whether the monitor is stopped when the timer 0 error occurs.	• OFF • ON (Default: OFF)
	Monitor buffer start address designation	Set the start address of the storage area for the monitored data.	1024 to 6909 (400H to 1AFDH), 9728 to 16381 (2600H to 3FFDH) (Default (CH1): 9728 (2600H)) (Default (CH2): 13056 (3300H))
	Monitor buffer size designation	Set a monitor data area size.	3 to 6656 (3H to 1A00H) (Default: 3328(D00H))
On-demand function specification	Buffer memory start address designation	Set the start address of the buffer memory used with the on-demand function.	1024 to 6911 (400H to 1AFFH), 9728 to 16383 (2600H to 3FFFH) (Default (CH1): 1024 (400H)) (Default (CH2): 2048 (800H))
	Data length designation	Set a data length to be transmitted using the on-demand function.	0 to 13312 (0H to 3400H) (Default: 0 (0H))
Send area specification	Send buffer memory start address designation	Set the start address of the area used as the send area.	1024 to 6911 (400H to 1AFFH), 9728 to 16383 (2600H to 3FFFH) (Default (CH1): 1024 (400H)) (Default (CH2): 2048 (800H))
	Send buffer memory size specification	Set the size of the area used as the send area.	1 to 6656 (1H to 1A00H) (Default: 512 (200H))

Item		Description	Setting range
Receiving area specification	Receiving buffer memory start address designation	Set the start address of the area used as the receive area.	1024 to 6911 (400H to 1AFFH), 9728 to 16383 (2600H to 3FFFH) (Default (CH1): 1536 (600H)) (Default (CH2): 2560 (A00H))
	Receiving buffer memory size specification	Set the size of the area used as the receive area.	1 to 6656 (1H to 1A00H) (Default: 512 (200H))
Conversion designation	ASCII-binary conversion designation	Set whether or not data to be communicated to a target device is communicated with ASCII code.	• Do not convert • Convert (Default: Do not convert)
Transparent code designation_Transparent code for sending specification (nth)[*6]	Transparent code	Set one-byte data for transmission control. Set '0' for both the transparent code and additional code when the transparent code for sending is not specified.	00H to FFH (Default: 00H: Do not designate (The transparent code and the additional code are both 0.))
	Additional code	Set one-byte data added immediately before the transparent code and additional code data. Set the additional code within the range of 01H to FFH when the transparent code for sending is specified. Set '0' for both the transparent code and additional code when it is not specified.	00H to FFH (Default: 00H: Do not designate (The transparent code and the additional code are both 0.))
Transparent code designation_Transparent code for receiving specification	Transparent code	Set one-byte data for transmission control. Set '0' for both the transparent code and additional code when the transparent code for receiving is not specified.	00H to FFH (Default: 00H: Do not designate (The transparent code and the additional code are both 0.))
	Additional code	Set one-byte data to be deleted at the data reception. (The immediately succeeding one-byte data is processed for reception.) Set the additional code within the range of 01H to FFH when the transparent code for receiving is specified. Set '0' for both the transparent code and additional code when it is not specified.	00H to FFH (Default: 00H: Do not designate (The transparent code and the additional code are both 0.))
Receiving end specification	Receive end data quantity designation	Set the receive end data quantity.	1 to 13310 (Default: 511)
	Receive end code designation	Set a receive end code. 0D0AH: CR, LF 00H to FFH: Arbitrary receive end code FFFFH: No designation	0000H to 00FFH, 0D0AH, FFFFH (Default: 0D0AH: CR, LF)
Protocol execution history specification option		Set whether execution logs of all protocol are stored or whether execution logs of only abnormally completed protocol are stored.	• Only protocol that completed abnormally • All protocol execution status and execution history (Default: Only protocol that completed abnormally)
Simple PLC Communication Setting[*7]		Set whether to use simple CPU communication.	Enable (Default: Enable)

*1 It can be set only on the CH1 side. When the test mode is selected, the communication protocol setting on the CH1 side and the setting on the CH2 side cannot be configured (will be invalid).

*2 It will be enabled only when "MELSOFT Connection" is selected for the communication protocol setting.

*3 It can be set on the CH2 side only. (Since the linked operation is not available for RJ71C24-R2, it is fixed to "Independent".)
When it is set to "Interlink", the communication protocol setting for the CH2 side cannot be configured.

*4 Set '8' for data communication using MC protocol (format 5), bidirectional protocol, or MODBUS slave (RTU).
For data communication using simple CPU communication, it differs depending on the device type of a communication destination.

*5 It can be set when "MODBUS Slave (RTU)" or "MODBUS Slave (ASCII)" is selected for the communication protocol setting.

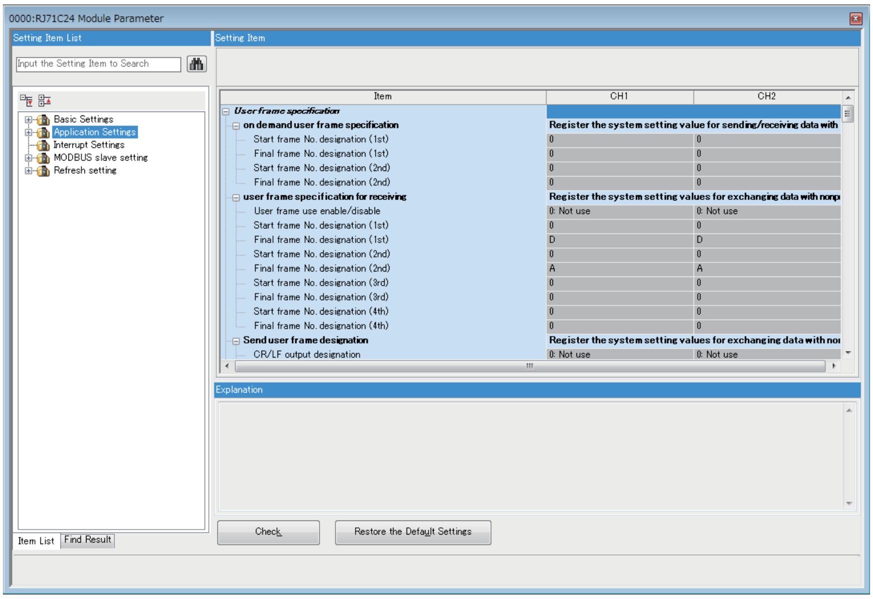
*6 n: 1 to 10

*7 It will be enabled when "Simple PLC Communication" is selected for the communication protocol setting.

Application settings

Configure the settings such as the user frame specification.

Window



Item	CH1	CH2
User frame specification		
on demand user frame specification	Register the system setting value for sending/receiving data with	
Start frame No. designation (1st)	0	0
Final frame No. designation (1st)	0	0
Start frame No. designation (2nd)	0	0
Final frame No. designation (2nd)	0	0
user frame specification for receiving	Register the system setting values for exchanging data with nonp	
User frame use enable/disable	0: Not use	0: Not use
Start frame No. designation (1st)	0	0
Final frame No. designation (1st)	D	D
Start frame No. designation (2nd)	0	0
Final frame No. designation (2nd)	A	A
Start frame No. designation (3rd)	0	0
Final frame No. designation (3rd)	0	0
Start frame No. designation (4th)	0	0
Final frame No. designation (4th)	0	0
Send user frame designation	Register the system setting values for exchanging data with no	
CR/LF output designation	0: Not use	0: Not use

Setting data

Item	Description	Reference
User frame specification	Sets the system setting value for data communication using a user frame.	Page 461 User frame specification
Modem function setting	Sets the system setting value for data communication using the modem function.	Page 462 Modem function setting
Programmable controller CPU monitoring function setting	Sets the system setting value to use the programmable controller CPU monitoring function.	Page 463 Programmable controller CPU monitoring function setting

User frame specification

Set the system setting value for data communication using a user frame.

Item		Description	Setting range
On-demand user frame specification	Start frame No. designation (nth)	Sets the user frame number to be used.	0000H to 801FH (Default: 0000H)
	Final frame No. designation (nth)	Sets the user frame number to be used.	0000H to 801FH (Default: 0000H)
User frame specification for receiving	User frame use enable/disable designation	Sets whether or not a user frame is used for data communication.	• Not use • Use (Default: Not use)
	Start frame No. designation (nth)	Sets the user frame number to be used.	0000H to 801FH (Default: 0000H)
	Final frame No. designation (nth)	Sets the user frame number to be used.	0000H to 801FH (Default: 0000H)
Send user frame designation	CR/LF output designation	Sets whether or not CR/LF is transmitted.	• Do not send • Send (Default: Do not send)
	Output start pointer designation	Sets the head position in the transmission frame number designation area to which the registration number of the user frame to be transmitted is written.	0 to 100 (Default: 0)
	Output quantity designation	Sets the number of user frame to be transmitted from the designated position in the output start pointer designation area.	0 to 100 (Default: 0)
	Send frame No. designation (nth)	Sets the user frame number to be transmitted from the designated position in the output start pointer designation area in the order of output.	0000H to FFFFH (Default: 0000H)
User frame receiving method designation	User frame receiving method designation (nth)	Sets the receive format of a user frame.	• Method 0 • Method 1 (Default: Method 0)
	Method 1 dedicated receive end data quantity designation (nth)	Sets the number of word/byte in the arbitrary data portion at the time of data reception using the user frame receive method 1.	0000H to FFFFH (Default: 0000H)

Modem function setting

Set the system setting value for data communication using the modem function.

Item		Description	Setting range
Modem function specification 1	Modem connection CH specification	Sets a channel to connecting a modem/TA	• None • CH1 • CH2 (Default: None)
	Connection re-try count designation	Sets a number of retries for the connection request.	1 to 5 (Default: 5)
	Connection retry interval designation	Sets an interval of the retry processing for the connection request. (Unit: s)	90 to 300 (Default: 180)
	Initialization/ connection timeout designation	Sets the wait time below. • Wait time until the modem/TA initialization is complete. • Wait time per one connection request (Unit: s)	1 to 60 (Default: 60)
	Number of initialization retries designation	Sets a number of initialization retries to the modem.	1 to 5 (Default: 3)
	Data No. for initialization designation	Sets the data number to be transmitted by the initialization request to the modem.	0000H, 07D0H to 07DDH (2000 to 2013), 09C4H to 09E1H (2500 to 2529) 8001H to 801FH (32769 to 32799) (Default: 0000H: No designation)
	Data No. for connection designation	Sets the data number for connection to be used for the processing to connect to a target device for data transmission.	0000H, 0BB8H to 0BD5H (3000 to 3029) 8001H to 801FH (32769 to 32799) (Default: 0000H: No designation)
	No-communication interval time designation	Sets the wait time before the line is closed when data communications with a target device is ceased after the line has been connected. 0: Infinite wait 1 to 120 (unit: minute)	0: Infinite wait 1 to 120 (0: Infinite wait)
	RS/CS control enable/disable designation	Sets whether or not the control to notify the reception capability of the host station to the target side with RS/CS signal is performed during data communication between C24 and a modem/TA.	• Control disabled • Control enabled (Default: Control enabled)
Modem function specification 2	Auto modem initialization designation	Sets whether or not the modem is automatically initialized.	• Do not automatically initialize • Automatically initialize (Default: Do not automatically initialize)
	Modem initialization DR (DSR) signal enable/disable designation	Sets how to treat DR signal when initializing a modem.	• DR signal enabled • DR signal disabled (Default: DR signal disabled)
Modem function specification 3	Line disconnection wait time specification (PLC CPU monitor)	Sets the time until data transmission from a modem on the host station side to the target device side is completed for data communication with the programmable controller CPU monitoring function.	0000H to FFFFH (Default: 0000H)
Remote password function	Remote password mismatch notification count designation	Sets the number of times to notify the CPU module that a remote password mismatch occurred during the unlock processing by a target device after the modem line has been connected.	0000H to FFFFH (Default: 0000H: No designation)
	Remote password mismatch notification accumulated count designation	Set the accumulated number of times to notify the CPU module that a remote password mismatch occurred during the unlock processing by a target device after C24 has started up.	0000H to FFFFH (Default: 0000H: No designation)

Programmable controller CPU monitoring function setting

Set the system setting value to use the programmable controller CPU monitoring function.

Item			Description	Setting range
Programmable controller CPU monitoring function designation			Sets the timing to transmit the programmable controller CPU monitoring results to a target device.	• Do not use the function • Fixed cycle send • Condition match send (Default: Do not use the function)
Cycle time units designation			Sets the unit of the cycle time.	• 100 ms • Second • Minute (Default: Minute)
Cycle time designation (programmable controller CPU monitoring interval time)			Set a cycle time. (The unit depends on the cycle time units designation.)	1 to 65535 (Default: 5)
Send pointer designation			Sets the pointer position of the user frame to be transmitted.	0 to 100 (Default: 0: No designation)
Output quantity designation			Sets the number of user frame to be transmitted.	0 to 100 (Default: 0: No designation)
Block monitoring device	Registered word block quantity designation		Sets the number of registered word blocks.	0 to 10 (Default: 0: No designation)
	Registered bit block quantity designation		Sets the number of registered bit blocks.	0 to 10 (Default: 0: No designation)
	CPU error monitoring designation		Sets whether or not the monitoring for abnormality on CPU module is performed.	• Do not monitor • Monitor (Default: Do not monitor)
	n-th block monitoring device*1	Monitoring device designation	Sets the device code to be monitored.	☞ Page 154 List of device code (Default: Do not designate)
		Start device No. specification	Sets the start number of device to be monitored.	☞ Page 154 List of device code (Default: 0)
		Number of read points specification	Sets the read point.	0000H to FFFFH (Default: 0000H)
		Monitoring condition specification (judgment condition specification)	Sets the judgment conditions for the monitoring condition value.	• Do not designate • =(Edge) • <>(Edge) • <=(Unsigned, Edge) • <(Unsigned, Edge) • >=(Unsigned, Edge) • >(Unsigned, Edge) • <=(Signed, Edge) • <(Signed, Edge) • >=(Signed, Edge) • >(Signed, Edge) • =(Level) • <>(Level) • <=(Unsigned, Level) • <(Unsigned, Level) • >=(Unsigned, Level) • >(Unsigned, Level) • <=(Signed, Level) • <(Signed, Level) • >=(Signed, Level) • >(Signed, Level) (Default: Do not designate)

Item			Description	Setting range
Block monitoring device	n-th block monitoring device[*1]	Monitoring condition value designation	Sets the condition and numeric value for the monitoring condition.	■When the monitoring device is a word device 0000H to FFFFH ■When the monitoring device is a bit device • 0000H: OFF • 0001H: ON (Default: 0000H)
		Send pointer designation	Sets the pointer position of the user frame to be transmitted.	0 to 100 (Default: 0: No designation)
		Output quantity designation	Sets the number of user frame to be transmitted.	0 to 100 (Default: 0: No designation)
CPU error monitoring designation	Send pointer designation		Sets the pointer position of the user frame to be transmitted.	0 to 100 (Default: 0: No designation)
	Output quantity designation		Sets the number of user frame to be transmitted.	0 to 100 (Default: 0: No designation)

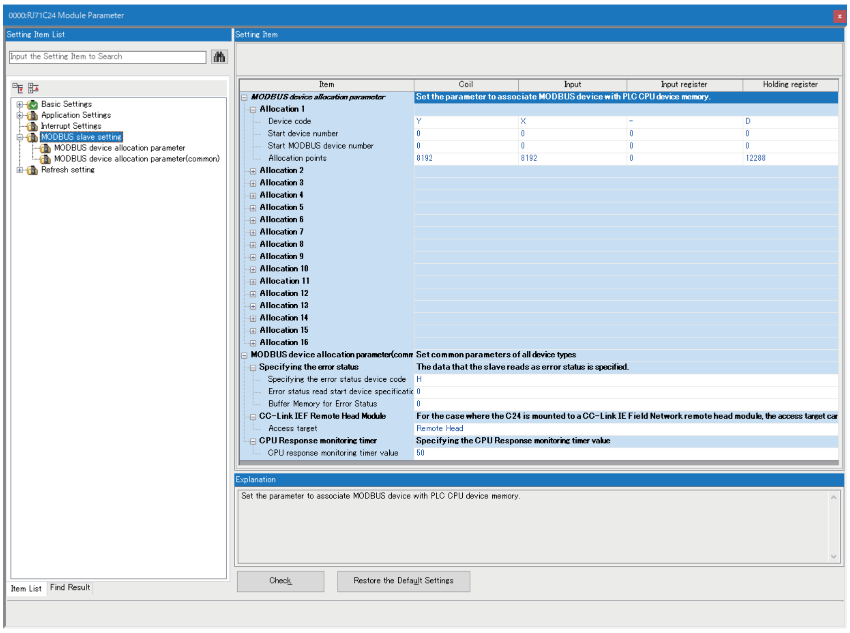
*1 n: 1 to 10

MODBUS slave settings

Configure the settings for communication using the MODBUS slave function.

This can be set when "MODBUS Slave (RTU)" or "MODBUS Slave (ASCII)" is selected for the communication protocol setting.

Window


0000:RJ71C24 Module Parameter
Setting Item List
Input the Setting Item to Search
Basic Settings
Application Settings
Interrupt Settings
MODBUS slave setting
MODBUS device allocation parameter
MODBUS device allocation parameter(common)
Refresh setting
Item List
Find Result
Setting Item
Explanation
Check
Restore the Default Settings

Item	Coil	Input	Input register	Holding register
MODBUS device allocation parameter	Set the parameter to associate MODBUS device with PLC CPU device memory.			
Allocation 1				
Device code	Y	X	-	D
Start device number	0	0	0	0
Start MODBUS device number	0	0	0	0
Allocation points	8192	8192	0	12288
Allocation 2				
Allocation 3				
Allocation 4				
Allocation 5				
Allocation 6				
Allocation 7				
Allocation 8				
Allocation 9				
Allocation 10				
Allocation 11				
Allocation 12				
Allocation 13				
Allocation 14				
Allocation 15				
Allocation 16				
MODBUS device allocation parameter(comm	Set common parameters of all device types			
Specifying the error status	The data that the slave reads as error status is specified.			
Specifying the error status device code	H			
Error status read start device specificatic	0			
Buffer Memory for Error Status	0			
CC-Link IEF Remote Head Module	For the case where the C24 is mounted to a CC-Link IE Field Network remote head module, the access target car			
Access target	Remote Head			
CPU Response monitoring timer	Specifying the CPU Response monitoring timer value			
CPU response monitoring timer value	50			

Set the parameter to associate MODBUS device with PLC CPU device memory.

Setting data

Setting item		Description	Setting range
MODBUS device allocation parameter	Allocation 1 to 16	Set parameters to associate MODBUS devices with device memories of a CPU module. ☞ Page 467 MODBUS device assignment parameters	—

Setting item			Description	Setting range
MODBUS device allocation parameter (common)	Specifying the error status	Specifying the error status device code	Specify data read as an error status by a slave (C24) when the read exception status (FC: 07) is received from a MODBUS master device. • For device designation: Designate a device code and a device number. • For buffer memory setting: Specify 'H' for the device code. Specify '0' and a buffer memory setting value (0 to 255) for the device number and "Buffer Memory for Error Status" respectively.	• SM • X • Y • M • L • F • V • B • TS • TC • STS • STC • CS • CC • LCS • LCC • SB • S • DX • DY • H (Default: H)
		Error status read head device specification		• H device: 0 • Devices other than H: Follow the device settings of the CPU parameter. (Default: 0)
		Buffer Memory for Error Status	Specify the setting value of a specified buffer memory when specifying the buffer memory for the specifying the error status device code. It is enabled only "H0" is entered for the specifying the error status device code and the number specification.	0 to 255 (Default: 0)
	CC-Link IEF Remote Head Module	Access target	Specify an access target when a C24 is connected to a CC-Link IE Field Network remote head module.	• Remote Head • Master Station (Default: Remote Head)
	CPU Response monitoring timer		Set a response monitoring time from a CPU module. If there is no response from the CPU module within a set time after a request message is received from a MODBUS master device, the response waiting state of a C24 is released.	• 0: 5s • 1 to 12000: Setting time (unit: 100 ms) (Default: 50)

MODBUS device assignment parameters

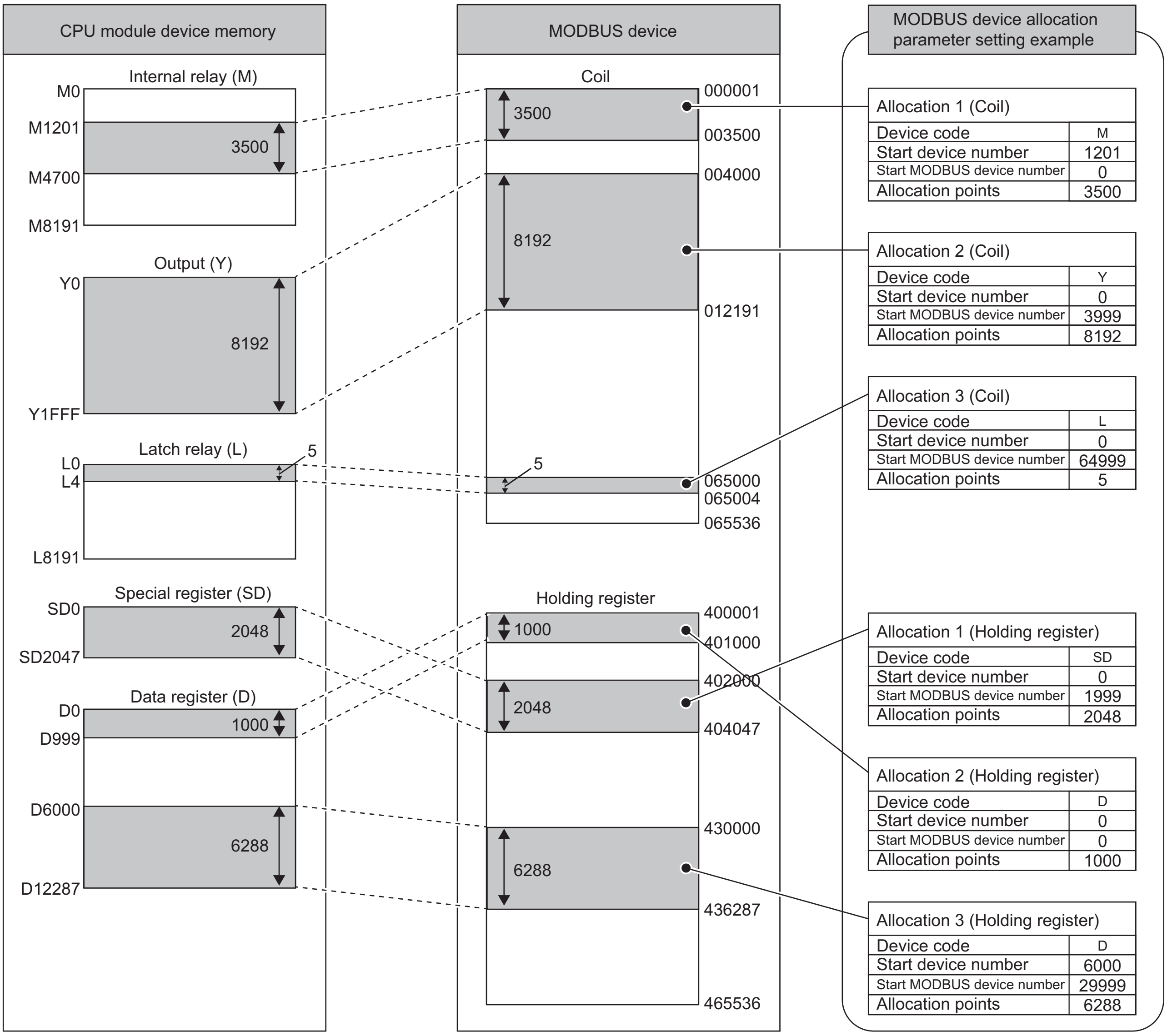
MODBUS device assignment parameters are used for associating MODBUS devices with device memories of a CPU module. By using them, direct access from a MODBUS master device to device memories of a CPU module is available.

The range of devices to be assigned must be set so that it does not exceed the range of devices that can be used for a CPU module.

The following items can be set for each allocation 1 to 16.

Item	Description	Setting range
Device code	Set a device of a CPU module to be assigned to a MODBUS device. Specify the start device number in hexadecimal regardless of the device type. Set 'H' for the device code when using the buffer memory. Set an address of 'User setting area for MODBUS' (Un\G49152 to Un\G53247) in hexadecimal (C000H to CFFFH) for the start device number.	• For using the buffer memory: HC000 to HCFFF • Devices other than the buffer memory (H): Follow the device settings of the CPU parameter. (Default: ☞ Page 470 Default assignment parameters)
Start device number		
Start MODBUS device number	Set the start number of a MODBUS device to which a device of a CPU module is assigned. A setting value of the start MODBUS device number can be calculated with the following formula: • Start MODBUS device number = Last five digits of the setting target MODBUS device number - 1	0 to 65535 (Default: ☞ Page 470 Default assignment parameters)
Allocation points	Set the number of device points of a CPU module assigned to a MODBUS device.	

The following shows the schematic diagram of the MODBUS device assignment parameter setting.



Allocation 1 (Coil)

Device code	M
Start device number	1201
Start MODBUS device number	0
Allocation points	3500

Allocation 2 (Coil)

Device code	Y
Start device number	0
Start MODBUS device number	3999
Allocation points	8192

Allocation 3 (Coil)

Device code	L
Start device number	0
Start MODBUS device number	64999
Allocation points	5

Allocation 1 (Holding register)

Device code	SD
Start device number	0
Start MODBUS device number	1999
Allocation points	2048

Allocation 2 (Holding register)

Device code	D
Start device number	0
Start MODBUS device number	0
Allocation points	1000

Allocation 3 (Holding register)

Device code	D
Start device number	6000
Start MODBUS device number	29999
Allocation points	6288

■Available devices

The following table lists the devices of a CPU module that can be set for MODBUS devices.

Device					MODBUS device (○: Assignable, ×: Not assignable)				
Device name		Symbol	Device code	Type	Coil	Input	Input register	Holding register	Extended file register
Special relay		SM	0002H	Bit	○	○	○*1	○*1	×
Special register		SD	0021H	Word	×	×	○	○	×
Input		X	0010H	Bit	○	○	○*1	○*1	×
Output		Y	0011H		○	○	○*1	○*1	×
Internal relay		M	0001H		○	○	○*1	○*1	×
Latch relay		L	0003H		○	○	○*1	○*1	×
Annunciator		F	0004H		○	○	○*1	○*1	×
Edge relay		V	0005H		○	○	○*1	○*1	×
Link relay		B	0014H		○	○	○*1	○*1	×
Data register		D	0020H	Word	×	×	○	○	×
Link register		W	0030H		×	×	○	○	×
Timer	Contact	TS	0041H	Bit	○	○	○*1	○*1	×
	Coil	TC	0040H		○	○	○*1	○*1	×
	Current value	TN	0042H	Word	×	×	○	○	×
Long timer	Current value	LTN	0052H	Double word	×	×	○*4	○*4	×
Retentive timer	Contact	STS	0049H	Bit	○	○	○*1	○*1	×
	Coil	STC	0048H		○	○	○*1	○*1	×
	Current value	STN	004AH	Word	×	×	○	○	×
Long retentive timer	Current value	LSTN	005AH	Double word	×	×	○*4	○*4	×
Counter	Contact	CS	0045H	Bit	○	○	○*1	○*1	×
	Coil	CC	0044H		○	○	○*1	○*1	×
	Current value	CN	0046H	Word	×	×	○	○	×
Long counter	Contact	LCS	0055H	Bit	○	○	○*1	○*1	×
	Coil	LCC	0054H		○	○	○*1	○*1	×
	Current value	LCN	0056H	Double word	×	×	○*2	○*2	×
Link special relay		SB	0015H	Bit	○	○	○*1	○*1	×
Link special register		SW	0031H	Word	×	×	○	○	×
Step relay*5		S	0008H	Bit	○	○	○*1	○*1	×
Direct access input		DX	0016H	Bit	○	○	○*1	○*1	×
Direct access output		DY	0017H		○	○	○*1	○*1	×
Index register	Index register	Z	0060H	Word	×	×	○	○	×
	Long index register	LZ	0062H	Double word	×	×	○*2	○*2	×
File register	Block switching method	R	0027H	Word	×	×	○	○	×
	Serial number access method	ZR	0028H		×	×	×	×	○*3
Refresh data register		RD	002CH	Word	×	×	○	○	×
Buffer memory		H	00A7H	Word	×	×	○	○	×

*1 When setting a bit type device for an input register and output register, set the number of assigned points to a multiple of 16.

*2 Two points of MODBUS devices are used.

*3 Fixed to the file register (ZR).

*4 The long timer (current value) and long retentive timer (current value) use four points of MODBUS devices. (☞ Page 469 Reading of the long timer and long retentive timer)

*5 The step relay is read-only. An attempt of writing will cause an error.

■Reading of the long timer and long retentive timer

When reading the long timer and long retentive timer, note that the following devices cannot be set:

- Long timer (contact)
- Long timer (coil)
- Long retentive timer (contact)
- Long retentive timer (coil)

When, however, the current value of the long timer or long retentive timer is set in units of four words, the contact and coil will be read simultaneously.

Response data	Description
1st word	The current value is stored.
2nd word	
3rd word	• b0: The value of a coil is stored. • b1: The value of a contact is stored. • b2 to b15: Used by the system.
4th word	Used by the system.

As shown above, the long timer and long retentive timer use the data configuration of four words per device. Consequently, specify the number of device points in units of four points.

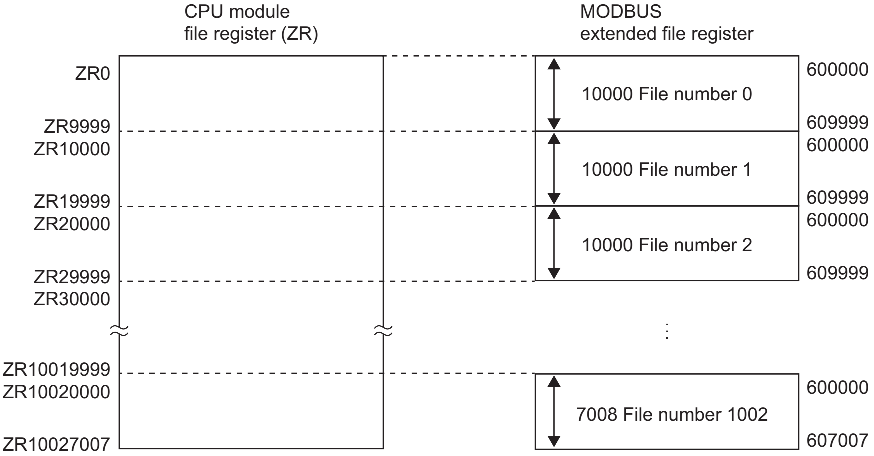
■Writing of the long timer and long retentive timer

Note that the long timer and long retentive timer cannot be set.

■MODBUS extended file register assignment

The assignment of the MODBUS extended file register to the CPU module is fixed to the file register (ZR).

The following shows the assignment to the file register of the CPU module. (The number of points for the file register shows the maximum size that can be set in the CPU module (when the R120CPU + NZ2MC-16MBS are mounted)).



■Size of a MODBUS extended file register

The size of a MODBUS extended file register depends on that of the file register (ZR) set in a CPU module.

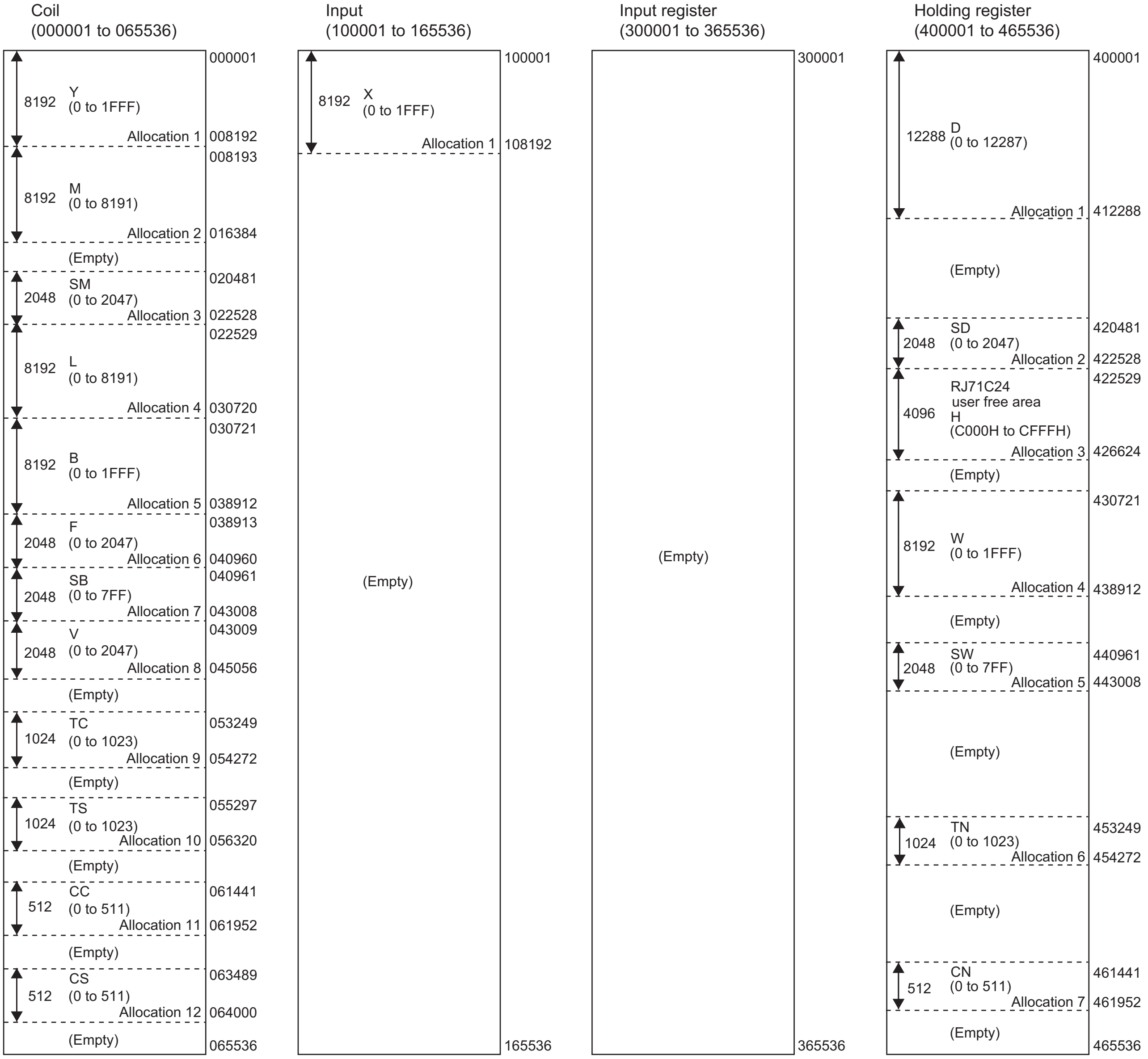
For details, refer to the following manual.

📖MELSEC iQ-R CPU Module User's Manual (Application)

Default assignment parameters

The MODBUS device assignment parameters include default assignment parameters as initial values.

The following diagram shows the MODBUS device assignment using the default assignment parameters.


Coil
(000001 to 065536)
000001
8192 Y (0 to 1FFF) Allocation 1
008192
008193
8192 M (0 to 8191) Allocation 2
016384
(Empty)
020481
2048 SM (0 to 2047) Allocation 3
022528
022529
8192 L (0 to 8191) Allocation 4
030720
030721
8192 B (0 to 1FFF) Allocation 5
038912
038913
2048 F (0 to 2047) Allocation 6
040960
040961
2048 SB (0 to 7FF) Allocation 7
043008
043009
2048 V (0 to 2047) Allocation 8
045056
(Empty)
053249
1024 TC (0 to 1023) Allocation 9
054272
(Empty)
055297
1024 TS (0 to 1023) Allocation 10
056320
(Empty)
061441
512 CC (0 to 511) Allocation 11
061952
(Empty)
063489
512 CS (0 to 511) Allocation 12
064000
(Empty)
065536
Input
(100001 to 165536)
100001
8192 X (0 to 1FFF) Allocation 1
108192
(Empty)
165536
Input register
(300001 to 365536)
300001
(Empty)
365536
Holding register
(400001 to 465536)
400001
12288 D (0 to 12287) Allocation 1
412288
(Empty)
420481
2048 SD (0 to 2047) Allocation 2
422528
422529
4096 RJ71C24 user free area H (C000H to CFFFH) Allocation 3
426624
(Empty)
430721
8192 W (0 to 1FFF) Allocation 4
438912
(Empty)
440961
2048 SW (0 to 7FF) Allocation 5
443008
(Empty)
453249
1024 TN (0 to 1023) Allocation 6
454272
(Empty)
461441
512 CN (0 to 511) Allocation 7
461952
(Empty)
465536

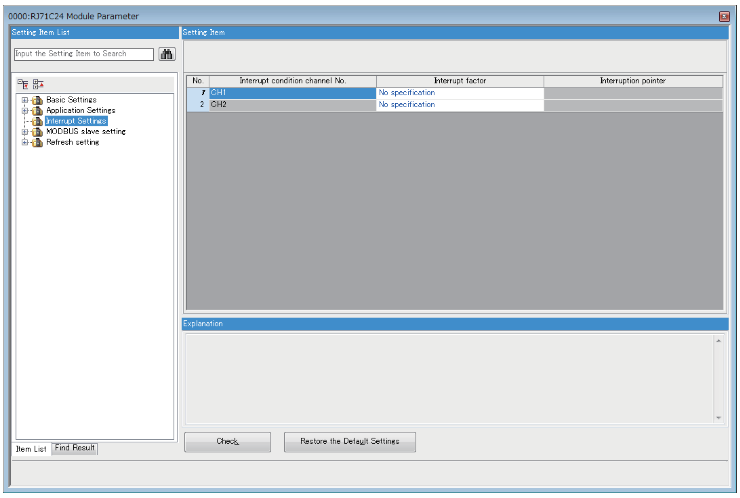
Point

- In the MODBUS device assignment parameters, the 'Start MODBUS device number' is the 'Last five digits of the setting target MODBUS device number - 1.'
- If a CC-Link IE Field Network remote head module is set as an access target, default values set for the MODBUS device assignment parameters cannot be used.

Interrupt settings

Configure the settings to receive data with an interrupt program.

Window



No.	Interrupt condition channel No.	Interrupt factor	Interruption pointer
1	CH1	No specification	
2	CH2	No specification	

Setting data

Item	Description	Setting range
Interrupt factor	Sets the condition for interruption occurrence.	• No specification • BUFRCVS instruction data reception (Default: No specification)
Interrupt pointer	Sets the interrupt pointer to be used by the program.	📖MELSEC iQ-R CPU Module User's Manual (Application)

Refresh setting

With this refresh settings, the read and write processing by a program is not necessary.
Set a device/module label of the CPU module which transfers data alternately with the buffer memory on C24.
The device/module label to be transmitted is selected from the following from "Target".

- Module label
- Refresh Data Register (RD)
- Device

Module label

The contents are transferred alternatively between the module label corresponding to each buffer memory and the buffer memory. Set "Valid" for the item to be transferred using module labels and "Invalid" for the item not to be transferred. By enabling "Send data_send area", the other transfer destinations will be set to "Valid".

Refresh data register (RD)

The contents are transferred alternatively between the refresh data register (RD) of a CPU module and the buffer memory. By setting the refresh data register (RD) to be transferred in "Top Device Name", the other transfer destinations will be automatically set.

Device

The contents are transferred alternatively between the designated device of a CPU module and the buffer memory. The devices that can be designated are X, Y, M, L, B, T, C, ST, D, W, R, ZR, and RD. When bit devices, X, Y, M, L, or B are used a number that can be divided by 16 points (for example, X10, Y120, M16, etc.) should be set. The buffer memory data is stored in the equivalent of 16 points from the designated device number.
(For example, when X10 is set, data is stored from X10 to X1F.)

Setting item

Item		Description	Reference
Transfer to intelligent function module	Send data_send area	A refresh setting for the send data storage area.	☞ Page 80 Transmitting Data to Target Device ☞ Page 99 Transmitting Data to Target Device
Transfer to CPU	LED lighting status, communication error status	A refresh setting for confirmation of LED lighting status and communication error status	☞ Page 483 Checking error status
	Transmission control status	A refresh setting for confirmation of transmission control status	☞ Page 489 Current operation status reading method
	Control signal status	A refresh setting for confirmation of control signal status	☞ Page 487 RS-232 control signal status reading method
	Communication result	A refresh setting for confirmation of communication result	☞ Page 488 Data communication status (transmission sequence status) reading method ☞ Page 509 ERR LED turns ON ☞ Page 310 Data Reception
	For receive data	A refresh setting for the receive data storage area	☞ Page 57 Receiving Data from Target Device ☞ Page 90 Receiving Data from Target Device
	Predefined protocol execution status	A refresh setting for confirmation of predefined protocol execution states	☞ Page 146 Protocol Execution Log Storage Function (for Predefined Protocol only) ☞ Page 509 ERR LED turns ON
	Sending user frame	A refresh setting for confirmation of user frame being transmitted	☞ Page 347 Settings of send user frames
	For user frame confirmation	A refresh setting for confirmation of user frame	—
	For modem function confirmation	A refresh setting for confirmation of modem function	☞ Page 193 Details of the buffer memory (for modern function)
	For remote password function	A refresh setting for the remote password function	☞ Page 197 Details of buffer memory (for the remote password function)
	Programmable controller CPU monitoring function	A refresh setting for the programmable controller CPU monitoring function	☞ Page 151 USING THE PROGRAMMABLE CONTROLLER CPU MONITORING FUNCTION
	For flash ROM (user frame) access	A refresh setting for the flash ROM (user frame) access	—
	Transmission setting, for mode switching confirmation	A refresh setting for confirmation of the transmission setting and the mode switching	☞ Page 409 SWITCHING THE MODE AFTER STARTING
Refresh group	Refresh group	A setting for the refresh group • At the Execution Time of END Instruction • At the Execution Time of Specified Program	—
	Program group No.	Designate the number which will be enabled when selecting 'At the Execution Time of Specified Program' for Refresh group.	
Refresh Timing (I/O)	Refresh timing	Sets the timing to transmit the I/O device data. • It follows the refresh timing (buffer memory).	

Refresh processing time

A refresh processing time (µs) is a constituent of the scan time of CPU module. For details on the scan time, refer to the following manual.

📖MELSEC iQ-R CPU Module User's Manual (Application)

The refresh processing time (µs) , which is taken for the refresh, is given by:

- Refresh processing time (µs) = Refresh read time (time for transferring refresh data to the CPU module) + Refresh write time (time for transferring refresh data to the Intelligent function module)

The refresh read time and refresh write time vary depending on the settings of "Target".

■When "Target" is "Module Label" or "Refresh Data Register (RD)"

The following table shows the refresh read time and refresh write time with an R□CPU used.

Classification	Refresh processing time
Refresh read time	80.80 µs
Refresh write time	20.48 µs

■When "Target" is "Device"

Calculate the refresh read time and refresh write time according to the number of items and the number of their transfer data (in units of word) that are set to be refreshed. For the calculation method, refer to the following manual.

📖MELSEC iQ-R CPU Module User's Manual (Application)

In addition, substitute the following values in the calculation formula to calculate the refresh processing time.

Item	Description
Number of refresh settings	Number of items that are set to be refreshed
Refresh time (A) of each of the 1st to nth set items	0.05 µs per one word[*1]
Refresh time (B) of each of the 1st to nth set items	0.01 µs per one word[*1]

[*1] This value is the time with an R□CPU used.

Ex.

When all the 48 items (1068 words in total) are set in the refresh read settings

$48 \times 0.98 + 0.05 \times 1068 + 11.6 = 112.04\ \mu s$

The refresh read time, therefore, is 112.04 µs.

Ex.

When all the 2 items (1022 words in total) are set in the refresh write settings

$2 \times 0.58 + 0.01 \times 1022 + 9.10 = 20.48\ \mu s$

The refresh write time, therefore, is 20.48 µs.

28.3 Module Extended Parameter

Set a module extended parameter. The module extended parameter includes user frame contents, modem initialization data, modem connection data, and simple CPU communication setting which are selected from the tree on the following screen.

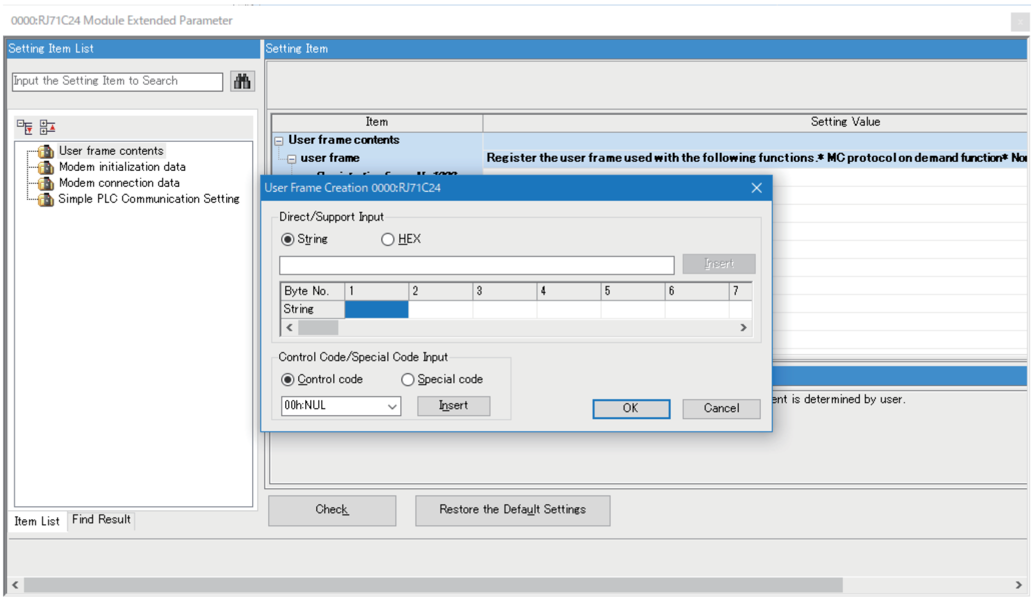
Navigation window ⇨ [Parameters] ⇨ [Module Information] ⇨ Module Name ⇨ [Module Extended Parameter]

User frame contents

Register the user frame which is used for the on-demand function using the MC protocol and the data transmit/receive function using nonprocedural protocol.

In the user frame contents, the setting value for each frame of the registration frame No. 1000 to 1199 is set.

Window


0000:RJ71C24 Module Extended Parameter
Setting Item List
Input the Setting Item to Search
User frame contents
Modem initialization data
Modem connection data
Simple PLC Communication Setting
Item List
Find Result
Setting Item
Item
Setting Value
User frame contents
user frame
Register the user frame used with the following functions.* MC protocol on demand function* No
User Frame Creation 0000:RJ71C24
Direct/Support Input
● String
○ HEX
Insert
Byte No.
String
Control Code/Special Code Input
● Control code
○ Special code
00h:NUL
Insert
OK
Cancel
ent is determined by user.
Check
Restore the Default Settings

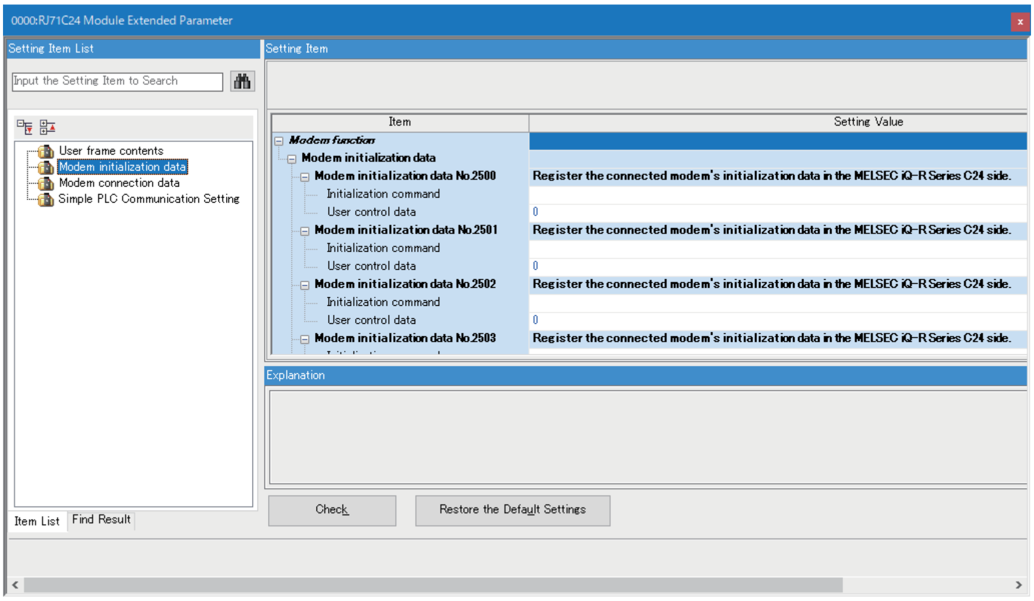
Setting data

Item	Description	Setting range
User frame	Sets the setting value for a user frame.	80 (maximum number of characters that can be entered)

Modem initialization data

In the modem initialization data, the data for initialization of the data number from 2500 to 2529 is set.

Window



0000:RJ71C24 Module Extended Parameter
Setting Item List
Input the Setting Item to Search
User frame contents
Modem initialization data
Modem connection data
Simple PLC Communication Setting
Item List
Find Result
Setting Item
Explanation
Check
Restore the Default Settings

Item	Setting Value
Modem function	
Modem initialization data	
Modem initialization data No.2500	Register the connected modem's initialization data in the MELSEC iQ-R Series C24 side.
Initialization command	
User control data	0
Modem initialization data No.2501	Register the connected modem's initialization data in the MELSEC iQ-R Series C24 side.
Initialization command	
User control data	0
Modem initialization data No.2502	Register the connected modem's initialization data in the MELSEC iQ-R Series C24 side.
Initialization command	
User control data	0
Modem initialization data No.2503	Register the connected modem's initialization data in the MELSEC iQ-R Series C24 side.

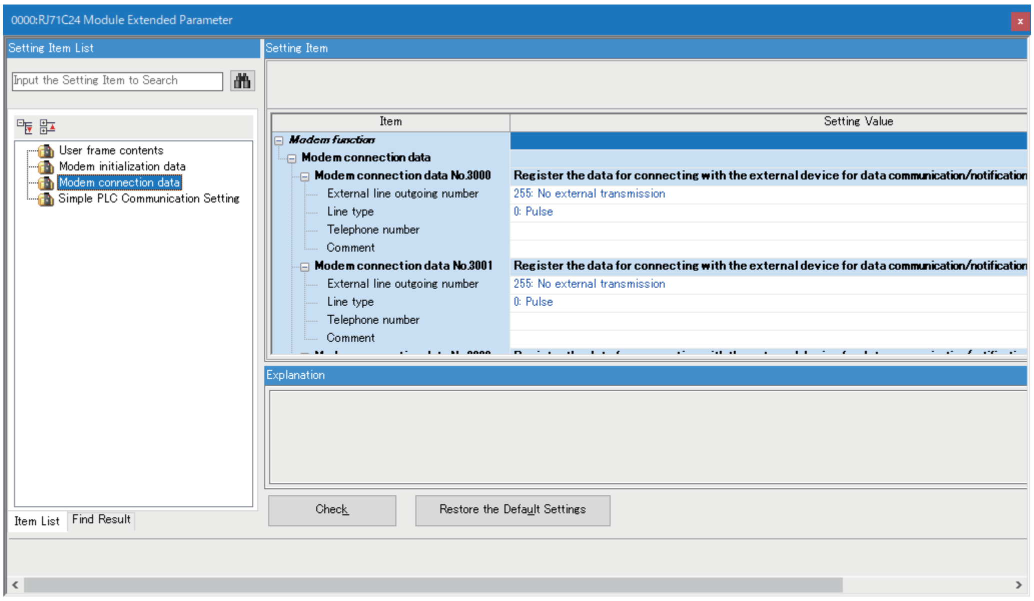
Setting data

Item	Description	Setting range
Initialization command	Sets a data code for initialization command.	78 (maximum number of characters that can be entered)
User control data	Sets an arbitrary data for a user to control the data to be registered.	0 to FFFF

Modem connection data

In the modem connection data, the data for connection of data number from 3000 to 3029 is set.

Window


0000:RJ71C24 Module Extended Parameter
Setting Item List
Input the Setting Item to Search
User frame contents
Modem initialization data
Modem connection data
Simple PLC Communication Setting
Item List
Find Result
Setting Item
Explanation
Check
Restore the Default Settings

Item	Setting Value
Modem function	
Modem connection data	
Modem connection data No.3000	Register the data for connecting with the external device for data communication/notification
External line outgoing number	255: No external transmission
Line type	0: Pulse
Telephone number	
Comment	
Modem connection data No.3001	Register the data for connecting with the external device for data communication/notification
External line outgoing number	255: No external transmission
Line type	0: Pulse
Telephone number	
Comment	

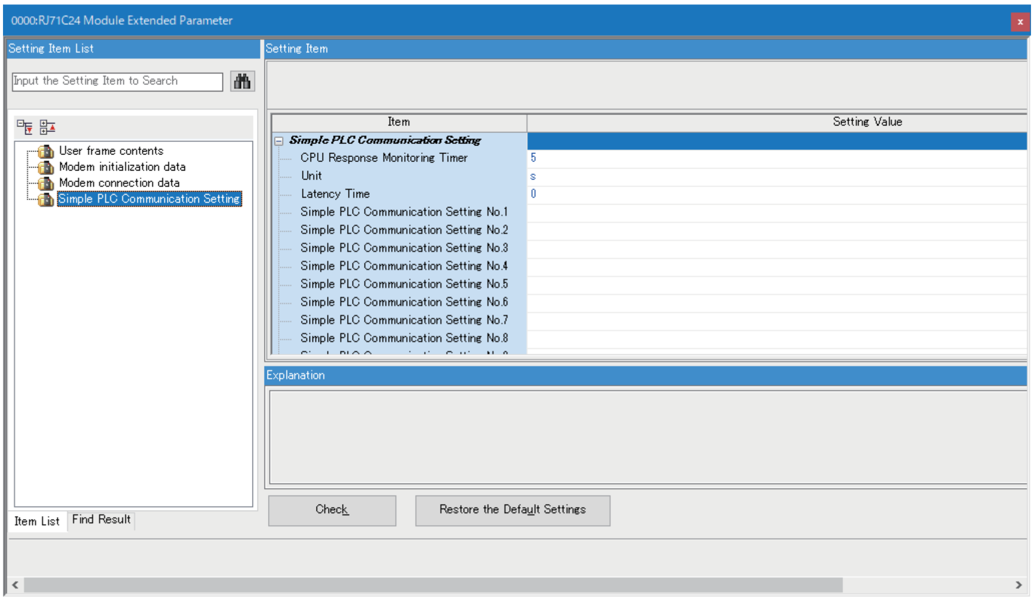
Setting data

Item	Description	Setting range
External line outgoing number	Sets an external line outgoing number of the C24 side for line connection to the side of target device for data transmission.	• 255: No external transmission • 0: 0 • 1: 1 • 2: 2 • 3: 3 • 4: 4 • 5: 5 • 6: 6 • 7: 7 • 8: 8 • 9: 9 • 10: * • 11: #
Line type	Sets a line type between the target device for data communication.	• 0: Pulse • 1: Tone • 2: ISDN
Telephone number	Sets a telephone number between the target device for data communication.	62 (maximum number of characters that can be entered)
Comment	Sets an arbitrary comment.	254 (maximum number of characters that can be entered)

Simple CPU communication setting

Set the following items when using the simple CPU communication function for a C24.

Window



0000:RJ71C24 Module Extended Parameter

Setting Item List

Input the Setting Item to Search

- User frame contents
- Modem initialization data
- Modem connection data
- Simple PLC Communication Setting

Setting Item

Item	Setting Value
Simple PLC Communication Setting	
CPU Response Monitoring Timer	5
Unit	s
Latency Time	0
Simple PLC Communication Setting No.1	
Simple PLC Communication Setting No.2	
Simple PLC Communication Setting No.3	
Simple PLC Communication Setting No.4	
Simple PLC Communication Setting No.5	
Simple PLC Communication Setting No.6	
Simple PLC Communication Setting No.7	
Simple PLC Communication Setting No.8	

Explanation

Setting data

Item	Description	Setting range
CPU Response Monitoring Timer	Set the time for monitoring the response from the CPU module. If there is no response from the CPU module within a set time, the response waiting state is released.	Unit: s • 1 to 16383 Unit: ms • 100 to 16383000
Unit	Select the unit of a CPU response monitoring timer.	• s • ms
Latency Time	Set a time to start communication after the CPU module is started. (Unit: s)	☞ Page 479 Simple CPU communication setting
Simple PLC Communication Setting No.□[*1]	Configure the settings to perform simple CPU communication. Set a communication pattern, communication destination, and devices to be sent and received.	

*1 □: 1 to 512

Point

"Simple PLC Communication Setting" can be set up to 512 with an engineering tool; however, when setting as a communication destination, up to 1 (RS-232) and 32 (RS-422/485) can be set for each channel.

Restriction

In GX Works3, if a project satisfies all the following conditions, the values in the simple CPU communication setting may become incorrect when opening the setting in version 1.072A or earlier.

- A serial communication module was placed in a project in version 1.072A or earlier.
- The above project was opened in version 1.075D or later, and the simple CPU communication setting was configured.

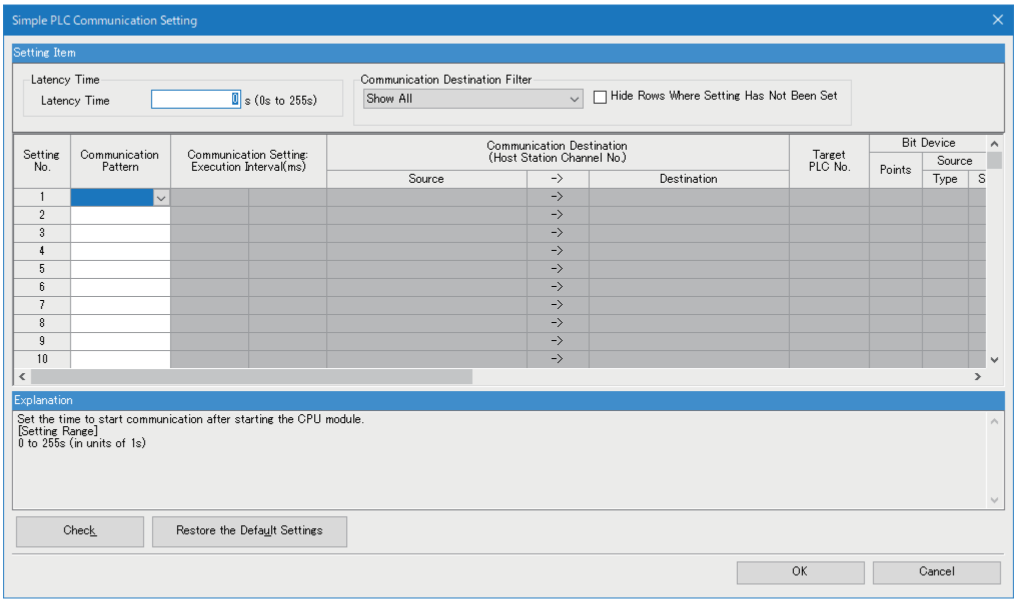
In this case, upgrade the version to 1.075D or later, then open the simple CPU communication setting.

Simple CPU communication setting

Set the following items to perform simple CPU communication.

A setting screen appears by double-clicking any one of "Simple PLC Communication Setting No.1" to "Simple PLC Communication Setting No.512" in the "Simple PLC Communication Setting" screen.

Displayed items



Simple PLC Communication Setting

Setting Item

Latency Time

Latency Time [0] s (0s to 255s)

Communication Destination Filter

Show All

□ Hide Rows Where Setting Has Not Been Set

Setting No.	Communication Pattern	Communication Setting: Execution Interval(ms)		Communication Destination (Host Station Channel No.)			Target PLC No.	Bit Device		
				Source	->	Destination		Points	Source	
									Type	S
1					->					
2					->					
3					->					
4					->					
5					->					
6					->					
7					->					
8					->					
9					->					
10					->					

Explanation

Set the time to start communication after starting the CPU module.
[Setting Range]
0 to 255s (in units of 1s)

Setting data

Item		Description	Setting range
Latency Time		Set the time required to start communications after completion of the CPU module startup. This setting is common to all the setting numbers and cannot be changed for each setting number. (Unit: s)	0 to 255 (Default: 0)
Setting No.		Setting No. 1 to 512 are available. Any setting number can be set. (A number that is not a sequential number can also be set.)	—
Communication Pattern		Set whether to read or write data.	• Read • Write (Default: Blank)
Communication Setting: Execution Interval (ms)	Communication Setting	Set the data communication timing.	• Fixed Interval • On Request (Default: Fixed Interval)
	Execution Interval	Set the execution interval of communications when "Fixed Interval" is set for the communication setting. (Unit: ms)	10 to 65535 (Default: 100)
Communication Destination (Host Station Channel No.)	Source	Set the transfer source. When selecting "Write" for the communication pattern, the host station is displayed.	☞ Page 481 Communication destination setting
	Destination	Set the transfer destination. When selecting "Read" for the communication pattern, the host station is displayed.	
Target PLC No.		Specify the PLC number of a communication destination. It will be the connected station when "Not Specified" is selected.	Not Specified (fixed)

Item		Description	Setting range
Bit Device	Points	Set the type, start number, and end number of a bit device for "Source," and the type and start number of a bit device for "Destination." The number of points is automatically displayed after entering these items.	Differs depending on the communication destination. (☞ Page 122 Device range)
	Type		
	Start		
	End		
Word Device	Points	Set the type, start number, and end number of a word device for "Source," and the type and start number of a word device for "Destination." The number of points is automatically displayed after entering these items.	
	Type		
	Start		
	End		
Communication Time-out Period (ms)		Set the time to start a retry after an error response is received from a communication destination, to regard the state as an error, or to wait for a response from a communication destination. Set it as follows: (Communication time-out period) ≥ (Execution interval) For details on the communication time-out period, refer to the following section. • When selecting "Fixed Interval" for the communication setting: ☞ Page 124 Operation when an error occurs • When selecting "On Request" for the communication setting: ☞ Page 129 Operation when an error occurs (Unit: ms)	10 to 65535 (Default: 5000)
Communication Retry Count		Set the number of times to repeat a waiting operation in the communication time-out period if there is an error response or no response from a communication destination. For details on the communication retry count, refer to the following section. • When selecting "Fixed Interval" for the communication setting: ☞ Page 124 Operation when an error occurs • When selecting "On Request" for the communication setting: ☞ Page 129 Operation when an error occurs	0 to 255 (Default: 3)
Monitoring Time At Error (s)		Set the communication time after a communication error occurs. For details on the monitoring time at error, refer to the following section. ☞ Page 125 Operation in an abnormal state (Unit: s)	1 to 300 (Default: 30)
Comment		Enter a comment for each setting number.	Up to 32 characters (Default: Blank)

Point

- The actual execution interval is affected by a communication destination and communication speed; therefore, it may become longer than the set execution interval.
- When setting "Latency Time," a data communication is performed after a latency time elapses.

■Latency time

A timing to start communication can be delayed by setting a latency time for the following cases:

- To avoid an error due to the concentrated communication and overlapped timing to start communication
- To start communication after the communication destination become available to communicate

During a latency time, '1' (preparing) is stored in 'Communication status' for each setting number.

Point

During a latency time, a communication at an execution interval set to "Fixed Interval" is ignored.

Communication destination setting

The following table lists the communication destination settings specified for simple CPU communication.

■Host station channel number, station number

Item	Description	Setting range
Host Station Channel No.[*1]	Set the channel number of the host station.	• CH1 (RS-232) • CH2 (RS-232) • CH1 (RS-422/485) • CH2 (RS-422/485)
Station No.	Set the station number when a communication destination is a MODBUS-compatible device. • 0: Broadcast (A request message is sent to all slave stations.) • 1 to 247: Station number of a slave to be communicated Cannot be set when a communication destination is a CPU module.	0 to 247

[*1] Displayed in the name of a communication destination.

Point

When setting "Station No." to '0' (broadcast), configure the settings as follows:

- "Communication Pattern": "Write"
- "Bit Device" and "Word Device": Cannot be set together. Select either one for each setting.
- "Communication Time-out Period": Set the value as the broadcast delay.

Broadcast is sent to all slave devices; therefore, consider each processing time of all slave devices and set the sufficient time for communication time-out period (broadcast delay).

If there is a slave device the communication time-out period of which is set insufficiently, an error may occur for another request to the slave device.

■Device type

Configure the setting of a communication destination for devices (transfer source and transfer destination).

Item	Description	Setting range
Device Type	Specify the device type of a communication destination.	• MELSEC-A/AnS (CPU COM) • MODBUS (RTU)-compatible device • MODBUS (ASCII)-compatible device
Option 1	Cannot be set.	—
Option 2	Cannot be set.	—

The following shows compatible devices for each device type.

Device type	Communication destination
MELSEC-A/AnS (CPU COM)	A0J2HCPU, A0J2HCPU-DC24V, A0J2HCPUP21, A0J2HCPUR21, A1NCPU, A1NCPUP21, A1NCPUP21-S3, A1NCPUR21, A1SCPUC24-R2, A1SHCPU, A1SJHCPU, A2ACPU, A2ACPU-S1, A2ACPUP21, A2ACPUP21-S1, A2ACPUP21-S3, A2ACPUP21-S4, A2ACPUR21, A2ACPUR21-S1, A2CCPU, A2CCPUC24, A2CCPUC24-PRF, A2CCPUP21, A2CCPUR21, A2CJCPU-S3, A2NCPU, A2NCPU-S1, A2NCPUP21, A2NCPUP21-S1, A2NCPUP21-S3, A2NCPUP21-S4, A2NCPUR21, A2NCPUR21-S1, A2SHCPU, A2UCPU, A2UCPU-S1, A2USCPU, A2USHCPU-S1, A3ACPU, A3ACPUP21, A3ACPUP21-S3, A3ACPUR21, A3NCPU, A3NCPUP21, A3NCPUP21-S3, A3NCPUR21, A3UCPU, A4UCPU
MODBUS (RTU)-compatible device	MODBUS slave (RTU)-compatible device
MODBUS (ASCII)-compatible device	MODBUS slave (ASCII)-compatible device

29 TROUBLESHOOTING

This chapter explains the errors which may occur when using C24 and the corrective actions.

29.1 Checking Module Status

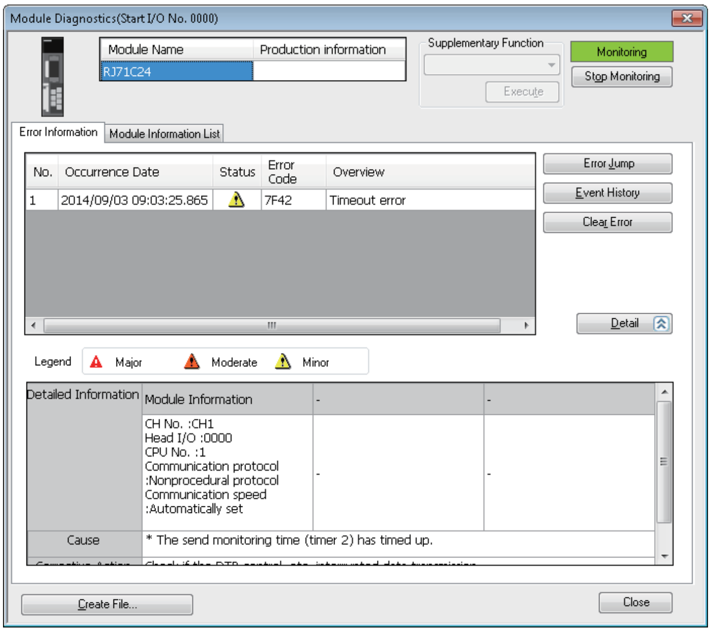
The following functions can be used in the "Module Diagnostics" screen of C24.

Function	Description
Error Information	Contents of error currently occurs are displayed. Errors detected on C24 and history of operation performed to the module can be checked by clicking the [Event History] button.
Module Information List	Various status information of C24 is displayed.

Error information

Check the content of error currently occurs and the corrective actions.

Window



Module Name	Production information
RJ71C24	

No.	Occurrence Date	Status	Error Code	Overview
1	2014/09/03 09:03:25.865	⚠	7F42	Timeout error

Detailed Information	Module Information	-	-
	CH No. :CH1 Head I/O :0000 CPU No. :1 Communication protocol :Nonprocedural protocol Communication speed :Automatically set	-	-
Cause	* The send monitoring time (timer 2) has timed up.		

Displayed items

Item	Description
Detailed Information	Up to three detail information of each error is displayed.
Cause	Detail reasons for error is displayed.
Corrective Action	Corrective actions for an error is displayed.

Module information list

Check each status information of C24 by switching to the [Module Information List] tab.

Checking error status

The transmission status and the communication error status for C24 can be checked with Engineering tool. (Contents of the 'CH□ side LED lighting status and communication error status' (Un\G513/514) are displayed.)

When a communication error occurs, refer to the following section and take corrective action.

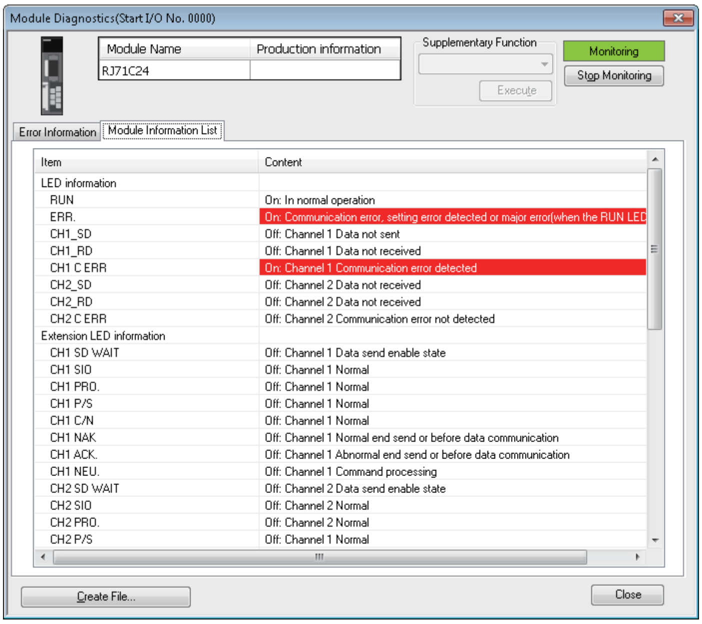
☞ Page 500 Troubleshooting by Symptom

Point

Each communications error status of SIO, PRO., P/S, and C/N turns ON when an error occurs.

To turn them OFF by clearing the communication error status, turn 'Error initialization request' (YE) ON.

Window



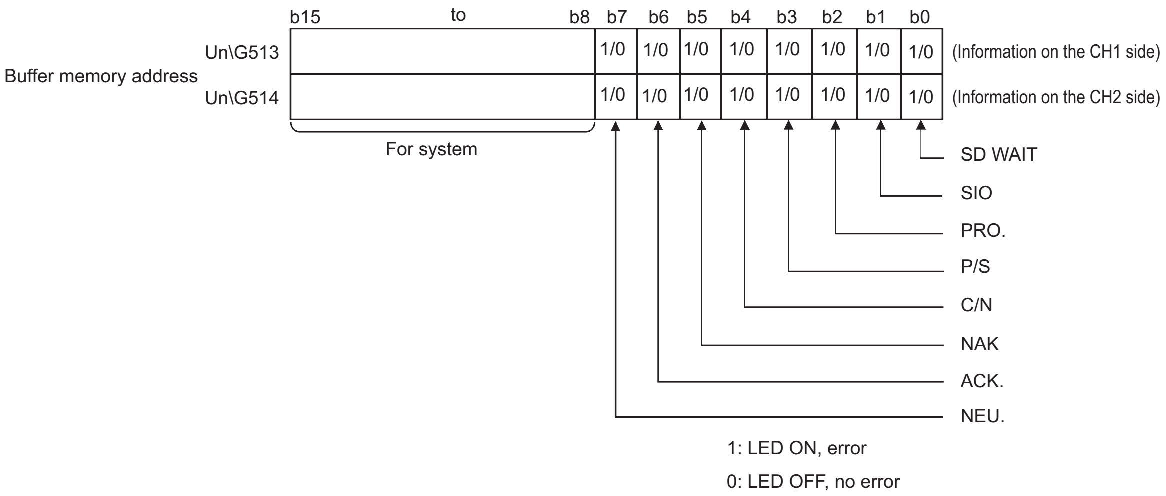
Module Diagnostics(Start I/O No. 0000)

Module Name	Production information
RJ71C24	

Item	Content
LED information	
RUN	On: In normal operation
ERR.	On: Communication error, setting error detected or major error(when the RUN LED
CH1_SD	Off: Channel 1 Data not sent
CH1_RD	Off: Channel 1 Data not received
CH1 C ERR	On: Channel 1 Communication error detected
CH2_SD	Off: Channel 2 Data not received
CH2_RD	Off: Channel 2 Data not received
CH2 C ERR	Off: Channel 2 Communication error not detected
Extension LED information	
CH1 SD WAIT	Off: Channel 1 Data send enable state
CH1 SIO	Off: Channel 1 Normal
CH1 PRO.	Off: Channel 1 Normal
CH1 P/S	Off: Channel 1 Normal
CH1 C/N	Off: Channel 1 Normal
CH1 NAK	Off: Channel 1 Normal end send or before data communication
CH1 ACK.	Off: Channel 1 Abnormal end send or before data communication
CH1 NEU.	Off: Channel 1 Command processing
CH2 SD WAIT	Off: Channel 2 Data send enable state
CH2 SIO	Off: Channel 2 Normal
CH2 PRO.	Off: Channel 2 Normal
CH2 P/S	Off: Channel 1 Normal

Displayed items

■Buffer memory area for the display contents



■SD WAIT

SD WAIT indicates "Transmission wait status".

Value	Status	Description
0001 (ON)	Waiting for data transmission	C24 got into a state in which data cannot be sent to a target device (waiting for data transmission) by the transmission control of C24. • The transmission cannot be started. • A transmission terminate request (receiving DC3/DR(DSR) signal OFF) is received from the target device during data transmission.
0000 (OFF)	Data transmission available	The transmission became available, and the data transmission is started/resumed.

■SIO

SIO indicates "SIO error status".

Value	Status	Description
0001 (ON)	Overrun or framing error occurred	Overrun or framing error occurred. Keeps the ON-status even after the communication returns to normal. Initialize the error information as necessary. ☞ Page 519 Error Code List
	OS area is full.	No free space is available in the OS area, and the received data was discarded. Keeps the ON-status even after the communication returns to normal. Initialize the error information as necessary. ☞ Page 519 Error Code List
0000 (OFF)	Normal	—

■PRO.

PRO. indicates "Character error status".

Value	Status	Description
0001(ON)	Error occurred	Character error occurred. Keeps the ON-status even after the communication returns to normal. Initialize the error information as necessary. ☞ Page 519 Error Code List
0000(OFF)	Normal	—

■P/S

P/S indicates "Status of parity error or sum check error".

Value	Status	Description
0001(ON)	Error occurred	Parity error or sum check error occurred. Keeps the ON-status even after the communication returns to normal. Initialize the error information as necessary. ☞ Page 519 Error Code List
0000(OFF)	Normal	—

■C/N

C/N indicates 'Access status with the CPU module'.

Value	Status	Description
0001(ON)	Error occurred	• The target device issued a data write request to the CPU module when online change is set to 'Disable' in the parameter settings with Engineering tool. For functions that cannot be used when online change is set to 'Disable', refer to the explanation section in each command list in the following manual. (📖 MELSEC Communication Protocol Reference Manual) • An error occurred during the access between C24 and the CPU module. Keeps the ON-status even after the communication returns to normal. Initialize the error information as necessary. ☞ Page 519 Error Code List
0000(OFF)	Normal	—

■NAK

NAK indicates "Abnormal completion status".

Value	Status	Description
0001(ON)	Abnormal completion transmission	C24 that could not receive data normally sent a response message (NAK message) to the target device to notice the abnormal completion.
0000(OFF)	Normal completion transmission or before data communication	• C24 that could receive data normally sent a response message (ACK message) to the target device to notice the normal completion. • Before data is communicated.

■ACK.

ACK. indicates "Normal completion status".

Value	Status	Description
0001(ON)	Normal completion transmission	C24 that could receive data normally sent a response message (ACK message) to the target device to notice the normal completion.
0000(OFF)	Abnormal completion transmission or before data communication	• C24 that could not receive data normally sent a response message (NAK message) to the target device to notice the abnormal completion. • Before data is communicated.

■NEU.

NEU. indicates "Command message to be received/command message in processing".

NEU. is valid only when MC protocol (format 1 to 5), MODBUS slave (RTU), or MODBUS slave (ASCII) is set in the communication protocol setting.

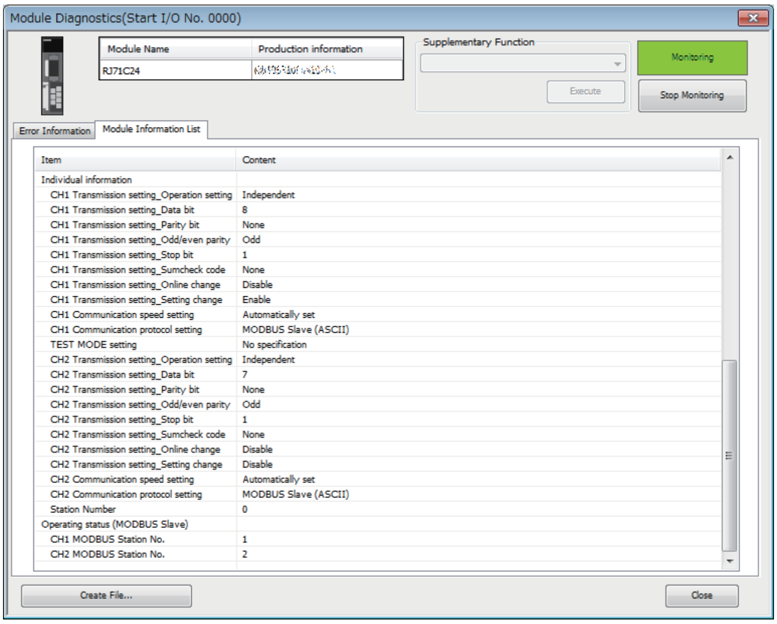
This is in the OFF-status when an item other than MC protocol (format 1 to 5), MODBUS slave (RTU), or MODBUS slave (ASCII) is set.

Value	Status	Description
0001(ON)	Neutral	Waiting for command message to be received.
0000(OFF)	Command reception	Command message is in process.

Checking transmission setting

Details of transmission settings can be monitored.

Window



Module Diagnostics(Start I/O No. 0000)

Module Name	Production information
RJ71C24	[illegible]

Item	Content
Individual information	
CH1 Transmission setting_Operation setting	Independent
CH1 Transmission setting_Data bit	8
CH1 Transmission setting_Parity bit	None
CH1 Transmission setting_Odd/even parity	Odd
CH1 Transmission setting_Stop bit	1
CH1 Transmission setting_Sumcheck code	None
CH1 Transmission setting_Online change	Disable
CH1 Transmission setting_Setting change	Enable
CH1 Communication speed setting	Automatically set
CH1 Communication protocol setting	MODBUS Slave (ASCII)
TEST MODE setting	No specification
CH2 Transmission setting_Operation setting	Independent
CH2 Transmission setting_Data bit	7
CH2 Transmission setting_Parity bit	None
CH2 Transmission setting_Odd/even parity	Odd
CH2 Transmission setting_Stop bit	1
CH2 Transmission setting_Sumcheck code	None
CH2 Transmission setting_Online change	Disable
CH2 Transmission setting_Setting change	Disable
CH2 Communication speed setting	Automatically set
CH2 Communication protocol setting	MODBUS Slave (ASCII)
Station Number	0
Operating status (MODBUS Slave)	
CH1 MODBUS Station No.	1
CH2 MODBUS Station No.	2

Displayed items

Item	Description
Operation setting	Independent, Interlink
Data bit	7, 8
Parity bit	None, Yes
Odd/even parity	Odd, Even
Stop bit	1, 2
Sum check code	None, Yes
Online change	Disable, Enable
Setting change	Disable, Enable
Test mode setting	No specification, Hardware test, Self-loopback test
Communication speed setting	1200, 2400, 4800, 9600 ,14400 ,19200 ,28800 ,38400, 57600, 115200, 230400 (bps)
Communication protocol	MELSOFT Connection, MC protocol (format 1 to 5), Predefined protocol, Nonprocedural protocol, Bidirectional protocol, For interlink operation setting, MODBUS Slave (RTU), MODBUS Slave (ASCII), Simple CPU Communication The communication protocol is valid when the test mode setting is set to "No specification".
Station number setting	0 to 31
MODBUS station number	1 to 247 The MODBUS station number is valid when the communication protocol is set to "MODBUS Slave." If it is invalid, '0' is displayed.

Intelligent function module monitor

Check the status information of C24 on the "Intelligent Function Module Monitor" screen.

For details of the "Intelligent Function Module Monitor" screen, refer to the following manual.

GX Works3 Operating Manual

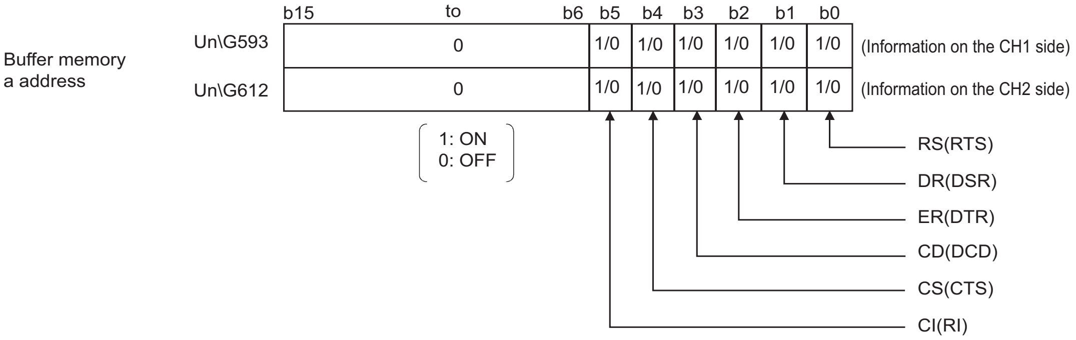
RS-232 control signal status reading method

The following shows how to read the control signal status stored in the buffer memory during communication using RS-232 interface.

This is a read operation to check the signal ON/OFF status of the RS-232 interface when a transmission problem occurs.

■RS-232 control signal status (Un\G596/612)

The RS-232 control signal status is stored as follows:



- Remarks

For details on signals for RS-232, refer to the following section.

MELSEC iQ-R Serial Communication Module User's Manual(Startup)

Signals output from the C24 side (RS(RTS) and ER(DTR)) are controlled by the system (operating system) of C24. They cannot be directly controlled by the program.

A lag of 100 ms at maximum occurs in the signal status stored in the buffer memory above.

Data communication status (transmission sequence status) reading method

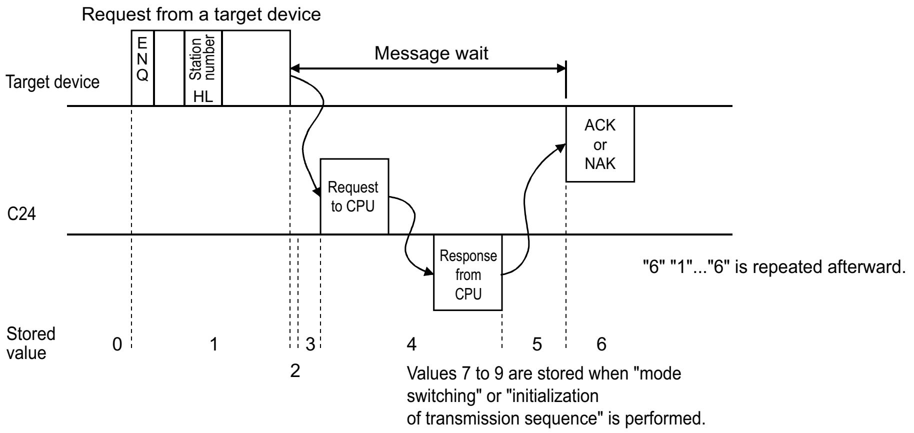
The following shows how to read the current data communication status using the MC protocol stored in the buffer memory. This read processing is for checking the data communication status using the MC protocol when a transmission problem occurs.

■Transmission sequence status (Un\G597/613)

The status of data communication using the MC protocol is stored as a numeric value as shown below.

		b15 to b6	
Buffer memory address	Un\G597	0 to 9	(Information on the CH1 side)
	Un\G613	0 to 9	(Information on the CH2 side)

The following shows the correspondence between the numerical values in the transmission sequence status storage area and the data communication status.



"6" "1"..."6" is repeated afterward.

Values 7 to 9 are stored when "mode switching" or "initialization of transmission sequence" is performed.

- Remarks

When the target interface is not set to use the MC protocol, '0' is stored in the 'Transmission sequence status' (Un\G597/613).

Current operation status reading method

The following shows how to read the current operation status of C24.

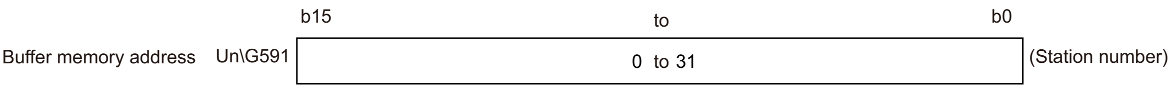
For details on the contents of each parameter setting, refer to the following manual.

☞ Page 454 PARAMETER SETTING

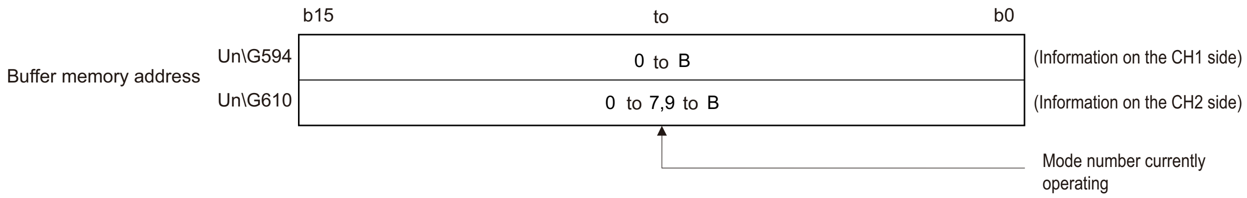
■Operation mode status storage area

The current operation status of C24 is stored in the following buffer memory areas.

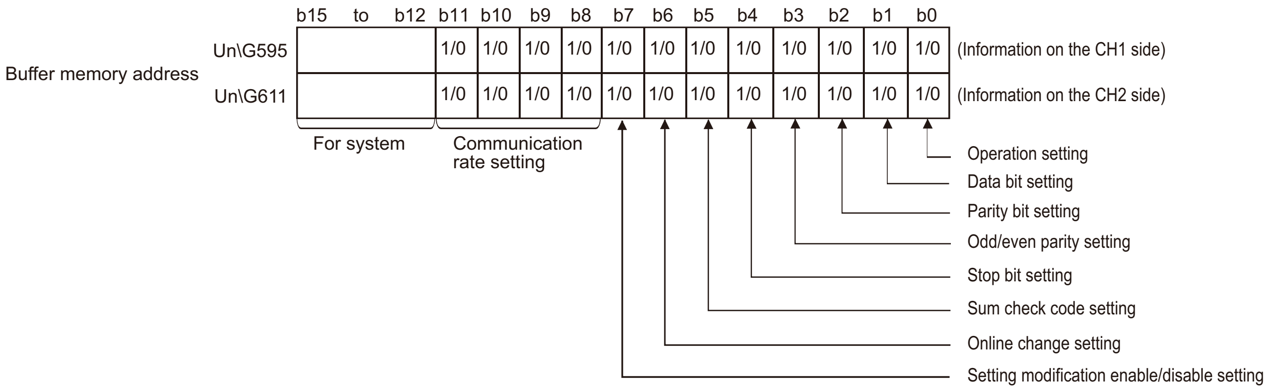
- Station number (instruction setting) (Un\G591)



- Communication protocol status (Un\G594/610)


b15
to
b0
Buffer memory address
Un\G594
Un\G610
0 to B
0 to 7,9 to B
(Information on the CH1 side)
(Information on the CH2 side)
Mode number currently operating

- Transmission setting status (Un\G595/611)

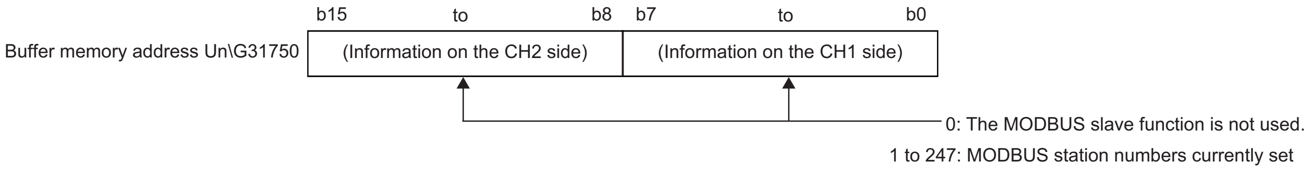


For the correspondence table for the communication speed setting, see the table below.

1: ON, 0: OFF

Communication speed setting	b11	b10	b9	b8
1200 bps	0	0	1	0
2400 bps	0	0	1	1
4800 bps	0	1	0	0
9600 bps	0	1	0	1
14400 bps	0	1	1	0
19200 bps	0	1	1	1
28800 bps	1	0	0	0
38400 bps	1	0	0	1
57600 bps	1	0	1	0
115200 bps	1	0	1	1
230400 bps	1	1	0	0

- MODBUS station number confirmation (Un\G31750)


b15
to
b8
b7
to
b0
Buffer memory address Un\G31750
(Information on the CH2 side)
(Information on the CH1 side)
0: The MODBUS slave function is not used.
1 to 247: MODBUS station numbers currently set

Check method for the communication status of the MODBUS slave function

The following explains the check method for the communication status of the MODBUS slave function.

When using the MODBUS slave function, the number of error occurrences during communication is stored in the buffer memory.

The communication status can be checked by reading the counter of the number of error occurrences by either of the following method:

- Checking in the "Intelligent Function Module Monitor" screen
- Accessing from a master device with any of the MODBUS standard functions (FC: 08H) (☞ Page 116 Function List)

Counter		Buffer memory		MODBUS standard function	
Type	**Description**	**CH1**	**CH2**	**Function name**	**Code**
Bus message count[*1]	Counts the number of messages detected on the line.	32512 (7F00H)	32576 (7F40H)	Return bus message count	08H (0BH)
Bus communication error count[*1]	Counts the number of the following error messages detected on the line. • CRC/LRC error message • Overrun/parity error • Short frame (less than 3 bytes) • Character overrun (256 bytes or more) Messages other than the above are counted by the bus message count.	32513 (7F01H)	32577 (7F41H)	Return bus communication error count	08H (0CH)
Exception error count	Counts the number of exception error occurrences. (Including broadcast messages)	32522 (7F0AH)	32586 (7F4AH)	Return bus exception error count	08H (0DH)
Slave message count	Counts the number of times that messages addressed to the host were processed. (Including when broadcast request messages were received)	32518 (7F06H)	32582 (7F46H)	Return slave message count	08H (0EH)
Slave no-response count	Counts the number of times that broadcast request messages were received.	32519 (7F07H)	32583 (7F47H)	Return slave no response count	08H (0FH)
Slave NAK count	Refers to the number of times that NAK responses were received in a master device. It is always '0' in RJ71C24s.	32520 (7F08H)	32584 (7F48H)	Return slave NAK count (0000H is always returned.)	08H (10H)
Slave busy count	Refers to the number of times that busy responses were received in a master device. It is always '0' in RJ71C24s.	32521 (7F09H)	32585 (7F49H)	Return slave busy count (0000H is always returned.)	08H (11H)
Character overrun count	Counts the number of times that the request message size exceeded the upper limit.	32514 (7F02H)	32578 (7F42H)	Return bus character overrun count	08H (12H)
				Return IOP overrun error count	08H (13H)
Message discard count	Counts the number of times that request messages were discarded when a request message was being processed by a slave or a request message to another station was received.	32515 (7F03H)	32579 (7F43H)	—	
Data discard count	Counts the number of times that improper data (such as a request message the data structure of which does not comply with the frame specifications) was discarded.	32516 (7F04H)	32580 (7F44H)		
Failed transmission count	Counts the number of times that transmission of response messages failed.	32517 (7F05H)	32581 (7F45H)		

[*1] The bus message count is in an exclusive relationship with the bus communication error count.

■Count range

0000H to FFFFH are counted.

The count is stopped if it has reached FFFFH. To continue, clear the counter.

■Methods for clearing a counter

A counter can be cleared by any of the following methods:

- Turning the power OFF and ON
- Resetting a CPU module
- Clearing the buffer memory to '0' by using a sequence program
- Receiving any of the following MODBUS standard functions:

MODBUS standard function		Counter that can be cleared
Clear counters and diagnostic register	08H (0AH)	• Bus message count • Bus communication error count • Exception error count • Slave message count • Slave no-response count • Character overrun count
Restart communications option	08H (01H)	
Clear overrun counter and flag	08H (14H)	Character overrun count

Error information clear

This section explains the factors causing the ERR LED to be turned ON and the error code initialization (clear) for C24.

Factors causing ERR LED to turn ON

When any of the following errors occurs, its error code is stored in the buffer memory corresponding to the interface (CH) where the error occurred, and the ERR LED turns ON.

The meanings of abbreviations in the table are as follows:

- MC: MC protocol
- Non: Nonprocedural protocol
- Bi: Bidirectional protocol
- Pd: Predefined protocol
- MD: MODBUS (slave function)
- S: Simple CPU communication

Error factor	Error code storage buffer memory			Applicable protocol					
	Address		Name	MC	Non	Bi	Pd	MD	S
	CH1	CH2							
Parameter setting error	Un\G515		Parameter setting error, mode switch error status	○	○	○	○	○	○
Mode switching error									
On-demand execution error	Un\G598	Un\G614	On-demand execution results	○	×	×	×	×	×
Data transmission error	Un\G599	Un\G615	Data transmission result	○	○	○	○	○	○
Data reception error	Un\G600	Un\G616	Data reception result	○	○	○	○	○	○
Modern function error	Un\G545		Modern function error code	○	○	○	×	×	×
MC protocol send error	Un\G602	Un\G618	MC protocol send error code	○	×	×	×	×	×
Monitoring device error	Un\G8709	Un\G8965	Programmable controller CPU monitoring function execution result	○	○	○	×	×	×
Predefined protocol function error	Un\G16450	Un\G16466	Predefined protocol function error code	×	×	×	○	×	×
MODBUS send/receive error	Un\G28674	Un\G28676	Exception code storage area	×	×	×	×	○	×

Factors causing ERR LED to flash

When any parameter error was detected, the ERR LED flashes, and the buffer memory address which is set out of the range of the parameter setting value is stored in the error address storage area (Un\G20224 to 20231).

Turning OFF the ERR LED and initializing the error codes

Turn 'Error initialization request' (YE) ON to turn the ERR LED OFF and initialize the error code.

- The buffer memory, which is a factor in turning the LED ON, is initialized.
- The protocol execution logs are not initialized.

Individual station test

When a problem occurs during data communication with a target device, perform an individual station test to check whether or not there is any problem in the operation of C24.

The individual station test consists of the following two tests:

- Hardware test
- Self-loopback test

Precautions

- To prevent any problems from occurring, set the CPU module in the STOP state during the individual station test.
- When either one of the interfaces needs to be tested, disregard the test result for the other interface.
- Cable removal and reconnection before and after the self-loopback test should be performed with the power to the station with C24 turned OFF.
- Before starting data communication with the target device after the individual station test, configure the respective setting and connect the cable by checking the specifications for the functions to be used.
- Do not execute the firmware update, when an individual station test is being performed.
- If an error occurs even though the settings have been configured correctly for C24 through an individual station test, please consult your local Mitsubishi representative, explaining a detailed description of the problem.

Hardware test

The hardware test is a test to check the memory of C24.

Perform the hardware test according to the procedure described below.

Operating procedure

1. Test mode setting (☞ Page 454 PARAMETER SETTING)

- Set the test mode setting to "Hardware test".
- Configure all of the transmission settings and communication speed setting for the two interfaces according to the transmission specifications for data communication with a target device.
- Set the CPU module to the STOP state, and write the parameters.

2. Performing hardware test

By restarting or resetting the CPU module, the test starts automatically in approximately one second. The RUN LED flashes during the test. (Set the CPU module in the STOP state.)

3. Checking results of hardware test

- When the result is normal

The RUN LED turns ON and the ERR LED turns OFF.

- When the result is abnormal

The RUN LED turns ON and the ERR LED turns ON.

4. Completing hardware test

- When the result is normal

Perform an Self-loopback test. (☞ Page 494 Self-loopback test)

Start data communication with the target device according to the following procedure.

· Configure the parameter settings with Engineering tool. (☞ Page 454 PARAMETER SETTING)

· Turn OFF the power for the station with C24, and connect the cable to communicate with the target device.

· Turn ON the power for the station with C24.

- When the result is abnormal

Check error contents on the items below on the "Intelligent Function Module Monitor" screen of Engineering tool.

Check the error descriptions on the screens by bit-decoding the monitored value (16-bit integer). Each bit corresponds to the following error description.

Buffer memory to be monitored			Description when corresponding bit is ON		Corrective action
Bit position	Address				
	CH1 side: Un\G513	CH2 side: Un\G514			
b0	—		—		—
b1	SIO		When interlink operation is set	Transmission setting error	Check the parameter setting and perform the test again.[1] ☞ Page 454 PARAMETER SETTING
b2	PRO			Protocol setting error	
b3	P/S		When independent operation is set	Transmission setting error	
b4	C/N			Protocol setting error	
b5 to b6	—		—		—
b7	NEU	—	Test completed		—
b8 to b13	—		—		—
b14	—	CH2 ERR	RAM error		Check the module installation status and perform the test again.[1]
b15	CH1 ERR	—	ROM error		

[1] When an error occurs even after performing the test again, please consult your local Mitsubishi representative.

Self-loopback test

The self-loopback test is a test to check the communication between C24 and the CPU module and the operation of the data transmit/receive function of C24.

Perform the self-loopback test according to the procedure below.

Operating procedure

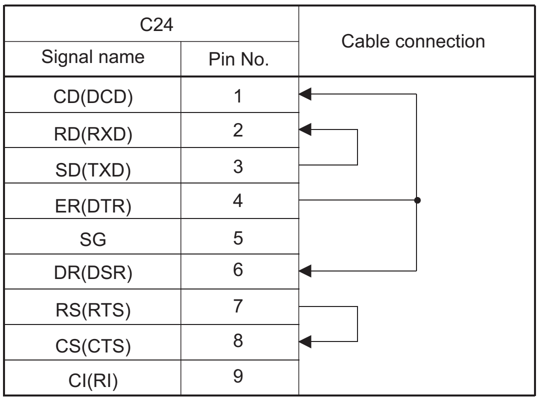
1. Connecting cable

- Connect cables to two interfaces as follows:

For the RS-232 interface, connect cables within the connector and install it to the interface.

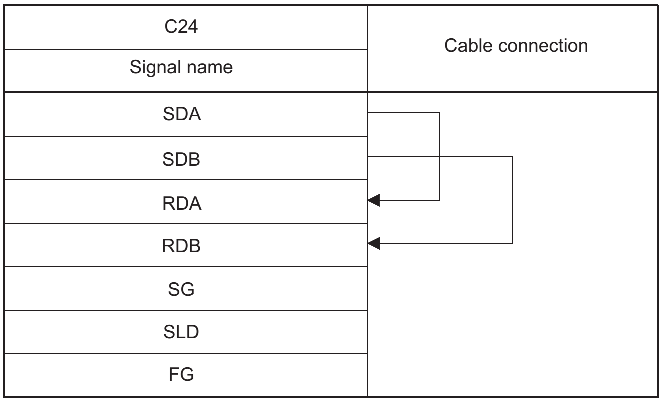
For the RS-422/485 interface, connect cables on the terminal block.

RS-232 interface (D-Sub 9P female)



C24		Cable connection
Signal name	Pin No.	
CD(DCD)	1	
RD(RXD)	2	
SD(TXD)	3	
ER(DTR)	4	
SG	5	
DR(DSR)	6	
RS(RTS)	7	
CS(CTS)	8	
CI(RI)	9	

RS-422/485 interface



C24	Cable connection
Signal name	
SDA	
SDB	
RDA	
RDB	
SG	
SLD	
FG	

2. Test mode setting (☞ Page 454 PARAMETER SETTING)

- Set the test mode setting to "Self-loopback test".
- Configure all of the transmission settings for the two interfaces according to the transmission specifications.
- Set the CPU module to the STOP state, and write the parameters.

3. Performing the self-loopback test

- By restarting or resetting the CPU module, the test starts automatically in approximately one second. The RUN LED flashes during the test. (Set the CPU module in the STOP state.)
- C24 performs the following tests in sequence and repeat them (one test cycle takes approximately one second). C24 ends the test when all test results become abnormal.

Test item	Description	LED flashes during the test
Checking communication with CPU module	Read and check the CPU module model.	RUN LED
Checking the transmit/receive function of interface	Send and receive data by changing data.	SD LED RD LED

4. Checking the results of self-loopback test

- When the result is normal

This test is performed repeatedly. When the ERR LED is OFF, C24 is operating normally.

- When the result is abnormal

The ERR LED turns ON.

5. Completing self-loopback test

- When the result is normal

Start data communication with the target device.

· Configure the parameter settings with Engineering tool. (☞ Page 454 PARAMETER SETTING)

· Turn OFF the power for the station with C24, and connect the cable to communicate with the target device.

· Turn ON the power for the station with C24.

- When the result is abnormal

Check an error description on the "Intelligent Function Module Monitor" screen of Engineering tool.

Check the error descriptions on the screens by bit-decoding the monitored value (16-bit integer). Each bit corresponds to the following error description.

Buffer memory to be monitored			Description when corresponding bit is ON	Corrective action
Bit position	Address			
	CH1 side: Un\G513	CH2 side: Un\G514		
b0 to b3	—		—	—
b4	C/N	—	Error occurred on the CPU module.	Remove the cause of error in the CPU module.
			The power supply capacity is insufficient.	Review the power supply capacity.
			Module is not mounted correctly.	Mount the module properly.
			Error on the base unit, extension cable, CPU module, or C24 occurred.	Check for each module and remove the cause of the error. Reconnect the cable correctly. Mount the module properly.
b5 to b13	—		—	—
b14	—	CH2 ERR	CH2 side communication error	Reconnect the cable correctly. Review the test wiring connection.
b15	CH1 ERR	—	CH1 side communication error	

Loopback test

A loopback test is a test to perform communication with the loopback test function of the MC protocol in order to check the connection between C24 and the target device, the communication function of each device, and the operation of the communication program of the target device.

Operating procedure

1. Connecting C24 to a target device

Turn OFF the power for the station with C24 and connect a communication cable to communicate with the target device in order to perform data communication using the MC protocol.

📖MELSEC iQ-R Serial Communication Module User's Manual(Startup)

2. Setting the parameter with Engineering tool

- Configure the parameter settings with Engineering tool, and write the parameters to the CPU module in order to perform data communication using the MC protocol.
- The following shows an example of the settings when performing the loopback test on the interface on the CH1 side.

Parameter setting item		Setting contents
Communication protocol setting		MC protocol (Format 1)
Communication speed setting		9600 bps
Transmission setting	Operation setting	Independent operation
	Data bit	7
	Parity bit	Yes
	Odd/even parity	Odd
	Stop bit	1
	Sum check code	Yes
	Online change	Disable
	Setting change	Disable
Station number setting		0

3. Performing loopback test

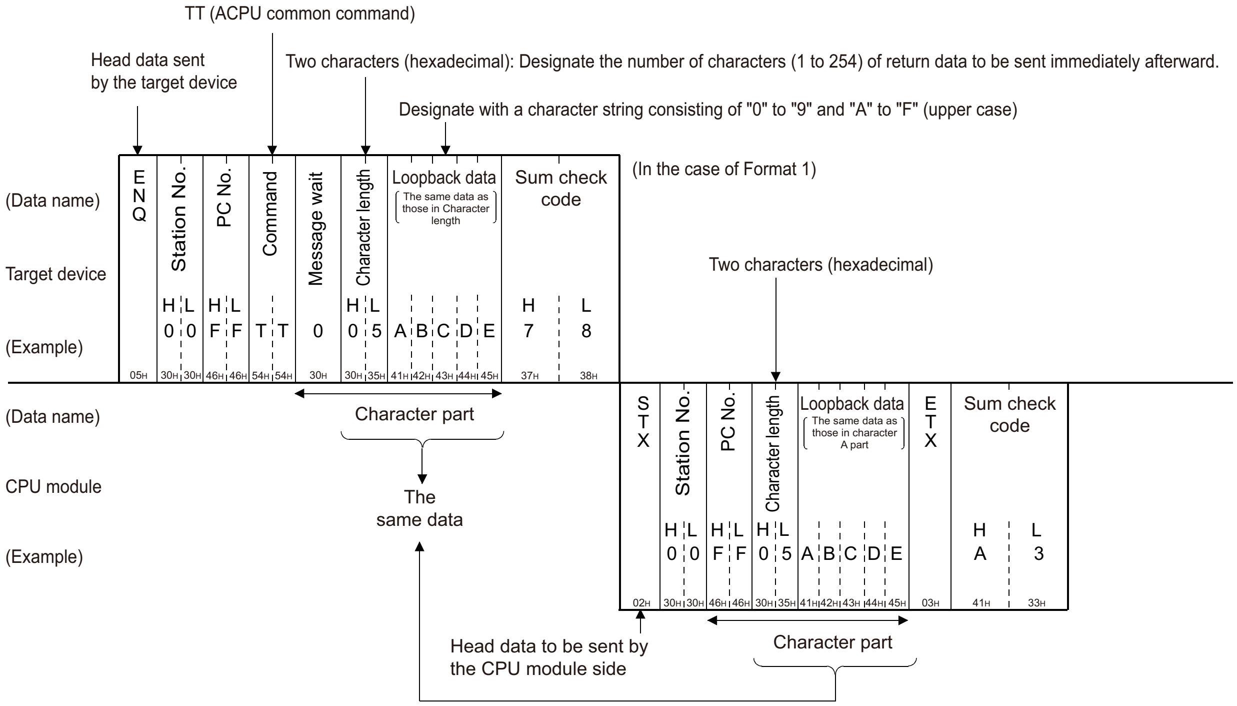
- By restarting or resetting the CPU module, the communication is enabled in approximately one second. (Set the CPU module in the STOP state.)
- Create a program for the loopback test on the target device side and send a command message as shown below to C24.

For communication test with 1C frame: TT command

For communication test with 2C/3C/4C frame[*1]: 0619 command

[*1] When performing communication test with 2C/3C/4C frame, refer to the following manual.
📖MELSEC Communication Protocol Reference Manual

An example of control procedure when the TT command is used in the loopback test (station number: 00, sum check code: Yes) is as shown below.


TT (ACPU common command)
Head data sent by the target device
Two characters (hexadecimal): Designate the number of characters (1 to 254) of return data to be sent immediately afterward.
Designate with a character string consisting of "0" to "9" and "A" to "F" (upper case)
(In the case of Format 1)
(Data name)
Target device
(Example)
ENQ
Station No.
PC No.
Command
Message wait
Character length
Loopback data
The same data as those in Character length
Sum check code
H L H L H L H L
0 0 F F T T 0 0 5 A B C D E 7 8
05H 30H 30H 46H 46H 54H 54H 30H 30H 35H 41H 42H 43H 44H 45H 37H 38H
Character part
The same data
Two characters (hexadecimal)
(Data name)
CPU module
(Example)
STX
Station No.
PC No.
Character length
Loopback data
The same data as those in character A part
ETX
Sum check code
H L H L H L H L
0 0 F F 0 5 A B C D E A 3
02H 30H 30H 46H 46H 30H 35H 41H 42H 43H 44H 45H 03H 41H 33H
Head data to be sent by the CPU module side
Character part

4. Checking loopback test result

Check for the target device side whether the data in the character area received from C24 in response to the data sent by the target device matches the data in the character area sent by the target device. (C24 sends back to the target device the received data, as is.)

- When the result is normal

The data in the character area that C24 sent back to the target device matches the one the target device had sent.

- When the result is abnormal

The data in the character area that C24 sent back to the target device does not match the one the target device had sent.

5. Completing loopback test

- When the result is normal

Configure the parameter settings with Engineering tool in accordance with the function to be used, and turn ON the power for the station with C24 to start data communication.

- When the result is abnormal

Check the items below, and take corrective actions.

Item	Corrective action
Reviewing the wiring of the connection cable	Check if the connection cable is wired correctly. 📖MELSEC iQ-R Serial Communication Module User's Manual(Startup)
Reviewing the parameter setting of C24	Check whether the C24 parameter settings (such as transmission setting and communication speed setting) match that of the target device. ☞ Page 454 PARAMETER SETTING
Check the detected error of C24	Check the error contents and take the corrective actions. ☞ Page 482 TROUBLESHOOTING

29.2 How to Clear Programmable Controller CPU Information

This section explains how to clear the programmable controller CPU information.

Programmable controller CPU information

- This is the information about the access target CPU module type used for communications using MC protocol and the MODBUS slave function.
  C24 obtains this information from the access target CPU module at the time of initial access, and stores it.
  When accessing for the second time or later, the processing to access to the CPU module get faster in accordance with this information.
- The programmable controller CPU information is cleared in the following cases:

When the programmable controller is powered OFF → ON, or when the CPU module is reset

When the programmable controller CPU information clear request is executed

If the programmable controller CPU information has not been correctly obtained

The following problems may occur:

- Accessible device range become narrowed. (Error code: 7140H)
- Some of commands and/or devices cannot be used. (Error code: 7142H, 714DH),

etc.

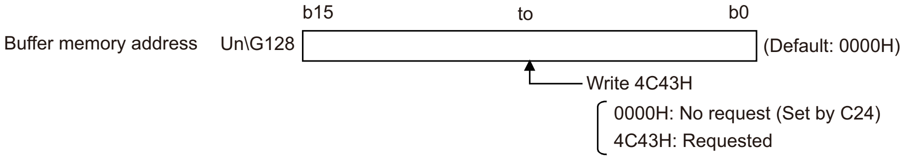
In the above case, execute the programmable controller CPU information clear request.

Restriction

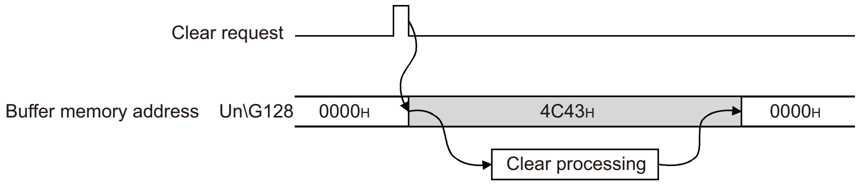
If initial access is made at startup of the access target CPU module or while the network is unstable, the programmable controller CPU information may not be correctly acquired.

Operation of the programmable controller CPU information clear request

- Write "4C43H" to the 'Programmable controller CPU information clear request' (Un\G128). (Set by the user)



- The programmable controller CPU information clear processing of C24 is performed.[*1]
- Upon completion of the clear processing, "0000H" is written to the 'Programmable controller CPU information clear request' (Un\G128). (Set by the C24)



[*1] The 'Transmission sequence status' (Un\G597/613) is also initialized.

Executing the programmable controller CPU information clear request

1. In the "Intelligent Function Module Monitor" screen, set "4C43H" to the 'Programmable controller CPU information clear request' (Un\G128).

2. Check the 'Programmable controller CPU information clear request' (Un\G128) turns to "0000H" with the Buffer Memory Batch Monitor function.

Precautions

- Execute the programmable controller CPU information clear after communication with the target device is stopped. In addition, do not perform data communication with the target device during the programmable controller CPU information clear processing. (If any data are received from the target device during the programmable controller CPU information clear processing, the data will be discarded.)
- When a modem is connected, cut off the line linked with the target device before executing the programmable controller CPU information clear. (The line is cut off at execution of the programmable controller CPU information clear.)
- Do not execute the UINI instruction during the programmable controller CPU information clear processing. If the programmable controller CPU information clear is attempted during execution of the UINI instruction, the clear processing will be executed after completion of the UINI instruction.

29.3 Troubleshooting by Symptom

The troubleshooting for any troubles generated during communication between C24 and the target device by trouble symptom are as shown below.

When any trouble occurs, check on the C24 condition at first, and check the relevant item in the table below.

Symptom	Symptom description	Protocol					
		MC	Non	Bi	Pd	MD	S
LED turns ON or flashes[*1]	☞ Page 502 SD LED does not flash even after a send request was made	○	○	○	○	—	○
	☞ Page 502 RD LED does not flash even though the target device is transmitting a message	○	○	○	○	○	○
	☞ Page 503 No response message is returned even though the target device transmitted a message and the RD LED flashes	○	—	○	—	○	—
	☞ Page 505 Read request signal does not turn ON even though the target device transmitted a message and the RD LED flashes	—	○	○	—	—	—
	☞ Page 507 The CPRTCL instruction execution is not completed although the RD LED flashes, or a receive wait timeout error (7D13H) occurs	—	—	—	○	—	—
	☞ Page 508 The target device transmitted a message and the RD LED flashes, but a time-out error (7A00H) occurs.	—	—	—	—	—	○
	☞ Page 508 RUN LED turns OFF	○	○	○	○	○	○
	☞ Page 508 ERR LED flashes	○	○	○	○	○	○
	☞ Page 509 ERR LED turns ON	○	○	○	○	○	○
	☞ Page 510 C ERR LED turns ON	○	○	○	○	○	○
Communication error occurred	☞ Page 511 Hardware information communication error "SIO" occurs[*2]	○	○	○	○	○	○
	☞ Page 512 Hardware information communication error "PRO." occurs[*2]	○	—	—	—	○	—
	☞ Page 513 Hardware information communication error "P/S" occurs[*2]	○	○	○	○	○	○
	☞ Page 513 Hardware information communication error "C/N" occurs[*2]	○	○	○	○	○	—
	☞ Page 514 Hardware information communication error "NAK" occurs[*2]	○	—	—	—	—	—
	☞ Page 514 Communication is intermittent	○	○	○	○	○	○
	☞ Page 515 Undecodable data are transmitted or received	○	○	○	○	○	○
	☞ Page 516 Whether the communication error is caused on C24 or target device is unclear	○	○	○	○	○	○
Communication via modem cannot be established[*3]	☞ Page 516 Communication is not established via modem	○	○	○	—	—	—
	☞ Page 517 Communication is not established with the ISDN sub-address when using a modem	○	○	—	—	—	—
The programmable controller CPU monitoring function does not operate in the specified cycle	☞ Page 517 Fixed cycle send is not performed normally	○	○	—	—	—	—
	☞ Page 517 Condition match send is not performed normally	○	○	—	—	—	—
Data cannot be received by an interrupt program	☞ Page 517 Data cannot be received by an interrupt program	—	○	○	—	—	—
The initial setting cannot be configured	☞ Page 518 Data cannot be written to flash ROM	○	○	○	○	—	○

[*1] For details on LED status, refer to the following manual.
📖MELSEC iQ-R Serial Communication Module User's Manual(Startup)

[*2] For details on how to check the hardware information communication error, refer to the following section.
☞ Page 483 Checking error status

[*3] If an error occurs while performing data communication using the modem function, first check the following items and determine whether or not the error is caused by the modem/TA connection.
If an error relating to the modem function occurred, take necessary corrective actions according to the reference sections shown below:
· Check of the ON/OFF status of current input/output signals on C24
· Check of the error code storage status for the modem function
· Check of the ON/OFF status of the DR terminal on the modem/TA (Refer to the instruction manual for modem/TA.)

Precautions

The considerations for data communication with the target device via either of C24 interfaces are shown below:

- When powering up the C24 side or the target device side, data reception error may occur at the target device side being connected.
- When the connected target device side starts up during data transmission, the data reception error occurs at the target device side.
- When the ERR LED on C24 turns ON due to data reception error, turn OFF the LED as necessary. (☞ Page 492 Turning OFF the ERR LED and initializing the error codes)
  Also, when a data reception error occurs on the target device side, take corrective actions in accordance with the instruction manuals for the target device.

■Error occurred on the C24 side

Handle the error as shown below when data reception error occurs on C24 side.

- For communications using the MC protocol
  C24 will disregard the received data or return a response message indicating an abnormal completion if it detects a data reception error after receiving the head data of the command message in the set format.
  If the data reception error is detected before the head data of the command message in the set format is received, the received data will be disregarded.
- When communicating using the predefined protocol
  C24 will store the error code in the 'Data reception result' (Un\G600/616) when a data reception error occurs.
  Reception error can be also checked with the send/receive data monitoring function.
  Take appropriate measures such as detecting a data reception error or clearing the received data, if necessary.
  For more details on the send/receive data monitoring function, refer to the following section.
  (☞ Page 419 USING SEND/RECEIVE DATA MONITORING FUNCTION)
- When communicating using nonprocedural protocol
  C24 will turns ON the reception abnormal detection (X4/XB) when it detects a data reception error.
  Take appropriate measures such as detecting a data reception error or clearing the received data, if necessary.
  (☞ Page 68 Reception error detection and check methods, ☞ Page 70 Receive data clear)
- For communications using bidirectional protocol
  C24 will return a response message indicating an abnormal completion if it detects a data reception error after receiving the head data of the communication message for the bidirectional protocol.
  If the reception error is detected before the head data of the communication message for the bidirectional protocol is received, the received data will be disregarded.
- When communicating using the MODBUS slave function
  C24 will disregard received data if it detects a CRC or LRC error during data reception.
  When using RTU mode, send request data as continuous data from a connected target device.
  If no-reception time occurs for 1.5 character times or more before all data is received, C24 will disregard received data and regard the head data of data received next as a new address field.
  When using ASCII mode, if there is no-reception time for one second or more, an error will occur.
- When communicating using the simple CPU communication function
  C24 will discard received data if it detects a reception error during data reception.

■Meaning of abbreviations in "Protocol" column

MC: MC protocol

Non: Nonprocedural protocol

Bi: Bidirectional protocol

Pd: Predefined protocol

MD: MODBUS (slave function)

S: Simple CPU communication

SD LED does not flash even after a send request was made

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
SD LED does not flash even after a send request was made.	The CS(CTS) signal is in the OFF-state, and C24 cannot transmit data to the external device.	• While the CS(CTS) signal is OFF, C24 does not transmit data to the target device. Connect so that it is constantly ON when the target device is ready to receive data.	○	○	○	○	—	○
	Due to the transmission control (DTR control, DC control), the external device cannot receive data.	• Wait until the external device is ready to receive data. • Check the external device or the cable condition when it takes too long for the external device to be ready to receive data.	○	○	○	○	—	○

RD LED does not flash even though the target device is transmitting a message

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
RD LED does not flash even though the target device is transmitting a message.	Signal lines are not connected correctly.	• Cross-connect the RD(RXD) and SD(TXD) signals between C24 and the target device. • When using MX Component, match the port opened by the OPEN statement of the target device with the cable connection port.	○	○	○	○	○	○
	The transmission control signals on the target device side have not turned ON.	• Connect the wiring so that the transmission control signals including DR(DSR) and CS(CTS) are ready. ■Remarks The Specifications of control signal vary depending on the device. Perform the wiring in accordance with the instruction manual of the device used.	○	○	○	○	—	○
	If the message passes through a modem or others, the signal is interrupted at the side of device relaying the message.	• Check the specification of the modem to review the settings and wiring, and correct them if necessary.	○	○	○	—	—	○

No response message is returned even though the target device transmitted a message and the RD LED flashes

Symptom	Cause	Corrective action	MC	No n	Bi	Pd	MD	S
The target device transmitted a message and the RD LED flashes, but C24 does not return the response message.	Communication protocol setting is incorrect.	• Review the communication protocol setting. • Review the program whether unintended mode switching is performed when the buffer memory mode switching area is used for mode switching. ■Remarks The communication protocol setting contents and the communication protocol currently in operation can be checked at the following addresses. • 'Communication protocol status (parameter setting)' (Un\G592/608) set by the parameter • 'Communication protocol status (current)' (Un\G594/610) currently in operation	○	—	—	—	○	—
	The CD terminal has been turned OFF when the CD terminal check is enabled.	• Change the wiring so that the CD terminal remains ON all the time.	○	—	○	—	—	—
	The header byte of the message is not data needed for the set protocol and format.	Check the data transmitted from the target device with the Circuit Trace function of Engineering tool, and take the following corrective actions. • If the format of message sent from the target device does not match with the respective protocol and format, correct the massage format according to the relevant protocol and format used. ■Remarks C24 skips all the messages transmitted until the header byte specified for each protocol and format (for example, "ENQ(05H)" for format 1) is received.	○	—	○	—	○	—
	C24 station number setting and message station number designation do not match.	Check the data transmitted from the target device with the Circuit Trace function of Engineering tool, and take the following corrective actions. • Match the station number setting of C24 with the station number designation of the message. ■Remarks C24 performs the data reception processing when the station No. designation is the host station after the header byte specified for each protocol and format is received.	○	—	—	—	—	—
	The MODBUS station number and the slave station number of a MODBUS message do not match.	Check the data transmitted from the target device with the Circuit Trace function of Engineering tool, and match the MODBUS station number of C24 with the station number designation of the message.	—	—	—	—	○	—
	The C24 data communications monitoring time is set to "Unlimited wait", or the monitoring time is too long.	• Set or shorten the monitoring time and retransmit the message from the target device and determine the trouble from the contents of the time-out error. ■Remarks Error detection by the monitoring time (timer 0 to timer 2) is possible in the following cases. • When part of a message is skipped. • When transmission from the target device is interrupted. • A link error occurred while accessing another station via CC-Link IE TSN, CC-Link IE Controller Network, CC-Link IE Field Network, MELSECNET/H, or MELSECNET/10. For details of the monitoring time, refer to the following manual. ☞ Page 259 CHANGING DATA COMMUNICATION MONITORING TIME	○	—	○	—	—	—

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
The target device transmitted a message and the RD LED flashes, but C24 does not return the response message.	The route setting of the access target CPU module is incorrect.	• Check the route of the access target CPU module, and set the route again.	○	—	—	—	—	—
	The CPU response monitoring time of the C24 is too long when using the MODBUS slave function.	• Shorten the CPU response monitoring time and send the message again from the target device, and identify the cause of the error from the contents of the timeout error.	—	—	—	—	○	—
	When using the MODBUS slave function, "Master Station" is selected for "Access target" in "CC-Link IEF Remote Head Module" and a Q series CPU module is accessed.	• When accessing a Q series CPU module via CC-Link IE Field Network, the first communication is delayed for a CPU response monitoring timer value. Shorten the CPU response monitoring time as necessary.	—	—	—	—	○	—
	"Communication data reception enable/disable designation" in the module parameter is set to "Receive disable."	• Change the setting in the module parameter to "Receive enable" or 'Communication data reception enable/disable designation' (Un\G292/452) to 'Receive enable' (0).	○	—	—	—	○	—

Read request signal does not turn ON even though the target device transmitted a message and the RD LED flashes

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
The target device transmitted a message and the RD LED flashes, but 'CH□ Reception data read request' (X3/XA) does not turn ON.	Communication protocol setting is incorrect.	• Review the communication protocol setting. • Review the program whether unintended mode switching is performed when the buffer memory mode switching area is used for mode switching. ■Remarks The communication protocol setting contents and the communication protocol currently in operation can be checked at the following addresses. • 'Communication protocol status (parameter setting)' (Un\G592/608) set by the parameter • 'Communication protocol status (current)' (Un\G594/610) currently in operation	—	○	○	—	—	—
	The CD terminal has been turned OFF when the CD terminal check is enabled.	• Change the wiring so that the CD terminal remains ON all the time.	—	○	○	—	—	—
	When DC control is selected, the necessary DC codes are not transmitted.	Check the data transmitted from the target device with the Circuit Trace function of Engineering tool, and take the following corrective actions. • During DC1/DC3 control, if the target device transmits 'DC3(13H)' and interrupts the transmission processing of C24, transmit 'DC1(11H) at first and then the next message after the transmission processing of C24 resumed. • During DC2/DC4 control, if the target device transmits the next message without adding 'DC2(12H)' after transmitting 'DC4(14H)', transmit a message with 'DC2(12H) added.	—	○	○	—	—	—
	The end code has not been received, or the fixed length data has not been received.	Check the data transmitted from the target device with the Circuit Trace function of Engineering tool, and take the following corrective actions. • Match the end code set by the receive end code designation of C24 with the one sent from the target device. • Match the data quantity set by the receive end data quantity of C24 with the data length sent from the target device. • For communication with the user frame, send the message with the final frame added to the end from the target device. ■Remarks When proper data are being transmitted from the target device, take the corrective action by referring to the troubleshooting below. • ☞ Page 511 Hardware information communication error "SIO" occurs • ☞ Page 508 ERR LED flashes	—	○	○	—	—	—
	The C24 setting and message format do not match.	The interpretation of data count, or message format, etc. varies depending on the settings below. Review the settings of each item and the operation by the settings in accordance with the relevant descriptions. • ASCII-BIN conversion enable/disable • Transparent code setting • User frame setting	—	○	○	—	—	—

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
The target device transmitted a message and the RD LED flashes, but 'CH□ Reception data read request' (X3/XA) does not turn ON.	The C24 data communications monitoring time is set to "Unlimited wait", or the monitoring time is too long.	• Set or shorten the monitoring time and retransmit the message from the target device and determine the trouble from the contents of the time-out error. ■Remarks In the cases below, errors can be detected by setting each monitoring time (timer 0 to 2). • When part of a message is skipped. • When transmission from the target device is interrupted. For details of the monitoring time, refer to the following manual. ☞ Page 259 CHANGING DATA COMMUNICATION MONITORING TIME	—	○	○	—	—	—
	"Communication data reception enable/disable designation" in the module parameter is set to "Receive disable."	• Change the setting in the module parameter to "Receive enable" or 'Communication data reception enable/disable designation' (Un\G292/452) to 'Receive enable' (0).	—	○	○	—	—	—

The CPRTCL instruction execution is not completed although the RD LED flashes, or a receive wait timeout error (7D13H) occurs

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
When the CPRTCL instruction was executed for the protocol with the communication type set to "Receive Only" or "Send & Receive," the RD LED flashes but the instruction execution is not completed, or a receive wait timeout error (7D13H) occurs.	Incorrect protocol number is specified in the control data.	• Review the protocol number specified in the control data for the CPRTCL instruction.	—	—	—	○	—	—
	The CD terminal has been turned OFF when the CD terminal check is enabled.	• Change the wiring so that the CD terminal remains ON all the time.	—	—	—	○	—	—
	The user-edited protocol setting with the Predefined protocol support function of Engineering tool is inconsistent with the packet format of the target device.	Check the packet format of the target device, and review the protocol setting. • In the packet setting, are the element type and element order incorrect? • Is each element setting item (Code type, Data length, Data flow, Calculating range, etc.) incorrect?	—	—	—	○	—	—
	C24 has not received the expected packet.	Check the data transmitted from the target device with the Circuit Trace function of Engineering tool, and correct the message if it's incorrect. When proper data are being transmitted from the target device, take the corrective action by referring to the troubleshooting below. • ☞ Page 508 ERR LED flashes • ☞ Page 514 Communication is intermittent	—	—	—	○	—	—
	C24 has cleared the data received before the protocol execution. (On the "Protocol Detailed Setting" screen of Engineering tool predefined protocol support function, "Clear receive data before protocol execution" is designated.)	• When data may be received before protocol execution, unselect the checkbox for "Clear OS area (receive data area) before protocol execution" on the "Protocol Detailed Setting" screen of Engineering tool predefined protocol support function.	—	—	—	○	—	—
	The C24 data communications monitoring time is set to "Unlimited wait", or the monitoring time is too long.	• Perform cancel processing to complete the CPRTCL instruction execution. • Set or shorten the monitoring time and retransmit the message from the external device and determine the trouble from the contents of the time-out error. ■Remarks In the cases below, errors can be detected by setting the receive wait time for the "Receive Setting" on the "Protocol Detailed Setting" screen of Engineering tool predefined protocol support function. • When part of a message is skipped. • When transmission from the target device is interrupted.	—	—	—	○	—	—
	"Communication data reception enable/disable designation" in the module parameter is set to "Receive disable."	• Change the setting in the module parameter to "Receive enable" or 'Communication data reception enable/disable designation' (Un\G292/452) to 'Receive enable' (0).	—	—	—	○	—	—

The target device transmitted a message and the RD LED flashes, but a time-out error (7A00H) occurs.

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
The target device transmitted a message and the RD LED flashes, but a time-out error (7A00H) occurs.	"Various control specification" in the module parameter is incorrect.	• Review the settings in "Various control specification" in the module parameter according to the settings of a communication destination.	—	—	—	—	—	○
	"Communication data reception enable/disable designation" in the module parameter is set to "Receive disable."	• Change the setting in the module parameter to "Receive enable" or 'Communication data reception enable/disable designation' (Un\G292/452) to 'Receive enable' (0).	—	—	—	—	—	○

RUN LED turns OFF

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
RUN LED on C24 turns OFF	The module is not mounted properly.	• Mount the C24 on the base unit properly.	○	○	○	○	○	○
	Power module 5 VDC consumption current is insufficient.	• Calculate the 5 VDC consumption current of each mounted module. If the current capacity is insufficient, review power module selection.	○	○	○	○	○	○
	Extraneous noise causes abnormal operation of C24.	• Review the wiring so that the shield is single-point grounded. • When ground is used with other devices, ground the C24 independently.	○	○	○	○	○	○

ERR LED flashes

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
ERR LED flashes.	The communication protocol setting or transmission setting is incorrect.	• Read the error code from the buffer memory, and check the error contents and change to the correct setting.	○	○	○	○	○	○
	The parameter settings are incorrect.	• Review the parameter settings with Engineering tool. Then, write the parameter settings to the CPU module, and reset it. • Review each settings for the parameters set in the buffer memory, and turn ON the 'CH□ mode switching request' (Y2/Y9). For more details on 'CH□ mode switching request', refer to the following section. (☞ Page 413 I/O Signals for Handshake with CPU Module and Buffer Memory)	○	○	○	○	○	○
	The protocol setting data stored in the CPU module is not for MELSEC iQ-R series C24.	• Write the protocol setting data for MELSEC iQ-R series C24 to the CPU module.	—	—	—	○	—	—

- Remarks

The error code when ERR LED flashes is stored in the buffer memory shown below.

Error factor	Error code storage buffer memory		
	Address		Name
	CH1	CH2	
Parameter setting error	Un\G515		Parameter setting error, mode switching error status
Parameter error (Module parameter)	Un\G20224 to 20231		Error address storage area

ERR LED turns ON

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
ERR LED turns ON.	The online mode switching designation is incorrect.	• Read the error code from the buffer memory, and check the error contents and correct the designation contents.	○	○	○	○	○	○
	The receive data clear request is executed during the dedicated instruction execution.	• Read the execution result (error code) of control data for the dedicated instruction of which the abnormal completion signal has turned ON, and take the corrective action corresponding the error contents.	○	○	○	○	—	○
	The power of C24 was turned OFF while writing to the flash ROM.	• Write the data again to the flash ROM.	○	○	○	○	○	○
	An error is detected.	• Check if any symptom corresponding to the troubleshooting by symptom occurs, and remove the cause of the error. (☞ Page 500 Troubleshooting by Symptom)	○	○	○	○	○	○
When the predefined protocol is executed, the ERR LED turns ON.	The communication protocol setting is set to other than "Predefined protocol".	• Set the communication protocol setting to "Predefined protocol".	—	—	—	○	—	—
	No protocol setting data has been written.	• Write the protocol setting data with the Predefined protocol support function of Engineering tool, and then execute the CPRTCL instruction.	—	—	—	○	—	—
	The protocol corresponding to the protocol number specified in the control data for the CPRTCL instruction is not registered.	• Read the protocol setting from C24 with the Predefined protocol support function of Engineering tool. • On the "Add Protocol" screen, check for the registered protocol and protocol No., and correct them if incorrect.	—	—	—	○	—	—
	Multiple CPRTCL instructions were simultaneously executed to the same channel.	• Review the program so that the next CPRTCL instruction is executed once one CPRTCL instruction execution is completed.	—	—	—	○	—	—
	Incorrect control data are specified for the CPRTCL instruction.	• Review the control data for the CPRTCL instruction.	—	—	—	○	—	—
	The edited protocol setting with the Predefined protocol support function of Engineering tool is inconsistent with the packet format of the target device.	Check the packet format of the target device, and review the protocol setting taking the following into consideration. • When a Conversion variable exists in the received packet, are conversion contents or the settings for sign character, number of decimals, or number of digits incorrect? • When a check code exists in the received packet, are the processing method or the settings for code type, data length, data flow, complement calculation, or calculating range incorrect. • When a Length exists in the received packet and a Conversion variable is included in the calculating range of the Length, are the code type or the settings for data length, data flow, or calculating range incorrect?	—	—	—	○	—	—
	An expected packet has not been received, and the monitoring timeout error occurred.	• Check the data transmitted from the target device with the Circuit Trace function of Engineering tool, and correct it if the message is in correct.	—	—	—	○	—	—
	Communication error occurred.	• Check the error description with the System Monitor function, and remove the cause of the error.	—	—	—	○	—	—

- Remarks

The error code when ERR LED turns ON is stored in the buffer memory shown below.

Error factor	Error code storage buffer memory		
	Address		Name
	CH1	CH2	
Modern function error	Un\G545		Modern function error
Monitoring device error	Un\G8709	Un\G8965	Programmable controller CPU monitoring function execution result
Predefined protocol function error	Un\G16450	Un\G16466	Predefined protocol function error code
MODBUS slave function error	Un\G31998 to Un\G32255		MODBUS operating status Error log

C ERR LED turns ON

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
C ERR LED turns ON.	An error occurred during the on-demand execution.	• Read the error code from the buffer memory, and check the error contents and correct the designation contents.	○	—	—	—	—	—
	C24 detected an error while transmitting data.	• Read the error code from the buffer memory, and take the corrective action corresponding to the error contents.	○	○	○	○	○	○
	C24 detected an error while receiving data.	• Read the error code from the buffer memory, and take the corrective action corresponding to the error contents.	○	○	○	○	○	○

- Remarks

The error code when C ERR LED turns ON is stored in the buffer memory shown below.

Error factor	Error code storage buffer memory		
	Address		Name
	CH1	CH2	
On-demand execution error	Un\G598	Un\G614	On-demand execution results
Data transmission error	Un\G599	Un\G615	Data transmission result
Data reception error	Un\G600	Un\G616	Data reception result
Modern function error	Un\G545		Modern function error code
MC protocol send error	Un\G602	Un\G618	MC protocol send error code
Monitoring device error	Un\G8709	Un\G8965	Programmable controller CPU monitoring function execution result
Predefined protocol function error	Un\G16450	Un\G16466	Predefined protocol function error code
MODBUS slave function error	Un\G31998 to Un\G32255		MODBUS operating status Error log
Simple CPU communication function error	Un\G32872 to Un\G45159		Diagnostic information area for simple CPU communication Latest error code

Hardware information communication error "SIO" occurs

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
Hardware information communication error "SIO" occurs.	The stop bit settings do not match.	• Match the stop bit settings of C24 to that of the target device.	○	○	○	○	○	○
	Transmission speed is too fast and the next data is transmitted before C24 completes processing of the receive data.	• Decrease the transmission speed and restart data communications.	○	○	○	○	○	○
	Data larger than the receive buffer size was received.	• Use the DTR and DC controls to interrupt transmission before the buffer becomes full. Perform RS/CS control when the modem function is used. • Lengthen the transmission interval and take sufficient time for the read processing of the CPU module side. ■Remarks For the nonprocedural protocol, data up to the end code received at first or fixed length data is stored in the nonprocedural reception buffer memory area, and C24 turns ON the read request to the CPU module. When the next data is sent while the read request is ON, the data is temporarily stored in the operating system receive buffer. When the operating system receive buffer becomes full, C24 skips the rest of the data and turns ON the "SIO" LED.	—	○	○	○	—	—
	During multidrop connection, two or more devices transmitted data at the same time.	• Connect the target devices to C24 on a 1:1 basis and perform a communications test. If C24 can communicate normally with each of all target devices, two or more devices transmit data simultaneously during multidrop communication. Therefore, interlock the devices not to transmit at the same time.	○	○	—	○	○	○

Hardware information communication error "PRO." occurs

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
Hardware information communication error signal "PRO." turns ON.	Communications were performed with a control procedure different from the C24 communication protocol setting.	Check the data transmitted from the target device with the Circuit Trace function of Engineering tool, and take the following corrective actions. • Check the C24 communication protocol setting and the message from the target device and match the settings, or correct the message and restart data communications.	○	—	—	—	○	—
	Some of the data are different from the control procedure.		○	—	—	—	○	—
	Designated command does not exist.		○	—	—	—	○	—
	The device number designation is not the character count corresponding to the command.		○	—	—	—	○	—
	The characters in the message include a data other than the "A to Z", "0 to 9", "(SP)", and control data.	Check the data transmitted from the target device with the Circuit Trace function of Engineering tool, and correct it if the message is in correct. ■Remarks Only the data '0 to 9' and 'A to F' can be handled as character area data during communications using MC protocol (ASCII mode) or the MODBUS slave function (ASCII mode). Therefore, when transmitting a character string as data, convert the ASCII code of each character string to 2-byte ASCII code. • Transmitting the character "G" The ASCII code for "G" is 47H, and is transmitted as the two bytes 34H, 37H. • Transmitting the character "A" The ASCII code for "A" is 41H, and is transmitted as the two bytes 34H, 31H. (When 41H, ASCII code for "A" is transmitted unchanged, C24 ASCII↔BIN conversion will convert it to AH(10) and pass it to the CPU module.)	○	—	—	—	○	—
	A device number outside the designated range was designated.	• When specifying a device, check "Device Setting" of the parameters written to the CPU module and correct it to a device No. within the designated range and restart data communications.	○	—	—	—	○	—
	A request message with a function code or sub code not supported by a C24 was received.	Check the function codes and sub codes supported by the C24, and review the request message to be sent.	—	—	—	—	○	—
	The number of access points or size of a MODBUS device in a received request message is out of range.	Check the MODBUS device assignment parameters set for the C24, and review the request message to be sent.	—	—	—	—	○	—
	The size of device data specified in a received request message does not match that of write device data.	Review the contents of the request message.	—	—	—	—	○	—
	The size of a received request message is less than 4 bytes/8 bytes, or exceeds 256 bytes/512 bytes.		—	—	—	—	○	—
	An illegal character was received after the end code CR.		—	—	—	—	○	—

Hardware information communication error "P/S" occurs

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
Hardware information communication error signal "P/S" turns ON.	The parity bit settings do not match.	• Review the data format between C24 and the target device, and match the settings.	○	○	○	○	○	○
	Sum check codes do not match.	• Calculate to check if the sum check code transmitted from the target device is correct, and correct the sum check code if it's incorrect.	○	—	—	—	—	—
	When CH1 and CH2 are set for the interlink operation, there is an interface to which the cable is not connected.	• When using CH1 and CH2 for the interlink operation, connect the cable to both of the interfaces. • When using CH1 and CH2 for the independent operation, reconfigure the setting to the independent operation if it is set to the interlink operation. ■Remarks If the cable is not connected to one of the interfaces when CH1 and CH2 are used for the interlink operation, the data may be destroyed due to noise and may result in 'data cannot be decoded' or 'cause of communications error'.	○	○	—	—	○	—

Hardware information communication error "C/N" occurs

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
Hardware information communication error signal "C/N" turns ON.	PC number is specified to the self-station (FF), or a station other than the station number set with the network parameter.	• Change the PC number to the self-station (FF) or the station number set with the network parameter, and restart data communications.	○	—	—	—	—	—
	Routing parameters between the CPU module with C24 mounted and the communication destination CPU module are not set.	• Check the routing parameters and set them up to the communication destination CPU module.	○	—	—	—	—	—
	An error occurred on CC-Link IE TSN, CC-Link IE Controller Network, CC-Link IE Field Network, MELSECNET/H, or MELSECNET/10.	• Check the error contents by the status of SB and SW related to CC-Link IE Controller Network and CC-Link IE Field Network with the monitor function of Engineering tool etc., and take corrective actions in accordance with the reference manuals for respective network.	○	—	—	—	○	—
	An error occurred on the CPU module of the host station on CC-Link IE TSN, CC-Link IE Controller Network, CC-Link IE Field Network, MELSECNET/H, or MELSECNET/10.							
	Module installation address designation for communications with Intelligent function module is incorrect.	• Change the transmission message designation data.	○	—	—	—	—	—
	A command that cannot be transmitted during RUN (program, parameters, etc.) was transmitted, or "Online change" is set to be disabled.	• Stop the CPU module and restart data communications.	○	—	—	—	○	—
		• Enable the "Online change" by command and restart data communications.	○	—	—	—	○	—

Hardware information communication error "NAK" occurs

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
Hardware information communication error signal "NAK" turns ON.	Refer to section which describes the communication error signal ("C/N", "P/S", "PRO", "SIO") that are turned ON simultaneously when the "NAK" signal is turned ON.	• Take corrective action corresponding to the contents of the error.	○	—	—	—	—	—
	■Remarks When the "NAK" signal is turned ON, the error code is added to "NAK" and is transmitted from C24 to the target device. Take the corrective action depending on the error code listed below. ☞ Page 519 Error Code List In addition, the error code when "NAK" is transmitted is stored to the C24 buffer memory shown below, so it can be checked at the C24 side as well. • Error code storage buffer memory CH1 side: Un\G602 CH2 side: Un\G618							

Communication is intermittent

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
Communications are intermittently established and lost.	During multidrop connection, two or more devices transmitted data at the same time.	• Connect the target devices to C24 on a 1:1 basis and perform a loopback test. If C24 can communicate normally with each of all target devices, two or more devices transmit data simultaneously during multidrop communication. Therefore, interlock the devices not to transmit at the same time.	○	○	○	○	○	○
	The signal cable wiring has a loose connection.	• Replace the cable, or make the connections firm.	○	○	○	○	○	○
	The CD(DCD) signal turns ON/OFF repeatedly when the CD terminal check is enabled.	• Change the wiring so that the CD(DCD) signal remains ON all the time, or change the setting so that the CD terminal check is disabled.	○	○	○	○	—	○
	When half-duplex communications is set, the ON/OFF timing of each signal does not match.	• Control at the target device side so that the ON/OFF timing of each signal matches. For details, refer to the following section. (☞ Page 275 DATA COMMUNICATIONS USING HALF-DUPLEX COMMUNICATIONS)	—	○	○	○	—	○
	The next transmission processing was requested before a previous transmission processing had been completed.	• Make sure to obtain handshake in the program.	○	○	○	○	—	—
	The next reception processing was requested before a previous reception processing had been completed.							
	Communication via modem is disrupted.	• Since communication via modem may be disrupted due to low line quality or instantaneous interruption, create a program for reconnecting the line in case of unexpected line disruption. • If this symptom occurs frequently, reduce the communication rate of the modem, and retry the communication.	○	○	○	—	—	—
	The data sent from C24 are received back to the RS-422/485 interface (CH2) side.	• Disable the echo back with the Echo back enable/disable designation before communication.	○	○	○	○	—	○

When a normal completion response message cannot be received

The following explains how to take corrective actions when a normal completion response message cannot be received upon command message transmission.

■An abnormal completion response message is received

When a target device received an abnormal completion response message in response to the sent command message, take corrective action according to the error code in the response message. (☞ Page 519 Error Code List)

When the following problem occurs at the time of access to other station CPU module, clear the programmable controller CPU information and retry. (☞ Page 498 How to Clear Programmable Controller CPU Information)

- Accessible device range became narrow. (Error code: 7140H)
- Some of commands and/or devices cannot be used (Error code: 7142H, 714DH) etc.

■Response messages cannot be received

Change the setting value for the response monitoring time (timer 1, 5 seconds by default). (☞ Page 454 PARAMETER SETTING)

If response messages still cannot be received even after changing the setting value, check the communication protocol setting for the relevant interface, the ON status of the indicator LED, and the connection cable.

■The first part of a response message cannot be received

Increase the setting value of the transmission wait time (the default value is 0 ms). (☞ Page 454 PARAMETER SETTING)

If response messages still cannot be received even after increasing the setting value, reducing the processing time from the end of transmission processing to the start of reception processing on the target device side is required.

Undecodable data are transmitted or received

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
Data that cannot be decoded was transmitted and received.	During multidrop connection, two or more devices transmitted data at the same time.	• Connect the target devices to C24 on a 1:1 basis and perform a loopback test. If C24 can communicate normally with each of all target devices, two or more devices transmit data simultaneously during multidrop communication. Therefore, interlock the devices not to transmit at the same time.	○	○	○	○	○	○
	The parity bit settings do not match.	• Match the settings of C24 with that of the target device. ■Remarks If '7 bits' is set to the data bit with the parity bit set to 'Yes' at one side and '8 bits' is set to the data bit with the parity bit set to 'None' at the other side, the transmitted/received bit number will be the same. Therefore, data transmission/reception can be possible without error.	○	○	○	○	○	○
	The bit length settings for the stop bit do not match.		○	○	○	○	○	○
	Transmission speed settings do not match.	• Match the transmission speed of C24 with that of the target device.	○	○	○	○	○	○
	During multidrop connection, the terminating resistor is not connected correctly.	• Connect the terminating resisters to the stations of both ends. (110 Ω or 330 Ω for C24)	○	○	○	○	○	○
	When CH1 and CH2 are set for the interlink operation, there is an interface to which the cable is not connected.	• When using CH1 and CH2 for the interlink operation, connect the cable to both of the interfaces. • When using CH1 and CH2 for the independent operation, reconfigure the setting to the independent operation if it is set to the interlink operation. ■Remarks If the cable is not connected to one of the interfaces when CH1 and CH2 are used for the interlink operation, the data may be destroyed due to noise and may result in 'data cannot be decoded' or 'cause of communications error'.	○	○	—	—	○	—

Whether the communication error is caused on C24 or target device is unclear

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
Whether the communications error is caused on the C24 side or the target device side is unclear.	—	Perform the following tests on the C24 side to investigate where the cause is. • Check the hardware. C24 is mounted correctly. The pins of C24 has no abnormalities such as bent. • Perform the individual station test on C24. (☞ Page 492 Individual station test) No errors occur by the hardware test. No errors occur by the self-loopback test. • Check the CPU module condition. No errors that stop the CPU module operation occur. ■Remarks The following methods are available for testing if normal communications are possible with C24 alone. • For single C24, set CH1 and CH2 to the nonprocedural protocol with self-loopback wiring. When data transmitted from one channel and data received at the other channel match, the transmission/reception processing is performed normally. • When two C24s are used, set one C24 to the nonprocedural protocol and the other C24 to the MC protocol, and transmit an MC protocol message from the station which is set to the nonprocedural protocol. If the station which is set to the MC protocol returns the transmission message normally, transmission/reception processing is performed normally.	○	○	○	○	○	○

Communication is not established via modem

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
Auto modem initialization cannot be performed.	A modem initialization error occurs.	• Check the error code and take corrective actions.	○	○	○	—	—	—
	Modem connection CH specification is not performed.	• Set the modem connection CH specification.	○	○	○	—	—	—
	The parameter settings are incorrect.	• Set the parameters to match the host station modem/TA.	○	○	○	—	—	—
		Review the parameter settings below with Engineering tool. Then, write the parameter settings to the CPU module, and reset it. • Communication speed setting • Transmission setting • Communication protocol setting • Station number setting						

Communication is not established with the ISDN sub-address when using a modem

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
Data cannot be communicated with ISDN sub-address	Sub-address is incorrect.	• Review the sub-address.	○	○	—	—	—	—
	The modem does not support ISDN sub-address.	• Use modem supporting ISDN sub-address.	○	○	—	—	—	—
	Delimiter code for sub-address is incorrect.	• Review the delimiter code in accordance with the manual for the TA.	○	○	—	—	—	—

Fixed cycle send is not performed normally

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
The programmable controller CPU monitoring function does not operate in the cycle time designated by the fixed cycle send setting.	The setting value of the fixed cycle send is incorrect.	• Review the settings for the fixed cycle send. (Check if the cycle shorter than the one calculated with the monitoring timing of the CPU module has been set, etc.) (☞ Page 155 Monitoring timing of CPU module)	○	○	—	—	—	—
	The cycle time is affected by some causes.	• Remove the causes or designate the cycle time taking the causes into consideration.	○	○	—	—	—	—

Condition match send is not performed normally

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
The programmable controller CPU monitoring function does not operate in the cycle time designated by the condition match send setting.	The setting value of the condition match send is incorrect.	• Review the settings for the condition match send. (Check if the setting value of the programmable controller CPU monitoring function designation is incorrect.) (☞ Page 157 Condition match send)	○	○	—	—	—	—
	The cycle time is affected by some causes.	• Remove the causes or designate the cycle time taking the causes into consideration.	○	○	—	—	—	—

Data cannot be received by an interrupt program

Symptom	Cause	Corrective action	MC	Non	Bi	Pd	MD	S
Data cannot be received by an interrupt program.	C24 receives data by an instruction other than BUFRCVS.	• Use BUFRCVS instruction to receive interrupt data.	—	○	○	—	—	—
	The parameter settings for C24 from [Module Parameter] ⇨ [Interrupt Settings] are not configured, or the settings are incorrect.	• Review the parameter settings from [Module Parameter] ⇨ [Interrupt Settings].	—	○	○	—	—	—

Data cannot be written to flash ROM

Symptom	Cause	Corrective action	MC	No n	Bi	Pd	MD	S
Data cannot be written to the flash ROM.	The setting change is not set to be enabled in the parameter settings with Engineering tool.	• Set the setting change to 'Enable' with Engineering tool.	○	○	○	○	—	○
Data written to flash ROM are not valid.	The power of C24 was turned OFF while writing to the flash ROM.	• Write the data again to the flash ROM.	○	○	○	○	—	○
	Communication with Engineering tool was interrupted while writing to the flash ROM.							

29.4 Error Code List

This section explains the error codes, error contents, and corrective actions for errors may occur during data communication and when using the modem function.

The signal name in the table shows the LED lighting status and communication error status stored in the 'CH□ side LED lighting status, communication error status' (Un\G513/514) when error occurs.

The CH□ side LED lighting status and communication error status can be checked with the System Monitor function or the Intelligent Function Module Monitor function of Engineering tool.

Note that "NAK" turns ON for errors which occur by the MC protocol communication and the signal name is not indicated.

For the error code when communicating using 1C frame, refer to the following manual.

📖MELSEC-L Serial Communication Module User's Manual (Basic)

The meanings of abbreviations in the table are as follows:

- MC: MC protocol
- Non: Nonprocedural protocol
- Bi: Bidirectional protocol
- Pd: Predefined protocol
- MD: MODBUS (slave function)
- S: Simple CPU communication

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
2242H	Module parameter error	• Incorrect setting value was set to the module parameter.	—	• Set the correct value to the module parameter, and rewrite it to the CPU module.	○	○	○	○	○	○
2250H	Module extended parameter error	■The written module extended parameter setting is any of the following: • The module extended parameter for other module was written to the CPU module. • The predefined protocol setting has an error. • The data for the set item is corrupted.	CHn ERR.	• Write module extended parameters for C24 to the CPU module. • Check the predefined protocol setting in the protocol setting data error (buffer memory: 4086H to 4089H), and correct and write it to C24, and execute the CPRTCL instruction. • Replace the module if error occurs even after rewriting.	○	○	○	○		○
20E0H	Invalid module	• The module cannot communicate with the CPU module.	—	• The hardware failure of the CPU module may have been occurred. Please consult your local Mitsubishi representative.						
3000H	Modbus device duplication parameter allocation error	• MODBUS device or CPU device specified in MODBUS device allocation parameter overlaps with another allocation setting.	—	• Correct the duplicate MODBUS device allocation settings and write to the CPU again.					○	
3001H	MODBUS device upper limit value over error	• The start MODBUS device number + assigned points in the MODBUS device allocation parameter exceeds the maximum value (65535) allowed for the MODBUS device.	—	• Check the parameter error information (address: 7C13H to 7C15H), and review the start MODBUS device number and the number of assigned points, set in the corresponding MODBUS device allocation parameter.					○	

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
3002H	Buffer memory assigned range error	• The assigned range of the C24 buffer memory set with the MODBUS device allocation parameter exceeds the range of the user free area.	—	• Check the parameter error information (address: 7C13H to 7C15H), and review the start device number and the number of assigned points, set in the corresponding MODBUS device allocation parameter.					○	
3003H	Device upper limit value over error	• The start device number + assigned points in the MODBUS device allocation parameter exceeds the maximum value allowed for the CPU device.	—	• Check the parameter error information (address: 7C13H to 7C15H), and review the start device number and the number of assigned points, set in the corresponding MODBUS device allocation parameter.					○	
3004H	Device allocation number error	• The bit device assigned to the input register or holding register is not in units of 16 points.	—	• Check the parameter error information (address: 7C13H to 7C15H), and set the number of assigned points set in the corresponding MODBUS device allocation parameter in units of 16 points.					○	
3005H	CC-Link IEF remote head module access target error	• "Master station" was set in the access target, when the C24 is not mounted on the CC-Link IE Field network remote head module.	—	• Review the value set for the access target of the CC-Link IEF remote head module in the module parameter.					○	
3C00H	System error	• The operating system of C24 detected some error.	—	☞Page 541 Corrective actions for system error						
4000H to 4FFFH	CPU module error	• Errors detected in the CPU module	—	📖MELSEC iQ-R CPU Module User's Manual (Application)						○
7102H	System error	• The operating system of C24 detected some error.	—	☞Page 541 Corrective actions for system error						
7103H	PLC access error	• Cannot communicate with CPU module.	C/N	• Increase the monitoring time (timer 1). • Perform a self-loopback test, and check if the data communication with the CPU module can be established.	○					
7104H	System error	• The operating system of C24 detected some error.	—	☞Page 541 Corrective actions for system error						
7140H[*1]	Request data error	• The request exceeded the limit of processing points that are predetermined for each command and allowed for one transmission. • A word device was designated in a bit unit command. • The requested range exceeded the last device number. (CPU module's last device number ≥ head device number + number of device points in message sent from the target device)	PRO	• Check and correct the transmission message from the target device and restart data communications. • If no error is found in the device range, clear the programmable controller CPU information and retry.	○					

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7141H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7142H[*1]	Device name error	• A device that cannot be specified for the corresponding command was specified.	PRO	• Check and correct the transmission message from the target device and restart data communications. • If no error is found in the device name, clear the programmable controller CPU information and retry.	○					
7143H[*1]	Device number error	• The head device number is out of range.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○					
7144H[*1]	Monitor registration error	• Monitor was requested before monitor registration.	PRO	• Register the device to be monitored before issuing a monitor request.	○					
7145H[*1]	Monitor PC No. error	• The registered PC No. and PC No. in the monitor request are different.	C/N	• Check and correct the transmission message from the target device and restart data communications. • Perform the monitor registration again.	○					
7146H[*1]	Monitor CPU model error	• The CPU model differs at monitor registration and monitor request.	C/N	• Check and correct the transmission message from the target device and restart data communications. • Perform the monitor registration again.	○					
7147H[*1]	Number of monitor registration point exceeded error	• The number of monitor registration points exceeds the range.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○					
7148H[*1]	Extension R block No. error	• A nonexistent extension file register block No. was designated. • A block No. being used as an extension comment area, sampling trace area, or status latch area was designated.	PRO	• Check and correct the designated block No. and restart data communications.	○					
714AH[*1]	Online disabled	• A write command was designated when the Online Change is disabled. • Parameter or program was written during RUN.	C/N	• Change the setting to enable online change and restart data communications. • Stop the CPU module and restart data communications.	○					
714DH	Use not possible command error	• An unsupported command was requested for the access destination CPU module.	C/N	• Check and correct the transmission message from the target device and restart data communications. • If no error is found in the command, clear the programmable controller CPU information and retry.	○					

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
714EH	Monitor network No. error	• The network No. differs at monitor registration and monitor request.	C/N	• Check and correct the other station access valid module No. in the network parameters, and restart data communications. • Perform the monitor registration again.	○					
7150H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7151H	PC No. designation error	• A value other than "FF" or "0 to 120 (00H to 78H)" was specified for the PC number.	C/N	• Check and correct the transmission message from the target device and restart data communications.	○					
7153H	Frame length error	• The request exceeded number of processing points that can be communicated with one session set for each command.	CHn ERR.	• Modify the setting of the transmission message from the target device so that the request will not exceed the limit of processing points that are allowed for one transmission and predetermined for each MC protocol command. Then, retry data communications.	○					
7155H	Monitor unregistration error	• Monitor was requested before monitor registration.	PRO	• Register the device to be monitored before issuing a monitor request.	○					
7156H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7160H	PLC access error	• Cannot confirm the CPU model.	C/N	• If the CPU module has an error, remove the CPU module error and restart data communications.	○					
7161H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7164H	Request details error	• The request details or device specification method is incorrect.	—	• Check and correct the transmission message/ request from the target device, and restart data communications.	○				○	
7167H	Online disabled	• A write command was designated for a online change prohibit setting.	C/N	• Change the setting to enable online change and restart data communications. • Stop the CPU module and restart data communications.	○					
7168H	Online disabled	• A command that cannot be executed during RUN was specified.	C/N	• Stop the CPU module and restart data communications.						
7169H	CPU module error	• Cannot communicate normally with CPU module.	C/N	• If the CPU module has an error, remove the CPU module error and restart data communications.	○					
716CH	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
716FH	Device error	• A nonexistent device was designated. • Device that cannot be specified by the corresponding command was designated.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○					

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7170H	1C frame error	• The number of access points exceeds the range.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○					
7171H	1C frame error	• A device that cannot be specified was specified.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○					
7172H	1C frame error	• The number of monitor registration points is not correct. For example, "0" is specified.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○	○	○			
7173H	Monitor registration error	• Monitor registration was performed to a CPU module other than R/Q/L/QnACPU using 2C/3C/4C frame.	PRO	• For CPU module other than R/Q/L/QnACPU, use 1C frame for monitor registration and monitor.	○					
7327H	CPU response monitoring timer setting error	• The CPU response monitoring timer value in the buffer memory (address:700EH) is incorrect.	—	• Review the CPU response monitoring timer value.					○	
7330H	Device code error	• The device code value specified as a MODBUS device allocation parameter is incorrect.	—	• Review the device code value.					○	
7331H	MODBUS device upper limit value over error	• The start MODBUS device number + assigned points in the MODBUS device allocation parameter exceeds the maximum value (65535) allowed for the MODBUS device.	—	• Review the start MODBUS device number and the number of assigned points.					○	
7332H	MODBUS device assigned range overlap error	• MODBUS device or CPU device ranges set with the MODBUS device allocation parameters are overlapped.	—	• Review the start MODBUS device number and the number of assigned points.					○	
7333H	Buffer memory assigned range error	• The assigned range of the C24 buffer memory set with the MODBUS device allocation parameter exceeds the range of the user free area.	—	• Review the start device number and the number of assigned points.					○	
7334H	Device upper limit value over error	• The start device number + assigned points in the MODBUS device allocation parameter exceeds the maximum value allowed for the CPU device.	—	• Review the start device number and the number of assigned points.					○	
7335H	Error status read device setting error	• The specification of the error status read device is incorrect.	—	• Review the setting of the error status read device.					○	
7336H	CC-Link IEF remote head module access target value error	• The access target specification value is other than 0 and 1 when CC-Link IE Field network remote head module is installed.	—	• Set the access target specification value to 0 or 1 when CC-Link IE Field network remote head module is installed.					○	

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7337H	CC-Link IEF remote head module access target error	• The access target (when mounted to CC-Link IE Field network remote head module) (address: 700FH) was set when the C24 is not mounted on the CC-Link IE Field network remote head module.	—	• Review the access target station or the specified access target value (when mounted to CC-Link IE Field network remote head module).					○	
7338H	Buffer memory setting error	• Writing to the system area (use prohibited) in the buffer memory.	—	• Check whether writing to the system area (use prohibited) in the buffer memory was executed by the sequence program.					○	
7339H	Modbus station number invalid	• Invalid Modbus station number is set.	—	• Set the valid Modbus station number.					○	
7340H	Device allocation number error	• The bit device assigned to the input register or holding register is not in units of 16 points.	—	• Set the number of assigned points in units of 16 points.					○	
7380H	CPU response monitoring timer timeout	• The CPU response monitoring timer timed out in the slave function.	CHn ERR. C/N	• If an error has occurred in the programmable controller CPU, remove the error. • Set a larger value for the CPU response monitoring timer.					○	
7381H	Function code error	• The request message with a function code that is not supported by the C24 MODBUS slave function was received.	CHn ERR. PRO.	• Confirm the function codes supported by the MODBUS slave function, and review the request message to be sent.					○	
7382H	Sub-code error	• The request message with a sub-code that is not supported by the C24 MODBUS slave function was received.	CHn ERR. PRO.	• Confirm the sub-codes supported by the MODBUS slave function, and review the request message to be sent.					○	
7383H	MODBUS device specification error	• The MODBUS device allocation parameters have not been set for the MODBUS device specified in the received request message. • Accessed a device that the requested CPU module does not support.	CHn ERR. PRO.	• Set the MODBUS device allocation parameters for the MODBUS device specified in the request message. • Review the device code specified in the MODBUS device allocation parameter for the MODBUS device specified in the request message.					○	
7384H	MODBUS device specification error	• The range of the MODBUS device specified in the received request message exceeds the valid range of the MODBUS device allocation parameter. • Accessed a device number that the requested CPU module does not support.	CHn ERR. PRO.	• Set an adequate MODBUS device allocation parameter so that it will satisfy the MODBUS device range specified in the received request message. • Review the assignment of the CPU device set in the MODBUS device allocation parameter for the MODBUS device specified in the request message.					○	
7385H	MODBUS device specification error	• The range of MODBUS devices specified in the received request message exceeds the maximum value of MODBUS devices.	CHn ERR. PRO.	• Check the specification of the MODBUS device on the master side from which the request message was sent.					○	

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7386H	MODBUS device specification error	• The number of access points for the MODBUS device specified in the received request message exceeds the maximum access points allowed for the relevant function.	CHn ERR. PRO.	• Check the specification of the MODBUS device on the master side from which the request message was sent.					○	
7388H	No setting for error status read device	• No error status read device was set for read exception status (FC: 07).	CHn ERR. PRO.	• Set an error status read device.					○	
7390H	Byte count specification error	• The number of write points in the received request message does not match the specified number of bytes.	CHn ERR. PRO.	• Review the number of write points and the number of bytes on the master side from which the request message was sent.					○	
7391H	Receive data size error	• The write device data size in the received request message is not consistent with the specified number of bytes.	CHn ERR. PRO.	• Review the specified contents of the write device data size and number of bytes on the master side that sent the request message.					○	
7392H	Reference type error	• The reference number value specified in the received request message (FC:20, FC: 21) is incorrect.	CHn ERR. PRO.	• Review the specification of the reference number on the master side from which the request message was sent.					○	
7393H	Data unit error	• The contents of the data unit in the received request message are incorrect. • The size of the received request message is smaller than the minimum size or greater than the maximum size required for the relevant function code.	CHn ERR. PRO.	• Review the contents of the request message on the master side from which the request message was sent.					○	
7394H	Write error during online change	• A request message for writing was received in the state of write prohibited during online change.	C/N	• Do not issue a write request message while writing is prohibited during online change. • Use the parameter to change to the setting for write during online change.					○	
7397H	No reception monitoring time timeout	• No reception for a 1.5 character time or 1 second or more was detected during message reception, and the message was discarded.	CHn ERR. PRO.	• Review the setting of the device from which the relevant message was sent. • Check the relevant device. • Disconnect an erroneous device if any.					○	
7398H	Short frame error	• The received message size (excluding the start character in the ASCII mode) was less than 4 or 8 bytes.	CHn ERR. PRO.	• Review the setting of the device from which the relevant message was sent. • Check the relevant device. • Disconnect an erroneous device if any.					○	
7399H	Character overrun error	• The received message size (excluding the start character in the ASCII mode) exceeded 256 or 512 bytes.	CHn ERR. PRO.	• Review the contents of the message issued by the station that sent the relevant message. • Check the relevant device. • Disconnect an erroneous device if any.					○	

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
739AH	ASCII-binary conversion error	• An ASCII code that cannot be converted to binary was received.	CHn ERR. PRO.	• Review the contents of the message issued by the station that sent the relevant message. • Check the relevant device. • Disconnect an erroneous device if any.					○	
739BH	End code error	• An illegal character was received after the end code CR.	CHn ERR. PRO.	• Review the contents of the message issued by the station that sent the relevant message. • Check the relevant device. • Disconnect an erroneous device if any.					○	
7403H	CS signal OFF	• The CS signal was OFF at the time of request or response message transmission, resulting in the failure of the transmission. • A cable is disconnected. • A cable is faulty.	CHn ERR. PRO.	• Check if cable is disconnected. • When using RJ71C24 and RJ71C24-R2 please check the cable connection and correct the wiring so that the CS signal of RS-232 is always ON.					○	
7411H	CRC/LRC error	• The CRC/LRC in the received message does not match the CRC/LRC calculated by the C24.	CHn ERR. P/S	• Do not turn OFF or disconnect the device from the network while it is sending a message. • Review the contents of the message issued by the relevant station. • Check the relevant device. • Disconnect the erroneous device if any. • Review the line status. • Take preventive measures against noise.					○	
7412H	Transmission monitoring timer timeout	• The transmission monitoring timer timed out.	CHn ERR.	• Check if cable is disconnected. • When using RJ71C24 and RJ71C24-R2 please check the cable connection and correct the wiring so that the CS signal of RS-232 is always ON. • Check that no data is being received from the line when transmitting					○	
7817H	Remote password error	C24 is in the remote password locked out state.	—	• Wait for awhile, and retransmit the password.	○					
7A00H	Time-out error	Transmission has failed due to resend timeout.	—	• Check the cable connection to the communication destination. • Check the settings for the communication destination. • If an error has occurred at C24, refer to the manual for C24 to check the error, and take corrective actions according to the error code. • Check if any errors occur on the communication destination.						○

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7A01H	Transmission error	Transmission has failed.	—	• Check the cable connection to the communication destination. • Check the settings for the communication destination. • If an error has occurred at C24, refer to the manual for C24 to check the error, and take corrective actions according to the error code. • Check if any errors occur on the communication destination.						○
7A02H	CPU response monitoring timer error	The request to the CPU module has failed.	—	• Review the monitoring time for the CPU response monitoring timer. • Review the device/label access service processing setting.						○
7A03H	Abnormal response error	An abnormal response was received from the communication destination.	—	• Check the exception code in the buffer memory.						○
7A04H	Received frame error	An incorrect frame was received from the communication destination.	—	• Check the cable connection to the communication destination. • Check the settings for the communication destination. • Take preventive measures against noise. • Check if any errors occur on the communication destination.						○
7A05H	Parameter setting error	The parameter settings are incorrect.	—	• Check the "Various control specification" in "Basic Settings".						○
7A06H	Own station channel error	The channel being used does not support the simple CPU communication.	—	• Available channel number depends on the firmware version of C24. Check the firmware version.						○
7A07H	Device specification error	The device specified as the communication destination is out of specification range.	—	• Check the type, start number and end number of the destination device specified as the communication destination in parameter setting.						○
7A0FH	Simple CPU communication error	The simple CPU communication cannot be executed.	—	• The type of the communication destination depends on the firmware version of C24. Check the firmware version. • Take preventive measures against noise. • If the same error code is displayed even after executing again, please consult your local Mitsubishi representative.						○
7D00H	Protocol No. setting error	• The protocol number is out of range in the control data for CPRTCL instruction.	CHn ERR.	• Review the protocol number.				○		

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7D01H	Protocol continuous execution count setting error	• The number of protocols to be executed continuously is out of range in an argument of the CPRTCL instruction.	CHn ERR.	• Review the number of protocols to be executed continuously.				○		
7D02H	Protocol not-ready error	• The CPRTCL instruction was executed while the 'predefined protocol ready' (X1D) is OFF. • The CPRTCL instruction was executed while the protocol setting data is being written. • The CPRTCL instruction was executed with abnormal protocol setting data.	CHn ERR.	• Execute the CPRTCL instruction after the 'predefined protocol ready' (X1D) turned ON. • Do not execute the CPRTCL instruction while the protocol setting data is being written. (Set the CPU module in STOP state, and write protocol setting data.) • Write the protocol setting data again to C24, and then execute the CPRTCL instruction. • If this error persists even after rewriting, replace the module.				○		
7D10H	Protocol unregistration error	• The protocol No. which is not registered in C24 is specified in the control data for the CPRTCL instruction. • The CPRTCL instruction was executed without the protocol setting data written.	CHn ERR.	• Check if the specified protocol No. is correct. • Check the 'Protocol registration' (Un\G16529 to 16536) to see if the specified protocol number is registered. • Write the protocol setting data, and then execute the CPRTCL instruction.				○		
7D11H	Protocol simultaneous execution error	• The CPRTCL instruction and 'CH1 protocol execution request' (Y3) or 'CH2 protocol execution request' (YA) were executed simultaneously on the same channel.	CHn ERR.	• Correct the program not to execute the CPRTCL instruction and 'CH1 protocol execution request' (Y3)/'CH2 protocol execution request' (YA) simultaneously on the same channel.				○		
7D12H	Send monitoring time timeout error	• Send monitoring time has timed out. • Although the transmission was retried the specified number of retries, it did not succeed.	CHn ERR.	• Check if the transmission is interrupted due to some reasons such as DTR control. • Check whether the CS signal is ON. • Check if the cables are disconnected.				○		
7D13H	Receive wait timeout error	• Receive wait time has timed out.	CHn ERR.	• Check if the reception is interrupted due to some reasons such as DTR control. • Check if the cables are disconnected. • Check if any errors occur on the target device side. ■Using the circuit trace function (send/receive data monitoring function), check if; • Transmissions from the target device are interrupted. • Any data are missing due to a data reception error. • The data (packet) sent from the target device is incorrect.				○		

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7D14H	Response monitoring timeout error	• Response monitoring time (timer 1) has timed out during data transmission. • Data to be sent to the target device was not obtained from the CPU module within the response monitoring time (timer 1).	CHn ERR.	• Increase the setting value for the response monitoring time (timer 1). (Default: 5 seconds) • If an error occurred on the CPU module, refer to the following manual. (📖MELSEC iQ-R CPU Module User's Manual (Application))				○		
7D15H	Response monitoring timeout error	• Response monitoring time (timer 1) has timed out during data reception. • Data received from the target device was not stored into the CPU module within the response monitoring time (timer 1).	CHn ERR.	• Increase the setting value for the response monitoring time (timer 1). (Default: 5 seconds) • If an error occurred on the CPU module, refer to the following manual. (📖MELSEC iQ-R CPU Module User's Manual (Application))				○		
7D16H	Protocol cancel request error	• A cancel request was accepted during protocol execution, and the CPRTCL instruction was completed abnormally. • The CPRTCL instruction was completed abnormally because protocol setting data were written during CPRTCL instruction execution. • The STOP → RUN operation was performed on the CPU module.	CHn ERR.	• Check the cancelled Predefined protocol in the CPRTCL instruction's control data (number of executions), and eliminate the cause of the cancellation. • Clear the error information by the 'Error initialization request' (YE) as necessary. • To prevent coincidental execution of the CPRTCL instruction, set the CPU module in the STOP state before writing the protocol setting data. • Execute the CPRTCL instruction again.				○		
7D17H	Packet size error	• A packet of more than 2048 bytes was received.	CHn ERR.	• Check the data length of the packet sent from the target device. • Divide the packet data into several parts and send them separately.				○		
7D18H	Insufficient digit number error	• When data is received using a protocol including a packet that contains Conversion variable (Variable number of data), the data (the number of digits) corresponding to the Conversion Variable is insufficient.	CHn ERR.	• Review data sent from the target device. • Check the packet format of the target device to see if the number of digits is set correctly.				○		
7D19H	Abnormal digit number error	• When data is received using a protocol including a packet that contains Conversion Variable (Fixed number of data and variable number of digits), the corresponding data is 0 bytes (0 digits) or exceeds the maximum number of digits.	CHn ERR.	• Review data sent from the target device. • Check the packet format of the target device to see if the number of digits is set correctly.				○		

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7D1AH	Data length error	• In the data received from the target device, the Length value does not match the data length of the Conversion Variable(s).	CHn ERR.	• Check the packet format of the target device to see if the conversion variable is set correctly. ■Check the data sent from the target device to see if; • The Length value is incorrect. • Any of the data corresponding to the conversion variable is missing.				○		
7D1BH	Value range error	• When data is received using a protocol including a packet that contains Conversion Variable, data for the Conversion Variable exceeds the range of value allowed for C24.	CHn ERR.	• Review data sent from the target device. ■Check the packet format of the target device to see if the conversion size is set incorrectly. • If it exceeds the range for word data, change the conversion size setting to double word. • If it exceeds the range for double word data, change the element to Non-conversion Variable.				○		
7D20H	Data length size error Data count size error	• The value set in the data length storage area or the data count storage area is out of range.	CHn ERR.	• Check the maximum data length can be set to the data length storage area, and specify the value which is equal to the maximum data length or less. • Check the maximum number of data can be set to the data count storage area, and specify the value which is equal to the maximum number of data or less.				○		
7D21H	Decimal point place specification error	• An out-of-range decimal point position is set when the number of decimals is set to Variable point. • The number of decimals is larger than the number of digits for each data.	CHn ERR.	• Correct the decimal point position. • Check the number of digits setting, and set the decimal point position so that the number of decimals is less than the number of digits.				○		
7E01H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7E03H to 7E05H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7E07H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7E08H	Execution currently disabled error	• The UINI instruction was executed during execution of the programmable controller CPU information clear.	—	• Execute the UINI instruction while the programmable controller CPU information clear is not executed.	○	○	○	○		
7E09H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7E0AH	Transmission setting error	• The specification details for the transmission settings, communication protocol settings and station number settings were incorrect.	—	• Review the parameter settings. • When mode switching is executed, check that the specified contents are correct. • If the problem is still not solved, please consult your local Mitsubishi representative.	○	○	○	○	○	○
7E40H	Command error	• A nonexistent command or subcommand was designated.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○					
7E41H	Data length error	• A data length exceeding the number of points that can communicate during random read/write was designated.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○					
7E42H	Data count error	• Number of processing points exceeds the range allowed for the command.	PRO	• Modify the setting of the transmission message from the target device so that the request will not exceed the limit of processing points that are allowed for one transmission and predetermined for each MC protocol command. Then, retry data communications.	○					
7E43H	Device error	• A nonexistent device was designated. • Device that cannot be specified by the corresponding command was designated. • The start device number is out of range.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○	○				
7E44H	Timer 1 time-out error	• A response message was not returned within the response monitoring time (timer 1).	C/N	• Increase the monitoring time for the response monitoring time (timer 1). • When accessing other station's CPU, check if the routing parameters are correct.	○					
7E47H	Continuous request error	• The next request was received before a response message was returned for the preceding request.	PRO	• Do not send requests continuously from the target device. • Adjust the monitoring time for the response monitoring time (timer 1) to the timeout time set for the target device side.	○				○	
7E48H	Request error	• A request was received during parameter communication.	PRO	• Wait for awhile and transmit the request again.	○				○	
7E4FH	Number of device points error	• Number of access points is incorrect.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○	○				

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7E50H	User frame No. designation error	• A user frame No. outside the range was designated.	PRO	• Check and correct the transmission message from the target device and restart data communications. • Review the frame No.	○	○				
7E51H	User frame unregistration error	• An unregistered user frame number was designated.	PRO	• Perform the frame registration beforehand. • Change the frame No. and restart data communications. • Review the frame No.	○	○				
7E52H	User frame overwrite error	• An overwrite request was issued to the frame number which has been already registered.	PRO	• Change the write destination to an unregistered frame number. • When overwriting, delete the given number beforehand.	○					
7E53H	Requested frame data error	• A nonexistent command was designated in a subcommand. • Byte count exceeding the number of bytes that can be requested was designated.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○					
7E54H	Modification not-allowed error	• The setting change enable/disable setting is set to be disabled.	PRO	• Set the setting change enable/disable setting to 'Enable'.	○	○	○	○	○	○
7E55H	User frame data error	• The user frame variable data has an error.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○	○				
7E57H	Flash ROM write error	• Writing to the flash ROM was not performed normally.	—	• Rewrite the module extended parameter to the CPU module or C24. • Rewrite the predefined protocol setting to the CPU module or C24. • If the error occurs after rewriting, replace the module.	○	○	○	○		○
		• Power is turned OFF during writing to Flash ROM.	CHn ERR.							
7E58H	Mode modification error	• At the time of mode modification, the mode No. or transmission specifications are incorrect.	PRO	• Check and correct the transmission message from the target device and restart data communications. • Review the parameter settings.	○	○	○	○	○	○
7E59H	Number of writing to flash ROM exceeded error	• Writing to flash ROM has been performed 1000 times since the power was turned ON. • Writing to flash ROM has been performed 100,000 times.	CHn ERR.	• Review the program. • Write "0" to the flash ROM write count area (address: 2401H) and restart data communication. • When the number of writes reached 100,000, replace the module.	○	○	○	○		○
7E5EH	PLC CPU monitoring function registration error	• The programmable controller CPU monitoring function is registered for the communication protocol that does not support the function.	CHn ERR.	• Correct the communication protocol setting.			○	○	○	○

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7E60H	Device monitoring duplicate registration error	• Programmable controller CPU monitoring registration was performed twice.	—	• Cancel the programmable controller CPU monitoring and perform the programmable controller CPU monitoring registration again.	○	○				
7E61H	Cycle time units range error	• The setting value for the cycle time units is out of range.	—	• Correct the cycle time units setting value.	○	○				
7E62H	PLC CPU monitoring function error	• The setting value for the programmable controller CPU monitoring function is out of range.	—	• Correct the programmable controller CPU monitoring function setting value.	○	○				
7E63H	Transmission method setting error	• The transmission method setting for programmable controller CPU monitoring is incorrect.	—	• Correct the transmission method setting value.	○	○				
7E64H	Registration point range error	• The total number of word blocks and bit blocks registered for programmable controller CPU monitoring is out of range.	—	• Correct the setting value so that the total number of registered word blocks and bit blocks is within 1 to 10.	○	○				
7E66H	Monitoring condition error of monitoring device	• The monitoring condition is out of range.	—	• Correct the monitoring condition.	○	○				
7E67H	CPU error monitoring setting value error	• The setting value (fixed value) for CPU error monitoring is out of range.	—	• Correct the setting value (fixed) for CPU error monitoring.	○	○				
7E68H	Transmission condition range error of monitoring device	• The setting value for the transmission condition (condition match send) is out of range.	—	• Correct the setting value for the transmission condition (condition match send).	○	○				
7E69H	CPU error monitoring setting value error	• The setting value (fixed value) for CPU error monitoring is out of range.	—	• Correct the setting value (fixed) for CPU error monitoring.	○	○				
7E6AH	Monitoring device read point range error	• The read point of the monitoring device is out of range.	—	• Correct the setting value for the monitoring device read point.	○	○				
7E6BH to 7E6DH	CPU error monitoring setting value error	• The setting value (fixed value) for CPU error monitoring is out of range.	—	• Correct the setting value (fixed) for CPU error monitoring.	○	○				
7E6FH	Modem connection error	• Modem is connected to an incorrect channel.	—	• Correct the destination to which data is sent with the modem function.	○	○				
7E70H	CPU error	• The communication with the CPU module was not performed normally.	C/N	• If the CPU module side has an error, remove the error and restart data communications.	○	○		○	○	○
7E71H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7EC1H to 7EC2H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7EC3H	Double send request error	• A send request was sent again during processing of the send request.	—	• Provide an interlock for send requests.		○	○			

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7EC4H	Transmission data count error, Send and receive buffer setting error	• Data exceeding the send buffer memory size was transmitted. • Send/receive buffer settings are out of range.	—	• Make the send data quantity smaller than the buffer memory size. • Increase the value for the send buffer memory length designation. • Designate the start address and size of transmit/receive buffer within the range of the user setting area.	○	○	○			
7EC5H	Flash ROM access error	• Writing to flash ROM was requested when the setting change enable/disable setting is set to be disabled with Engineering tool.	—	• Set the setting change enable/disable setting to 'Enable' with Engineering tool, write to the CPU, and re-execute after the CPU reset or powering OFF → ON.	○	○	○	○		○
7EC6H	Flash ROM access error	• The flash ROM read/write request contents are incorrect.	—	• Check the data at the time of read/write request, and specify the correct data.	○	○	○	○		○
7F00H	Other module response error	• An error response has been returned from CPU module.	—	• An error code is included in the response message of MC protocol. Refer to the manual for the relevant CPU module to check the error, and review the request message of MC protocol. • If this error has occurred at the execution of G(P).CPRTCL instruction, refer to the manual for the relevant CPU module to check the error, and review the CPU device number specified in the data storage area for Non-conversion Variable or Conversion Variable. • If this error occurs at the using the MODBUS slave function, refer to error log (buffer memory:7CFEH to 7DFFH),and review the MODBUS request message or MODBUS device allocation parameter.	○			○	○	○
7F01H	Other module response error	• An error response has been returned from a module other than CPU module.	—	• An error code is included in the response message of MC protocol. Refer to the manual for the relevant module to check the error, and review the request message of MC protocol. • If this error occurs at the using the MODBUS slave function, refer to error log (buffer memory:7CFEH to 7DFFH),and review the MODBUS request message or MODBUS device allocation parameter.	○				○	○

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7F02H	CPU module error	• An error was detected in the CPU module.	—	• Check the error of the CPU module and take corrective action using the module diagnostics.	○			○		
7F03H to 7F06H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7F07H	CPU module error	• An error was detected in the CPU module.	—	• Check the error of the CPU module and take corrective action using the module diagnostics.						
7F20H	ASCII→BIN conversion error	• When the nonprocedural or bidirectional protocol is used, an ASCII code that cannot be converted to binary code was received. • When the nonprocedural or bidirectional protocol is used, an odd number of bytes of data was received during data communications using ASCII ↔ BIN conversion.	PRO	• Check and correct the transmission message from the target device and restart data communications. • When the nonprocedural or bidirectional protocol is used and data is communicated using ASCII ↔ BIN conversion, be sure to send data in units of even bytes.		○	○			
		• When data is received by Conversion Variable of the predefined protocol function, the data corresponding to the Conversion Variable cannot be converted to binary numbers. • When data is received by a Check code (ASCII hexadecimal or ASCII decimal) of the predefined protocol function, the data corresponding to the check code cannot be converted to binary numbers.	PRO	■If the predefined protocol was edited, check if the following values match the packet format of the target device: • Conversion, sign character, number of decimals, delimiter, number of digit for conversion variable • Code type and data length of check code				○		
7F21H	Receive header area error	• The command (frame) part designation has an error. • An ASCII code that cannot be converted to binary code was received.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○					
7F22H	Command error	• A nonexistent command, subcommand, or device was designated. • The remote password length is incorrect.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○					
7F23H	MC protocol message error	• There is no data (ETX, CR-LF, etc.) following the character area, or the designation was incorrect.	PRO	• Check and correct the transmission message from the target device and restart data communications.	○					

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7F24H	Sum check error	• The sum check code calculated does not match the sum check code received. • The horizontal parity code calculated does not match the horizontal parity code received. • The calculated check code does not match the received check code.	P/S	• Review the sum check code of the target device. • Review the horizontal parity code of the target device. • Review CRC-16 of the target device. ■If the predefined protocol was edited, check if the following values match the packet format of the target device: • Processing method or code type of the check code • Data length • Data flow • Complement calculation • Calculating range	○	○		○		○
7F25H	Data length error	• The length of the received data exceeds the receive area size.	CHn ERR.	• Correct the "data length" that is transmitted from the target device. • Change the word/byte units designation to 'byte units'. • Increase the value for the receive buffer memory length designation.			○			
7F26H	Command error	• Another command was received before the unlock processing when the remote password is registered.	CHn ERR.	• Perform data communications after the unlock processing is normally completed.	○					
7F30H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7F31H	Simultaneous transmission error	• C24 and the target device started transmissions simultaneously.	—	• Perform processing according to the predefined arrangement with the target device. If necessary, change the 'simultaneous transmission data valid/ invalid designation' (Un\G155/315) setting.			○			
7F32H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7F40H	Time-out error	• No-reception monitoring time (timer 0) has timed out.	PRO	• Check if any data is missing due to a data reception error. • Check if the DTR control, etc. interrupted data reception.	○	○	○			
7F41H	Time-out error	• Response monitoring time (timer 1) has timed out.	—	• Check the status of the target device, and restart data communications, as required.	○		○			
7F42H	Time-out error	• Send monitoring time (timer 2) has timed out.	—	• Check if the DTR control, etc. interrupted data transmission.	○	○	○			
7F50H to 7F54H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						

<table>
<tr><th rowspan="2">Error code (hexadecimal)</th><th rowspan="2">Error</th><th rowspan="2">Description</th><th rowspan="2">Signal name</th><th rowspan="2">Corrective action</th><th colspan="6">Protocol</th></tr>
<tr><th>MC</th><th>Non</th><th>Bi</th><th>Pd</th><th>MD</th><th>S</th></tr>
<tr><td>7F67H</td><td>Overrun error</td><td>• C24 received the next data before it completed receive processing of the previous data.</td><td>SIO</td><td>• Decrease the transmission speed and restart data communications.
• Check whether momentary power failure has occurred on the station with C24. (can be checked "AC DOWN counter" of the special register.)
In case momentary power failure has occurred, remove the cause.
• Perform the registration, read, and delete to/from the flash ROM while the data communication is not performed.</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td></tr>
<tr><td>7F68H</td><td>Framing error</td><td>• The stop bit settings do not match.
• A malfunction occurred on the line because the power for the target station was ON/OFF.
• Noise occurs on the line.
• During multidrop connection, two or more devices transmitted data at the same time.</td><td>SIO</td><td>• Match the settings of C24 with that of the target device.
• Clear the error information by 'Error initialization request' (YE).
In addition, when communicating data with the nonprocedural protocol, execute the INPUT instruction, and read and discard the unnecessary data.
• Take measures against noise.
• During multidrop connection, provide the interlock so that data will not be transmitted simultaneously by multiple devices.</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td></tr>
<tr><td>7F69H</td><td>Parity error</td><td>• The parity bit settings do not match.
• A malfunction occurred on the line because the power for the target station was ON/OFF.
• Noise occurs on the line.
• During multidrop connection, two or more devices transmitted data at the same time.</td><td>P/S</td><td>• Match the settings of C24 with that of the target device.
• Clear the error information by 'Error initialization request' (YE).
In addition, when communicating data with the nonprocedural protocol, execute the INPUT instruction, and read and discard the unnecessary data.
• Take measures against noise.
• During multidrop connection, provide the interlock so that data will not be transmitted simultaneously by multiple devices.</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td></tr>
</table>

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7F6AH	Buffer full error	• The operating system receive buffer overflowed and received data was skipped.	SIO	• Use the DTR and DC controls and perform data communications so that the buffer does not become full. (Perform RS/CS control when the modem function is used.) • When the reception data read request signal turns ON, execute read operation immediately.	○	○	○	○	○	○
7F6BH	CD(DCD) signal control error	• Data was received when the CD(DCD) signal is OFF with the CD terminal check enabled.	CHn ERR.	• Review the CD(DCD) signal control of the target device. (Data is sent when the CD(DCD) signal is ON.) • Perform data communications with the CD terminal check disabled.	○	○	○	○		○
7F6CH	Transmission error	• Data cannot be sent because the line is not connected.	CHn ERR.	• Send data after line connection processing on the interface side that uses the modem function.	○	○	○			
7F9AH	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7FABH	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7FC6H	System error	• An error was detected in the CPU built-in memory.	—	☞ Page 541 Corrective actions for system error						
7FC7H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7FC9H to 7FCCH	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7FCFH	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7FD1H	Modem function error	• The connection re-try count designation is incorrect.	CHn ERR.	• Set the connection re-try count to 1 to 5.	○	○	○			
7FD2H	Modem function error	• The connection retry interval designation is incorrect.	CHn ERR.	• Set the connection retry interval to 90 to 300 (seconds).	○	○	○			
7FD3H	Modem function error	• The initialization/ connection timeout designation is incorrect.	CHn ERR.	• Set the time for the initialization/connection timeout to 1 to 60 (seconds).	○	○	○			
7FD4H	Modem function error	• The number of initialization retries designation is incorrect.	CHn ERR.	• Set the number of initialization retires to 1 to 5.	○	○	○			
7FD5H	Modem function error	• The data No. for initialization designation is incorrect.	CHn ERR.	• Designate the registered number of data for initialization or '0'.	○	○	○			
7FD6H	Modern function error	• The data for connection of the number specified in the data No. for connection designation is incorrect.	CHn ERR.	• Review the line type.	○	○	○			
7FD7H	Modem function error	• The data for connection of the number specified in the data No. for connection designation is incorrect.	CHn ERR.	• Review the external line outgoing number.	○	○	○			
7FD8H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7FD9H	Modern function error	• The line is connected to the target, or the modem has not been initialized.	CHn ERR.	• Upon the completion of the modem initialization, connect the line to the target device (via public line) while the line has not been connected yet.	○	○	○			
7FDAH	Modem function error	• The data No. for connection designation is incorrect.	CHn ERR.	• Designate the registered number of data for connection.	○	○	○			
7FDBH	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7FDCH	Modem function error	• The data for connection of the number specified in the data No. for connection designation is incorrect.	CHn ERR.	• Review the data for connection.	○	○	○			
7FDDH	Modem function error	• The connection timeout error occurred.	CHn ERR.	• Review the telephone number in the data for connection. • Check whether the connection target is connectable. • Review the setting value for the connection timeout. • Review the data for initialization.	○	○	○			
7FDEH	Modem function error	• Connection to the modem/TA failed, or a modem/TA is not connected.	CHn ERR.	• Check if the modem/TA has any failure by referring to the instruction manual for the modem/TA. • Connect the modem/TA to C24. • Check the connection cable by referring to the instruction manual for the modem/TA. • Check for any abnormality according to the following procedure. (☞ Page 541 Corrective actions for system error)	○	○	○			
7FDFH	Modem function error	• The modem connection channel No. has not been set.	CHn ERR.	• Set the modem connection CH specification to 1 or 2.	○	○	○			
7FE0H	Modem function error	• The designated value for modem connection channel is incorrect.	—	• Set the modem connection CH specification to 1 or 2.	○	○	○			
7FE1H	Modem function error	• The user frame No. designation is incorrect. • The output start pointer designation is incorrect.	CHn ERR.	• Review the user frame No. • Designate the registered user frame No. • Register the designated data for initialization. • Set the output start pointer designation to 1 to 100. • Designate the position (n-th) to which the number of data for initialization is designated. • Designate the number of data for initialization to the designated position (n-th).	○	○	○			

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7FE2H	Modem function error	• The output frame No. designation is incorrect.	CHn ERR.	• Designate the registered number of data for initialization to the range of the designated output quantity starting from the position of the output start pointer designation.	○	○	○			
7FE3H	Modem function error	• The registration data byte count designation is incorrect.	CHn ERR.	• Set the data for initialization to 1 to 78. • Set the data for connection to 80.	○	○	○			
7FE4H	Modern function error	• The data for connection of the number specified in the data No. for connection designation is incorrect.	CHn ERR.	• Review the data for connection.	○	○	○			
7FE5H	Modem function error	• The no-communication interval time designation is incorrect.	CHn ERR.	• Set the no-communication interval time designation to 0 to 120.	○	○	○			
7FE6H	Remote password error	• Remote passwords do not match.	—	• Review the remote password and restart data communications.	○					
7FE7H	Remote password error	• The remote password is not registered. • The remote password processing command was received on the interface side where the modem function is not used. • The MELSOFT Connection is selected in the Communication protocol setting for the corresponding channel.	—	• A remote password processing command is not necessary when a remote password is not registered. • Unlock the remote password by MELSOFT. Alternatively, change the Communication protocol setting for the corresponding channel to MC protocol.	○					
7FE8H	Remote password error	• The number of remote password mismatch exceeded the designated value of the address 200DH.	CHn ERR.	• Clear the 'accumulated count of unlock process abnormal completion' (Un\G8956) to '0'.	○					
7FE9H	Modem function error	• The number of callback denial exceeded the designated value of address 2002H.	CHn ERR.	• Clear the callback denial accumulated count (address: 22F1H) to '0'.	○	○	○			
7FECH	Module extended parameters do not exist.	• Reading data has failed since module extended parameters do not exist.	—	• Write module extended parameters to C24 or CPU module.						
7FEDH	Simultaneous execution error	• More than one target device accessed to the parameter file at the same time.	—	• Do not access to the parameter file from more than one target device simultaneously.						
7FEFH	Parameter setting error	• The parameter settings with MELSOFT are incorrect. • A network module having the firmware version not supporting the Simple PLC communication function is used although the Simple PLC communication is selected as communication protocol setting.	—	• Correct the parameter settings, write it to the CPU module, and restart it. • Use a network module having the firmware version that supports Simple PLC Communication.	○	○	○	○	○	○
7FF0H	Dedicated instructions simultaneous execution error	• Dedicated instructions were executed simultaneously.	—	• Do not execute the dedicated instructions simultaneously.	○	○	○	○		

Error code (hexadecimal)	Error	Description	Signal name	Corrective action	Protocol					
					MC	Non	Bi	Pd	MD	S
7FF1H	Control data error	• Setting value is incorrect. • Setting value is out of the range.	—	• Correct the setting value in the control data.	○	○	○		○	
7FF2H	Communication protocol setting error	• The instruction cannot be executed with the current communication protocol.	—	• Correct the setting value of the communication protocol.	○	○	○	○	○	○
7FF3H to 7FF4H	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7FF5H	Transmission cancel at mode switching/ transmission sequence initialization	■The processing in operation was cancelled due to performance of the following functions: • Mode switching, transmission sequence initialization, receive data clear (by buffer memory), user frame use enable/ disable designation, UINI instruction, programmable controller CPU information clear	—	• Do not execute these functions during various processes in progress.	○	○	○	○	○	○
7FF7H	Simultaneous access error	• Multiple target devices accessed to a station on the same network at the same time.	—	• Do not access to a station on the same network from multiple target device at the same time.	○					
7FFBH	Channel error	• Incorrect channel was designated with a dedicated instruction.	—	• Review the channel for a dedicated instruction.	○	○	○	○		
7FFCH	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						
7FFDH	NULL character transmission error	• Only NULL character data was transmitted when NULL characters automatic deletion is designated.	—	• Review the data to be transmitted.		○	○			
7FFFH	System error	• The operating system of C24 detected some error.	—	☞ Page 541 Corrective actions for system error						

*1 An error code when the target CPU is one of MELSEC-A series.

Corrective actions for system error

1. Check if the C24, the power module, and the CPU module are mounted correctly.

2. Check if the environment of the C24 usage is within the range of the general specifications for the CPU module.

3. Check if the power supply capacity is sufficient.

4. Check if the hardware of the C24 and the CPU module is normal according to the respective module manual. In case of a malfunction, please contact your local Mitsubishi representative for service.

5. If the problem is still not solved, please consult your local Mitsubishi representative, explaining a detailed description of the problem.

29.5 Event History List

The table below shows the events saved by C24.

Event code	Event type	Description	Cause
10200	Security	Remote password lock	The remote password is locked.
10201		Remote password unlock	The remote password is unlocked.
10202		Failure of remote password unlock	The remote password was failed to be unlocked.
27000	Operation	Transmission sequence initialization	The transmission sequence used by the MC protocol was initialized.
27001		Programmable controller CPU information clear	The programmable controller CPU information clear request was executed.
27002		Error clear request	The error clear request was executed.
27003		Mode switching request	The mode switching request was executed.
27004		Modem initialization request	The modem initialization request was executed.
27005		Connection request	The connection request for the modem was executed.
27006		Modem disconnection request	The disconnection request for the modem was executed.
27007		Parameter initialization request	The parameter initialization request was executed.
27008		Connection request from MELSOFT products	By MELSOFT connection, the access via Serial communication module was performed.
27010		Module extended parameter open error	Opening module extended parameter in CPU module has failed.
27011		Module extended parameter read error	Reading module extended parameter in CPU module has failed.

29.6 MODBUS Exception Code List

Exception codes are error codes common to the MODBUS protocol that are stored in a response message when a slave returns an abnormal response to a request message from a master device.

When a target device receives an exception code from a C24, refer to the following table to take a corrective action.

If processing on a C24 is completed abnormally, an exception code is stored in the error log area (Un\G31998 to 32255) of the buffer memory.

The following table lists the exception codes.

Code	Name	Description	Corrective action	
			Target device side	C24 side
01h	Illegal Function	The slave station received an unsupported function code.	Check function codes supported by the C24, and modify the request message to be issued.	—
02h	Illegal Data Address	The specified address of the MODBUS device is erroneous.	Check the MODBUS device type and size supported by the C24, and correct the specified address in the request message to be issued.	—
03h	Illegal Data Value	A value contained in the data field of the request message is incorrect.	Review the data field of the issued request message.	—
04h	Slave Device Failure	An unrecoverable error occurred while the slave station was attempting to perform the requested action.	Review the data field of the issued request message.	Remove the cause of the error occurred on the C24 side. Refer to the error log of the C24, and solve the problem according to the error code.
05h	Acknowledge	As the slave station is executing another processing, a long duration of time is required to complete the requested processing.	Not issued by the C24.	
06h	Slave Device Busy	As the slave station is executing another processing, the requested processing cannot be executed.	Not issued as a response message by the C24.	
07h	NAK Error	The requested program function cannot be executed on a slave station.		
08h	Memory Parity Error	A parity error was detected on a slave station during access to the extension file register.	Not issued by the C24.	
0Ah	Gateway Path Unavailable	The gateway device is not available for use.		
0Bh	Gateway Target Device Failed To Respond	There is no response from the slave station connected ahead of the gateway device.		

APPENDIX

Appendix 1 Module Label

The input/output signal and buffer memory of C24 can be set by using module label.

Module label configuration

A name of module label is defined in the configuration below:

"Instance name"_"Module number"."Channel number"."Label name"

"Instance name"_"Module number"."Label name"

"Instance name"_"Module number"."Channel number"."Label name"_D

"Instance name"_"Module number"."Label name"_D

Ex.

C24_1.stnCH1.uSet_ReceivedDataCount_Reception_D

■Instance name

The instance name for RJ71C24(-R2/R4) is "C24".

■Module number

A module number is a number starting from 1, which is added to identify a module that has the same instance name.

■Channel number

A channel number is a number added to the module label corresponding to a buffer memory other than the CH1/CH2 common area.

For module label corresponding to a buffer memory other than the CH1/CH2 common area, the following content is added as the channel number.

Channel number	Content to be added
CH1	stnCH1
CH2	stnCH2

■Label name

This is a module unique label name.

■_D

This indicates that the module label is for direct access. Without this symbol means a label for refresh. There are some differences between refresh and direct access as shown below.

Type	Description	Access timing
Refresh	Values written to/read from a module label are reflected to the module in batch at the time of refresh. This shortens program execution time.	At the time of refresh
Direct access	Values written to/read from a module label are immediately reflected to the module. Although the program execution time is longer than refresh, the responsiveness will be increased.	At the time of writing to/ reading from module label

Appendix 2 Input/Output Signal List

This section explains the input/output signals of C24.

The input/output signal assignments in the table below are when the start I/O number of C24 is '0000'.

The devices starting with X indicate input signals from a C24 to CPU module.

The devices starting with Y indicate output signals from a CPU module to C24.

Point

Do not output (turn ON) the signals of "Use prohibited" in the input/output signal list shown below.

If any of the "Use prohibited" signals is output, the programmable controller system may malfunction.

Input signals

Device number	Signal name	Reference
X0[*1]	CH1 Transmission normal completion	—
X1[*1]	CH1 Transmission abnormal completion	
X2[*1]	CH1 Transmission processing	
X3[*2]	CH1 Reception data read request	Page 57 Receiving Data from Target Device Page 90 Receiving Data from Target Device
X4[*2]	CH1 Reception abnormal detection	
X5	CH1 Protocol execution completion	—
X6[*3]	CH1 Switching mode	Page 409 SWITCHING THE MODE AFTER STARTING
X7[*1]	CH2 Transmission normal completion	—
X8[*1]	CH2 Transmission abnormal completion	
X9[*1]	CH2 Transmission processing	
XA[*2]	CH2 Reception data read request	Page 57 Receiving Data from Target Device Page 90 Receiving Data from Target Device
XB[*2]	CH2 Reception abnormal detection	
XC	CH2 Protocol execution completion	—
XD[*3]	CH2 Switching mode	Page 409 SWITCHING THE MODE AFTER STARTING
XE[*4]	CH1 Error occurrence	—
XF[*4]	CH2 Error occurrence	
X10	Modem initialization completion	Page 192 List of input/output signals for the modem function
X11	Dialing	
X12	Connection	
X13	Initialization/connection abnormal completion	
X14	Modem disconnection completion	
X15 to X16	Use prohibited	—
X17[*1]	User frame read completion from flash ROM	
X18[*1]	User frame write completion from flash ROM	
X19	Use prohibited	
X1A	CH1 Global signal	MELSEC Communication Protocol Reference Manual
X1B	CH2 Global signal	
X1C	Parameter initialization completion	—
X1D[*5]	Predefined protocol ready	Page 52 Execution Condition of Predefined Protocol Communication
X1E[*6]	Module READY	—
X1F[*7]	Watchdog timer error	

[*1] The device does not turn ON/OFF by executing a dedicated instruction when using a function that corresponds to the input signal.

[*2] The device turns ON/OFF by executing a dedicated instruction when using a function that corresponds to the input signal.
(ON→OFF: Data read completion)

[*3] The 'CH□ Switching mode' (X6/XD) turns ON when the following function is performed.
· Mode switching, transmission sequence initialization, reception data clear (by buffer memory), user frame use enable/disable designation, UINI instruction, programmable controller CPU information clear
While the 'CH□ Switching mode' (X6/XD) is ON, do not issue a communication request to a target interface.
(The transmission/reception processing of C24 is stopped while the 'CH□ Switching mode' (X6/XD) is ON.)

[*4] The C ERR LED of the same channel also turns ON when an error occurs.
'CH□ Error occurrence' (XE/XF) turns OFF when 'Error initialization request' (YE) turns ON or the data transmission/reception processing is performed normally.

[*5] 'Predefined protocol ready' (X1D) is a signal that turns ON when predefined protocol communication becomes ready.

[*6] Module READY is a signal that sends a signal of the applicability of accessing to C24 from the CPU module.
Use it as an interlock signal for a program.

[*7] Restart the CPU module when the watchdog timer error is turned ON.
If the error still occurs after restarting, refer to the following section and take corrective actions.
☞ Page 508 RUN LED turns OFF

Output signals

Device number	Signal name	Reference
Y0	CH1 Transmission request	—
Y1	CH1 Reception data read completion	
Y2	CH1 Mode switching request	Page 409 SWITCHING THE MODE AFTER STARTING
Y3	CH1 protocol execution request	—
Y4 to Y6	Use prohibited	
Y7	CH2 Transmission request	
Y8	CH2 Reception data read completion	
Y9	CH2 Mode switching request	Page 409 SWITCHING THE MODE AFTER STARTING
YA	CH2 protocol execution request	—
YB to YD	Use prohibited	
YE	Error initialization request	Page 492 Turning OFF the ERR LED and initializing the error codes
YF	Use prohibited	—
Y10	Modem initialization request	Page 192 List of input/output signals for the modem function
Y11	Connection request	
Y12	Modem disconnection request	
Y13 to Y16	Use prohibited	—
Y17	User frame read request from flash ROM	
Y18	User frame write request to flash ROM	
Y19 to Y1B	Use prohibited	
Y1C	Parameter initialization request	
Y1D to Y1F	Use prohibited	

Appendix 3 Buffer Memory

The buffer memory is the memory of C24 used for storing data (setting values, monitoring values, etc.) transmitted to or received from the CPU module.

Initial values are set to the buffer memory.

When changing the initial values, set the parameters with Engineering tool.

For details, refer to the following section.

☞ Page 454 PARAMETER SETTING

Configuration of buffer memory

A buffer memory consists of a user area and a system area as shown below.

■User area

- User area is an area where a user can read/write data.
- The user area consists of areas for storing setting values for data communication, for actual data communication, and for storing communication status and communication error information.
- Read/write data from/to the user area in accordance with the instructions in the corresponding reference section.

■System area

System area is an area used by the C24 system.

Buffer memory assignment list

A buffer memory is configured with 16 bits per address.

Names, default values, etc. of each address of the buffer memory are as shown below.

☞ Page 550 Buffer memory list

■Meaning of abbreviations in "Protocol" column

MC:	MC protocol
Non:	Nonprocedural protocol
Bi:	Bidirectional protocol
Pd:	Predefined protocol
MD:	MODBUS (slave function)
S:	Simple CPU communication

■Meaning of symbols in "Protocol" column

The following symbols are assigned to protocols related to the setting values of a corresponding area and to areas used for control by user settings.

R, W: An area where data can be read/written from/to the CPU module and an target device.

R: An area where data can only be read from the CPU module and an target device.

—: System area used by the system or area not used by the corresponding protocol.

Precautions

- Do not write data in the "System area" of the buffer memory.
  If data is written to any of the system areas, the programmable controller system may malfunction.
  The system area exists in the user area partially, therefore, take caution when reading from/writing to the buffer memory.
- To use the dedicated instructions, change the following initial settings (default values) for data communications by setting parameters with an Engineering tool, or by executing the CSET instruction in the program.

Initial setting		Setting item
Communication using MC protocol	Setting the unit of data length sent by the on-demand function	• "Word/byte units designation" (addresses: 96H/136H)
	Setting the buffer memory used by the on-demand function	• "Buffer memory start address designation" (addresses: A0H/140H) • "Send buffer memory length designation" (addresses: A3H/143H)
Communication using nonprocedural/bidirectional protocols	Setting the unit of data length to be sent and received	• "Word/byte units designation" (addresses: 96H/136H)
	Setting the data send area	• "Send buffer memory start address designation" (addresses: A2H/142H) • "Send buffer memory length designation" (addresses: A3H/143H)
	Setting the data receive area	• "Receive buffer memory length designation" (addresses: A7H/147H)

For details on the registration operation by an Engineering tool, refer to the following section.

☞ Page 454 PARAMETER SETTING

For the CSET instruction, refer to the following manual.

📖MELSEC iQ-R Programming Manual (Module Dedicated Instructions)

Buffer memory list

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
0, 1 (0H, 1H)		Use prohibited	System area	—							
2 (2H)		For flash ROM (user frame) access	Register/read/delete instructions 0: No request 1: Registration request 2: Read request 3: Deletion request	0	R, W		—				—
3 (3H)			Frame No. direction 0: No designation Other than 0: Frame number	0							
4 (4H)			Registration/read/delete result storage 0: Normal completion Other than 0: Abnormal completion	0							
5 (5H)			Registration data byte count designation 0: No designation Other than 0: Number of registered data bytes (maximum 80 bytes)	0							
6 to 45 (6H to 2DH)			User frame 0: No designation Other than 0: Registration data (maximum 80 bytes)	0							
46 (2EH)		For designation of the modem function-1	Modem connection CH specification 0: None 1: CH1 2: CH2	0	R, W			—			Page 193 Buffer memory
47 (2FH)		Use prohibited	System area	—							
48 (30H)		For designation of the modem function-1	Connection re-try count designation 1 to 5: Number of retries	3 (3 times)	R, W			—			Page 193 Buffer memory
49 (31H)			Connection retry interval designation 90 to 300: Connection retry interval (unit: s)	180 (180 seconds)							
50 (32H)			Initialization/connection timeout designation 1 to 60: Timeout time (unit: s)	60 (60 seconds)							
51 (33H)			Number of initialization retries designation 1 to 5: Number of retries	3 (3 times)							
52 (34H)			Data No. for initialization designation 0H : Send designated data for initialization by the send user frame designation area. 7D0H to 801FH: Data number for initialization	7D0H							
53 (35H)			Data No. for connection designation 0H: No designation BB8H to 801FH: Data number for connection	0							
54 (36H)		Use prohibited	System area	—							
55 (37H)		For designation of the modem function-1	No-communication interval time designation 0: Infinite wait 1 to 120: No communication interval (line disconnection wait time) (unit: min)	30 (30 minutes)	R, W			—			Page 193 Buffer memory
56 (38H)			RS/CS control enable/disable designation 0: Control disabled 1: Control enabled	1							
57 to 127 (39H to 7FH)		Use prohibited	System area	—							

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
128 (80H)		For programmable controller CPU information clear	Programmable controller CPU information clear request 0: No request 4C43H: Requested	0	R, W	—			R, W	—	Page 498 How to Clear Programmable Controller CPU Information
129 to 143 (81H to 8FH)		Use prohibited	System area	—							
144 (90H)	304 (130H)	For designation of mode switching	Switching mode No. designation 0000H: Match the setting in MELSOFT. 0001H: MC protocol (Format 1) 0002H: MC protocol (Format 2) 0003H: MC protocol (Format 3) 0004H: MC protocol (Format 4) 0005H: MC protocol (Format 5) 0006H: Nonprocedural protocol 0007H: Bidirectional protocol 0009H: Predefined protocol 000AH: MODBUS slave (RTU) 000BH: MODBUS slave (ASCII) 00FFH: MELSOFT connection	0	R, W					—	Page 413 I/O Signals for Handshake with CPU Module and Buffer Memory
145 (91H)	305 (131H)	For designation of mode switching	Transmission specifications after switching designation Designates transmission specifications after switching when b15 of this area is 1 (ON). • Operation setting (b0) 0: Independent 1: Interlink • Data bit (b1) 0: 7 bits 1: 8 bits For the MODBUS slave (RTU), specify '1: 8 bits.' • Parity bit (b2) 0: None 1: Yes • Odd/even parity (b3) 0: Odd 1: Even • Stop bit (b4) 0: 1 bit 1: 2 bits • Sum check code (b5) 0: None 1: Yes • Online change (b6) 0: Disable 1: Enable • Setting change (b7) 0: Disable 1: Enable • Communication speed (b8 to b11) 1200 bps to 230400 bps • For system (b12 to b14) All 0 • Transmission specifications direction after switching (b15) 0: Match settings in MELSOFT 1: Match settings in this area	0							
146 (92H)	306 (132H)	Signal specification	RS and DTR signal status designation 0: OFF 1: ON RS (RTS) signal (b0) ER (DTR) signal (b2) For system (b1), (b3) to (b15)	5H	R, W					—	MELSEC iQ-R Serial Communication Module User's Manual(Startup)

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
147 (93H)	307 (133H)	For designation of transmission control	DTR/DSR (ER/DR), DC control designation • Transmission control (b0) 0: DTR/DSR control 1: DC code control • DC1/DC3 control (b8) 0: Control disabled 1: Control enabled • DC2/DC4 control (b9) 0: Control disabled 1: Control enabled	0	R, W				—		Page 268 DATA COMMUNICATIONS USING DC CODE TRANSMISSION CONTROL
148 (94H)	308 (134H)		DC1/DC3 (Xon/Xoff) code designation • DC1 code (b0 to b7) 00H to FFH: DC1 code • DC3 code (b8 to b15) 00H to FFH: DC3 code	1311H							
149 (95H)	309 (135H)		DC2/DC4 code designation • DC2 code (b0 to b7) 00H to FFH: DC2 code • DC4 code (b8 to b15) 00H to FFH: DC4 code	1412H							
150 (96H)	310 (136H)	For designation of communication control	Word/byte units designation 0: Word units 1: Byte units	0	R, W			—	—		Page 257 CHANGING DATA LENGTH UNITS TO BYTE UNITS (WORD/ BYTE UNITS SETTING)
151 (97H)	311 (137H)		CD terminal check designation (for RS-232) 0: Check 1: Do not check	1	R, W				—		MELSEC iQ-R Serial Communication Module User's Manual(Startup)
152 (98H)	312 (138H)		Communication method designation (for RS-232) 0: Full-duplex communication 1: Half-duplex communication	0							Page 275 DATA COMMUNICATIONS USING HALF-DUPLEX COMMUNICATIONS
153 (99H)	313 (139H)	For designation of half-duplex communications control (for RS-232)	Simultaneous transmission priority/non-priority designation 0: Priority 1 to 255: Non-priority (transmission wait time, unit: 100 ms)	0	R, W				—		Page 275 DATA COMMUNICATIONS USING HALF-DUPLEX COMMUNICATIONS
154 (9AH)	314 (13AH)		Retransmission time transmission method designation 0: Do not resend 1: Resend	0							
155 (9BH)	315 (13BH)	For designation of communication control	Simultaneous transmission data valid/invalid designation • Receive data enable/disable (b0) 0: Enable 1: Disable • Send data enable/disable (b8) 0: Enable 1: Disable	0	—		R, W	—			

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
156 (9CH)	316 (13CH)	For designation of data communication time monitoring	No-reception monitoring time (timer 0) designation 0H: Unlimited wait 28H to FA0H: Monitoring time (unit: byte)	0	R, W			—			Page 259 CHANGING DATA COMMUNICATION MONITORING TIME
157 (9DH)	317 (13DH)		Response monitoring time (timer 1) designation 0H: Unlimited wait 1H to BB8H: Monitoring time (unit: 100ms)	32H (5 seconds)	R, W	—	R, W		—		
158 (9EH)	318 (13EH)		Send monitoring time (timer 2) designation 0H: Unlimited wait 1H to BB8H: Monitoring time (unit: 100ms)	708H (3 minutes)	R, W			—			
159 (9FH)	319 (13FH)	Use prohibited	System area	—							
160 (A0H)	320 (140H)	For designation of on-demand function	Buffer memory start address designation (400H to 1AFFH, 2600H to 3FFFH)	CH1:400H CH2:800H	R, W	—					Page 299 ON-DEMAND DATA COMMUNICATIONS USING USER FRAMES MELSEC Communication Protocol Reference Manual
161 (A1H)	321 (141H)		Data length designation (0000H to 3400H)	0							
162 (A2H)	322 (142H)	For designation of send area	Send buffer memory start address designation (400H to 1AFFH, 2600H to 3FFFH)	CH1: 400H CH2: 800H	—	R, W		—			Page 422 Send/receive Data Monitoring Function Settings
163 (A3H)	323 (143H)		Send buffer memory length designation (0001H to 1A00H)	200H							
164 (A4H)	324 (144H)	For data reception	Receive end data quantity designation 0001H to 33FEH: Receive end data quantity	1FFH	—	R, W	—				
165 (A5H)	325 (145H)		Receive end code designation FFFFH: No designation for receive end code 0H to FFH: Receive end code	0D0AH: CR+LF							
166 (A6H)	326 (146H)	For designation of receive area	Receiving buffer memory start address designation (400H to 1AFFH, 2600H to 3FFFH)	CH1: 600H CH2: A00H	—	R, W		—			Page 422 Send/receive Data Monitoring Function Settings
167 (A7H)	327 (147H)		Receive buffer memory length designation (0001H to 1A00H)	200H							
168 (A8H)	328 (148H)	For data reception	Receive data clear request 0: No request 1: Requested	0	—	R, W	—				
169 (A9H)	329 (149H)	For designation of on-demand user frame	Start frame No. designation, 1st 0: No designation Other than 0: Designated	0	R, W	—					Page 300 User Frame On-Demand Data Transmission and Buffer Memory Used
170 (AAH)	330 (14AH)		Start frame No. designation, 2nd 0: No designation Other than 0: Designated	0							
171 (ABH)	331 (14BH)		Final frame No. designation, 1st 0: No designation Other than 0: Designated	0							
172 (ACH)	332 (14CH)		Final frame No. designation, 2nd 0: No designation Other than 0: Designated	0							

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
173 (ADH)	333 (14DH)	For designation of receive user frame	User frame use enable/disable designation 0: Not use 1: Use 2: Data communication possible (Set by C24)	0	—	R, W	—				Page 321 Setting of receive user frames
174 to 177 (AEH to B1H)	334 to 337 (14EH to 151H)		Start frame No. designation (1st to 4th) 0H: No designation 1H or more: Start frame No.	0							
178 to 181 (B2H to B5H)	338 to 341 (152H to 155H)		Final frame No. designation (1st to 4th) 0H: No designation 1H or more: Final frame No.	0DH 0AH 00H 00H							
182 (B6H)	342 (156H)	Sending user frame	Sending user frame 0: Has not been sent 1 to 100: User frame being sent (nth)	0	—	R	—				Page 347 Settings of send user frames
183 (B7H)	343 (157H)	For designation of send user frame	CR/LF output designation 0: Do not send 1: Send	0	—	R, W	—				
184 (B8H)	344 (158H)		Output start pointer designation 0: No designation 1 to 100: Send from nth	0							
185 (B9H)	345 (159H)		Output quantity designation 0: No designation 1 to 100: Output n	0							
186 to 285 (BAH to 11DH)	346 to 445 (15AH to 1BDH)		Send frame number designation (A maximum of 100 can be designated) 0H: No designation 1H to C01FH: Designated	0							
286 (11EH)	446 (1BEH)	For designation of send wait time	Transmission wait time designation 0: No wait time 1H to FH: Wait time (unit: 10 ms)	0	R, W	—					Page 267 Transmission Wait Time Setting
287 (11FH)	447 (1BFH)	For designation of transparent code	Transparent code for sending specification, 1st 0000H: No specification Other than 0000H: Specified (below) • Transparent code (b0 to b7) 00H to FFH: Transparent code • Additional code (b8 to b15) 01H to FFH: Additional code	0	—	R, W		—			Page 358 TRANSPARENT CODES AND ADDITIONAL CODES
288 (120H)	448 (1C0H)		Transparent code for receiving specification 0000H: No specification Other than 0000H: Specified (below) • Transparent code (b0 to b7) 00H to FFH: Transparent code • Additional code (b8 to b15) 01H to FFH: Additional code	0							
289 (121H)	449 (1C1H)	For conversion designation	ASCII-binary conversion designation 0: Do not convert 1: Convert	0	—	R, W		—			Page 379 COMMUNICATING WITH ASCII CODE (ASCII-BIN CONVERSION)
290 (122H)	450 (1C2H)	For communication control specification	Echo back enable/prohibit specification for RS-422/485 interface 0: Echo back enable 1: Echo back prohibit	0	R, W				—		Page 436 RS-422/485 INTERFACE ECHO BACK ENABLE/ PROHIBIT SPECIFICATION

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
291 (123H)	451 (1C3H)	For NULL character automatic removal designation	NULL character automatic removal designation for transmission 0: Auto delete disabled 1: Auto delete enabled	0	—	R, W		—			Page 440 AUTOMATIC REMOVAL OF NULL CHARACTERS INCLUDED IN SEND DATA
292 (124H)	452 (1C4H)	For erroneous reception prevention designation	Communication data reception enable/disable designation 0: Receive enable 1: Receive disable	0	R, W						Page 438 ERRONEOUS NOISE SIGNAL RECEPTION PREVENTION FUNCTION
293 to 303 (125H to 12FH)	453 to 511 (1C5H to 1FFH)	Use prohibited	System area	—							
512 (200H)		For confirmation of station number setting status	Station number (parameter setting)	Depends on parameter setting	R				—		Page 483 Module information list
513 (201H)		For confirmation of LED lighting status and communication error status	CH1 side LED lighting status and communication error status 0: LED OFF, no error 1: LED ON, error SD WAIT (b0) SIO (b1) PRO. (b2) P/S (b3) C/N (b4) NAK (b5) ACK. (b6) NEU. (b7) For system (b8) to (b14) CH1 ERR. (b15)	Depends on module status.					R		
514 (202H)			CH2 side LED lighting status and communication error status 0: LED OFF, no error 1: LED ON, error SD WAIT (b0) SIO (b1) PRO. (b2) P/S (b3) C/N (b4) NAK (b5) ACK. (b6) NEU. (b7) CH2.ERR. (b14) For system (b8) to (b13),(b15)	Depends on module status.							

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
515 (203H)		For confirmation of parameter setting and mode switching	• Parameter setting error and mode switching error status 0: No error Other than 0: Parameter setting error and mode switching error • CH1 Communication protocol setting number (b0) 0: Normal 1: Error • CH1 Communication speed setting (b1) 0: Normal 1: Error • CH1 Mode switching at setting change prohibit (b3) 0: Normal 1: Error • CH2 Communication protocol setting number (b4) 0: Normal 1: Error • CH2 Communication speed setting (b5) 0: Normal 1: Error • CH2 Mode switching at setting change prohibit (b7) 0: Normal 1: Error • Setting station No. (b14) 0: Normal 1: Error • Interlink operation setting (b15) 0: Normal 1: Error	0	R						Page 487 Intelligent function module monitor
516 (204H)		For confirmation of user frame	Number of registered user frames 0: No registration 1 to 200: Number of registered frames	Depends on registration status.	R		—				—
517 to 541 (205H to 21DH)			User frame registration status (for registration number check) 0: No registration 1: Registered Bit corresponding to registration No. is 0(OFF)/1(ON). Registration No. 3E8H (1000): Address 205H (b0) to Registration No. 4AFH(1199): Address 211H (b7)	Depends on registration status.							
542 (21EH)		For confirmation of user frame	Number of registered default registration frames (for system)	Depends on registration status.							Page 290 Default registration frame (read only)
543, 544 (21FH, 220H)		Use prohibited	System area	—							

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
545 (221H)		For confirmation of the modem function	Modern function error code (error code when the modem function is being used) 0: Normal completion Other than 0 (error code) : Abnormal completion	0	R, W			—			Page 193 Buffer memory
546 (222H)			Modem function sequence status 0: Idle 1: Waiting for initialization 2: Initializing modem 3: Waiting 4: Checking password 5: Communicating 7: Modem disconnected	0	R			—			
547 (223H)			Number of data registration for connection 0: No registration 1 to 30: Number of registrations	Depends on registration status.							
548, 549 (224H, 225H)			Data registration status for connection (for confirmation of registration No.) 0: No registration 1: Registered Bit corresponding to registration No. is 0(OFF)/ 1(ON). Registration No. BB8H (3000) : Address 224H (b0) to Registration No. BD5H (3029) : Address 225H (b13)	Depends on registration status.							
550 (226H)			Number of data registration for initialization 0: No registration 1 to 30: Number of registrations	Depends on registration status.							
551, 552 (227H, 228H)		For confirmation of the modem function	Data registration status for initialization (for confirmation of registration No.) 0: No registration 1: Registered Bit corresponding to registration No. is 0(OFF)/ 1(ON). Registration No.9C4H (2500) : Address 227H (b0) to Registration No. 9E1H (2529) : Address 228H (b13)	Depends on registration status.	R			—			
553 to 589 (229H to 24DH)		Use prohibited	System area	—							
590 (24EH)		For confirmation of test mode status	Test mode setting status 0000H: No specification 0001H: Hardware Tests 0002H: Self-loopback test	Depends on parameter setting.	R						—
591 (24FH)		For confirmation of station number setting status	Station number (instruction setting) (0 to 31)	Depends on module status.	R				—	R	Page 487 Intelligent function module monitor

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
592 (250H)	608 (260H)	For confirmation of transmission control status	Communication protocol status (parameter setting) 0: MELSOFT Connection 1: MC protocol (Format 1) 2: MC protocol (Format 2) 3: MC protocol (Format 3) 4: MC protocol (Format 4) 5: MC protocol (Format 5) 6: Nonprocedural protocol 7: Bidirectional protocol 8: (For interlink operation) 9: Predefined protocol AH: MODBUS slave (RTU) BH: MODBUS slave (ASCII) 10H: Simple CPU communication	Depends on parameter setting.	R						—
593 (251H)	609 (261H)		Transmission setting status (parameter setting) • Operation setting (b0) 0: Independent 1: Interlink • Data bit (b1) 0: 7 bits 1: 8 bits • Parity bit (b2) 0: None 1: Yes • Odd/even parity (b3) 0: Odd 1: Even • Stop bit (b4) 0: 1 bit 1: 2 bits • Sum check code (b5) 0: None 1: Yes • Online change (b6) 0: Disable 1: Enable • Setting change (b7) 0: Disable 1: Enable • Communication speed (b8 to b11) 1200 bps to 230400 bps • For system (b12 to b15) All 0	Depends on parameter setting.	R						—

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
594 (252H)	610 (262H)	For confirmation of transmission control status	Communication protocol status (current) 0: MELSOFT connection 1: MC protocol (Format 1) 2: MC protocol (Format 2) 3: MC protocol (Format 3) 4: MC protocol (Format 4) 5: MC protocol (Format 5) 6: Nonprocedural protocol 7: Bidirectional protocol 8: (For interlink operation) 9: Predefined protocol AH: MODBUS slave (RTU) BH: MODBUS slave (ASCII) 10H: Simple CPU communication	Depends on module status.	R						Page 487 Intelligent function module monitor
595 (253H)	611 (263H)		Transmission status (current) • Operation setting (b0) 0: Independent 1: Interlink • Data bit (b1) 0: 7 bits 1: 8 bits • Parity bit (b2) 0: None 1: Yes • Odd/even parity (b3) 0: Odd 1: Even • Stop bit (b4) 0: 1 bit 1: 2 bits • Sum check code (b5) 0: None 1: Yes • Online change (b6) 0: Disable 1: Enable • Setting change (b7) 0: Disable 1: Enable • Communication speed (b8 to b11) 1200 bps to 230400 bps • For system (b12 to b15) All 0	Depends on module status.							

596
(254H)
	612
	(264H)
		Control signal status	RS-232 control signal status
			0: OFF status 1: ON status
			RS(RTS) (b0)
			DR(DSR) (b1)
			ER(DTR) (b2)
			CD(DCD) (b3)
			CS(CTS) (b4)
			CI(RI) (b5)
			Not used (b6 to b15) All 0
				Depends on signal status.	R						

597 (255H)	613 (265H)	For confirmation of communication result	Transmission sequence status (For confirmation of MC protocol communication status) 0: Waiting for receiving command message 1: Receiving command message 2: Command message reception complete 3: Waiting to access CPU module 4: Accessing CPU module 5: CPU module access complete 6: Response message transmission	0	R	—					Page 487 Intelligent function module monitor
598 (256H)	614 (266H)		On-demand execution results 0: Normal completion Other than 0: Abnormal completion (error code)	0	R, W	—					Page 265 Send Monitoring Time (timer 2) Setting Page 491 Error information clear
599 (257H)	615 (267H)		Data transmission result 0: Normal completion Other than 0: Abnormal completion (error code)	0	R, W				—		
600 (258H)	616 (268H)		Data reception result 0: Normal completion Other than 0: Abnormal completion (error code)	0							
601 (259H)	617 (269H)	Use prohibited	System area	—							

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
602 (25AH)	618 (26AH)	For confirmation of communication result	MC protocol send error code (excludes 1C frame communication) 0: No error 1 or more: Send error code	0	R, W	—					—
603 (25BH)	619 (26BH)		Receive user frame (nth) 0: Has not been received 1 to 4: Combination of user frame No. designations for reception message	0	—	R	—				
604 to 607 (25CH to 25FH)	620 to 1023 (26CH to 3FFH)	Use prohibited	System area	—							
1024 (400H)	2048 (800H)	Send/receive area[*1]	Send data quantity designation 0: No designation 1 or more: Number of send data	0	R, W				—		Page 60 Receive area and receive data arrangement Page 81 Send area and send data arrangement
1025 to 1535 (401H to 5FFH)	2049 to 2559 (801H to 9FFH)		Send data specification Data to be sent to a target device	0							
1536 (600H)	2560 (A00H)		Receive data quantity (Number of data for which read is requested) 0: No receive data 1 or more: Receive data quantity	0							
1537 to 2047 (601H to 7FFH)	2561 to 3071 (A01H to BFFH)		Receive data Data received from a target device	0							
3072 to 6911 (C00H to 1AFFH)		For user	User setting area (3840 words) Determined by the user.	0	R, W				—		—

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
6912 to 6952 (1B00H to 1B28H)		For user registration	For registration No. 8001H	The user registration area is used together for the following usage, with data written by the user according to the purpose of use by the TO instruction, etc. ■When data is communicated using user frames • User frame ■When data is communicated using the modem function • Data for initialization • Data for connection							Page 193 Buffer memory
6953 to 6993 (1B29H to 1B51H)			For registration No. 8002H								
6994 to 7034 (1B52H to 1B7AH)			For registration No. 8003H								
7035 to 7075 (1B7BH to 1BA3H)			For registration No. 8004H								
7076 to 7116 (1BA4H to 1BCCH)			For registration No. 8005H								
7117 to 7157 (1BCDH to 1BF5H)			For registration No. 8006H								
7158 to 7198 (1BF6H to 1C1EH)			For registration No. 8007H								
7199 to 7239 (1C1FH to 1C47H)			For registration No. 8008H								
7240 to 7280 (1C48H to 1C70H)			For registration No. 8009H								
7281 to 7321 (1C71H to 1C99H)			For registration No. 800AH								
7322 to 7362 (1C9AH to 1CC2H)			For registration No. 800BH								
7363 to 7403 (1CC3H to 1CEBH)			For registration No. 800CH								
7404 to 7444 (1CECH to 1D14H)			For registration No. 800DH								
7445 to 7485 (1D15H to 1D3DH)			For registration No. 800EH								
7486 to 7526 (1D3EH to 1D66H)			For registration No. 800FH								
7527 to 7567 (1D67H to 1D8FH)			For registration No. 8010H								
7568 to 7608 (1D90H to 1DB8H)			For registration No. 8011H								
7609 to 7649 (1DB9H to 1DE1H)			For registration No. 8012H								
7650 to 7690 (1DE2H to 1E0AH)			For registration No. 8013H								
7691 to 7731 (1E0BH to 1E33H)			For registration No. 8014H								
7732 to 7772 (1E34H to 1E5CH)			For registration No. 8015H								
7773 to 7813 (1E5DH to 1E85H)			For registration No. 8016H								

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
7814 to 7854 (1E86H to 1EAEH)		For user registration	For registration No. 8017H	The user registration area is used together for the following usage, with data written by the user according to the purpose of use by the TO instruction, etc. ■When data is communicated using user frames • User frame ■When data is communicated using the modem function • Data for initialization • Data for connection							Page 193 Buffer memory
7855 to 7895 (1EAFH to 1ED7H)			For registration No. 8018H								
7896 to 7936 (1ED8H to 1F00H)			For registration No. 8019H								
7937 to 7977 (1F01H to 1F29H)			For registration No. 801AH								
7978 to 8018 (1F2AH to 1F52H)			For registration No. 801BH								
8019 to 8059 (1F53H to 1F7BH)			For registration No. 801CH								
8060 to 8100 (1F7CH to 1FA4H)			For registration No. 801DH								
8101 to 8141 (1FA5H to 1FCDH)			For registration No. 801EH								
8142 to 8182 (1FCEH to 1FF6H)			For registration No. 801FH								
8183 to 8198 (1FF7H to 2006H)		Use prohibited	System area	—							
8199 (2007H)		For designation of the modem function -2	Auto modem initialization designation 0: Do not automatically initialize 1: Automatically initialize	0	R, W			—			Page 193 Buffer memory
8200 (2008H)			Modem initialization DR (DSR) signal enable/ disable designation 0: DR signal enabled 1: DR signal disabled	1							
8201 (2009H)			Complete signal handling for modem function designation 0: Do not turn ON/OFF X13, X14. 1: Turn ON/OFF X13, X14.	1							
8202, 8203 (200AH, 200BH)		Use prohibited	System area	—							
8204 (200CH)		For remote password function	Remote password mismatch notification count designation 0H: No designation 1H to FFFFH: Number of times for notification	0	R, W	—					Page 193 Buffer memory
8205 (200DH)			Remote password mismatch notification accumulated count designation 0H: No designation 1H to FFFFH: Cumulative number of times for notification	1							
8206 (200EH)		For designation of the modem function -3	Line disconnection wait time specification (PLC CPU monitor) 0000H to FFFFH: Wait time (unit: s)	0	R, W		—				Page 193 Buffer memory
8207 (200FH)	8459 to 8463 (210BH to 210FH)	Use prohibited	System area	—							
8208 (2010H)	8464 (2110H)	Interrupt specification	Receive interrupt-issued designation 0: Do not issue interrupt. 1: Issue interrupt.	0	—	R, W		—			Page 471 Interrupt settings
8209 (2011H)	8465 (2111H)	Use prohibited	System area	—							

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
8210 (2012H)	8466 (2112H)	For designation of transmission control	Transmission control start free space designation 64 to 4095: transmission control start free space	64	R, W				—		Page 269 C24 DTR control contents and free OS area specification
8211 (2013H)	8467 (2113H)		Transmission control end free space designation 263 to 4096: transmission control end free space	263							
8212 (2014H)	8468 (2114H)		Nonprocedural no-reception monitoring time method designation 0: Method 0 1: Method 1	0	—	R, W	—				
8213 to 8215 (2015H to 2017H)	8469 to 8471 (2115H to 2117H)	Use prohibited	System area	—							
8216 (2018H)	8472 (2118H)	Send/receive data monitoring function	Send/receive data monitoring designation 0000H: No monitor/stop instruction 0001H: Monitor start instruction 0002H: Monitoring (set by C24.) 1002H: Monitoring stopped (set by C24.) 100FH: Monitor setting error (set by C24.)	0	R, W						Page 420 Send/receive data monitoring operation
8217 (2019H)	8473 (2119H)		Monitor option specification 0: OFF 1: ON Stop at buffer full specification (b0) 0 timer error outbreak stop designation (b2) For system (b1), (b3) to (b15)	0							
8218 (201AH)	8474 (211AH)		Monitor buffer start address designation (400H to 1AFDH, 2600H to 3FFDH)	CH1: 2600H CH2: 3300H							
8219 (201BH)	8475 (211BH)		Monitor buffer size designation (0003H to 1A00H)	0D00H							
8220 to 8223 (201CH to 201FH)	8476 to 8479 (211CH to 211FH)	Use prohibited	System area	—							
8224 to 8227 (2020H to 2023H)	8480 to 8483 (2120H to 2123H)	For designation of user frame receiving method	User frame receiving method designation (1st to 4th) 0: Method 0, 1: Method 1	0	—	R, W	—				Page 321 Setting of receive user frames
8228 to 8231 (2024H to 2027H)	8484 to 8487 (2124H to 2127H)		Method 1 dedicated receive end data quantity designation (1st to 4th) 0 or more: Method 1 dedicated receive end data quantity designation	0							
8232 to 8239 (2028H to 202FH)	8488 to 8495 (2128H to 212FH)	Use prohibited	System area	—							
8240 to 8248 (2030H to 2038H)	8496 to 8504 (2130H to 2138H)	For designation of transparent code	Transparent code for sending specification (2nd to 10th) 0000H: No specification Other than 0000H: Specified (below) • Transparent code (b0 to b7) 00H to FFH : Transparent code • Additional code (b8 to b15) 01H to FFH : Additional code	0	—	R, W		—			Page 359 Registering Transparent Codes and Additional Codes

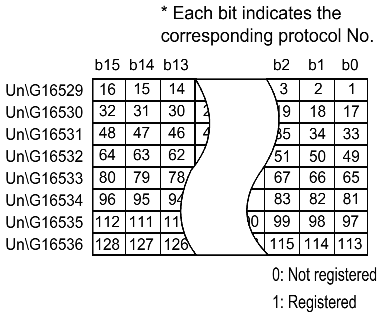
Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
8249 to 8255 (2039H to 203FH)	8505 to 8511 (2139H to 213FH)	Use prohibited	System area	—							
8256 (2040H)	8512 (2140H)	For designation of programmable controller CPU monitoring function	Cycle time units designation 0: 100 ms 1: s 2: min	2	R		—				Page 170 Settings for Using the Programmable Controller CPU Monitoring Function
8257 (2041H)	8513 (2141H)		Cycle time designation (programmable controller CPU monitoring interval time) 0H: No designation 1H to FFFFH: Programmable controller CPU monitoring interval time	5H							
8258 (2042H)	8514 (2142H)		Programmable controller CPU monitoring function designation 0: Do not use the function. 1: Fixed cycle send 2: Condition match send	0							
8259 (2043H)	8515 (2143H)		System area	—							
8260 (2044H)	8516 (2144H)		Send pointer designation (For fixed cycle send and nonprocedural data transmission) 0: No specification 1 to 100: Output head point (send from the nth) Send the user frames designated in the following send frame number designation areas from the designated pointer position. (addresses: CH1 side = BAH to 11DH, CH2 side = 15AH to 1BDH)	0	—	R	—				
8261 (2045H)	8517 (2145H)		Output quantity designation (for fixed cycle send and nonprocedural data transmission) 0: No specification 1 to 100:Output quantity (Specify the number of frame to be transmitted.)	0							
8262 (2046H)	8518 (2146H)		Data No. for connection designation (for fixed cycle send) 0: No designation 0BB8H to 0BD5H, 8001H to 801FH: Data No. for connection	0	R		—				
8263 to 8268 (2047H to 204CH)	8519 to 8524 (2147H to 214CH)	Use prohibited	System area	—							

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
8269 (204DH)	8525 (214DH)	For designation of programmable controller CPU monitoring function	Registered word block quantity designation 0: No designation 1 to 10: Number of blocks of word devices (Up to 10 blocks can be designated in total.)	0	R		—				Page 170 Settings for Using the Programmable Controller CPU Monitoring Function
8270 (204EH)	8526 (214EH)		Registered bit block quantity designation 0: No designation 1 to 10: Number of blocks of bit devices (Up to 10 blocks can be designated in total.)	0							
8271 (204FH)	8527 (214FH)		CPU error monitoring designation 0: Do not monitor. 1: Monitor.	0							
8272 (2050H)	8528 (2150H)		1st block monitoring device Monitoring device designation 0: No designation 90H to CCH: Device code	0							
8273, 8274 (2051H, 2052H)	8529, 8530 (2151H, 2152H)		1st block monitoring device Start device No. specification 0 or more: Start device number	0							
8275 (2053H)	8531 (2153H)		1st block monitoring device Number of read points specification 0: No specification 1 or more: read point	0							
8276 (2054H)	8532 (2154H)		1st block monitoring device Monitoring condition specification (judgment condition specification) 0: No specification 1 or more: Monitoring condition	0							
8277 (2055H)	8533 (2155H)	For designation of programmable controller CPU monitoring function	1st block monitoring device Monitoring condition value designation • At bit device 0: OFF 1: ON • At word device 0 to FFFFH: Monitoring condition value	0	R		—				Page 170 Settings for Using the Programmable Controller CPU Monitoring Function
8278 (2056H)	8534 (2156H)		1st block monitoring device Send pointer designation (for condition match send and nonprocedural data transmission) 0: No designation 1 to 100: Output head point (send from the nth) Send the user frames designated in the following send frame number designation areas from the designated pointer position. (address: CH1 side = BAH to 11DH, CH2 side = 15AH to 1BDH)	0	—	R	—				
8279 (2057H)	8535 (2157H)		1st block monitoring device Output quantity designation (for condition match send and nonprocedural data transmission) 0: No designation 1 to 100: Output quantity (Designate the number of frame to be transmitted.)	0							
8280 (2058H)	8536 (2158H)		1st block monitoring device Data No. for connection designation (for condition match send) 0: No designation 0BB8H to 0BD5H, 8001H to 801FH: Data No. for connection	0	R		—				
8281 to 8361 (2059H to 20A9H)	8537 to 8617 (2159H to 21A9H)		2nd to 10th block monitoring devices The structure of each area is the same as the 1st block monitoring device area. ☞ Page 579 Monitoring device areas for the programmable controller CPU monitoring function								

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
8362 to 8421 (20AAH to 20E5H)	8618 to 8677 (21AAH to 21E5H)	Use prohibited	System area	—							
8422 (20E6H)	8678 (21E6H)	For designation of programmable controller CPU monitoring function	CPU error monitoring designation Send pointer designation (for condition match send and nonprocedural data transmission) 0: No designation 1 to 100: Output head point (send from the nth) Send the user frames designated in the following send frame number designation areas from the designated pointer position. CH1 side = BAH to 11DH CH2 side = 15AH to 1BDH	0	—	R	—				Page 170 Settings for Using the Programmable Controller CPU Monitoring Function
8423 (20E7H)	8679 (21E7H)		CPU error monitoring designation Output quantity designation (for condition match send and nonprocedural data transmission) 0: No designation 1 to 100: Output quantity (number of frame to be transmitted)	0							
8424 (20E8H)	8680 (21E8H)		CPU error monitoring designation Data No. for connection designation (for condition match send) 0: No designation 0BB8H to 0BD5H, 8001H to 801FH: Data No. for connection	0	R		—				
8425 to 8458 (20E9H to 210AH)	8681 to 8703 (21E9H to 21FFH)	Use prohibited	System area	—							
8704 to 8707 (2200H to 2203H)	8960 to 8963 (2300H to 2303H)	Use prohibited	System area	—							
8708 (2204H)	8964 (2304H)	Programmable controller CPU monitoring function	Programmable controller CPU monitoring function operation status 0: Not executed 1: Wait for programmable controller CPU monitoring time 2: Accessing programmable controller CPU 3: Sending monitoring results	0	R		—				Page 170 Settings for Using the Programmable Controller CPU Monitoring Function
8709 (2205H)	8965 (2305H)		Programmable controller CPU monitoring function execution result (current) 0: Normal completion Other than 0: Abnormal completion (error code)	0							
8710 (2206H)	8966 (2306H)		Programmable controller CPU monitoring function number of transmission 0: Not executed 1 or more: Number of transmissions	0							
8711 (2207H)	8967 (2307H)		Monitoring condition attained block No. 0: No block of which monitoring condition is satisfied 1 to 10: Registration order of word/bit block (nth) 4096 : CPU abnormal monitoring block The latest block No. for which monitoring condition is satisfied is stored.	0							

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
8712 to 8948 (2208H to 22F4H)	8968 to 9215 (2308H to 23FFH)	Use prohibited	System area	—							
8949 to 8954 (22F5H to 22FAH)		Use prohibited	System area	—							
8955 (22FBH)		For the remote password function	Accumulated count of unlock process normal completion 0 or more: Accumulated count of normal completion	0	R, W	—					Page 193 Buffer memory
8956 (22FCH)			Accumulated count of unlock process abnormal completion 0 or more: Accumulated count of abnormal completion	0							
8957, 8958 (22FDH, 22FEH)		Use prohibited	System area	—							
8959 (22FFH)		For the remote password function	Accumulated count of lock process by line disconnection 0 or more: Accumulated count of lock process by line disconnection	0	R, W	—					Page 193 Buffer memory
9216 (2400H)		Use prohibited	System area	—							
9217 (2401H)		For storage of number of flash ROM write	Flash ROM write count 0 to 1000: Number of rewriting	0	R						—
9218 to 9727 (2402H to 25FFH)		Use prohibited	System area	—							
9728 to 16383 (2600H to 3FFFH)		For user	User setting area 2 (6656 words) (Send/receive data monitoring function default buffer) Usage is determined by the user.	0	R, W						—
16384 (4000H)	16416 (4020H)	For designation of predefined protocol function control data	Number of consecutive protocol executions 0: No protocol executed 1 to 8: Number of protocols to be executed continuously	0	—			R, W	—		
16385 (4001H)	16417 (4021H)	Use prohibited	System area	—							
16386 (4002H)	16418 (4022H)	For designation of predefined protocol function control data	Execution results 0: Normal completion Other than 0: Abnormal completion (error code)	0	—			R	—		—
16387 (4003H)	16419 (4023H)		Result of the execution count 0: No protocol executed 1 to 8: Number of executed protocol for continuous execution	0	—			R	—		
16388 to 16395 (4004H to 400BH)	16420 to 16427 (4024H to 402BH)		Protocol number 0: No designation 1 to 128, 201 to 207: Protocol number to be executed	0	—			R, W	—		
16396 to 16403 (400CH to 4013H)	16428 to 16435 (402CH to 4033H)		Matched packet No.1 to 8 0: No matched packet 1 to 16: Matched packet No.	0	—			R	—		

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
16404 to 16415 (4014H to 401FH)	16436 to 16447 (4034H to 403FH)	Use prohibited	System area	—							
16448 (4040H)	16464 (4050H)	For designation of predefined protocol function control data	Protocol cancel 0: No cancel direction (default) 1: Cancel request (set by user) 2: Cancel completed (set by C24)	0	—			R, W	—		—
16449 (4041H)	16465 (4051H)	For confirmation of predefined protocol function execution status	Protocol execution status 0: Unexecuted (default) 1: Waiting for transmission 2: Sending 3: Waiting for data reception 4: Receiving 5: Completed	0	—			R	—		Page 142 DEBUG SUPPORT FUNCTION Page 519 Error Code List
16450 (4042H)	16466 (4052H)		Predefined protocol function error code 0: Normal Other than 0: Error (error code)	0							
16451 (4043H)	16467 (4053H)		Protocol execution count 0: No log 1 to 65535: Number of executions	0							—
16452 to 16463 (4044H to 404FH)	16468 to 16479 (4054H to 405FH)	Use prohibited	System area	—							
16480 to 16517 (4060H to 4085H)		Use prohibited	System area	—							
16518 (4086H)		Protocol setting data error information	Protocol No. 1 to 128: Protocol No. 65535: Unidentified	0	—			R	—		Page 39 Packet elements Page 519 Error Code List
16519 (4087H)			Setting type 0: Packet setting or element setting 1: Protocol detailed setting 65535: Unidentified	0							
16520 (4088H)			Packet No. 0: Send packet 1 to 16: Receive packet No. 65535: Unidentified Valid when the Setting type value is 0.	0							
16521 (4089H)			Element No. 1 to 32: Element No. 65535: Unidentified Valid when the Setting type value is 0.	0							
16522 to 16527 (408AH to 408FH)		Use prohibited	System area	—							

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
16528 (4090H)		For confirmation of protocol setting data	Number of registered protocols 0: No registration 1 to 128: Number of registrations	0	—			R	—		Page 30 DATA COMMUNICATION USING PREDEFINED PROTOCOL
16529 to 16536 (4091H to 4098H)			Protocol registration 0: No registration 1: Registered The bit corresponding to each protocol No. turns ON or OFF.  * Each bit indicates the corresponding protocol No. b15 b14 b13 b2 b1 b0 Un\G16529 16 15 14 3 2 1 Un\G16530 32 31 30 2 9 18 17 Un\G16531 48 47 46 4 35 34 33 Un\G16532 64 63 62 51 50 49 Un\G16533 80 79 78 67 66 65 Un\G16534 96 95 94 83 82 81 Un\G16535 112 111 11 0 99 98 97 Un\G16536 128 127 126 115 114 113 0: Not registered 1: Registered	0							—
16537 to 16607 (4099H to 40DFH)		Use prohibited	System area	—							
16608, 16609 (40E0H, 40E21)	16624, 16625 (40F0H, 40F1H)	Use prohibited	System area								
16610 (40E2H)	16626 (40F2H)	For designation of protocol execution history	Execution log options History storage condition 0 (OFF): Error logs of only protocol that completed abnormally are stored. 1 (ON): All protocol execution status and execution logs are stored.	0	—			R, W	—		Page 146 Protocol Execution Log Storage Function (for Predefined Protocol only)
16611 to 16623 (40E3H to 40EFH)	16627 to 16638 (40F3H to 40FEH)	Use prohibited	System area	—							
16639 (40FFH)	18431 (47FFH)	Use prohibited	System area								
16640 (4100H)	18432 (4800H)	For confirmation of protocol execution history	Number of stored protocol execution logs 0: No log 1 to 32: Number of stored logs	0	—			R	—		Page 146 Protocol Execution Log Storage Function (for Predefined Protocol only)
16641 (4101H)	18433 (4801H)		Protocol execution log write pointer 0: No log 1 to 32: Latest protocol execution log No.	0							
16642 (4102H)	18434 (4802H)	Use prohibited	System area	—							

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
16643 (4103H)	18435 (4803H)	For confirmation of protocol execution history	Execution log 1 Protocol No. 0: No log 1 to 128: Protocol No. 201 to 207: Functional protocol No.	0	—			R	—		Page 146 Protocol Execution Log Storage Function (for Predefined Protocol only)
16644 to 16659 (4104H to 4113H)	18436 to 18451 (4804H to 4813H)		Execution log 1 Target device model 0: No protocol executed Other than 0: Target device model (Up to 32 bytes are stored in ASCII codes.)	0							
16660 to 16675 (4114H to 4123H)	18452 to 18467 (4814H to 4823H)		Execution log 1 Protocol name 0: No protocol executed Other than 0: Protocol name (Up to 32 bytes in ASCII codes are stored.)	0							
16676 (4124H)	18468 (4824H)		Execution log 1 Communication type 0: No protocol executed 1: Send only 2: Receive only 3: Send and receive 14: Functional protocol 15: Unregistered protocol	0							
16677 (4125H)	18469 (4825H)		Execution log 1 Protocol execution status 0: Unexecuted 1: Waiting for transmission 2: Sending 3: Waiting for reception 4: Receiving 5: Completed	0							
16678 (4126H)	18470 (4826H)		Execution log 1 Execution result 0: Normal completion Other than 0 (error code): Abnormal completion	0							
16679 (4127H)	18471 (4827H)		Execution log 1 Matched packet No. 0: Error occurred, or Communication type of the executed protocol is "Send only". 1 to 16: Matched receive packet number	0							
16680 (4128H)	18472 (4828H)		Execution log 1 Number of send retries 0: No retry 1 to 10: Number of retries	0							
16681 (4129H)	18473 (4829H)	Use prohibited	System area	—							
16682 (412AH)	18474 (482AH)	For confirmation of protocol execution history	Execution log 1 start time and date 0: No log Upper 8 bits: Month Lower 8 bits: Last 2 digits of year b15 to b8 b7 to b0 Month (01H to 12H) | Year (00H to 99H), last 2 digits	0	—			R	—		Page 146 Protocol Execution Log Storage Function (for Predefined Protocol only)
16683 (412BH)	18475 (482BH)		Execution log 1 start time and date 0: No log Upper 8 bits: Hour Lower 8 bits: Day b15 to b8 b7 to b0 Hour (00H to 23H) | Day (01H to 31H)	0							

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
16684 (412CH)	18476 (482CH)	For confirmation of protocol execution history	Execution log 1 start time and date 0: No log Upper 8 bits: Second Lower 8 bits: Minute b15 to b8 b7 to b0 Second (00H to 59H) | Minute (00H to 59H)	0	—			R	—		Page 146 Protocol Execution Log Storage Function (for Predefined Protocol only)
16685 (412DH)	18477 (482DH)		Execution log 1 start time and date 0: No log Lower 8 bits: First 2 digits of year Lower 8 bits: Day of week (0: Sun to 6: Sat) b15 to b8 b7 to b0 Year (00H to 99H), first 2 digits | Day of week (00H to 06H) 00H (Sunday) to 06H (Saturday)	0							
16686 (412EH)	18478 (482EH)		Execution log 1 end time and date 0: No log Upper 8 bits: Month Lower 8 bits: Last 2 digits of year b15 to b8 b7 to b0 Month (01H to 12H) | Year (00H to 99H), last 2 digits	0							
16687 (412FH)	18479 (482FH)		Execution log 1 end time and date 0: No log Upper 8 bits: Hour Lower 8 bits: Day b15 to b8 b7 to b0 Hour (00H to 23H) | Day (01H to 31H)	0							
16688 (4130H)	18480 (4830H)		Execution log 1 end time and date 0: No log Upper 8 bits: Second Lower 8 bits: Minute b15 to b8 b7 to b0 Second (00H to 59H) | Minute (00H to 59H)	0							
16689 (4131H)	18481 (4831H)		Execution log 1 end time and date 0: No log Lower 8 bits: First 2 digits of year Lower 8 bits: Day of week (0: Sun to 6: Sat) b15 to b8 b7 to b0 Year (00H to 99H), first 2 digits | Day of week (00H to 06H) 00H (Sunday) to 06H (Saturday)	0							
16690 to 18177 (4132H to 4701H)	18482 to 19969 (4832H to 4E01H)		Execution log 2 to 32 Same as Execution log 1	0							
18178 to 18430 (4702H to 47FEH)	19970 to 20223 (4E02H to 4EFFH)	Use prohibited	System area	—							
20224 to 20231 (4F00H to 4F07H)		Error address storage area	Stores the buffer memory address of which parameter setting value is set outside the range. (Up to 8)	0	—			R	—		Page 491 Error information clear
20232 to 20479 (4F08H to 4FFFH)		Use prohibited	System area	—							
20480 to 24575 (5000H to 5FFFH)		Send/receive area for predefined protocol function	Predefined protocol function buffer	0	—			R, W	—		Page 491 Error information clear

Address Dec (Hex)		Purpose	Name	Default value	Protocol						Reference
CH1	CH2				MC	Non	Bi	Pd	MD	S	
24576 to 28671 (6000H to 6FFFH)		Use prohibited	System area	—							
28672 to 32639 (7000H to 7F7FH)		☞ Page 573 Buffer memory list for MODBUS		—							
32640 to 45159 (7F80H to B067H)		☞ Page 577 Buffer memory list for simple CPU communication		—							
45160 to 49151 (B068H to BFFFH)		Use prohibited	System area	—							
49152 to 53247 (C000H to CFFFH)		For user	User setting area for MODBUS	0	—				R, W	—	Page 467 MODBUS device assignment parameters
53248 to 57343 (D000H to DFFFH)		For user	User setting area for simple CPU communication	0	—					R, W	—
57344 to 65535 (E000H to FFFFH)		Use prohibited	System area	—							

*1 The area can be used as a user setting area (the send and receive data storage areas).

Buffer memory list for MODBUS

Address Dec (Hex)		Purpose	Name		Default value	Target	Reference
CH1	CH2						
28672 to 28673 (7000H to 7001H)		MODBUS status storage area	System area		—		
28674 (7002H)	28676 (7004H)		Error code	CH1/CH2 side exception code storage area	0	R	Page 543 MODBUS Exception Code List
28675 (7003H)	28677 (7005H)			System area	—		
28678 to 28679 (7006H to 7007H)			System area		—		
28680 to 28681 (7008H to 7009H)		Setting area for MODBUS	System area		—		
28682 (700AH)			Device for error status	Device code	000A7H	R, W	Page 465 MODBUS slave settings
28683 (700BH)				System area	—		
28684 to 28685 (700CH to 700DH)				Start device number	0H	R, W	
28686 (700EH)			CPU response monitoring timer value 0: 5s Value other than '0': Setting time = Setting value × 100 ms		32H(5s)	R, W	
28687 (700FH)			Access target specification when a CC-Link IE Field Network remote head module is mounted		0	R, W	
28688 (7010H)			Buffer memory for error status		0	R, W	
28689 (7011H)			MODBUS station number setting Upper 8 bits: CH2 side (1 to 247) Lower 8 bits: CH1 side (1 to 247)		0101H		
28690 to 30975 (7012H to 78FFH)		Use prohibited	System area		—		

Address Dec (Hex)		Purpose	Name		Default value	Target	Reference
CH1	CH2						
30976 (7900H)		MODBUS device assignment parameter	Coil assignment 1	Device code	0	R, W	Page 467 MODBUS device assignment parameters
30977 (7901H)				System area	—		
30978 to 30979 (7902H to 7903H)				Start device number	0	R, W	
30980 (7904H)				Start coil number	0		
30981 (7905H)				Number of assigned points	0		
30982 to 31071 (7906H to 795FH)			Coil assignment 2 to 16	Same as coil assignment 1	0		
31072 (7960H)			Input assignment 1	Device code	0	R, W	
31073 (7961H)				System area	—		
31074 to 31075 (7962H to 7963H)				Start device number	0	R, W	
31076 (7964H)				Start input number	0		
31077 (7965H)				Number of assigned points	0		
31078 to 31167 (7966H to 79BFH)			Input assignment 2 to 16	Same as input assignment 1	0		
31168 (79C0H)			Input register assignment 1	Device code	0	R, W	
31169 (79C1H)				System area	—		
31170 to 31171 (79C2H to 79C3H)				Start device number	0	R, W	
31172 (79C4H)				Start input register number	0		
31173 (79C5H)				Number of assigned points	0		
31174 to 31263 (79C6H to 7A1FH)			Input register assignment 2 to 16	Same as input register assignment 1	0		
31264 (7A20H)			Holding register assignment 1	Device code	0	R, W	
31265 (7A21H)				System area	—		
31266 to 31267 (7A22H to 7A23H)				Start device number	0	R, W	
31268 (7A24H)				Start holding register number	0		
31269 (7A25H)				Number of assigned points	0		
31270 to 31359 (7A26H to 7A7FH)			Holding register assignment 2 to 16	Same as holding register assignment 1	0		
31360 to 31748 (7A80H to 7C04H)		Use prohibited	System area		—		

Address Dec (Hex)		Purpose	Name		Default value	Target	Reference
CH1	CH2						
31749 (7C05H)		MODBUS operating status	System area		—		
31750 (7C06H)			MODBUS station number confirmation Upper 8 bits: CH2 side (1 to 247) Lower 8 bits: CH1 side (1 to 247) When the MODBUS slave function is not used, '0' is stored.		0	R	Page 489 Current operation status reading method
31751 to 31762 (7C07H to 7C12H)			System area		—		
31763 (7C13H)			Parameter error information	MODBUS device allocation parameter error code storage area	0	R	Page 467 MODBUS device assignment parameters
31764 (7C14H)				MODBUS device allocation parameter setting result storage area Device type error 0: Not assigned 1: Coil 2: Input 4: Input register 5: Holding register	0		
31765 (7C15H)				MODBUS device allocation parameter setting result storage area Assigned group No. error	0		
31766 to 31769 (7C16H to 7C19H)				System area	—		
31770 to 31997 (7C1AH to 7CFDH)			System area		—		
31998 (7CFEH)			Error log	Number of errors occurred	0	R	Page 543 MODBUS Exception Code List
31999 (7CFFH)				Error log write pointer	0		
32000 (7D00H)				Error log 1 detailed error code	0		
32001 (7D01H)				Error log 1 exception code	0		
32002 (7D02H)				Error log 1 function code	0		
32003 (7D03H)				Error log 1 CH	0		
32004 (7D04H)				Error log 1 station number	0		
32005 to 32006 (7D05H to 7D06H)				System area	—		
32007 (7D07H)				Error log 1 function 0: No error 3: Slave function 4: Others	0	R	
32008 to 32255 (7D08H to 7DFFH)				Error log 2 to 32 (Same as error log 1)	0		
32256 to 32511 (7E00H to 7EFFH)		Use prohibited	System area		—		

Address Dec (Hex)		Purpose	Name		Default value	Target	Reference
CH1	CH2						
32512 (7F00H)	32576 (7F40H)	MODBUS communication status	CH1/CH2 communication status	Diagnostic data Bus message count	0	R, W	Page 116 Function List Page 490 Check method for the communication status of the MODBUS slave function
32513 (7F01H)	32577 (7F41H)			Diagnostic data Bus communication error count	0		
32514 (7F02H)	32578 (7F42H)			Diagnostic data Character overrun count	0		
32515 (7F03H)	32579 (7F43H)			Diagnostic data Message discard count	0		
32516 (7F04H)	32580 (7F44H)			Diagnostic data Data discard count	0		
32517 (7F05H)	32581 (7F45H)			Diagnostic data Failed transmission count	0		
32518 (7F06H)	32582 (7F46H)			Diagnostic data Slave message count	0		
32519 (7F07H)	32583 (7F47H)			Diagnostic data Slave no-response count	0		
32520 (7F08H)	32584 (7F48H)			Diagnostic data Slave NAK count	0		
32521 (7F09H)	32585 (7F49H)			Diagnostic data Slave busy count	0		
32522 (7F0AH)	32586 (7F4AH)			Diagnostic data Exception error count	0		
32523 (7F0BH)	32587 (7F4BH)			Diagnostic data Communications event count	0	R	
32524 (7F0CH)	32588 (7F4CH)			Diagnostic data 2nd byte of end code	0AH		
32525 (7F0DH)	32589 (7F4DH)			Diagnostic data Communications mode 0: Online · mode 1: Force listen · only · mode	0		
32526 to 32542 (7F0EH to 7F1EH)	32590 to 32606 (7F4EH to 7F5EH)			System area	—		
32543 (7F1FH)	32607 (7F5FH)			Communications event log count	0	R	Page 116 Function List
32544 to 32575 (7F20H to 7F3FH)	32608 to 32639 (7F60H to 7F7FH)			Communication event log 1 to 64 b15 to b8 b7 to b0 7F20H | Log 2 | Log 1 7F21H | Log 4 | Log 3 · · · 7F3FH | Log 64 | Log 63 '1' is stored in the communication event corresponding to each bit.	0		

Buffer memory list for simple CPU communication

Address Dec (Hex)		Purpose	Name	Default value	Target	Reference
CH1	CH2					
32640 to 32671 (7F80H to 7F9FH)		Communication status area for simple CPU communication	Request to start communication at request for each setting number A request contact to start data transmission when "On Request" is selected for the communication setting is stored. OFF → ON: Requested ON → OFF: Completed For 512 settings (1 bit per setting)	0	R, W	—
32672 to 32703 (7FA0H to 7FBFH)			Request to stop communication for each setting number A request contact to stop data transmission when "Fixed Interval" is selected for the communication setting is stored. OFF → ON: Requested ON → OFF: Completed For 512 settings (1 bit per setting)	0	R, W	—
32704 to 32735 (7FC0H to 7FDFH)			Request to restart communication for each setting number A request contact to restart data transmission when "Fixed Interval" is selected for the communication setting is stored. OFF → ON: Requested ON → OFF: Completed For 512 settings (1 bit per setting)	0	R, W	—
32736 to 32767 (7FE0H to 7FFFH)			Execution status flag for each setting number The data transmission/reception status is stored. ON: Communicating OFF: Communication stop For 512 settings (1 bit per setting)	0	R	—
32768 to 32799 (8000H to 801FH)			Preparation completion for each setting number The preparation completion status of the simple CPU communication is stored. ON: Ready OFF: Not set, not ready For 512 settings (1 bit per setting)	0	R	—
32800 to 32831 (8020H to 803FH)			System area	—		

Address Dec (Hex)		Purpose	Name		Default value	Target	Reference
CH1	CH2						
32832 to 32863 (8040H to 805FH)		Diagnostic information area for simple CPU communication	Setting status of each setting number The setting status of each setting number is stored. ON: Set OFF: Not set For 512 settings (1 bit per setting)		0	R	—
32864 (8060H)			Number of settings		0	R	—
32865 (8061H)			Error clear request A request to clear the latest error code and exception codes for setting number 1 to 512 diagnostic information is stored. 0 → 1: Requested 1 → 0: Completed		0	R, W	—
32866 to 32871 (8062H to 8067H)			System area		—		
32872 (8068H)			Setting No.1 diagnostic information	Setting No.	0	R	☞ Page 132 Checking Simple CPU Communication Status
32873 (8069H)				Communication pattern	0	R	
32874 (806AH)				Communication setting	0	R	
32875 (806BH)				Communication destination	0	R	
32876 (806CH)				Communication status	0	R	
32877 (806DH)				Station number	0	R	
32878 (806EH)				Host station CH	0	R	
32879 (806FH)				Target PLC number	0	R	
32880 to 32881 (8070H to 8071H)				Execution interval [ms] (current value)	0	R	
32882 to 32883 (8072H to 8073H)				Execution interval [ms] (maximum value)	0	R	
32884 to 32885 (8074H to 8075H)				Execution interval [ms] (minimum value)	0	R	
32886 to 32887 (8076H to 8077H)				Number of times for normal completion	0	R	
32888 to 32889 (8078H to 8079H)				Number of times for abnormal completion	0	R	
32890 to 32891 (807AH to 807BH)				Number of retries	0	R	
32892 (807CH)				Latest error code	0	R	
32893 (807DH)				Exception code	0	R	
32894 to 32895 (807EH to 807FH)				System area	—		
32896 to 45159 (8080H to B067H)			Setting No.2 to 512 diagnostic information (same as setting No.1 diagnostic information)				

■Addresses of setting No.2 diagnostic information and later

Each address of setting No.2 diagnostic information and later can be calculated with the following formula:

- Address of the buffer memory corresponding to setting No.1 + ((Target setting number - 1) × 24)

Ex.

The address of 'Latest error code' of setting No.512 is as follows:

- 32892+((512-1)×24)=45156

Monitoring device areas for the programmable controller CPU monitoring function

■CH1 side buffer memory address: decimal (hexadecimal)

N-th block monitoring device										Name
1	2	3	4	5	6	7	8	9	10	
8272 (2050H)	8281 (2059H)	8290 (2062H)	8299 (206BH)	8308 (2074H)	8317 (207DH)	8326 (2086H)	8335 (208FH)	8344 (2098H)	8353 (20A1H)	Monitoring device designation
8273 to 8274 (2051H to 2052H)	8282 to 8283 (205AH to 205BH)	8291 to 8292 (2063H to 2064H)	8300 to 8301 (206CH to 206DH)	8309 to 8310 (2075H to 2076H)	8318 to 8319 (207EH to 207FH)	8327 to 8328 (2087H to 2088H)	8336 to 8337 (2090H to 2091H)	8345 to 8346 (2099H to 209AH)	8354 to 8355 (20A2H to 20A3H)	Start device No. specification
8275 (2053H)	8284 (205CH)	8293 (2065H)	8302 (206EH)	8311 (2077H)	8320 (2080H)	8329 (2089H)	8338 (2092H)	8347 (209BH)	8356 (20A4H)	Number of read points specification
8276 (2054H)	8285 (205DH)	8294 (2066H)	8303 (206FH)	8312 (2078H)	8321 (2081H)	8330 (208AH)	8339 (2093H)	8348 (209CH)	8357 (20A5H)	Monitoring condition specification (judgment condition specification)
8277 (2055H)	8286 (205EH)	8295 (2067H)	8304 (2070H)	8313 (2079H)	8322 (2082H)	8331 (208BH)	8340 (2094H)	8349 (209DH)	8358 (20A6H)	Monitoring condition value designation
8278 (2056H)	8287 (205FH)	8296 (2068H)	8305 (2071H)	8314 (207AH)	8323 (2083H)	8332 (208CH)	8341 (2095H)	8350 (209EH)	8359 (20A7H)	Send pointer designation (for condition match send, data transmission)
8279 (2057H)	8288 (2060H)	8297 (2069H)	8306 (2072H)	8315 (207BH)	8324 (2084H)	8333 (208DH)	8342 (2096H)	8351 (209FH)	8360 (20A8H)	Output quantity designation (for condition match send, data transmission)
8280 (2058H)	8289 (2061H)	8298 (206AH)	8307 (2073H)	8316 (207CH)	8325 (2085H)	8334 (208EH)	8343 (2097H)	8352 (20A0H)	8361 (20A9H)	Data No. for connection designation (for condition match send)

■CH2 side buffer memory address: decimal (hexadecimal)

N-th block monitoring device										Name
1	2	3	4	5	6	7	8	9	10	
8528 (2150H)	8537 (2159H)	8546 (2162H)	8555 (216BH)	8564 (2174H)	8573 (217DH)	8582 (2186H)	8591 (218FH)	8600 (2198H)	8609 (21A1H)	Monitoring device designation
8529 to 8530 (2151H to 2152H)	8538 to 8539 (215AH to 215BH)	8547 to 8548 (2163H to 2164H)	8556 to 8557 (216CH to 216DH)	8565 to 8566 (2175H to 2176H)	8574 to 8575 (217EH to 217FH)	8583 to 8584 (2187H to 2188H)	8592 to 8593 (2190H to 2191H)	8601 to 8602 (2199H to 219AH)	8610 to 8611 (21A2H to 21A3H)	Start device No. specification
8531 (2153H)	8540 (215CH)	8549 (2165H)	8558 (216EH)	8567 (2177H)	8576 (2180H)	8585 (2189H)	8594 (2192H)	8603 (219BH)	8612 (21A4H)	Number of read points specification
8532 (2154H)	8541 (215DH)	8550 (2166H)	8559 (216FH)	8568 (2178H)	8577 (2181H)	8586 (218AH)	8595 (2193H)	8604 (219CH)	8613 (21A5H)	Monitoring condition specification (judgment condition specification)
8533 (2155H)	8542 (215EH)	8551 (2167H)	8560 (2170H)	8569 (2179H)	8578 (2182H)	8587 (218BH)	8596 (2194H)	8605 (219DH)	8614 (21A6H)	Monitoring condition value designation
8534 (2156H)	8543 (215FH)	8552 (2168H)	8561 (2171H)	8570 (217AH)	8579 (2183H)	8588 (218CH)	8597 (2195H)	8606 (219EH)	8615 (21A7H)	Send pointer designation (for condition match send, data transmission)
8535 (2157H)	8544 (2160H)	8553 (2169H)	8562 (2172H)	8571 (217BH)	8580 (2184H)	8589 (218DH)	8598 (2196H)	8607 (219FH)	8616 (21A8H)	Output quantity designation (for condition match send, data transmission)
8536 (2158H)	8545 (2161H)	8554 (216AH)	8563 (2173H)	8572 (217CH)	8581 (2185H)	8590 (218EH)	8599 (2197H)	8608 (21A0H)	8617 (21A9H)	Data No. for connection designation (for condition match send)

Appendix 4 Dedicated Instructions

Dedicated instructions are used to simplify programming when using intelligent function module functions.

For details, refer to the following manual.

📖MELSEC iQ-R Programming Manual (Module Dedicated Instructions)

Dedicated instruction list

List of dedicated instruction

The following table lists the dedicated instructions explained.

Purpose	Dedicated instruction	Function overview
On-demand function transmission	ONDEMAND*1	Sends data using the on-demand function of MC protocol.
Predefined protocol function	CPRTCL	Executes the protocol setting data registered using the Predefined protocol support function of Engineering tool.
Nonprocedural protocol communication	OUTPUT*1	Transmits data for the designated data count.
	INPUT*1	Reads received data.
Bidirectional protocol communication	BIDOUT*1	Transmits data for the designated data count.
	BIDIN*1	Reads received data.
Communication status confirmation	SPBUSY	Reads the status of data transmission/reception using the dedicated instruction.
Data transmission/reception	BUFRCVS*1	Receives data by an interrupt program using nonprocedural or bidirectional protocol communication.
	PRR*1	Transmits data with the user frame according to the contents defined in the send user frame designation area in data communications using the nonprocedural protocol.
Receive data clear	CSET	Performs the receive data clear without stopping the transmission processing in data communications using the nonprocedural protocol.
Programmable controller CPU monitoring function		Registers/cancels programmable controller CPU monitoring in order to use the programmable controller CPU monitoring functions.
Setting the units of send/receive data quantity		Sets the unit (word/byte) of data quantity to be sent and received.
User frame registration to flash ROM (registration/reading of user frame)	PUTE	Registers a user frame.
	GETE	Reads a user frame.
Mode switching	UINI	Changes the mode, transmission specifications, and host station No. of C24.

*1 To change the following setting values of the buffer memory used by the dedicated instructions, use Engineering tool or execute the CSET instruction (initial setting) before starting data communication. (📖MELSEC iQ-R Programming Manual (Module Dedicated Instructions))
·'Word/byte units designation' (Un\G150/310)
·'Buffer memory start address designation' (Un\G160/321)
·'Send buffer memory start address designation' (Un\G162/322)
·'Send buffer memory length designation' (Un\G163/323)
·'Receiving buffer memory start address designation' (Un\G166/326)
·'Receive buffer memory length designation' (Un\G167/327)
The dedicated instructions are executed base on the setting values of the buffer memory at the module startup or those which were changed by the CSET instruction (initial setting).

Precautions

Do not change data (control data, request data, etc.) designated by a dedicated instruction until the execution of that instruction is completed.

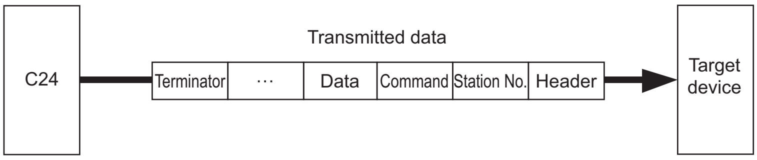
Appendix 5 Operation Image and Data Structure of Predefined Protocol

Operation image of each communication type of protocol

In the predefined protocol function, communication with target devices is performed through the communication type "Send only", "Receive only", or "Send & receive". This section explains the respective operation images.

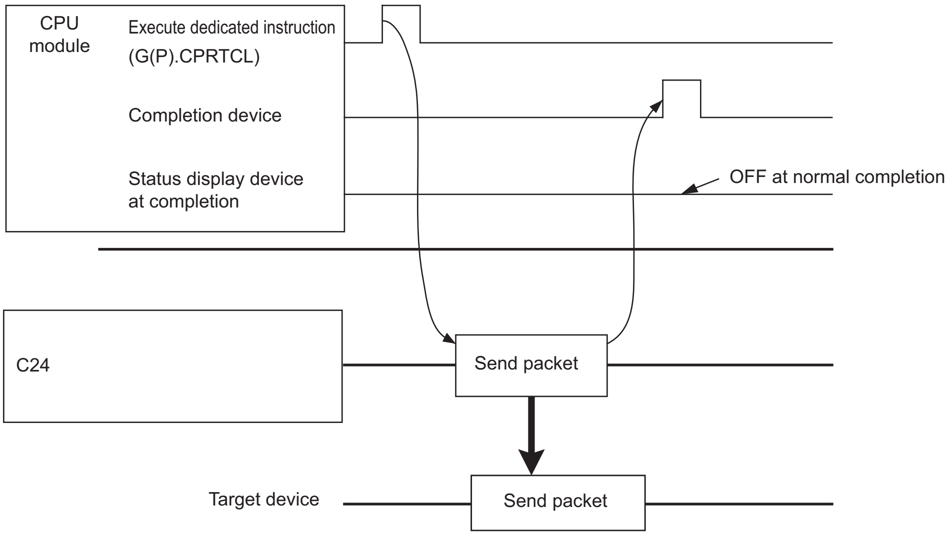
When communication type is "Send Only"

The specified packet is transmitted once.



The operation image of "Send only" is as follows.

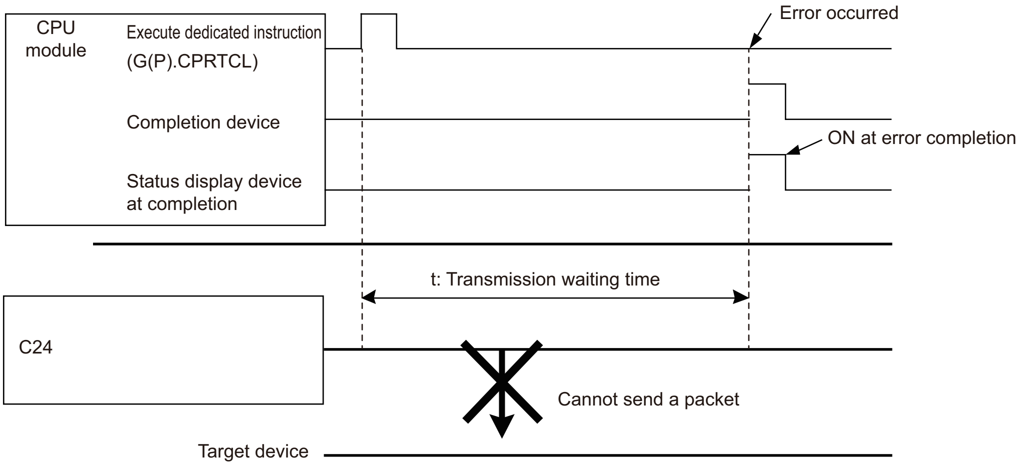
■Normal completion



■Error completion (transmission monitoring timeout error)

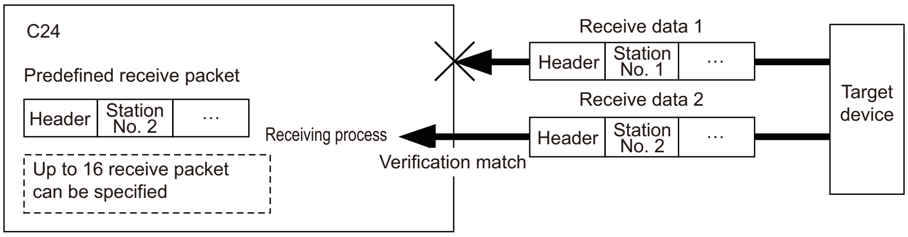
Example of setting

Standby time: 0, Number of send retries: 0, Monitoring time: other than 0


CPU module
Execute dedicated instruction (G(P).CPRTCL)
Completion device
Status display device at completion
Error occurred
ON at error completion
t: Transmission waiting time
C24
Cannot send a packet
Target device

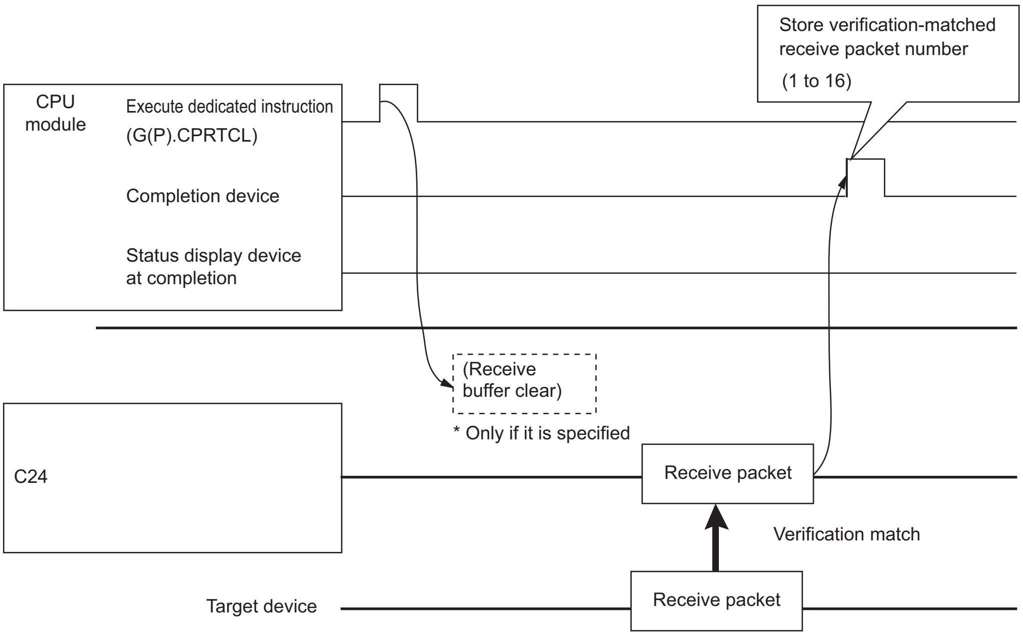
When communication type is "Receive Only"

When data are received from the target device, the process completes when the received data matches the receive packet and the receiving process is performed.

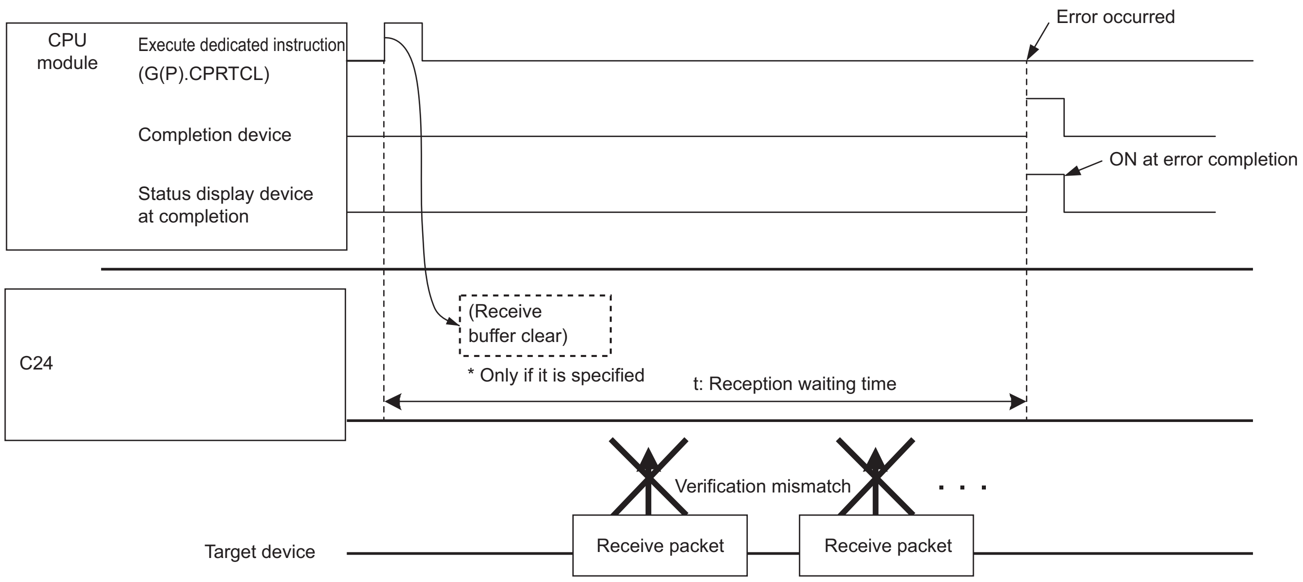


The operation image of "Receive only" is as follows.

■Normal completion


Store verification-matched receive packet number (1 to 16)
CPU module
Execute dedicated instruction (G(P).CPRTCL)
Completion device
Status display device at completion
(Receive buffer clear)
* Only if it is specified
C24
Receive packet
Verification match
Target device
Receive packet

■Error completion (receive wait timeout error)

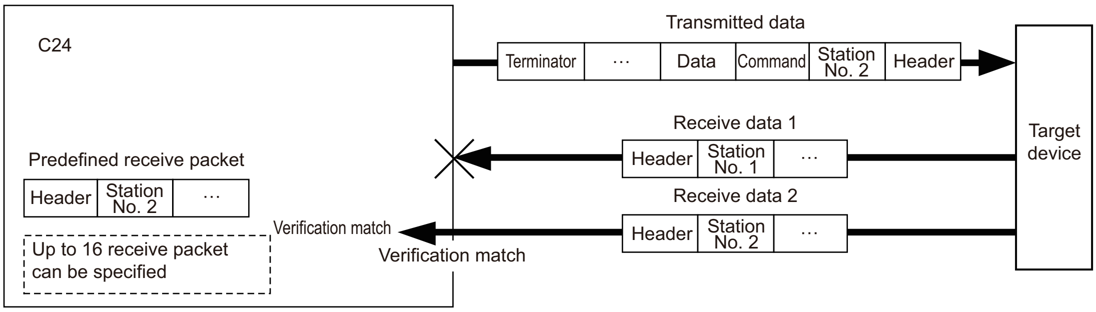


Point

- When variables are included in receive packet elements, variable parts are not verified.
- When more than one receive packet is specified, received data is verified with the receive packet information of the first registered packet in the order of registration. The receive processing is performed once received data match one of the receive packet number, and further verification is not performed.
- The receive packet number which is matched as the result of the verification is stored in the control data of the dedicated instruction (CPRTCL instruction).

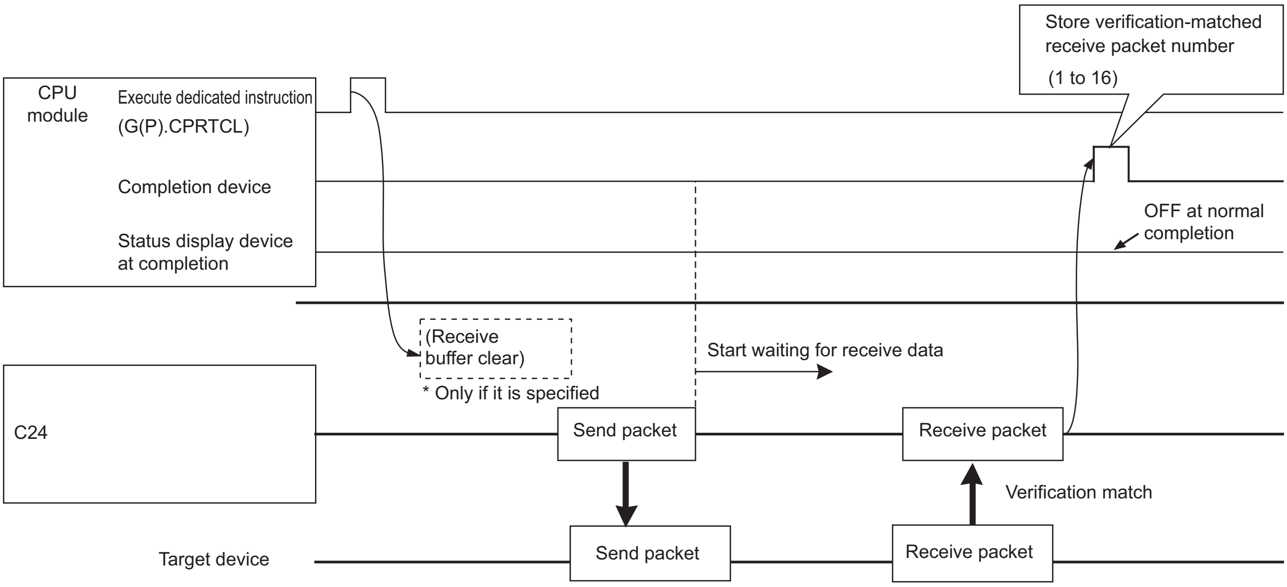
When communication type is "Send & Receive"

A packet is sent once, and the status changes to the data receive wait stat after the transmission completes normally. Then, data is received from the target device, and the process completes when the received data matches the receive packet and the receiving process is performed.

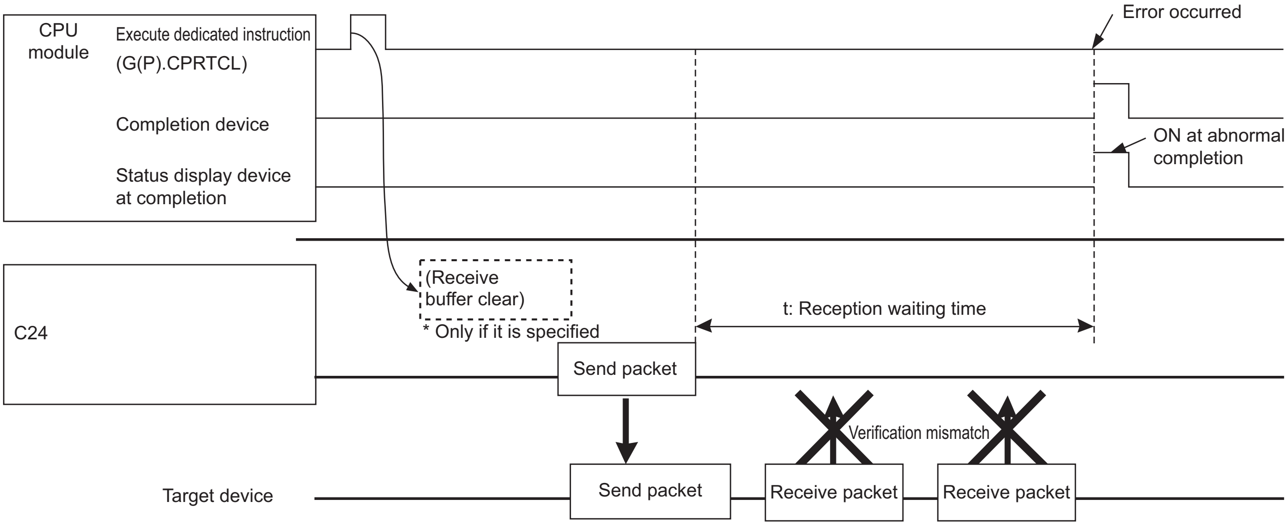


The operation image of "Send & receive" is as follows.

■Normal completion


Store verification-matched receive packet number (1 to 16)
CPU module
Execute dedicated instruction (G(P).CPRTCL)
Completion device
Status display device at completion
OFF at normal completion
(Receive buffer clear)
* Only if it is specified
Start waiting for receive data
C24
Send packet
Receive packet
Verification match
Target device
Send packet
Receive packet

■Error completion (receive wait timeout error)

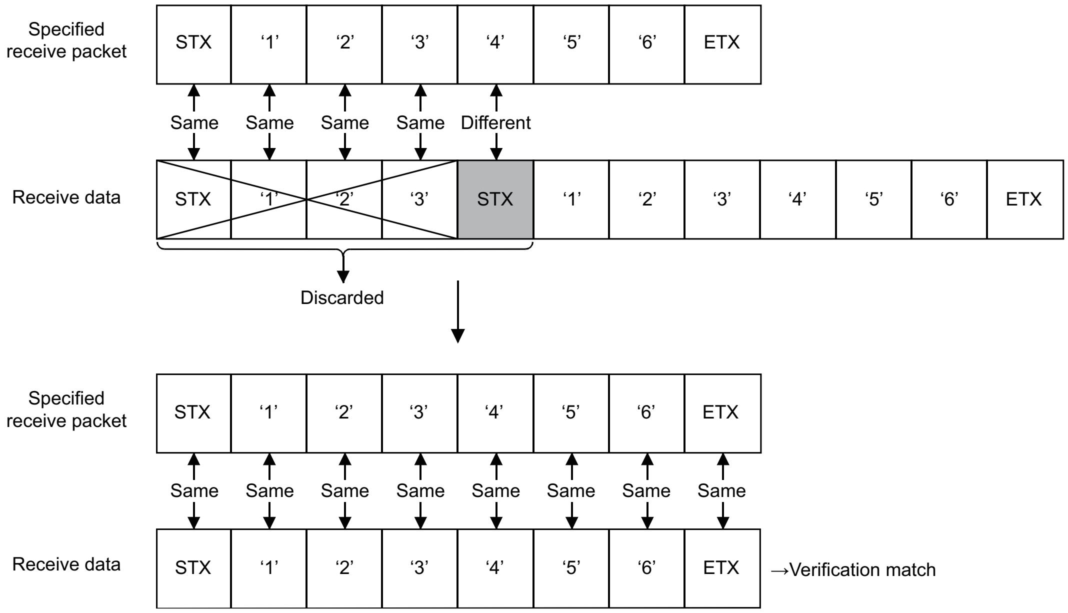

CPU module
Execute dedicated instruction (G(P).CPRTCL)
Completion device
Status display device at completion
Error occurred
ON at abnormal completion
C24
(Receive buffer clear)
* Only if it is specified
t: Reception waiting time
Send packet
Verification mismatch
Target device
Send packet
Receive packet
Receive packet

Point

- When variables are included in receive packet elements, variable parts are not verified.
- When more than one receive packet is specified, received data is verified with the receive packet information of the first registered packet in the order of registration. The receive processing is performed once received data match one of the receive packet number, and further verification is not performed.
- The receive packet number which is matched as the result of the verification is stored in the control data of the dedicated instruction (CPRTCL instruction).

Verification operation of receive packet

The following shows C24 operation when data that are different from the specified receive packet are received.
Receive data prior to the different data is discarded. Data is compared again from the start of the receive packet, and once the data matches the receive packet, the data receiving operation is processed.



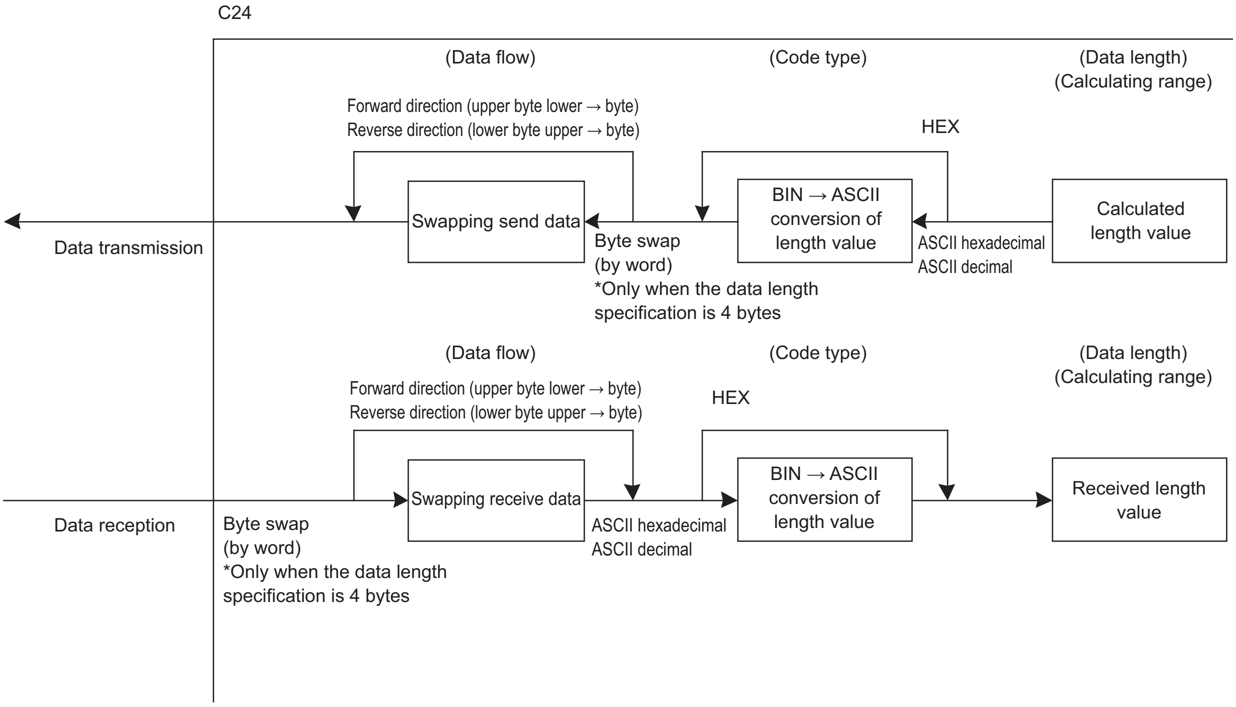
Data examples of packet elements

This section explains the processing procedures and practical data examples of elements that can be placed in a packet.

Length

■Processing procedure

C24 processes length according to the following procedure.


C24
(Data flow)
(Code type)
(Data length)
(Calculating range)
Forward direction (upper byte lower → byte)
Reverse direction (lower byte upper → byte)
HEX
Swapping send data
BIN → ASCII conversion of length value
Calculated length value
Data transmission
Byte swap
(by word)
*Only when the data length specification is 4 bytes
ASCII hexadecimal
ASCII decimal
(Data flow)
(Code type)
(Data length)
(Calculating range)
Forward direction (upper byte lower → byte)
Reverse direction (lower byte upper → byte)
HEX
Swapping receive data
BIN → ASCII conversion of length value
Received length value
Data reception
Byte swap
(by word)
*Only when the data length specification is 4 bytes
ASCII hexadecimal
ASCII decimal

■Data example

The following is an example in the case where the calculated value of length is 258 bytes in decimal (258 is 102H in hexadecimal).

- Data flow is "Forward direction"

Code type	Data length[*1]			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	'2' (32H)	'02' (30H 32H)	'102' (31H 30H 32H)	'0102' (30H 31H 30H 32H)
ASCII decimal	'8' (38H)	'58' (35H 38H)	'258' (32H 35H 38H)	'0258' (30H 32H 35H 38H)
HEX	02H	0102H	000102H	00000102H

*1 Values in '( )' indicate ASCII code.

- Data flow is "Reverse direction"

Code type	Data length[*1]			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	—	'20' (32H 30H)	'201' (32H 30H 31H)	'2010'(32H 30H 31H 30H)
ASCII decimal	—	'85' (38H 35H)	'852' (38H 35H 32H)	'8520' (38H 35H 32H 30H)
HEX	—	0201H	020100H	02010000H

*1 Values in '( )' indicate ASCII code.

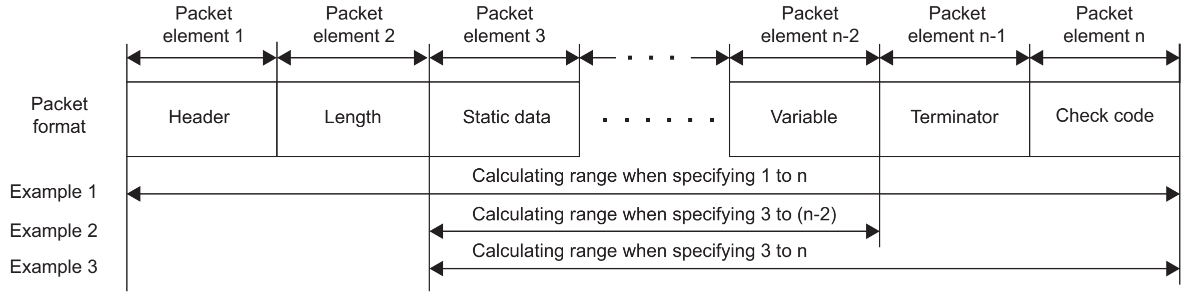
- Data flow is "Byte swap"

Code type	Data length[*1]			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	—	—	—	'1020' (31H 30H 32H 30H)
ASCII decimal	—	—	—	'2085' (32H 30H 38H 35H)
HEX	—	—	—	00000201H

*1 Values in '( )' indicate ASCII code.

■Calculating range

The following shows specification examples of the calculating range of Length.



Example 1: Calculating range when 1 and n are specified as the start and the end respectively.

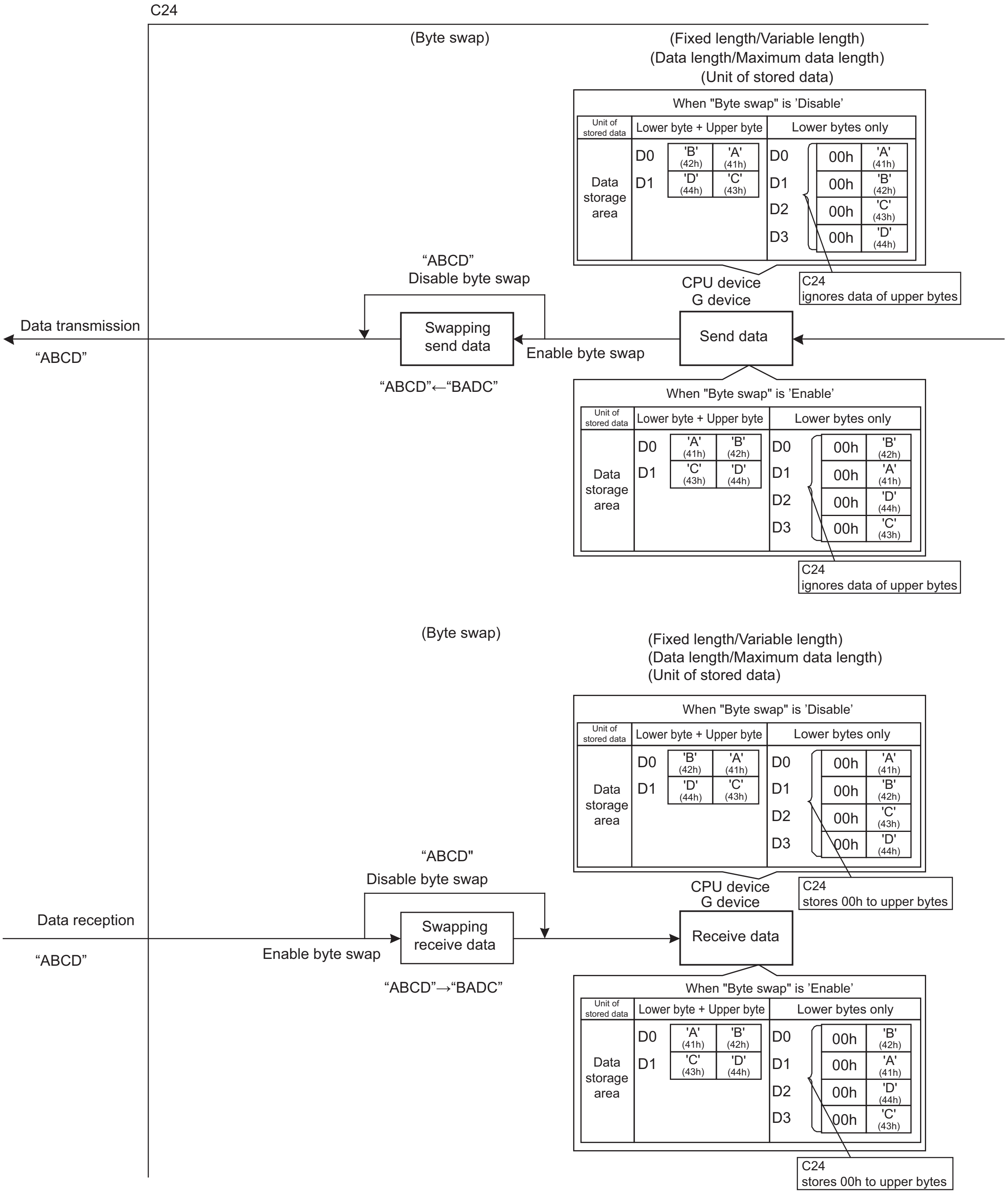
Example 2: Calculating range when 3 and n-2 are specified as the start and the end respectively.

Example 3: Calculating range when 3 and n are specified as the start and the end respectively.

Non-conversion variable

■Processing procedure

C24 processes non-conversion variable according to the following procedure.



■Data example

- The following table shows data to be stored in the data storage area when the string of send data is 'ABCD'.
  (Reference: A=41H, B=42H, C=43H, and D=44H in ASCII code)

Item	Description			
Fixed length/Variable length	Fixed length			
Data length	4 bytes			
Start address of data storage area	D0			
Unit of stored data	Lower byte + Upper byte		Lower bytes only	
Byte swap	Disable	Enable	Disable	Enable

Data to be stored in data storage area	D0=4241H
	D1=4443H
		D0=4142H
		D1=4344H
			D0=0041H
			D1=0042H
			D2=0043H
			D3=0044H
				D0=0042H
				D1=0041H
				D2=0044H
				D3=0043H

- The following table shows data to be stored in the data storage area when the string of send data is 'EFG'.
  (Reference: E=45H, F=46H, and G=47H in ASCII code)

Item	Description			
Fixed length/Variable length	Fixed length			
Data length	3 bytes			
Start address of data storage area	D0			
Unit of stored data	Lower byte + Upper byte		Lower bytes only	
Byte swap	Disable	Enable	Disable	Enable

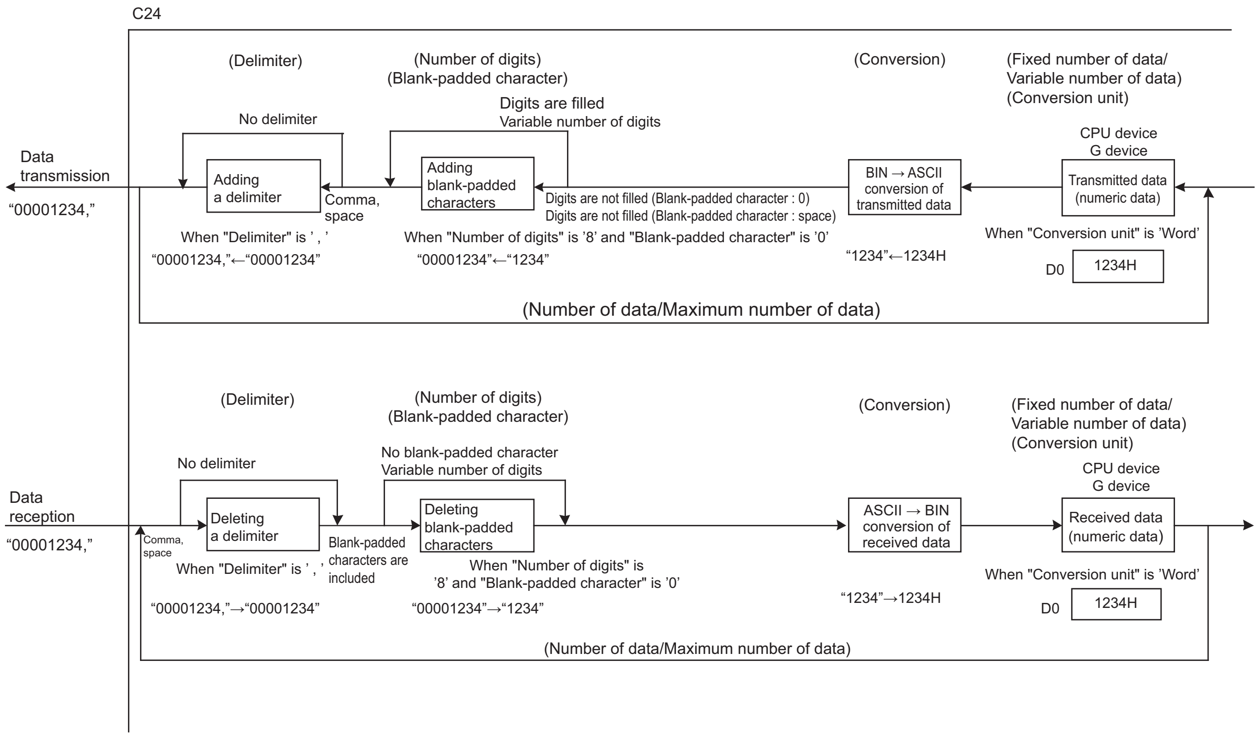
Data to be stored in data storage area	D0=4645H
	D1=0047H
		D0=4546H
		D1=4700H
			D0=0045H
			D1=0046H
			D2=0047H
			D3= (any data)
				D0=0046H
				D1=0045H
				D2=0047H
				D3= (any data)

Conversion variable

■Processing procedure

C24 processes conversion variable according to the following procedure.

- When "Conversion" is 'HEX → ASCII hexadecimal' or 'ASCII hexadecimal → HEX'



Blank-padded characters

When transmitting data, upper digits are filled with the data specified in "Blank-padded character" when the number of digits is less than that specified in "Number of Send Digits of Data".

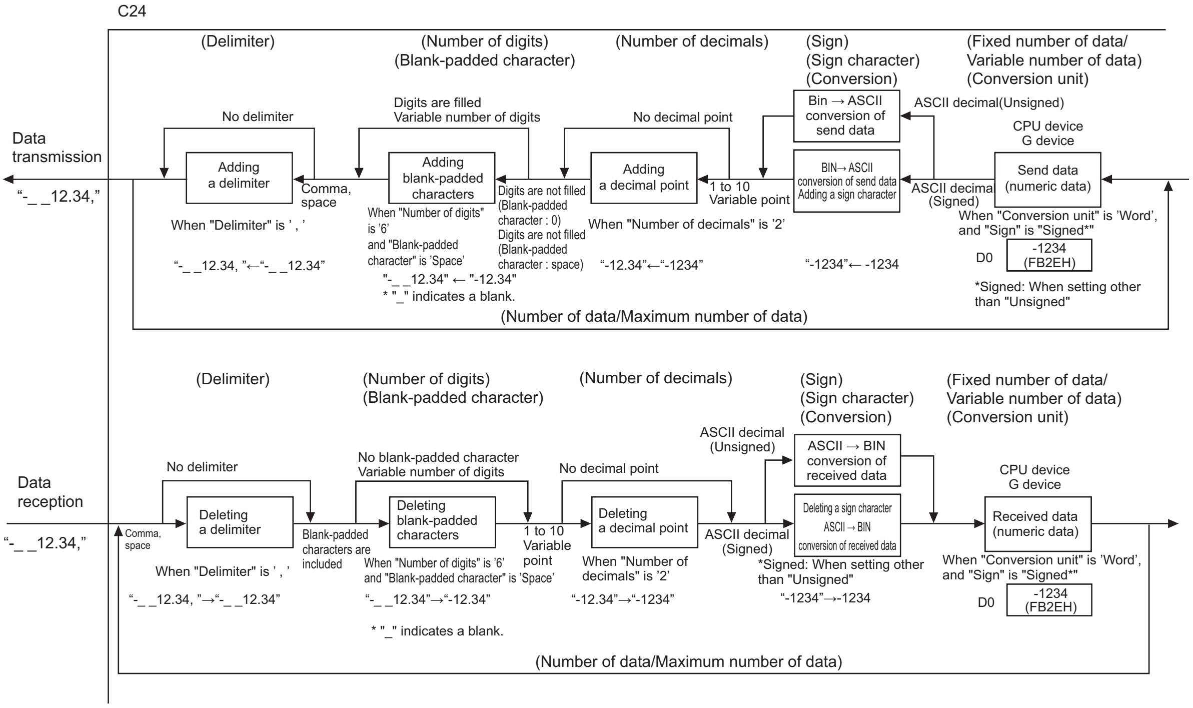
When receiving data, either '0' or '_(space)' is processed as a blank-padded character regardless of the specification in "Blank-padded character".

Ex.

Setting of "Number of Send Digits of Data" is '6' ('_' indicates a space character in the table)

Received data	Operation of C24
000120	Considers the first 3 digits as blank-padded characters.
_ _ 0120	Considers the first 3 digits as blank-padded characters.
0 _ 0120	Considers the first 3 digits as blank-padded characters.
_ _ _ 120	Considers the first 3 digits as blank-padded characters.
00012 _	Considers it to be an ASCII → BIN conversion error (7F20H).
_ _ _ 12 _	Considers it to be an ASCII → BIN conversion error (7F20H).
0001 _ 0	Considers it to be an ASCII → BIN conversion error (7F20H).

- When "Conversion" is "HEX → ASCII Decimal" or "ASCII Decimal → HEX"


C24
(Delimiter)
(Number of digits)
(Blank-padded character)
(Number of decimals)
(Sign)
(Sign character)
(Conversion)
(Fixed number of data/
Variable number of data)
(Conversion unit)
No delimiter
Digits are filled
Variable number of digits
No decimal point
Bin → ASCII conversion of send data
ASCII decimal(Unsigned)
CPU device
G device
Data transmission
Adding a delimiter
Adding blank-padded characters
Adding a decimal point
BIN→ASCII conversion of send data Adding a sign character
Send data (numeric data)
"-_ _12.34,"
Comma, space
Digits are not filled (Blank-padded character : 0) Digits are not filled (Blank-padded character : space)
1 to 10 Variable point
ASCII decimal (Signed)
When "Delimiter" is ' , '
When "Number of digits" is '6' and "Blank-padded character" is 'Space'
When "Number of decimals" is '2'
When "Conversion unit" is 'Word', and "Sign" is "Signed*"
D0
-1234 (FB2EH)
"-_ _12.34, "←"-_ _12.34"
"-_ _12.34" ← "-12.34"
* "_" indicates a blank.
"-12.34"←"-1234"
"-1234"← -1234
*Signed: When setting other than "Unsigned"
(Number of data/Maximum number of data)
(Delimiter)
(Number of digits)
(Blank-padded character)
(Number of decimals)
(Sign)
(Sign character)
(Conversion)
(Fixed number of data/
Variable number of data)
(Conversion unit)
ASCII decimal (Unsigned)
ASCII → BIN conversion of received data
No delimiter
No blank-padded character
Variable number of digits
No decimal point
CPU device
G device
Data reception
Deleting a delimiter
Deleting blank-padded characters
Deleting a decimal point
Deleting a sign character ASCII→BIN conversion of received data
Received data (numeric data)
"-_ _12.34,"
Comma, space
Blank-padded characters are included
1 to 10 Variable point
ASCII decimal (Signed)
When "Delimiter" is ' , '
When "Number of digits" is '6' and "Blank-padded character" is 'Space'
When "Number of decimals" is '2'
*Signed: When setting other than "Unsigned"
When "Conversion unit" is 'Word', and "Sign" is "Signed*"
"-_ _12.34, "→"-_ _12.34"
"-_ _12.34"→"-12.34"
"-12.34"→"-1234"
"-1234"→-1234
D0
-1234 (FB2EH)
* "_" indicates a blank.
(Number of data/Maximum number of data)

■Data example

The following table shows send data when a packet consists of [Header], [Conversion variable], [Terminator] and data stored in the data storage area is D0=837 (0345H), D1=18 (0012H). (Reference: 120345H =1180485 in decimal form)

- Data example 1

Item	Setting contents		
Fixed number of data/Variable number of data	Fixed number of data	Fixed number of data	Fixed number of data
Number of Send Data	1	1	1
Start address of data storage area	D0	D0	D0
Conversion unit	Word	Word	Word
Conversion	HEX → ASCII decimal	HEX → ASCII decimal	HEX → ASCII decimal
Number of Send Digits of Data	5	5	Variable number of digits
Blank-padded character	0	Space	—(Not applicable)
Sign	Unsigned	Signed	Signed
Sign character	—(Not applicable)	+	+
Number of decimals	No decimal point	2	No decimal point
Delimiter	No delimiter	Comma	Comma
Send data[*1]	[Header]00837[Terminator]	[Header]+_ _ 8.37, [Terminator]	[Header]+837, [Terminator]

*1 The "_" indicates a space.

- Data example 2

Item	Setting contents		
Fixed number of data/Variable number of data	Fixed number of data	Fixed number of data	Fixed number of data
Number of Send Data	1	2	2
Start address of data storage area	D0	D0	D0
Conversion unit	Double word	Word	Word
Conversion	HEX → ASCII decimal	HEX → ASCII decimal	HEX → ASCII decimal
Number of Send Digits of Data	10	5	5
Blank-padded character	0	Space	0
Sign	Signed	Unsigned	Signed
Sign character	+	—(Not applicable)	+
Number of decimals	8	No decimal point	2
Delimiter	No delimiter	No delimiter	Comma
Send data[*1]	[Header]+00.01180485[Terminator]	[Header]_ _ 837 _ _ _18[Terminator]	[Header]+008.37, +000.18[Terminator]

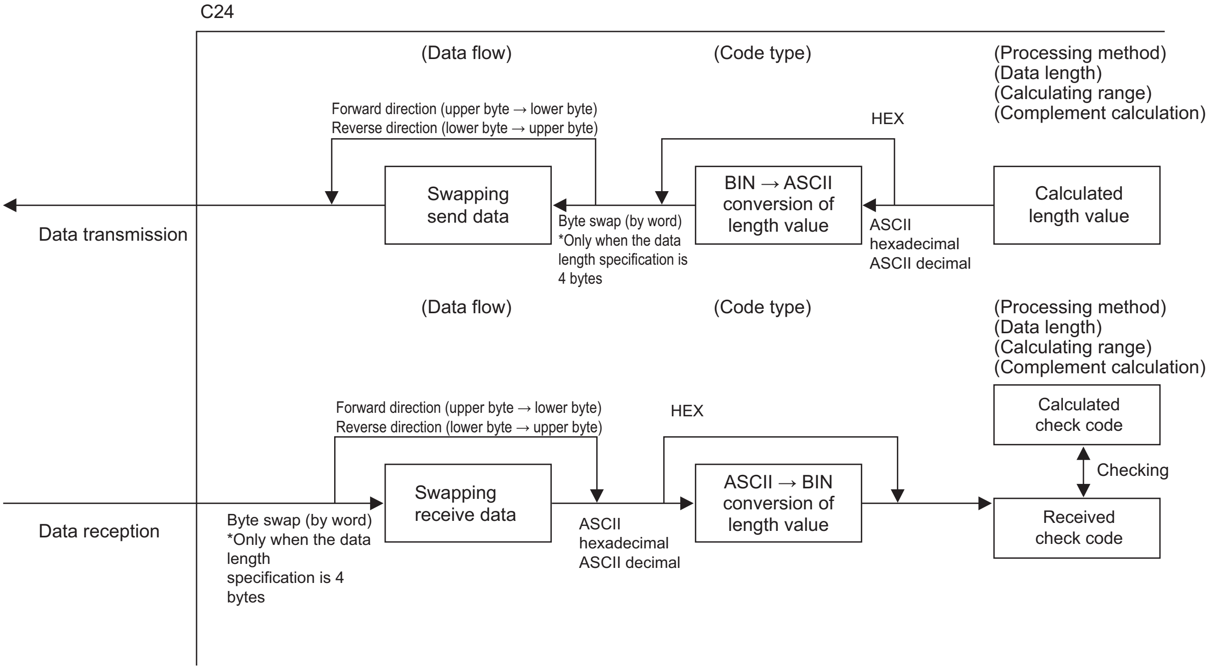
*1 '_' indicates a blank.

Check code

■Processing procedure

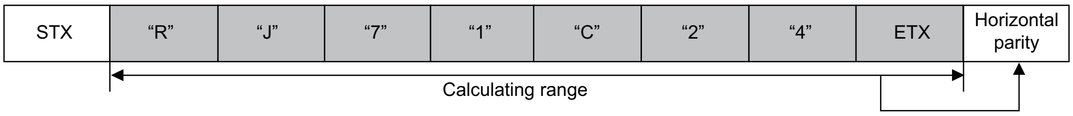
C24 processes check code according to the following procedure.

1. Calculates value according to the selection of "Processing method".
2. When "Complement calculation" is 'One's complement' or 'Two's complement', performs a 2-word-wise complement operation on the value calculated in the above 1.
3. When "Code type" is 'ASCII decimal', extracts the lowest one word from the value calculated in the above 2, and performs the hexadecimal to decimal conversion.



■Calculation procedure for horizontal parity

The following show procedures for calculating horizontal parities using the following sample data.



For the packet shown above

```
"R"  (52H)  0101 0010
                XOR
"J"  (4AH)  0100 1010 =  0001 1000
                             XOR
"7"  (37H)               0011 0111  =  0010 1111
                                          XOR
"1"  (31H)                             0011 0001  =  0001 1110
                                                        XOR
"C"  (43H)                                           0100 0011  =  0101 1101
                                                                      XOR
"2"  (32H)                                                         0011 0010  =  0110 1111
                                                                                    XOR
"4"  (34H)                                                                       0011 0100  =  0101 1011
                                                                                                  XOR
ETX  (03H)                                                                                     0000 0011  =  0101 1000  (Binary)

                                                                                  Horizontal parity = 58    (Hexadecimal)
```

■ Data flow: Forward direction

- "No Complement Calculation" is designated (58H in hexadecimal is 88 in decimal)

Code type	Data length*1			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	"8" (38H)	"58" (35H 38H)	"058"(30H 35H 38H)	"0058" (30H 30H 35H 38H)
ASCII decimal	"8" (38H)	"88" (38H 38H)	"088" (30H 38H 38H)	"0088" (30H 30H 38H 38H)
HEX	58H	0058H	000058H	00000058H

*1 Values in '( )' indicate ASCII code.

- "One's complement" is designated (One's complement for 0000 0058H is FFFF FFA7H)
  When "Code type" is "ASCII decimal", the lower one word is extracted and converted from hexadecimal to decimal. (FFA7H in hexadecimal is 65447 in decimal.)

Code type	Data length*1			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	"7"(37H)	"A7" (41H 37H)	"FA7" (46H 41H 37H)	"FFA7" (46H 46H 41H 37H)
ASCII decimal	"7" (37H)	"47" (34H 37H)	"447" (34H 34H 37H)	"5447" (35H 34H 34H 37H)
HEX	A7H	FFA7H	FFFFA7H	FFFFFFA7H

*1 Values in '( )' indicate ASCII code.

- "Two's complement" is designated (Two's complement for 0000 0058H is FFFF FFA8H)
  When "Code type" is "ASCII decimal", the lower one word is extracted and converted from hexadecimal to decimal. (FFA8H in hexadecimal is 65448 in decimal.)

Code type	Data length*1			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	"8" (38H)	"A8" (41H 38H)	"FA8" (46H 41H 38H)	"FFA8" (46H 46H 41H 38H)
ASCII decimal	"8" (38H)	"48" (34H 38H)	"448" (34H 38H 38H)	"5448" (35H 34H 34H 38H)
HEX	A8H	FFA8H	FFFFA8H	FFFFFFA8H

*1 Values in '( )' indicate ASCII code.

■ Data flow: Reverse direction

- "No Complement Calculation" is designated (58H in hexadecimal is 88 in decimal)

Code type	Data length*1			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	—	"85" (38H 35H)	"850" (38H 35H 30H)	"8500" (38H 35H 30H 30H)
ASCII decimal	—	"88" (38H 38H)	"880" (38H 38H 30H)	"8800" (38H 38H 30H 30H)
HEX	—	5800H	580000H	58000000H

*1 Values in '( )' indicate ASCII code.

- "One's complement" is designated (One's complement for 0000 0058H is FFFF FFA7H)
  When "Code type" is "ASCII decimal", the lower one word is extracted and converted from hexadecimal to decimal. (FFA7H in hexadecimal is 65447 in decimal.)

Code type	Data length*1			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	—	"7A" (37H 41H)	"7AF" (37H 41H 46H)	"7AFF" (37H 41H 46H 46H)
ASCII decimal	—	"74" (37H 34H)	"744" (37H 34H 34H)	"7445" (37H 34H 34H 35H)
HEX	—	A7FFH	A7FFFFH	A7FFFFFFH

*1 Values in '( )' indicate ASCII code.

- "Two's complement" is designated (Two's complement for 0000 0058H is FFFF FFA8H)
  When "Code type" is "ASCII decimal", the lower one word is extracted and converted from hexadecimal to decimal. (FFA8H in hexadecimal is 65448 in decimal.)

Code type	Data length*1			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	—	"8A" (38H 41H)	"8AF" (38H 44H 46H)	"8AFF" (38H 41H 46H 46H)
ASCII decimal	—	"84" (38H 34H)	"844" (38H 34H 34H)	"8445" (38H 34H 34H 35H)
HEX	—	A8FFH	A8FFFFH	A8FFFFFFH

*1 Values in '( )' indicate ASCII code.

■ Data Flow: Byte swap

- "No Complement Calculation" is designated (58H in hexadecimal is 88 in decimal)

Code type	Data length[1]			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	—	—	—	"0085" (30H 30H 38H 35H)
ASCII decimal	—	—	—	"0088" (30H 30H 38H 38H)
HEX	—	—	—	00005800H

[1] Values in '( )' indicate ASCII code.

- "One's complement" is designated (One's complement for 0000 0058H is FFFF FFA7H)
  When "Code type" is "ASCII decimal", the lower one word is extracted and converted from hexadecimal to decimal. (FFA7H in hexadecimal is 65447 in decimal.)

Code type	Data length[1]			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	—	—	—	"FF7A" (46H 46H 37H 41H)
ASCII decimal	—	—	—	"4574" (34H 35H 37H 34H)
HEX	—	—	—	FFFFA7FFH

[1] Values in '( )' indicate ASCII code.

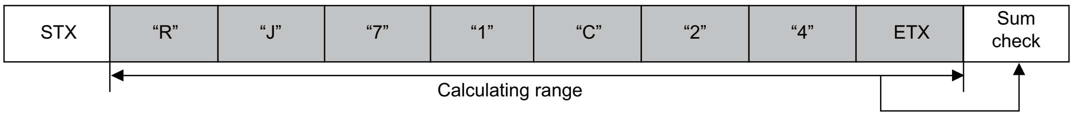
- "Two's complement" is designated (Two's complement for 0000 0058H is FFFF FFA8H)
  When "Code type" is "ASCII decimal", the lower one word is extracted and converted from hexadecimal to decimal. (FFA8H in hexadecimal is 65448 in decimal.)

Code type	Data length[1]			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	—	—	—	"FF8A" (46H 46H 38H 41H)
ASCII decimal	—	—	—	"4584" (34H 35H 38H 34H)
HEX	—	—	—	FFFFA8FFH

[1] Values in '( )' indicate ASCII code.

■Calculation procedure for sum check

The following show procedures for calculating sum check codes using the following sample data.



For the data shown above

Sum check value=52H+4AH+37H+31H+43H+32H+34H+03H=1B0H

■ Data flow: Forward direction

- "No Complement Calculation" is designated (1B0H in hexadecimal is 432 in decimal)

Code type	Data length*1			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	"0" (30H)	"B0" (42H 30H)	"1B0" (31H 42H 30H)	"01B0" (30H 31H 42H 30H)
ASCII decimal	"2" (32H)	"32" (33H 32H)	"432" (34H 33H 32H)	"0432" (30H 34H 33H 32H)
HEX	B0H	01B0H	0001B0H	000001B0H

*1 Values in '( )' indicate ASCII code.

- "One's complement" is designated (One's complement for 0000 01B0H is FFFF FE4FH)
  When "Code type" is "ASCII decimal", the lower one word is extracted and converted from hexadecimal to decimal. (FE4FH in hexadecimal is 65103 in decimal.)

Code type	Data length*1			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	"F" (46H)	"4F" (34H 46H)	"E4F" (45H 34H 46H)	"FE4F" (46H 45H 34H 46H)
ASCII decimal	"3" (33H)	"03" (30H 33H)	"103" (31H 30H 33H)	"5103" (35H 31H 30H 33H)
HEX	4FH	FE4FH	FFFE4FH	FFFFFE4FH

*1 Values in '( )' indicate ASCII code.

- "Two's complement" is designated (Two's complement for 0000 01B0H is FFFF FE50H)
  When "Code type" is "ASCII decimal", the lower one word is extracted and converted from hexadecimal to decimal. (FE50H in hexadecimal is 65104 in decimal.)

Code type	Data length*1			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	"0" (30H)	"50" (35H 30H)	"E50" (45H 35H 30H)	"FE50" (46H 45H 35H 30H)
ASCII decimal	"4" (34H)	"04" (30H 34H)	"104" (31H 30H 34H)	"5104" (35H 31H 30H 34H)
HEX	50H	FE50H	FFFE50H	FFFFFE50H

*1 Values in '( )' indicate ASCII code.

■ Data flow: Reverse direction

- "No Complement Calculation" is designated (1B0H in hexadecimal is 432 in decimal)

Code type	**Data length**[*1]			
	1 byte	**2 bytes**	**3 bytes**	**4 bytes**
ASCII hexadecimal	—	"0B" (30H 42H)	"0B1" (30H 42H 31H)	"0B10" (30H 42H 31H 30H)
ASCII decimal	—	"23" (32H 33H)	"234" (32H 33H 34H)	"2340" (32H 33H 34H 30H)
HEX	—	B001H	B00100H	B0010000H

*1 Values in '( )' indicate ASCII code.

- "One's complement" is designated (One's complement for 0000 01B0H is FFFF FE4FH)
  When "Code type" is "ASCII decimal", the lower one word is extracted and converted from hexadecimal to decimal. (FE4FH in hexadecimal is 65103 in decimal.)

Code type	**Data length**[*1]			
	1 byte	**2 bytes**	**3 bytes**	**4 bytes**
ASCII hexadecimal	—	"F4" (46H 34H)	"F4E" (46H 34H 45H)	"F4EF" (46H 34H 45H 46H)
ASCII decimal	—	"30" (33H 30H)	"301" (33H 30H 31H)	"3015" (33H 30H 31H 35H)
HEX	—	4FFEH	4FFEFFH	4FFEFFFFH

*1 Values in '( )' indicate ASCII code.

- "Two's complement" is designated (Two's complement for 0000 01B0H is FFFF FE50H)
  When "Code type" is "ASCII decimal", the lower one word is extracted and converted from hexadecimal to decimal. (FE50H in hexadecimal is 65104 in decimal.)

Code type	**Data length**[*1]			
	1 byte	**2 bytes**	**3 bytes**	**4 bytes**
ASCII hexadecimal	—	"05" (30H 35H)	"05E" (30H 35H 45H)	"05EF" (30H 35H 45H 46H)
ASCII decimal	—	"40" (34H 30H)	"401" (34H 30H 31H)	"4015" (34H 30H34H 35H)
HEX	—	50FEH	50FEFFH	50FEFFFFH

*1 Values in '( )' indicate ASCII code.

■ Data Flow: Byte Swap

- "No Complement Calculation" is designated (1B0H in hexadecimal is 432 in decimal)

Code type	Data length[*1]			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	—	—	—	"100B" (31H 30H 30H 42H)
ASCII decimal	—	—	—	"4023" (34H 30H 32H 33H)
HEX	—	—	—	0000B001H

*1 Values in '( )' indicate ASCII code.

- "One's complement" is designated (One's complement for 0000 01B0H is FFFF FE4FH)
  When "Code type" is "ASCII decimal", the lower one word is extracted and converted from hexadecimal to decimal. (FE4FH in hexadecimal is 65103 in decimal.)

Code type	Data length[*1]			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	—	—	—	"EFF4" (45H 46H 46H 34H)
ASCII decimal	—	—	—	"1530" (31H 35H 33H 30H)
HEX	—	—	—	FFFF4FFEH

*1 Values in '( )' indicate ASCII code.

- "Two's complement" is designated (Two's complement for 0000 01B0H is FFFF FE50H)
  When "Code type" is "ASCII decimal", the lower one word is extracted and converted from hexadecimal to decimal. (FE50H in hexadecimal is 65104 in decimal.)

Code type	Data length[*1]			
	1 byte	2 bytes	3 bytes	4 bytes
ASCII hexadecimal	—	—	—	"EF05" (45H 46H 30H 35H)
ASCII decimal	—	—	—	"1540" (31H 35H 34H 30H)
HEX	—	—	—	FFFF50FEH

*1 Values in '( )' indicate ASCII code.

■Calculation procedure for 16-bit CRC (for MODBUS)

This is an error check method to be used when data are transmitted/received with the RTU mode of the MODBUS protocol. The data length of CRC is fixed to 2 bytes (16 bits), and the CRC is calculated every 1 byte (8 bits) from the start of the calculating range according to the following procedure.

1. Load a 16-bit register whose bits are all '1'.
2. Exclusive OR (XOR) the first 1 byte (8 bits) of the calculating range with 8 bits in above 1.
3. Shift the result of the step 2. for one bit right.
4. If the least significant bit in above 2. is '1', exclusive OR (XOR) the result of 3. with the generator polynomial (A001H). If the least significant bit is '0', shift the result of step 3. one bit right (operation described in 3.) without the exclusive OR (XOR) operation.
5. Repeat steps 3. and 4. for 8 times.
6. Exclusive OR (XOR) the result of the above 5 with the next 1 byte (8 bits).
7. Repeat step 3 through 6 until all bytes have been processed. The final result is CRC value.
8. The CRC value is stored in a packet in the order of lower 8 bits → upper 8 bits.

Ex.

The calculation example of 16-bit CRC (for MODBUS)

Packet example:

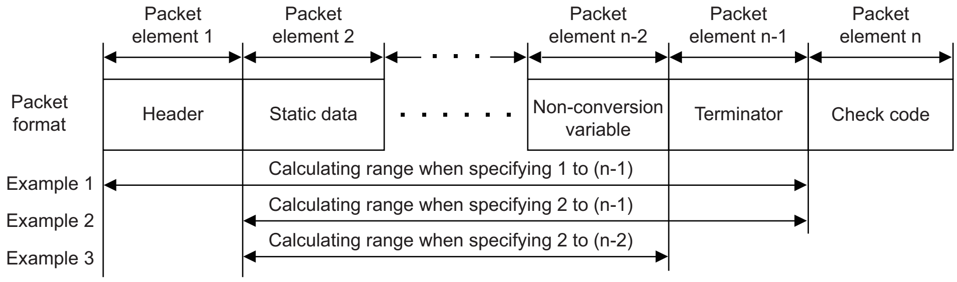
Station number	Function code	16-bit CRC	
02H	07H	41H	12H

Calculation method of 16-bit CRC (for MODBUS) for the above packet example

CRC error checking procedure	16-bit register (MSB)		Flag	Calculating procedure
(Load a 16-bit register whose bits are all '1') 02H (Station number) Exclusive OR (XOR)	1111 1111 - 1111 1111	1111 1111 0000 0010 1111 1101	-	1 to 2
Shift 1 Generator polynomial Exclusive OR (XOR)	0111 1111 1010 0000 1101 1111	1111 1110 0000 0001 1111 1111	1	3 to 4
Shift 2 Generator polynomial Exclusive OR (XOR)	0110 1111 1010 0000 1100 1111	1111 1111 0000 0001 1111 1110	1	5
Shift 3 Shift 4 Generator polynomial Exclusive OR (XOR)	0110 0111 0011 0011 1010 0000 1001 0011	1111 1111 1111 1111 0000 0001 1111 1110	0 1	
Shift 5 Shift 6 Generator polynomial Exclusive OR (XOR)	0100 1001 0010 0100 1010 0000 1000 0100	1111 1111 1111 1111 0000 0001 1111 1110	0 1	
Shift 7 Shift 8 Generator polynomial Exclusive OR (XOR)	0100 0010 0010 0001 1010 0000 1000 0001	0111 1111 0011 1111 0000 0001 0011 1110	0 1	
07H (Function code) Exclusive OR (XOR)	- 1000 0001	0000 0111 0011 1001	-	6
Shift 1 Generator polynomial Exclusive OR (XOR)	0100 0000 1010 0000 1110 0000	1001 1100 0000 0001 1001 1101	1	7
Shift 2 Generator polynomial Exclusive OR (XOR)	0111 0000 1010 0000 1101 0000	0100 1110 0000 0001 0100 1111	1	
Shift 3 Generator polynomial Exclusive OR (XOR)	0110 1000 1010 0000 1100 1000	0010 0111 0000 0001 0010 0110	1	
Shift 4 Shift 5 Generator polynomial Exclusive OR (XOR)	0110 0100 0011 0010 1010 0000 1001 0010	0001 0011 0000 1001 0000 0001 0000 1000	0 1	
Shift 6 Shift 7 Shift 7	0100 1001 0010 0100 0001 0010	0000 0100 1000 0010 0100 0001	0 0 0	
CRC value	12H	41H	-	8

■Check code calculation range

The following shows specification examples of the check code calculation code.



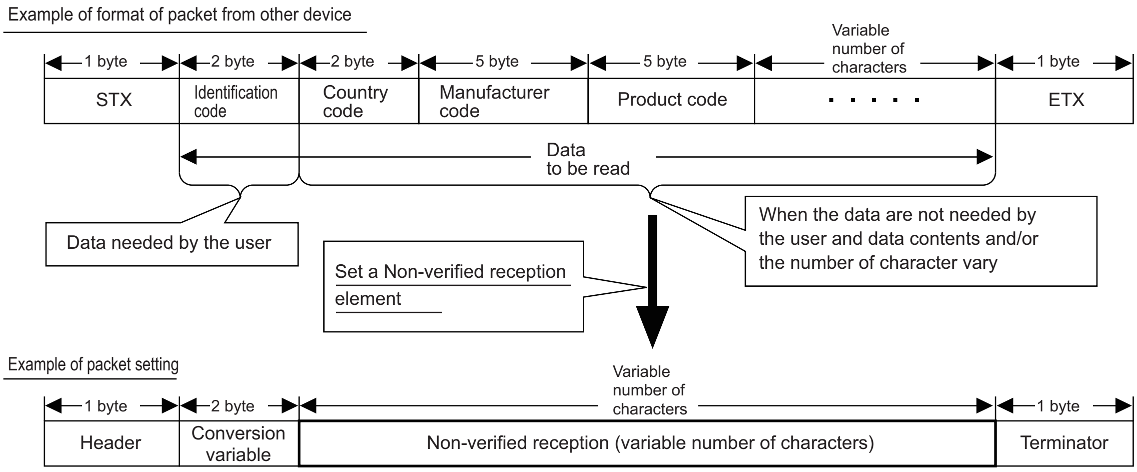
Example 1: Calculating range when 1 and n-1 are specified as the start and the end respectively.

Example 2: Calculating range when 2 and n-1 are specified as the start and the end respectively.

Example 3: Calculating range when 2 and n-2 are specified as the start and the end respectively.

Non-verified reception

The usage example for non-verified reception is shown below.


Example of format of packet from other device
1 byte
2 byte
2 byte
5 byte
5 byte
Variable number of characters
1 byte
STX
Identification code
Country code
Manufacturer code
Product code
ETX
Data to be read
Data needed by the user
Set a Non-verified reception element
When the data are not needed by the user and data contents and/or the number of character vary
Example of packet setting
1 byte
2 byte
Variable number of characters
1 byte
Header
Conversion variable
Non-verified reception (variable number of characters)
Terminator

Using a non-verified reception element has the following advantages in the case of the packet format shown above.

- The only necessary data can be stored in the device memory of a CPU module and buffer memory.
- A single protocol (packet) can handle receive packets that includes data whose contents vary each time.

Appendix 6 Processing Time

Processing time for nonprocedural protocol (guideline)

The processing time for the Output and Input instructions can be estimated by the following formula.

Note, however, that the processing time may become longer depending on other communication functions (e.g., communication using the MC protocol) and special functions (e.g., ASCII-binary conversion, transmission control) that are used at the same time.

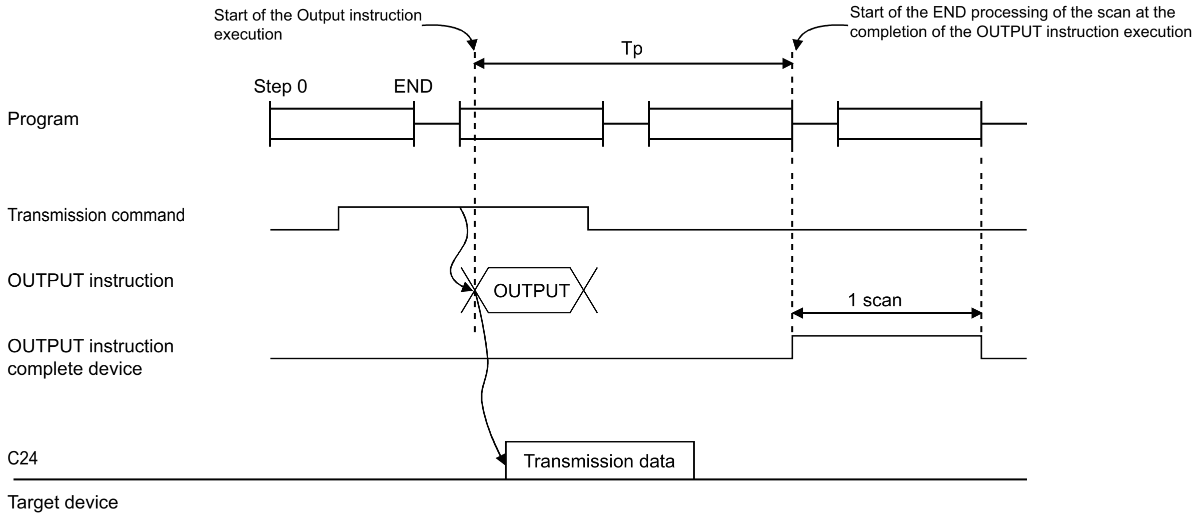
The value obtained from the following formula is considered to be the processing time taken for reception or transmission only when CH1 only is used (RS-232 connection).

■OUTPUT instruction

$Tp=St+(Dle/Cr\times1000+0.07)\times Ds+T1$

Tp[*1]	: The time from the start of execution of the Output instruction to the End processing of the sequence scan at the completion of the Output instruction execution (ms)
St	: Scan time
Dle	: Data length of 1 byte at data transmission (number of bits)
Cr	: Communication rate (bps)
Ds	: Number of bytes of send data
T1	: C24 T1=2.7

*1 This stands for the time range of processing (Tp).

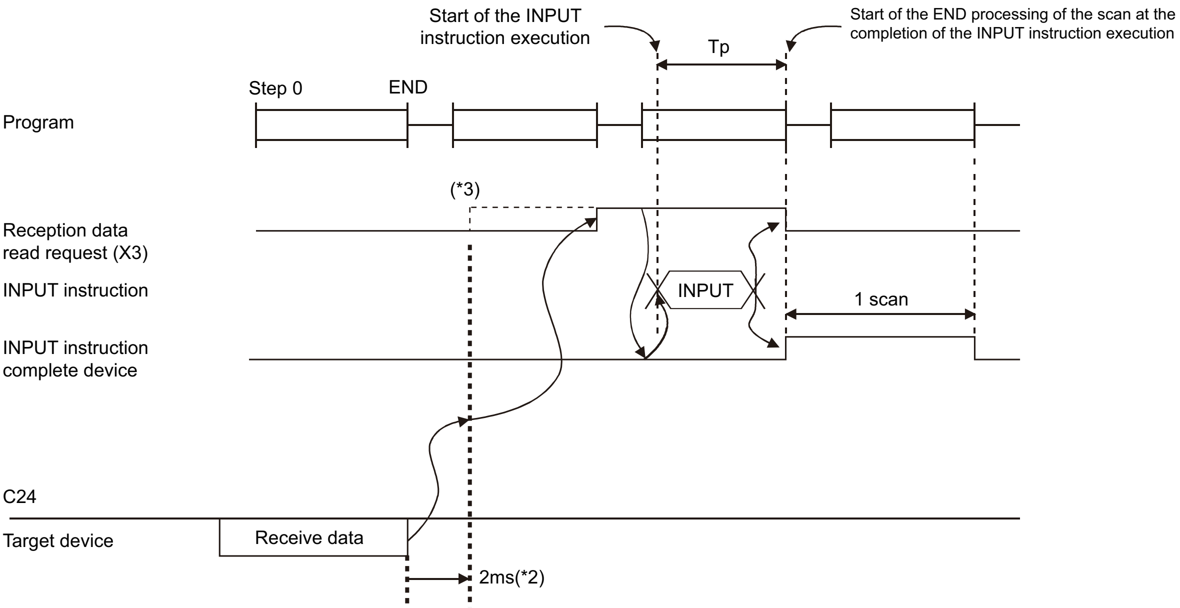

Start of the Output instruction execution
Start of the END processing of the scan at the completion of the OUTPUT instruction execution
Tp
Step 0
END
Program
Transmission command
OUTPUT instruction
OUTPUT
1 scan
OUTPUT instruction complete device
C24
Transmission data
Target device

■INPUT instruction

Tp = Sr+0.09 × Dr+T2

Tp[*1]	: Time from the start of the INPUT instruction execution to the start of End processing for the sequence scan at the completion of the INPUT instruction execution (ms)
Sr	: Scan time
Dr	: Number of bytes of reception data
T2	: C24 T2=6.3

*1 This indicates the range of processing time (Tp).



Start of the INPUT instruction execution
Start of the END processing of the scan at the completion of the INPUT instruction execution
Tp
Step 0
END
Program
(*3)
Reception data read request (X3)
INPUT instruction
INPUT
1 scan
INPUT instruction complete device
C24
Target device
Receive data
2ms(*2)

*2 It indicates the time from when C24 receives data from the line to when the 'CH1 reception data read request' (X3) turns ON when only read processing of 30 bytes data is performed with one of the channels on C24 used.

*3 When DX3 (direct mode) is used, C24 is accessed at the time of the instruction execution with the program.
DX3 (direct mode) speeds up reading input compared to X3 (refresh mode).
For differences between the modes, refer to the following manual.
MELSEC iQ-R CPU Module User's Manual (Application)

Appendix 7 When Connecting the Module to a Remote Head Module

This section describes the restrictions and communication example for when connecting the C24 to a remote head module.

Restrictions

This section describes the restricted functions and specifications of when connecting the C24 to a remote head module.

When creating a program

- Dedicated instructions cannot be used.
- Data cannot be received with an interrupt program.

Communication using the MC protocol

The command that can be used differs from that of the CPU module. For the command that can be used in the remote head module, refer to the following.

📖MELSEC Communication Protocol Reference Manual

Communication using the MODBUS slave function

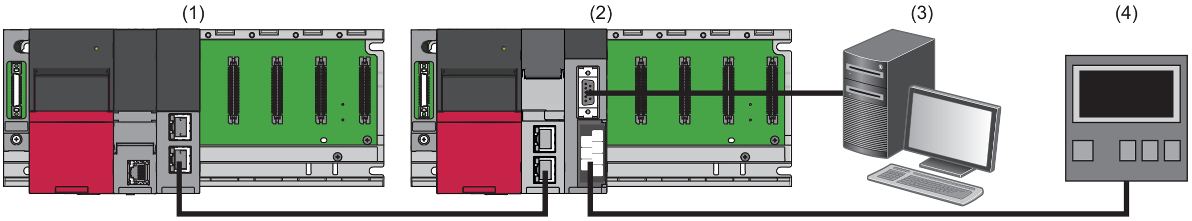
For the restrictions on communications using the MODBUS slave function, refer to the following section.

☞ Page 115 Specifying an access target

Communication example

This section describes the communication example for when connecting the C24 to a remote head module.

System configuration example



No.	Component	Network	Protocol
(1)	R04CPU	—	
	RJ71GF11-T2	CC-Link IE Field Network	—
(2)	RJ72GF15-T2	CC-Link IE Field Network	—
	RJ71C24	—	CH1: Nonprocedural protocol or bidirectional protocol CH2: Predefined protocol
(3)	External device (personal computer)	—	Nonprocedural protocol or bidirectional protocol
(4)	External device (temperature controller)	—	Predefined protocol

Communication procedure

The following shows the procedure to send/receive data.

■Data communication using the predefined protocol

❶ Writes send data to the link register (W) of the CPU module to store the send data in the data storage area set in the packet setting of the predefined protocol support function.

❷ Executes the REMTO instruction to write the number of consecutive protocol executions to the buffer memory of the C24.

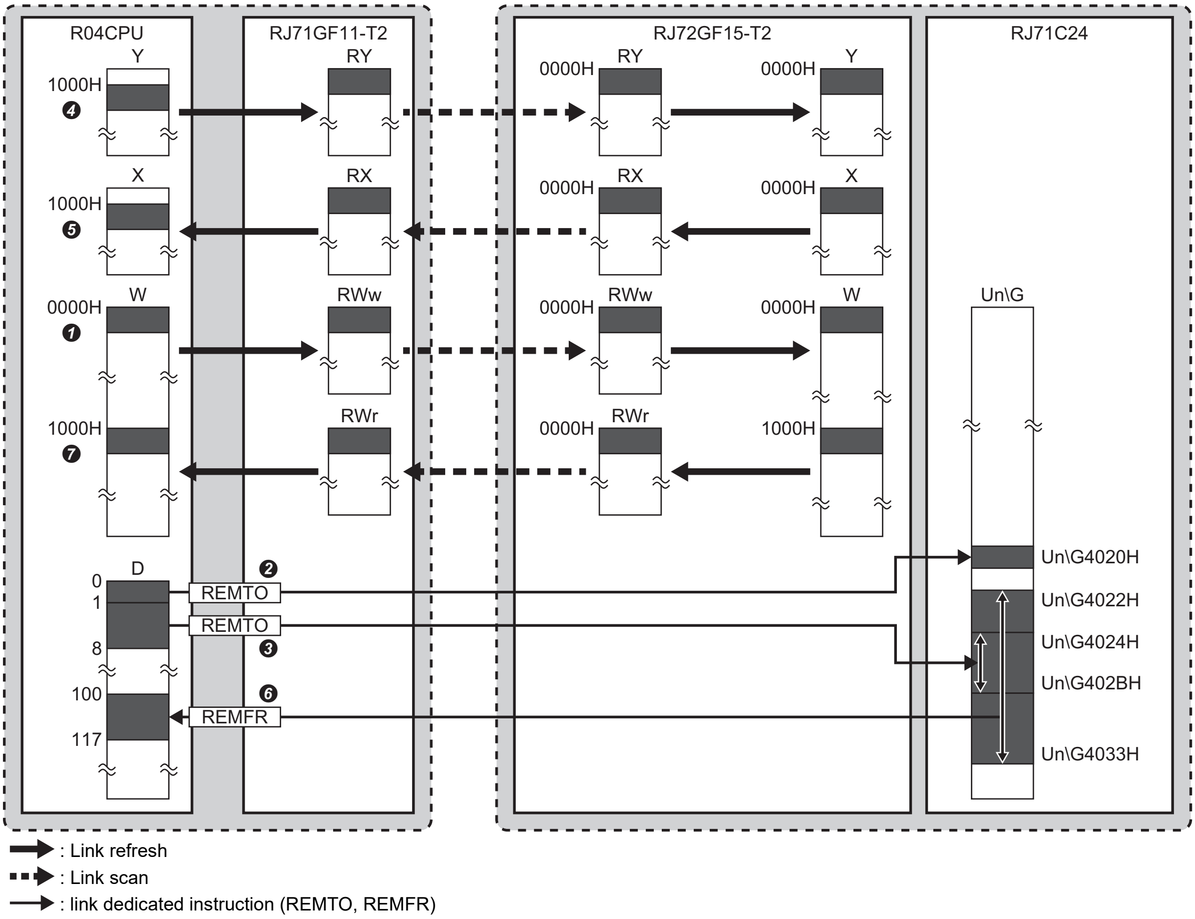
❸ Executes the REMTO instruction to write the number of a protocol to be executed to the buffer memory of the C24.

❹ Executes the protocol by 'CH2 protocol execution request' (YA).

❺ Checks the protocol execution by 'CH2 protocol execution completion' (XC).

❻ Reads the predefined protocol execution result, result of the execution count, and matched packet number from the buffer memory of the C24 with the REMFR instruction.

❼ Reads the data received from the external device which is transferred from the device of the remote head module to the link register (W) of the CPU module.



: Link refresh
: Link scan
: link dedicated instruction (REMTO, REMFR)

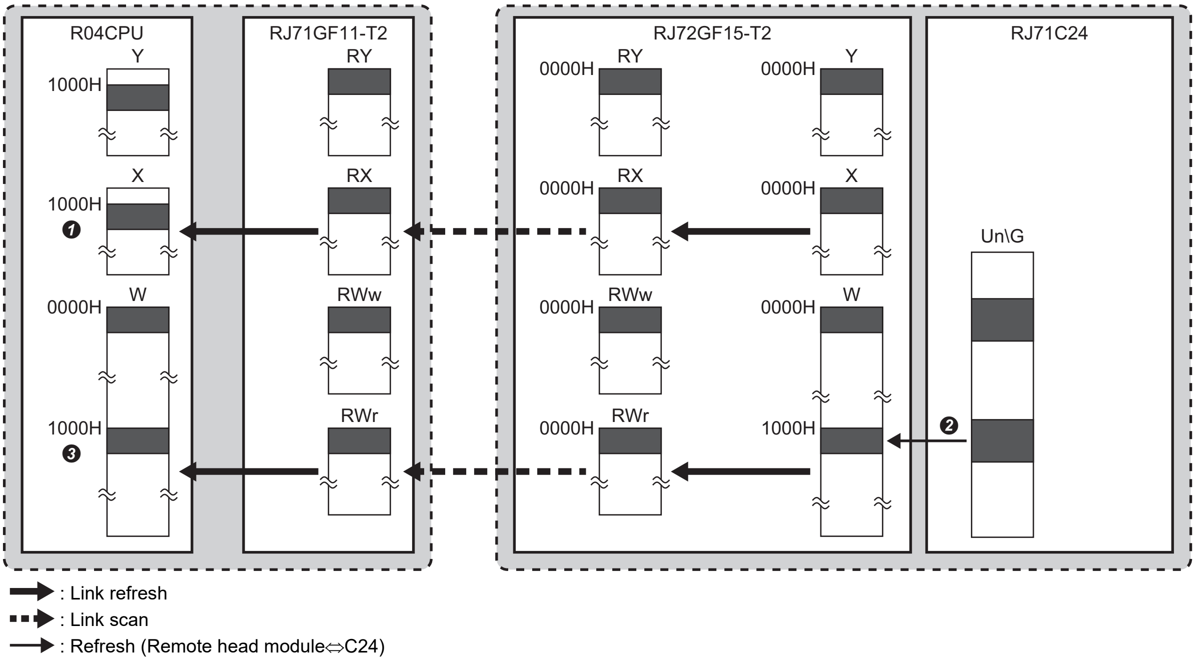
■Data communication using the nonprocedural or bidirectional protocol

- Receiving procedure

❶ Checks data received from the external device by 'CH1 reception data read request' (X3).

❷ Transfers the received data from the buffer memory of the C24 to the device of the remote head module by refresh (remote head module ⇔ C24).

❸ Reads the received data transferred to the link register (W) of the CPU module by link refresh.

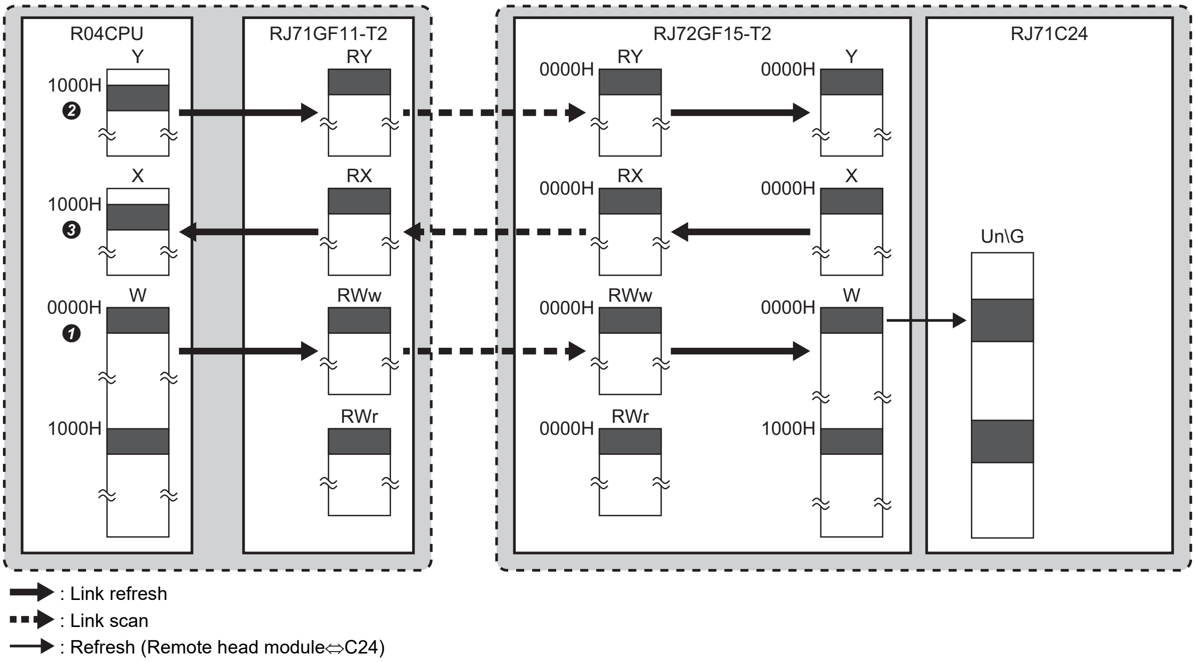


- Sending procedure

❶ Writes send data to the link register (W) of the CPU module to write the send data to the buffer memory of the C24.

❷ Issues the transmission request by 'CH1 transmission request' (Y0).

❸ 'CH1 transmission normal completion' (X0) turns ON when the execution of the transmission request completes.

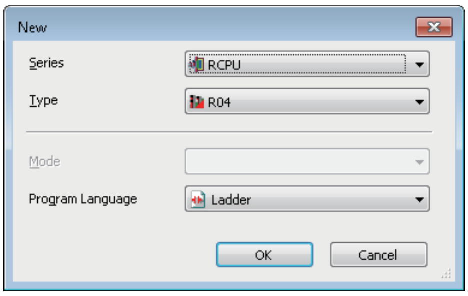


Setting in the master station on CC-Link IE Field Network

Connect the engineering tool to the CPU module of the master station on CC-Link IE Field Network and set parameters.

1. Set the CPU module as follows:

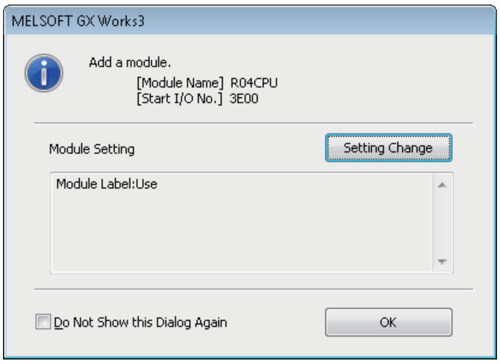
[Project] ⇨ [New]



New

Series	RCPU
Type	R04
Mode	
Program Language	Ladder

2. Click the [Setting Change] button to set to use module labels.



MELSOFT GX Works3

Add a module.
[Module Name] R04CPU
[Start I/O No.] 3E00

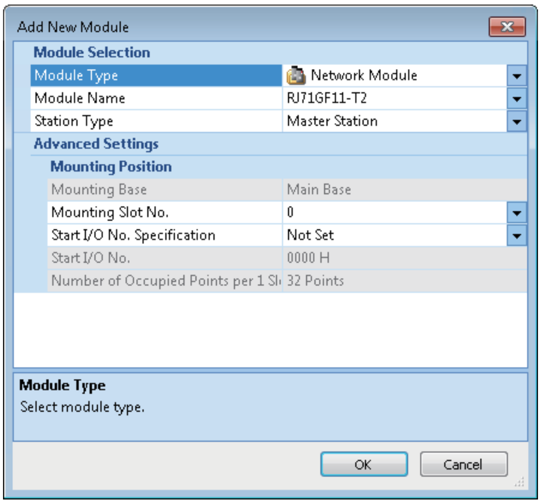
Module Setting

Module Label:Use

□ Do Not Show this Dialog Again

3. Set the CC-Link IE Field Network master/local module as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ right-click ⇨ [Add New Module]

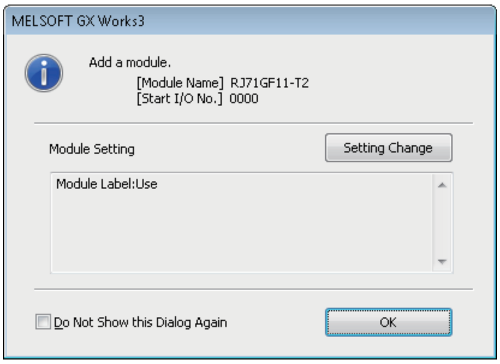


Add New Module

Module Selection	
Module Type	Network Module
Module Name	RJ71GF11-T2
Station Type	Master Station
Advanced Settings	
Mounting Position	
Mounting Base	Main Base
Mounting Slot No.	0
Start I/O No. Specification	Not Set
Start I/O No.	0000 H
Number of Occupied Points per 1 Sl	32 Points

Module Type
Select module type.

4. Click the [Setting Change] button to set to use module labels.



MELSOFT GX Works3

Add a module.
[Module Name] RJ71GF11-T2
[Start I/O No.] 0000

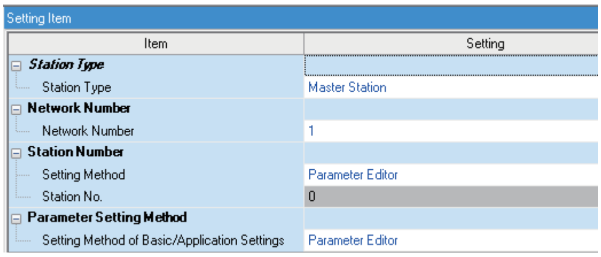
Module Setting [Setting Change]

Module Label:Use

□ Do Not Show this Dialog Again [OK]

5. Set the items in "Required Settings" of "Module Parameter" as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71GF11-T2] ⇨ [Module Parameter] ⇨ [Required Settings]

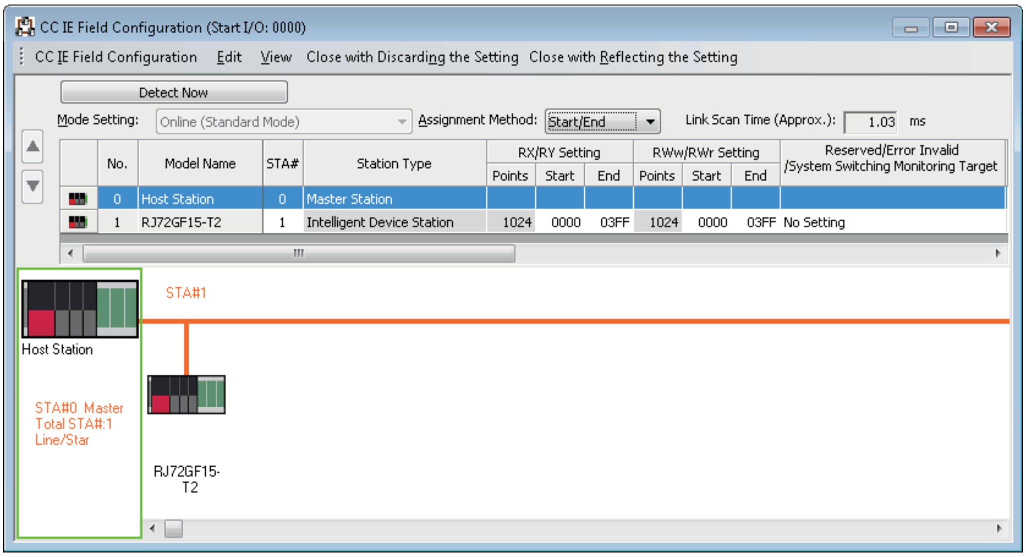


Setting Item

Item	Setting
⊟ ***Station Type***	
Station Type	Master Station
⊟ **Network Number**	
Network Number	1
⊟ **Station Number**	
Setting Method	Parameter Editor
Station No.	0
⊟ **Parameter Setting Method**	
Setting Method of Basic/Application Settings	Parameter Editor

6. Set the network configuration as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71GF11-T2] ⇨ [Module Parameter] ⇨ [Basic Settings] ⇨ [Network Configuration Settings]



CC IE Field Configuration (Start I/O: 0000)

CC IE Field Configuration Edit View Close with Discarding the Setting Close with Reflecting the Setting

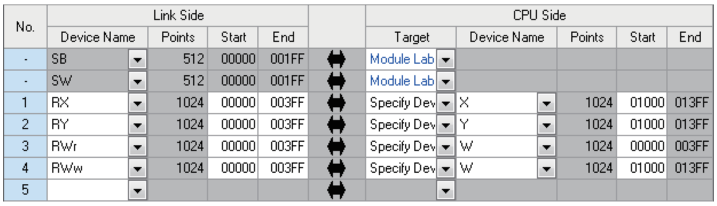
[Detect Now]

Mode Setting: Online (Standard Mode) Assignment Method: Start/End Link Scan Time (Approx.): 1.03 ms

	No.	Model Name	STA#	Station Type	RX/RY Setting			RWw/RWr Setting			Reserved/Error Invalid /System Switching Monitoring Target
					Points	Start	End	Points	Start	End	
	0	Host Station	0	Master Station							
	1	RJ72GF15-T2	1	Intelligent Device Station	1024	0000	03FF	1024	0000	03FF	No Setting

7. Set the refresh settings as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71GF11-T2] ⇨ [Module Parameter] ⇨ [Basic Settings] ⇨ [Refresh Setting]



No.	Link Side					CPU Side				
	Device Name	Points	Start	End		Target	Device Name	Points	Start	End
-	SB	512	00000	001FF	↔	Module Lab				
-	SW	512	00000	001FF	↔	Module Lab				
1	RX	1024	00000	003FF	↔	Specify Dev	X	1024	01000	013FF
2	RY	1024	00000	003FF	↔	Specify Dev	Y	1024	01000	013FF
3	RWr	1024	00000	003FF	↔	Specify Dev	W	1024	00000	003FF
4	RWw	1024	00000	003FF	↔	Specify Dev	W	1024	01000	013FF
5					↔					

8. Write the set parameters to the CPU module of the master station on CC-Link IE Field Network. Then, reset the CPU module or power off and on the system.

[Online] ⇨ [Write to PLC]

Point

In this example, default values are used for parameters that are not shown above. For the parameters, refer to the following.

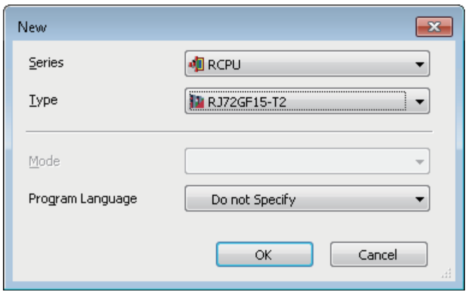
📖MELSEC iQ-R CC-Link IE Field Network User's Manual (Application)

Setting in the intelligent device station on CC-Link IE Field Network

Connect the engineering tool to the remote head module of the intelligent device station on CC-Link IE Field Network and set the parameters.

1. Set the remote head module as follows.

[Project] ⇨ [New]

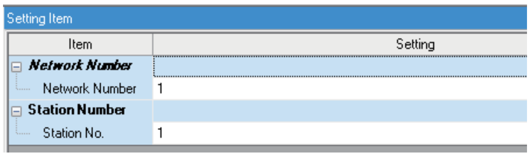


New

Series	RCPU
Type	RJ72GF15-T2
Mode	
Program Language	Do not Specify

2. Set the items in "Network Required Setting" of "CPU Parameter" as follows.

[Navigation window] ⇨ [Parameter] ⇨ [RJ72GF15-T2] ⇨ [CPU Parameter] ⇨ [Network Required Setting]

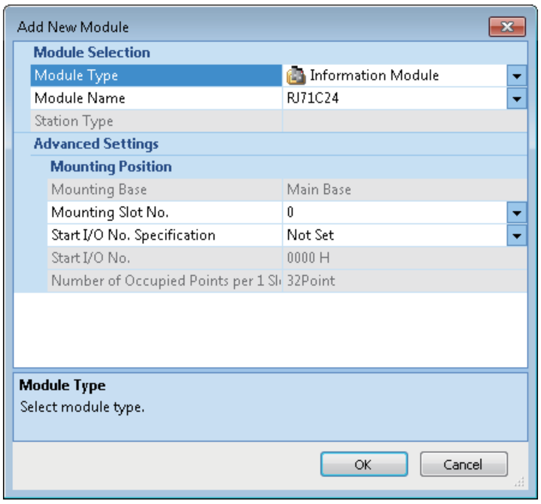


Setting Item

Item	Setting
Network Number	
Network Number	1
Station Number	
Station No.	1

3. Set the C24 as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ right-click ⇨ [Add New Module]

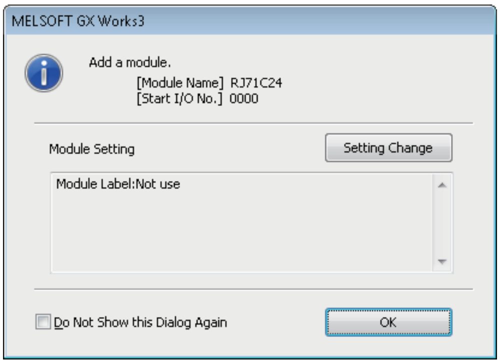


Add New Module

Module Selection	
Module Type	Information Module
Module Name	RJ71C24
Station Type	
Advanced Settings	
Mounting Position	
Mounting Base	Main Base
Mounting Slot No.	0
Start I/O No. Specification	Not Set
Start I/O No.	0000 H
Number of Occupied Points per 1 Sl	32Point

Module Type
Select module type.

4. Click the [OK] button.



MELSOFT GX Works3

Add a module.

[Module Name] RJ71C24
[Start I/O No.] 0000

Module Setting

Module Label:Not use

5. Set parameters of the C24 according to the protocol to be used.

☞ Page 614 Program example of a predefined protocol

☞ Page 619 Program examples of a nonprocedural protocol or bidirectional protocol

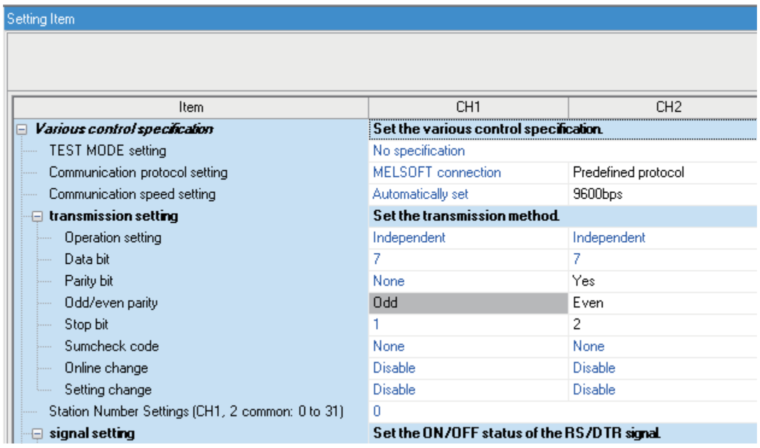
Program example of a predefined protocol

The following shows a program example to connect an external device (temperature controller) to the CH2 of a C24, and to execute a predefined protocol.

■Module parameter setting of a C24

1. Set the items in "Basic Settings" as follows:

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter] ⇨ [Basic Settings]



Setting Item

Item	CH1	CH2
⊟ ***Various control specification***	**Set the various control specification.**	
TEST MODE setting	No specification	
Communication protocol setting	MELSOFT connection	Predefined protocol
Communication speed setting	Automatically set	9600bps
⊟ **transmission setting**	**Set the transmission method.**	
Operation setting	Independent	Independent
Data bit	7	7
Parity bit	None	Yes
Odd/even parity	Odd	Even
Stop bit	1	2
Sumcheck code	None	None
Online change	Disable	Disable
Setting change	Disable	Disable
Station Number Settings (CH1, 2 common: 0 to 31)	0	
⊟ **signal setting**	**Set the ON/OFF status of the RS/DTR signal.**	

2. Write the set parameters to the remote head module of the intelligent device station on CC-Link IE Field Network. Then, reset the remote head module or power the system off and on.

[Online] ⇨ [Write to PLC]

Point

In this example, default values are used for parameters that are not shown above. For the parameters, refer to the following.

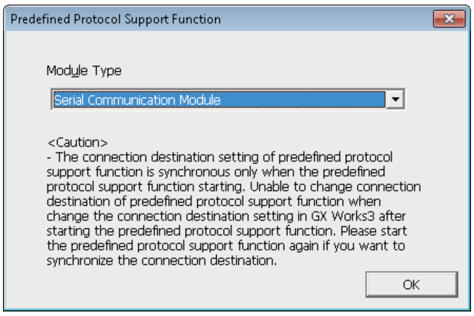
📖MELSEC iQ-R CC-Link IE Field Network User's Manual (Application)

☞ Page 454 PARAMETER SETTING

■Predefined protocol support function setting of a C24

1. Select [Tool]⇨[Predefined Protocol Support Function] from an Engineering tool.

2. Select "Serial Communication Module" in "Module Type" in the "Predefined Protocol Support Function" screen.



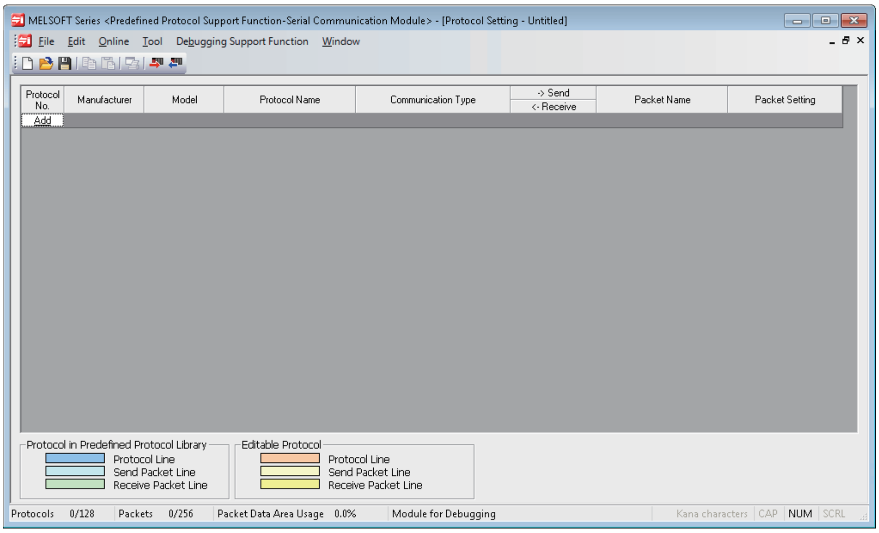
Predefined Protocol Support Function

Module Type

Serial Communication Module

<Caution>
- The connection destination setting of predefined protocol support function is synchronous only when the predefined protocol support function starting. Unable to change connection destination of predefined protocol support function when change the connection destination setting in GX Works3 after starting the predefined protocol support function. Please start the predefined protocol support function again if you want to synchronize the connection destination.

3. Select [File] ⇨ [New] in the "Predefined Protocol Support Function" screen.

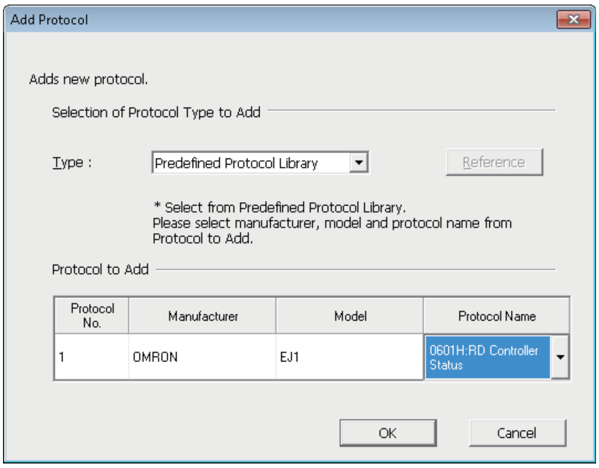

MELSOFT Series <Predefined Protocol Support Function-Serial Communication Module> - [Protocol Setting - Untitled]
File Edit Online Tool Debugging Support Function Window

Protocol No.	Manufacturer	Model	Protocol Name	Communication Type	-> Send <- Receive	Packet Name	Packet Setting
Add							

Protocol in Predefined Protocol Library
Protocol Line
Send Packet Line
Receive Packet Line
Editable Protocol
Protocol Line
Send Packet Line
Receive Packet Line
Protocols 0/128 Packets 0/256 Packet Data Area Usage 0.0% Module for Debugging
Kana characters CAP NUM SCRL

4. Select [Edit] ⇨ [Add Protocol] in the "Protocol Setting" screen.

5. Select "Predefined Protocol Library" in "Type" on the "Add Protocol" screen, and select the protocol to be used in "Protocol to Add".



Adds new protocol.

Selection of Protocol Type to Add

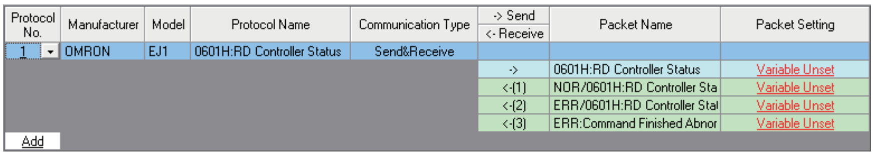
Type : Predefined Protocol Library

* Select from Predefined Protocol Library.
Please select manufacturer, model and protocol name from Protocol to Add.

Protocol to Add

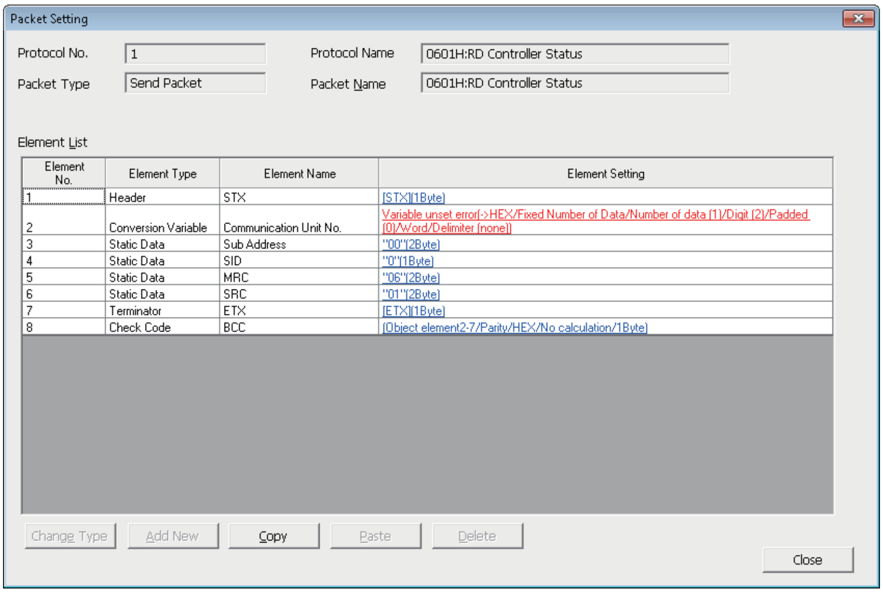
Protocol No.	Manufacturer	Model	Protocol Name
1	OMRON	EJ1	0601H:RD Controller Status

6. Click the "Variable Unset" cell of "Packet Setting" screen on the "Protocol Setting" screen.



Protocol No.	Manufacturer	Model	Protocol Name	Communication Type	-> Send <- Receive	Packet Name	Packet Setting
1	OMRON	EJ1	0601H:RD Controller Status	Send&Receive			
					->	0601H:RD Controller Status	Variable Unset
					<-(1)	NOR/0601H:RD Controller Sta	Variable Unset
					<-(2)	ERR/0601H:RD Controller Stat	Variable Unset
					<-(3)	ERR:Command Finished Abnor	Variable Unset
Add							

7. Click the "Variable unset error" cell on the "Packet Setting" screen



Protocol No.: 1
Protocol Name: 0601H:RD Controller Status
Packet Type: Send Packet
Packet Name: 0601H:RD Controller Status

Element List

Element No.	Element Type	Element Name	Element Setting
1	Header	STX	[STX][1Byte]
2	Conversion Variable	Communication Unit No.	Variable unset error[->HEX/Fixed Number of Data/Number of data (1)/Digit (2)/Padded (0)/Word/Delimiter (none)]
3	Static Data	Sub Address	"00"[2Byte]
4	Static Data	SID	"0"[1Byte]
5	Static Data	MRC	"06"[2Byte]
6	Static Data	SRC	"01"[2Byte]
7	Terminator	ETX	[ETX][1Byte]
8	Check Code	BCC	(Object element2-7/Parity/HEX/No calculation/1Byte)

8. Enter 'W0' to "Send Data Storage Area" on the "Element Setting" screen.

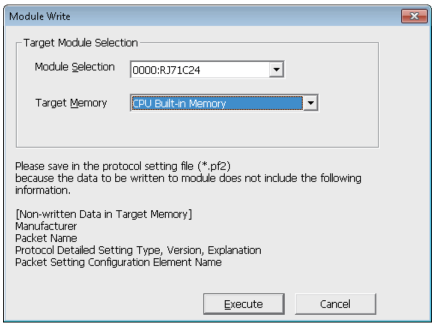
9. Set the packet settings in the "Variable Unset" cell of "Packet Setting" on the "Protocol Setting" screen with the same procedure (from step 5 to step 8).

Set the following values for the data storage area.

Packet name	Element number	Element name	Data storage area specification
NOR/0601H:RD Controller Status	2	Communication Unit No.	Set 'W1000'
	8	Operation Status	Set 'W1001'
	9	Related Information	Set 'W1002'
ERR/0601H; RD Controller Status	2	Communication Unit No.	Set 'W1000'
	7	Response Code	Set 'W1003'
ERR: Command Finished Abnormally	2	Communication Unit No.	Set 'W1000'
	4	End Code	Set 'W1004'

10. Select [Online]⇨ [Write to Module] from Engineering tool.

11. Select "CPU Built-in Memory" in "Target Memory" on the "Write to Module" screen, and click the [Execute] button.



Target Module Selection
Module Selection: 0000:RJ71C24
Target Memory: CPU Built-in Memory

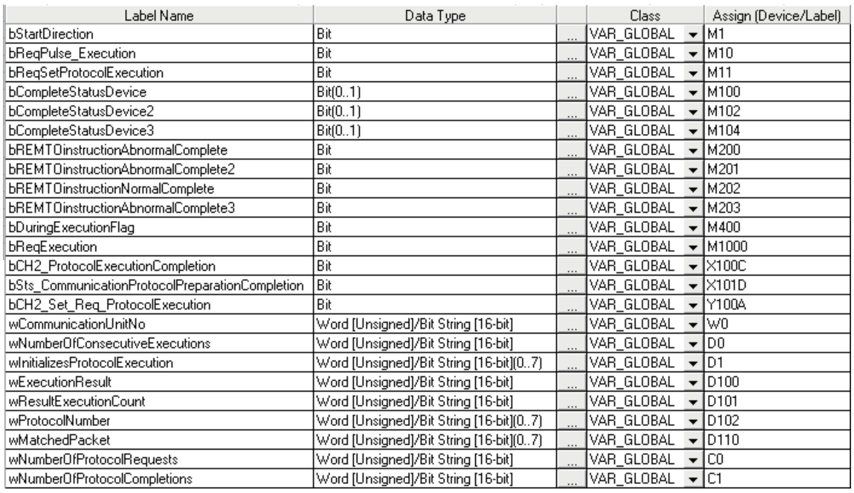
Please save in the protocol setting file (*.pf2)
because the data to be written to module does not include the following information.

[Non-written Data in Target Memory]
Manufacturer
Packet Name
Protocol Detailed Setting Type, Version, Explanation
Packet Setting Configuration Element Name

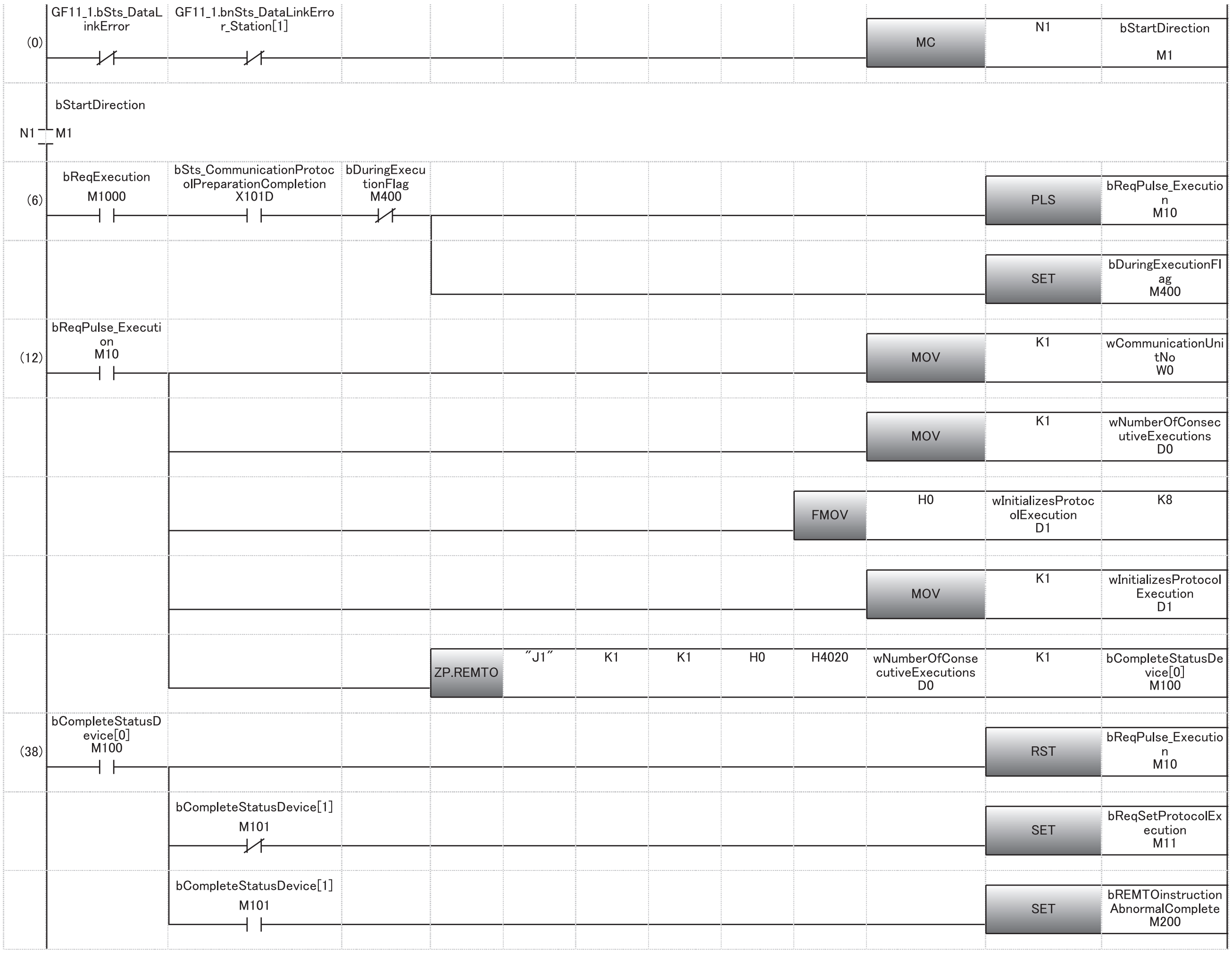
■Program example

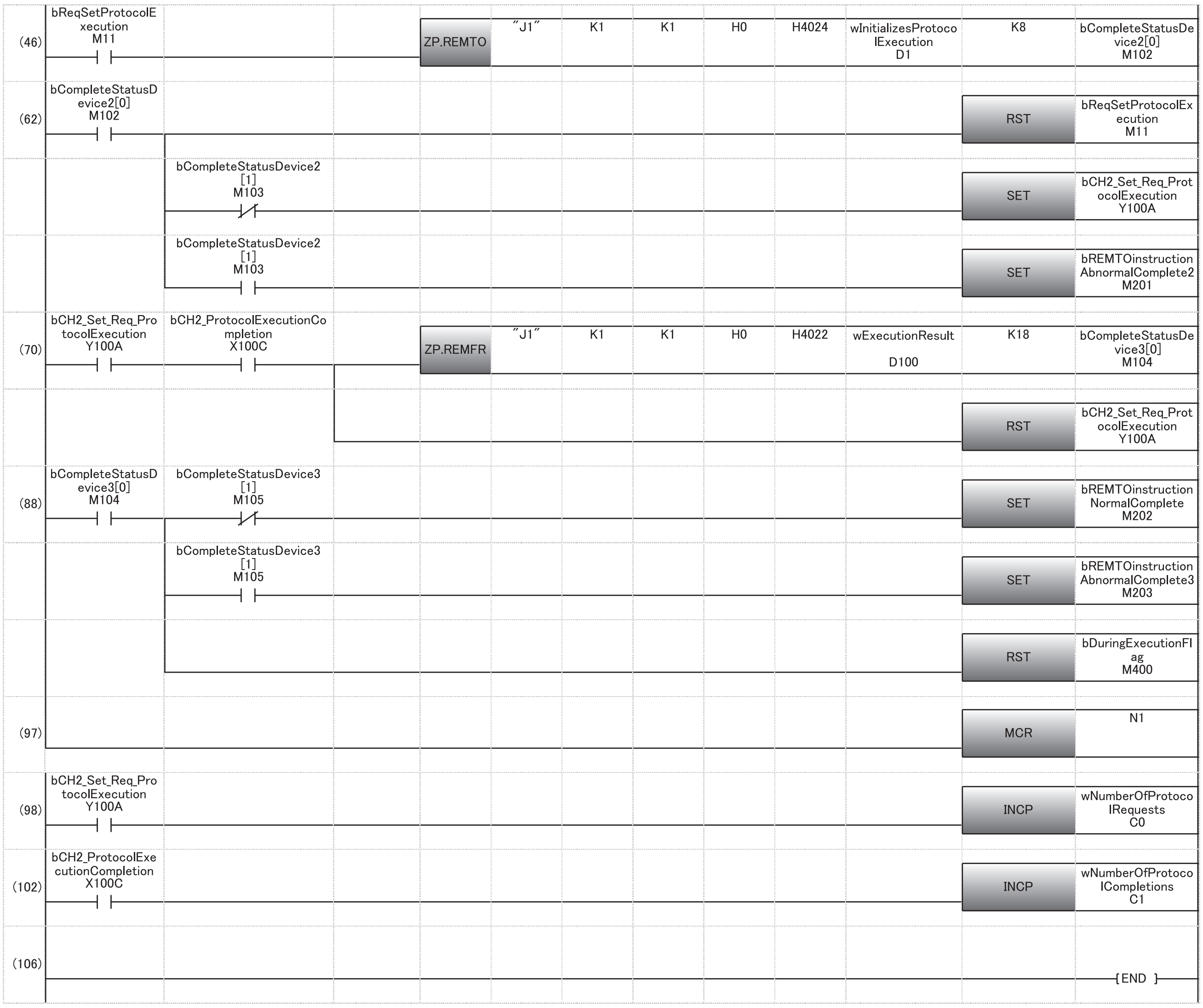
This program uses the module labels of the CC-Link IE Field Network master/local module. Write the program to the CPU module on the master station.

Category	Label name	Description	Device
Module label	GF11_1.bSts_DataLinkError	Data link error status of own station	SB0049
	GF11_1.bnSts_DataLinkError_Station[1]	Data link status of each station (station No.1)	SW00B0.0
Label to be defined	Define the global label as shown below.		



Label Name	Data Type		Class	Assign (Device/Label)
bStartDirection	Bit	...	VAR_GLOBAL	M1
bReqPulse_Execution	Bit	...	VAR_GLOBAL	M10
bReqSetProtocolExecution	Bit	...	VAR_GLOBAL	M11
bCompleteStatusDevice	Bit(0..1)	...	VAR_GLOBAL	M100
bCompleteStatusDevice2	Bit(0..1)	...	VAR_GLOBAL	M102
bCompleteStatusDevice3	Bit(0..1)	...	VAR_GLOBAL	M104
bREMTOinstructionAbnormalComplete	Bit	...	VAR_GLOBAL	M200
bREMTOinstructionAbnormalComplete2	Bit	...	VAR_GLOBAL	M201
bREMTOinstructionNormalComplete	Bit	...	VAR_GLOBAL	M202
bREMTOinstructionAbnormalComplete3	Bit	...	VAR_GLOBAL	M203
bDuringExecutionFlag	Bit	...	VAR_GLOBAL	M400
bReqExecution	Bit	...	VAR_GLOBAL	M1000
bCH2_ProtocolExecutionCompletion	Bit	...	VAR_GLOBAL	X100C
bSts_CommunicationProtocolPreparationCompletion	Bit	...	VAR_GLOBAL	X101D
bCH2_Set_Req_ProtocolExecution	Bit	...	VAR_GLOBAL	Y100A
wCommunicationUnitNo	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W0
wNumberOfConsecutiveExecutions	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
wInitializesProtocolExecution	Word [Unsigned]/Bit String [16-bit](0..7)	...	VAR_GLOBAL	D1
wExecutionResult	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D100
wResultExecutionCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D101
wProtocolNumber	Word [Unsigned]/Bit String [16-bit](0..7)	...	VAR_GLOBAL	D102
wMatchedPacket	Word [Unsigned]/Bit String [16-bit](0..7)	...	VAR_GLOBAL	D110
wNumberOfProtocolRequests	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	C0
wNumberOfProtocolCompletions	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	C1




(46)
bReqSetProtocolExecution M11
ZP.REMTO "J1" K1 K1 H0 H4024 wInitializesProtocolExecution D1 K8 bCompleteStatusDevice2[0] M102
(62)
bCompleteStatusDevice2[0] M102
RST bReqSetProtocolExecution M11
bCompleteStatusDevice2[1] M103
SET bCH2_Set_Req_ProtocolExecution Y100A
bCompleteStatusDevice2[1] M103
SET bREMTOinstructionAbnormalComplete2 M201
(70)
bCH2_Set_Req_ProtocolExecution Y100A
bCH2_ProtocolExecutionCompletion X100C
ZP.REMFR "J1" K1 K1 H0 H4022 wExecutionResult D100 K18 bCompleteStatusDevice3[0] M104
RST bCH2_Set_Req_ProtocolExecution Y100A
(88)
bCompleteStatusDevice3[0] M104
bCompleteStatusDevice3[1] M105
SET bREMTOinstructionNormalComplete M202
bCompleteStatusDevice3[1] M105
SET bREMTOinstructionAbnormalComplete3 M203
RST bDuringExecutionFlag M400
(97)
MCR N1
(98)
bCH2_Set_Req_ProtocolExecution Y100A
INCP wNumberOfProtocolRequests C0
(102)
bCH2_ProtocolExecutionCompletion X100C
INCP wNumberOfProtocolCompletions C1
(106)
END

(6) Sets send data and writes the number of consecutive protocol executions and the number of a protocol to be executed to the buffer memory of the C24 with the REMTO instruction.

(62) Issues the predefined protocol execution request when the REMTO instruction completes.

(70) Reads the execution result from the buffer memory of the C24 with the REMFR instruction when the execution of the predefined protocol completes.

Program examples of a nonprocedural protocol or bidirectional protocol

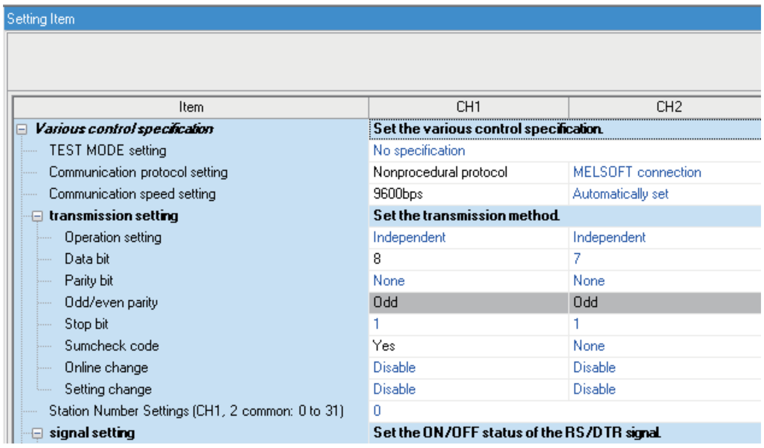
The following shows program examples to connect an external device (personal computer) to the CH1 of a C24, and to send/receive data by using a nonprocedural protocol or bidirectional protocol with input/output signals.

■Module parameter setting of a C24

1. Set the items in "Basic Settings" as follows:

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter] ⇨ [Basic Settings]

For communication using the bidirectional protocol, set the communication protocol setting to "Bidirectional protocol."

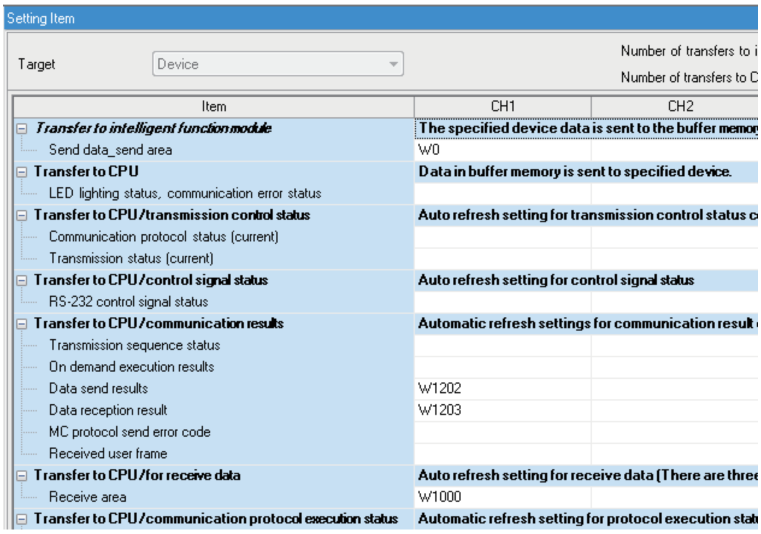


Setting Item

Item	CH1	CH2
Various control specification	Set the various control specification.	
TEST MODE setting	No specification	
Communication protocol setting	Nonprocedural protocol	MELSOFT connection
Communication speed setting	9600bps	Automatically set
transmission setting	Set the transmission method.	
Operation setting	Independent	Independent
Data bit	8	7
Parity bit	None	None
Odd/even parity	Odd	Odd
Stop bit	1	1
Sumcheck code	Yes	None
Online change	Disable	Disable
Setting change	Disable	Disable
Station Number Settings (CH1, 2 common: 0 to 31)	0	
signal setting	Set the ON/OFF status of the RS/DTR signal.	

2. Set the items in "refresh settings" of "Module Parameter" as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter] ⇨ [Refresh setting]



Setting Item

Target: Device

Number of transfers to i

Number of transfers to C

Item	CH1	CH2
Transfer to intelligent function module	The specified device data is sent to the buffer memor	
Send data_send area	W0	
Transfer to CPU	Data in buffer memory is sent to specified device.	
LED lighting status, communication error status		
Transfer to CPU/transmission control status	Auto refresh setting for transmission control status c	
Communication protocol status (current)		
Transmission status (current)		
Transfer to CPU/control signal status	Auto refresh setting for control signal status	
RS-232 control signal status		
Transfer to CPU/communication results	Automatic refresh settings for communication result	
Transmission sequence status		
On demand execution results		
Data send results	W1202	
Data reception result	W1203	
MC protocol send error code		
Received user frame		
Transfer to CPU/for receive data	Auto refresh setting for receive data (There are three	
Receive area	W1000	
Transfer to CPU/communication protocol execution status	Automatic refresh setting for protocol execution stat	

3. Write the set parameters to the remote head module of the intelligent device station on CC-Link IE Field Network. Then, reset the remote head module or power the system off and on.

[Online] ⇨ [Write to PLC]

Point

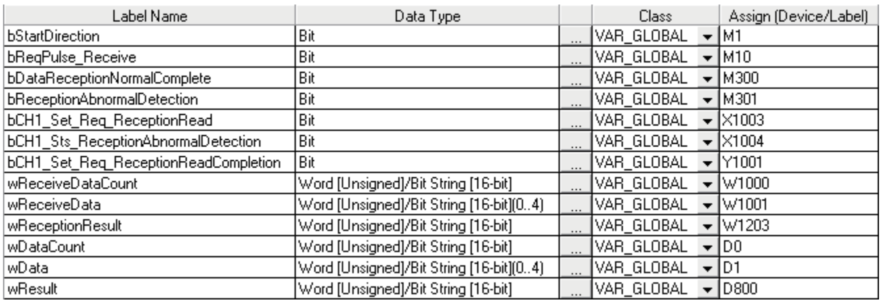
In this example, default values are used for parameters that are not shown above. For the parameters, refer to the following.

MELSEC iQ-R CC-Link IE Field Network User's Manual (Application)

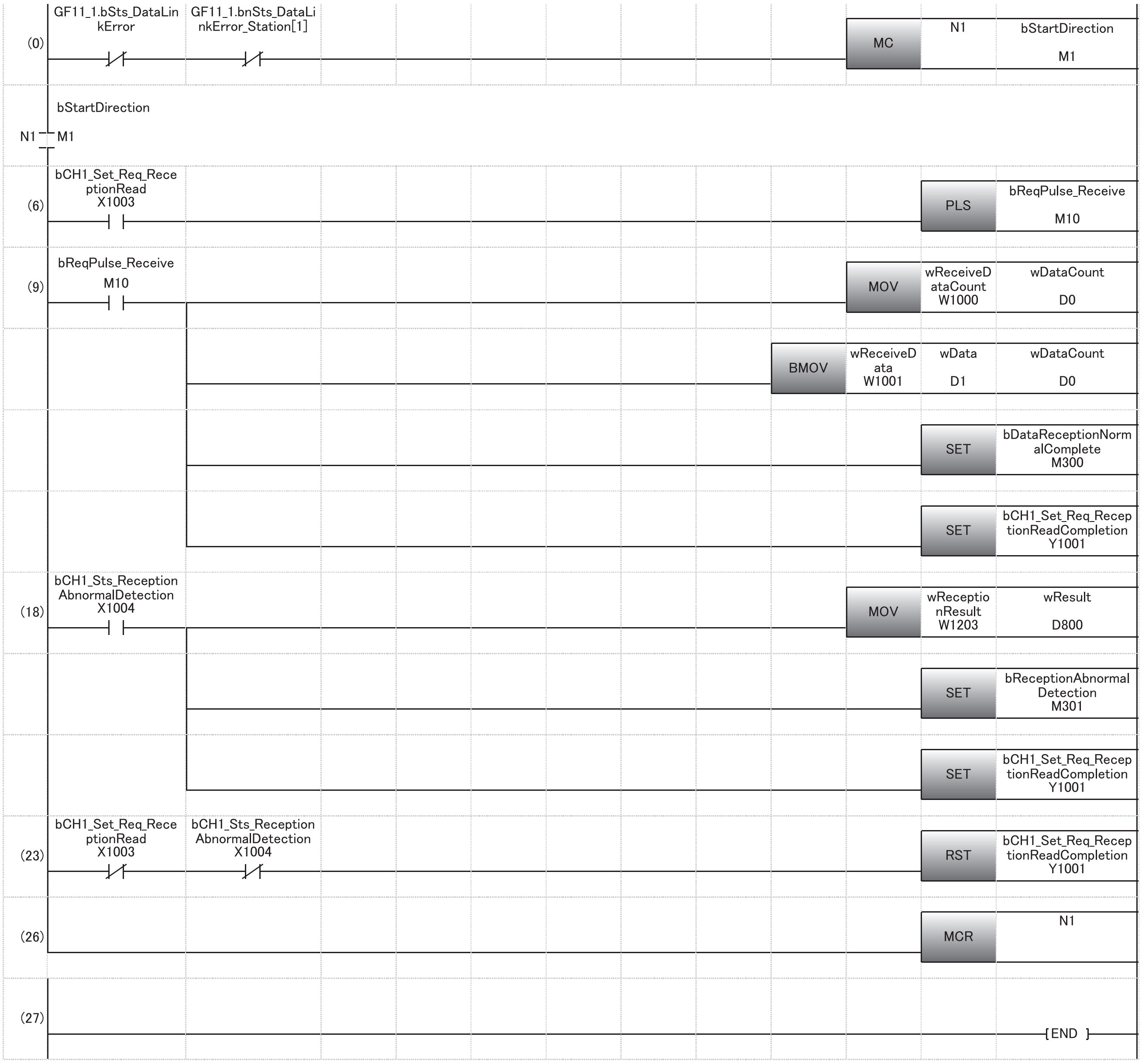
☞ Page 454 PARAMETER SETTING

■Program example for data reception

This program uses the module labels of the CC-Link IE Field Network master/local module. Write the program to the CPU module on the master station.

Category	Label name	Description	Device
Module label	GF11_1.bSts_DataLinkError	Data link error status of own station	SB0049
	GF11_1.bnSts_DataLinkError_Station[1]	Data link status of each station (station No.1)	SW00B0.0
Label to be defined	Define the global label as shown below. 		

Label Name	Data Type		Class	Assign (Device/Label)
bStartDirection	Bit	...	VAR_GLOBAL	M1
bReqPulse_Receive	Bit	...	VAR_GLOBAL	M10
bDataReceptionNormalComplete	Bit	...	VAR_GLOBAL	M300
bReceptionAbnormalDetection	Bit	...	VAR_GLOBAL	M301
bCH1_Set_Req_ReceptionRead	Bit	...	VAR_GLOBAL	X1003
bCH1_Sts_ReceptionAbnormalDetection	Bit	...	VAR_GLOBAL	X1004
bCH1_Set_Req_ReceptionReadCompletion	Bit	...	VAR_GLOBAL	Y1001
wReceiveDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W1000
wReceiveData	Word [Unsigned]/Bit String [16-bit](0..4)	...	VAR_GLOBAL	W1001
wReceptionResult	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W1203
wDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
wData	Word [Unsigned]/Bit String [16-bit](0..4)	...	VAR_GLOBAL	D1
wResult	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D800

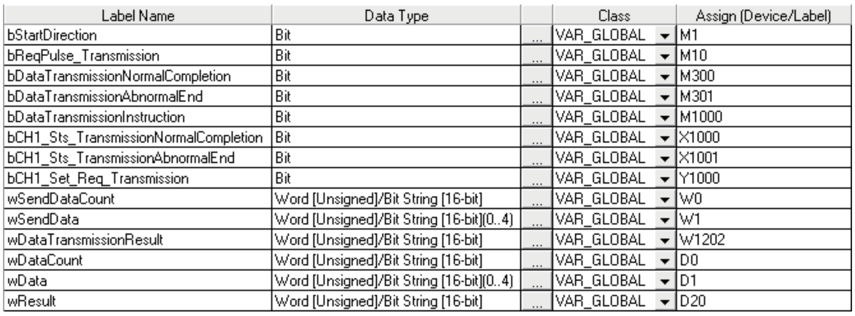

(0)
GF11_1.bSts_DataLinkError
GF11_1.bnSts_DataLinkError_Station[1]
MC
N1
bStartDirection
M1
bStartDirection
N1
M1
(6)
bCH1_Set_Req_ReceptionRead
X1003
PLS
bReqPulse_Receive
M10
(9)
bReqPulse_Receive
M10
MOV
wReceiveDataCount
W1000
wDataCount
D0
BMOV
wReceiveData
W1001
wData
D1
wDataCount
D0
SET
bDataReceptionNormalComplete
M300
SET
bCH1_Set_Req_ReceptionReadCompletion
Y1001
(18)
bCH1_Sts_ReceptionAbnormalDetection
X1004
MOV
wReceptionResult
W1203
wResult
D800
SET
bReceptionAbnormalDetection
M301
SET
bCH1_Set_Req_ReceptionReadCompletion
Y1001
(23)
bCH1_Set_Req_ReceptionRead
X1003
bCH1_Sts_ReceptionAbnormalDetection
X1004
RST
bCH1_Set_Req_ReceptionReadCompletion
Y1001
(26)
MCR
N1
(27)
END

(6) Reads the received data when the C24 receives data normally.

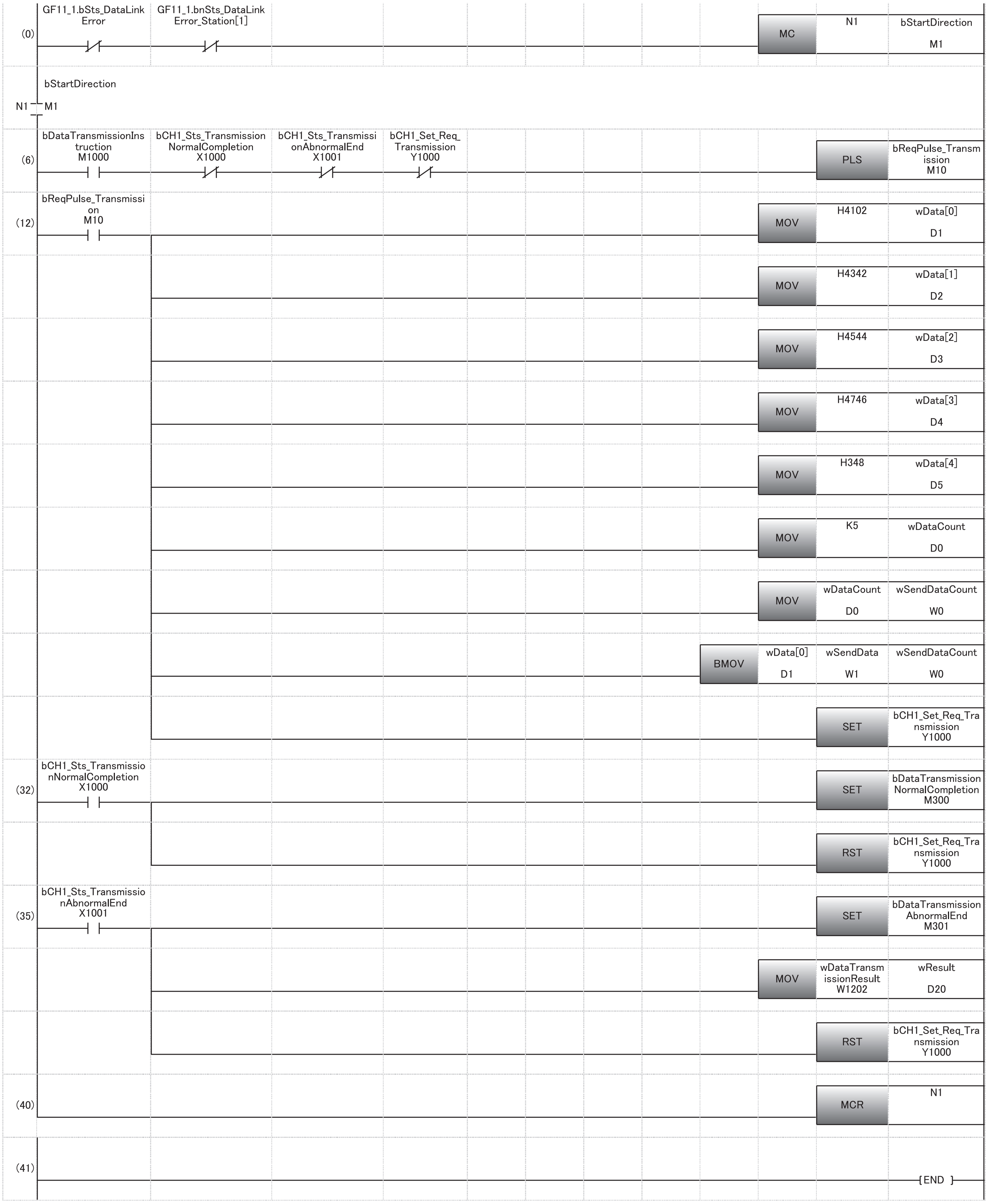
(18) Reads error description (error code) when the C24 detects an error at data reception. (Unnecessary for bidirectional protocols.)

■Program for data transmission

This program uses the module labels of the CC-Link IE Field Network master/local module. Write the program to the CPU module on the master station.

Category	Label name	Description	Device
Module label	GF11_1.bSts_DataLinkError	Data link error status of own station	SB0049
	GF11_1.bnSts_DataLinkError_Station[1]	Data link status of each station (station No.1)	SW00B0.0
Label to be defined	Define the global label as shown below. 		

Label Name	Data Type		Class	Assign (Device/Label)
bStartDirection	Bit	...	VAR_GLOBAL	M1
bReqPulse_Transmission	Bit	...	VAR_GLOBAL	M10
bDataTransmissionNormalCompletion	Bit	...	VAR_GLOBAL	M300
bDataTransmissionAbnormalEnd	Bit	...	VAR_GLOBAL	M301
bDataTransmissionInstruction	Bit	...	VAR_GLOBAL	M1000
bCH1_Sts_TransmissionNormalCompletion	Bit	...	VAR_GLOBAL	X1000
bCH1_Sts_TransmissionAbnormalEnd	Bit	...	VAR_GLOBAL	X1001
bCH1_Set_Req_Transmission	Bit	...	VAR_GLOBAL	Y1000
wSendDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W0
wSendData	Word [Unsigned]/Bit String [16-bit](0..4)	...	VAR_GLOBAL	W1
wDataTransmissionResult	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W1202
wDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
wData	Word [Unsigned]/Bit String [16-bit](0..4)	...	VAR_GLOBAL	D1
wResult	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D20


(0)
GF11_1.bSts_DataLinkError
GF11_1.bnSts_DataLinkError_Station[1]
MC N1 bStartDirection M1
N1 bStartDirection M1
(6)
bDataTransmissionInstruction M1000
bCH1_Sts_TransmissionNormalCompletion X1000
bCH1_Sts_TransmissionAbnormalEnd X1001
bCH1_Set_Req_Transmission Y1000
PLS bReqPulse_Transmission M10
(12)
bReqPulse_Transmission M10
MOV H4102 wData[0] D1
MOV H4342 wData[1] D2
MOV H4544 wData[2] D3
MOV H4746 wData[3] D4
MOV H348 wData[4] D5
MOV K5 wDataCount D0
MOV wDataCount D0 wSendDataCount W0
BMOV wData[0] D1 wSendData W1 wSendDataCount W0
SET bCH1_Set_Req_Transmission Y1000
(32)
bCH1_Sts_TransmissionNormalCompletion X1000
SET bDataTransmissionNormalCompletion M300
RST bCH1_Set_Req_Transmission Y1000
(35)
bCH1_Sts_TransmissionAbnormalEnd X1001
SET bDataTransmissionAbnormalEnd M301
MOV wDataTransmissionResult W1202 wResult D20
RST bCH1_Set_Req_Transmission Y1000
(40)
MCR N1
(41)
{END }

(6) Sets send data and issues a send request.

(32) Performs the processing when the transmission processing completes normally.

(35) Performs the processing when the transmission processing completes abnormally.

Appendix 8 Using MODBUS in a Redundant System

This section explains the restriction and communication examples when using MODBUS in a redundant system.
For the restrictions when using a C24 in a redundant system with a redundant extension base unit, refer to the following section.

☞ Page 661 Using the Module in the Redundant System with Redundant Extension Base Unit

Restrictions

Available modules

Use the following modules in a redundant system.

Model	Type	Firmware version
RnPCPU	Process CPU	'24' or later
R6RFM	Redundant function module	'01' or later
RJ71C24	Serial communication module	'13' or later
RJ71C24-R2		
RJ71C24-R4		

Available functions

When using a C24 in a redundant system, only the following functions can be used.

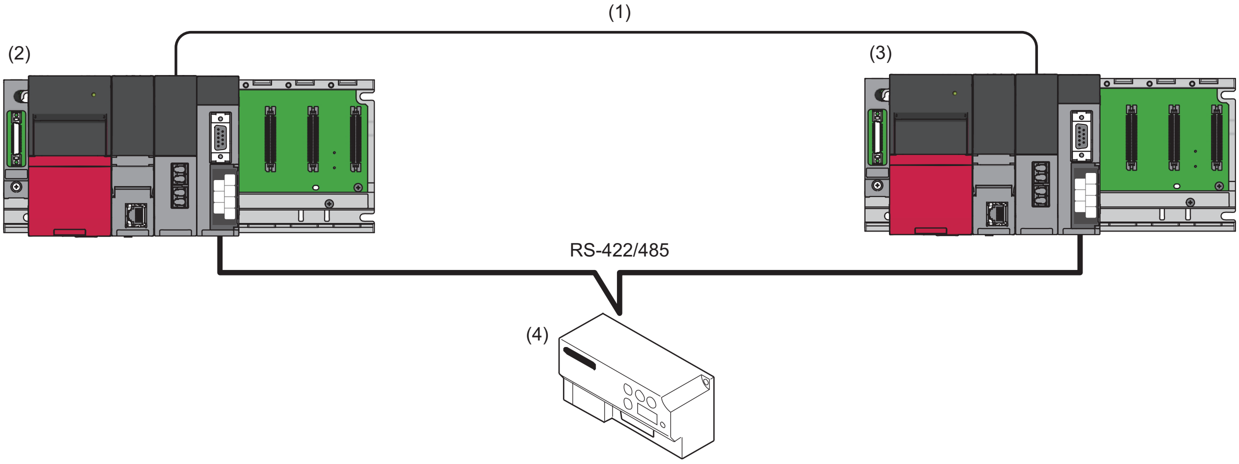
- Data communication using the predefined protocol
- Data communication using a MODBUS slave

Communication using a dedicated instruction

If the system is switched during dedicated instruction execution, the instruction may not be completed. Execute it again from the control system CPU module after system switching.

Redundant master station (single line)

Connect the master station of the control system and that of the standby system with a tracking cable.
If an error occurs in the master station of the control system, the master station of the standby system performs control and continues communications with a slave station.



(1) Tracking cable
(2) Master station (control system)
(3) Master station (standby system)
(4) MODBUS slave device

System switching

An execution result of the protocols registered for the predefined protocol support function (GP.CPRTCL) is monitored.
When an error occurs, the system switching instruction (SP.CONTSW) is executed in the process CPU (redundant mode) of the control system.[*1]

[*1] To execute this instruction, 'System switching by a user' (SM1646) must be turned ON (manual switching permitted) in advance.

Cases in which the system can be switched

The system is switched if any of the following errors occurs:

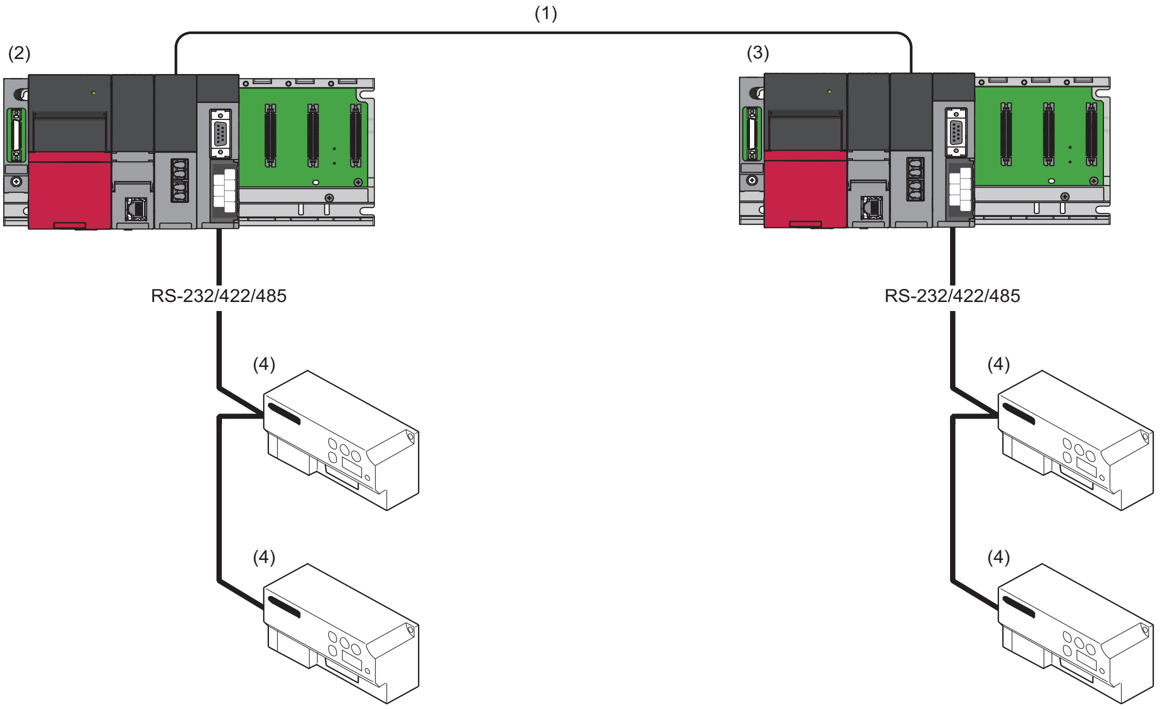
- Cable disconnection
- Communication error
- Power interruption

Redundant master station (redundant line)

Connect the master station of the control system and that of the standby system with a tracking cable.

Set a slave station for each master station.

If an error occurs in the master station of the control system, the master station of the standby system performs control and continues communications with slave stations.



(1) Tracking cable
(2) Master station (control system)
(3) Master station (standby system)
(4) MODBUS slave device

Precautions

- The configurations of the control system and the standby system must be the same.
- When using RS-232, only one MODBUS slave device can be connected.

System switching

An execution result of the protocols registered for the predefined protocol support function (GP.CPRTCL) is monitored.

When an error occurs, the system switching instruction (SP.CONTSW) is executed in the process CPU (redundant mode) of the control system.[*1]

[*1] To execute this instruction, 'System switching by a user' (SM1646) must be turned ON (manual switching permitted) in advance.

Cases in which the system can be switched

The system is switched if any of the following errors occurs:

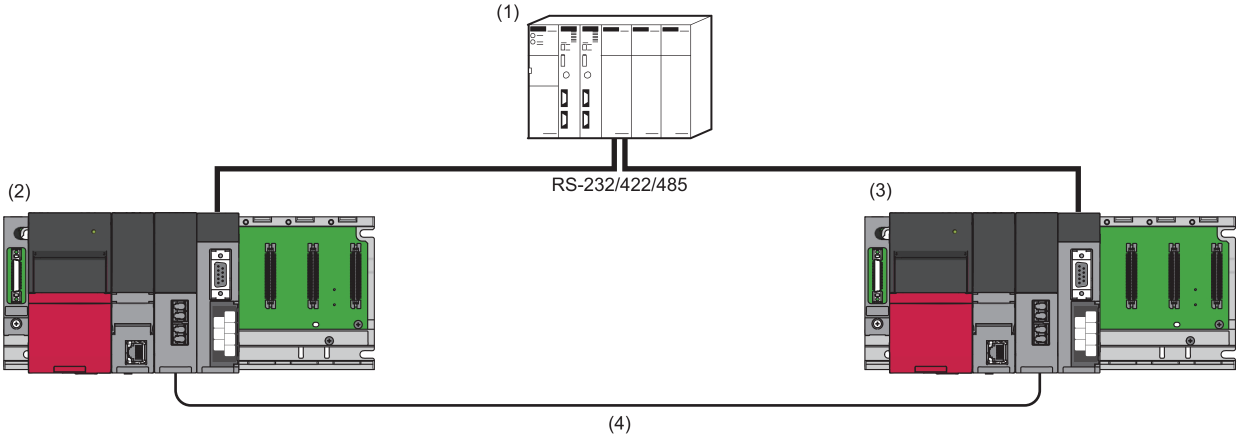
- Cable disconnection
- Communication error
- Power interruption

Redundant slave station (redundant line)

Connect the slave station of the control system and that of the standby system with a tracking cable.

Prepare a line to the master station for each of the slave stations of the control system and the standby system.

If an error occurs in the slave station of the control system, the slave station of the standby system performs control and continues communications with the master station.



(1) MODBUS master device
(2) Slave station (control system)
(3) Slave station (standby system)
(4) Tracking cable

System switching

Data for alive check is periodically sent from the master station to determine whether a communication error occurs, and the data is monitored in slave stations.

When an error occurs, the system switching instruction (SP.CONTSW) is executed in the process CPU (redundant mode) of the control system.[*1]

The master station determines which slave station is currently operating as the control system by reading the values in 'Control system judgment flag' (SM1634) and 'Standby system judgment flag' (SM1635) of each slave station.

[*1] To execute this instruction, 'System switching by a user' (SM1646) must be turned ON (manual switching permitted) in advance.

Cases in which the system can be switched

The system is switched if any of the following errors occurs:

- Cable disconnection
- Communication error
- Power interruption

Communication examples for a redundant master station (single line)

The following shows program examples to continue communications with a slave station by switching the master station of the standby system to the control system if an error occurs in the master station of the control system.

Point

Common sample programs can be used for redundant master stations (both single line and redundant line). Therefore, a redundant master station (single line) is used for explanation in communication examples shown below.

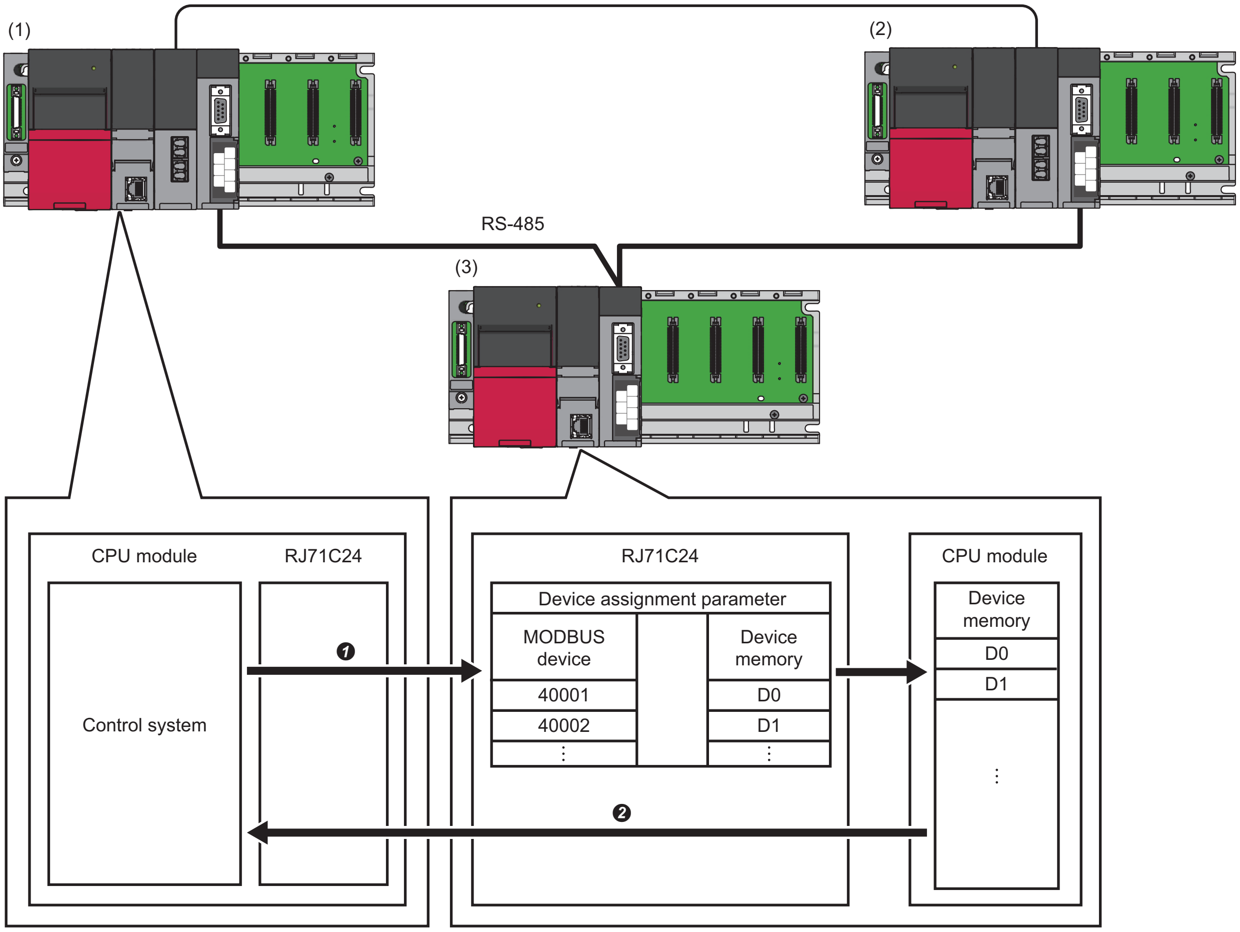
A program executed in a master station (standby system) is required for a redundant master station (single line). For details on this type of program, refer to the following section.

(☞ Page 639 Considerations for a redundant master station (single line))

Overview

The following shows the operation overviews of a sample program used for a communication example.

■Normal operation

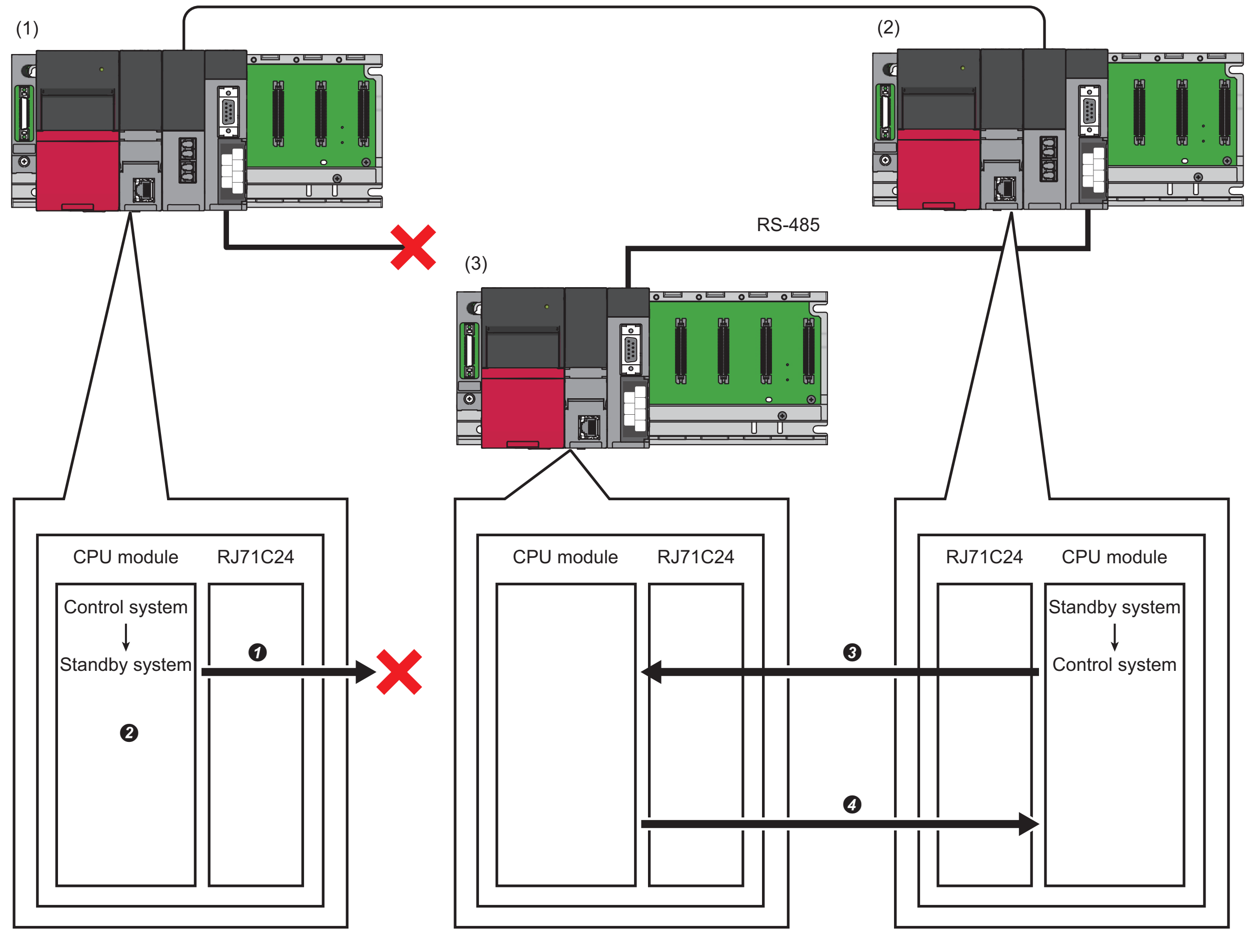


(1) Master station (control system): RJ71C24 (predefined protocol)
(2) Master station (standby system): RJ71C24 (predefined protocol)
(3) Slave station: RJ71C24 (MODBUS slave (RTU))

No.	Description
❶	The master station (control system) sends the following request message to the slave station with the GP.CPRTCL instruction. • Read holding registers (FC: 03) (request to read a value in the read holding register 400001)
❷	The slave station returns a value in D0 according to the MODBUS device assignment parameter settings. The master station (control system) stores received data in W5 according to protocol setting data.

The above operations are repeated every three seconds.

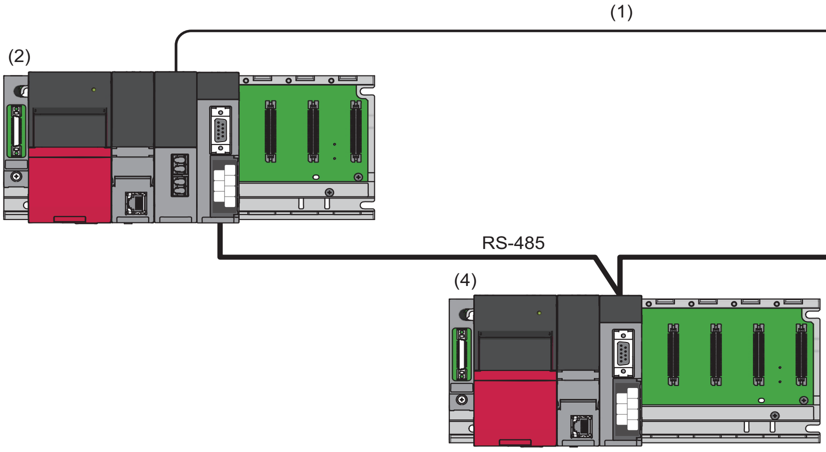
■When an error occurs



(1) Master station (control system): RJ71C24 (predefined protocol)
(2) Master station (standby system): RJ71C24 (predefined protocol)
(3) Slave station: RJ71C24 (MODBUS slave (RTU))

No.	Description
❶	The master station (control system) sends a request message to the slave station with the GP.CPRTCL instruction.
❷	If the GP.CPRTCL instruction completes with an error, the master station (control system) switches the system with the SP.CONTSW instruction.
❸	The master station (standby system) that has switched to the new control system sends a request message to the slave station with the GP.CPRTCL instruction in the same way as the master station (control system) that is the old control system.
❹	The slave station sends a response message to the master station (standby system) that has switched to the new control system.

System configuration



(1) Tracking cable
(2) Master station (control system): RJ71C24 (predefined protocol)
(3) Master station (standby system): RJ71C24 (predefined protocol)
(4) Slave station: RJ71C24 (MODBUS slave (RTU))

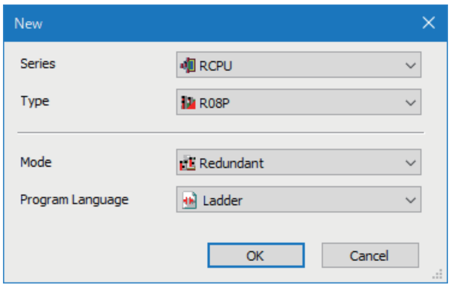
Type	Model	Type	Start I/O No.
Master station	RnPCPU	Process CPU	3E00H
	R6RFM	Redundant function module	0000H
	RJ71C24	Serial communication module	0020H
Slave station	RCPU	CPU module	3E00H
	RJ71C24	Serial communication module	0000H

Setting for a master station (control system)

The following shows the procedure for connecting to the CPU module on a master station and setting parameters in an engineering tool.

1. Set the CPU module as follows:

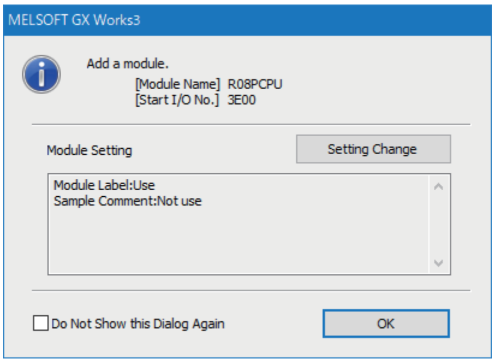
[Project] ⇨ [New]


New

Series: RCPU
Type: R08P
Mode: Redundant
Program Language: Ladder

2. Click the [Setting Change] button to set to use module labels.

3. Click the [OK] button to add module labels of the CPU module.


MELSOFT GX Works3

Add a module.
[Module Name] R08PCPU
[Start I/O No.] 3E00

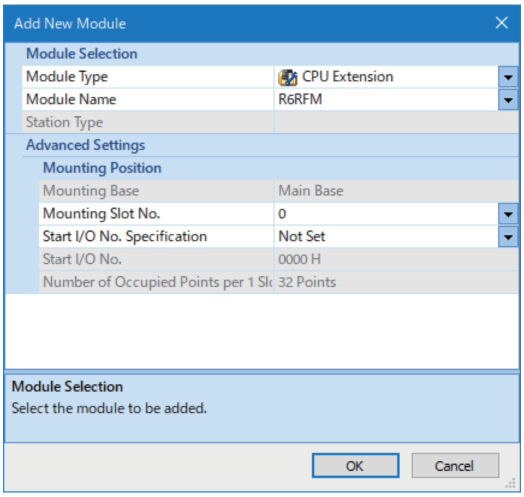
Module Setting

Module Label:Use
Sample Comment:Not use

□ Do Not Show this Dialog Again

4. Set the R6RFM as follows:

[Navigation window] ⇨ [Parameter] ⇨ [System Parameter] ⇨ [I/O Assignment] tab ⇨ [I/O Assignment Setting]



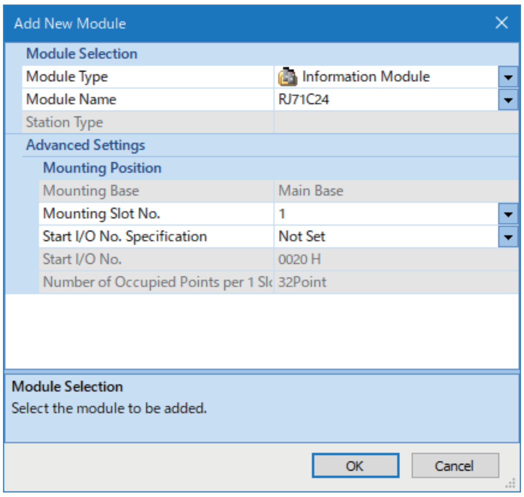
Add New Module

Module Selection	
Module Type	CPU Extension
Module Name	R6RFM
Station Type	
Advanced Settings	
Mounting Position	
Mounting Base	Main Base
Mounting Slot No.	0
Start I/O No. Specification	Not Set
Start I/O No.	0000 H
Number of Occupied Points per 1 Slo	32 Points

Module Selection
Select the module to be added.

5. Set the RJ71C24 as follows:

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ right-click ⇨ [Add New Module]

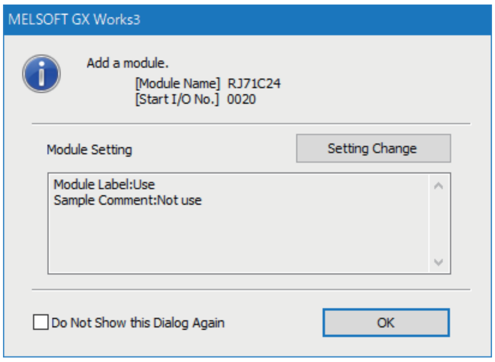


Add New Module

Module Selection	
Module Type	Information Module
Module Name	RJ71C24
Station Type	
Advanced Settings	
Mounting Position	
Mounting Base	Main Base
Mounting Slot No.	1
Start I/O No. Specification	Not Set
Start I/O No.	0020 H
Number of Occupied Points per 1 Slo	32Point

Module Selection
Select the module to be added.

6. Click the [OK] button to add module labels of the RJ71C24.



MELSOFT GX Works3

Add a module.
[Module Name] RJ71C24
[Start I/O No.] 0020

Module Setting

Module Label:Use
Sample Comment:Not use

□ Do Not Show this Dialog Again

7. Set the RS-485 (CH2) of the master station as follows:

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter] ⇨ [Basic Settings] ⇨ [Various control specification]

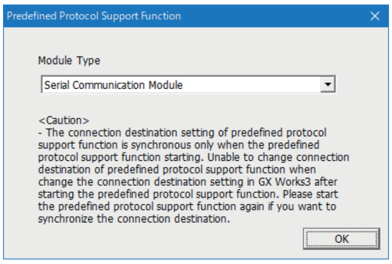
Item			Description
Various control specification	Communication protocol setting		Predefined protocol
	Communication speed setting		230400 bps
	Transmission setting	Operation setting	Independent
		Data bit	8
		Parity bit	None
		Odd/even parity	—
		Stop bit	1
		Sum check code	None
		Online change	Enable
		Setting change	Disable
	Station number setting		0
	MODBUS station number setting		—
	Communication control setting	Echo back enable/prohibit specification	Echo back prohibit

8. Write the set parameters to the CPU module that controls the master station.

[Online] ⇨ [Write to PLC]

9. Select [Tool] ⇨ [Predefined Protocol Support Function] in an engineering tool.

10. Select "Module Type" ⇨ "Serial Communication Module."

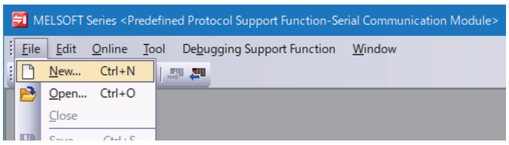


Module Type

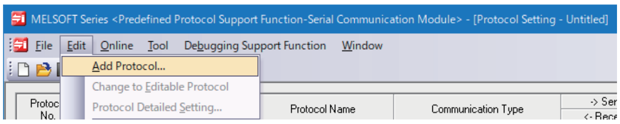
Serial Communication Module

<Caution>
- The connection destination setting of predefined protocol support function is synchronous only when the predefined protocol support function starting. Unable to change connection destination of predefined protocol support function when change the connection destination setting in GX Works3 after starting the predefined protocol support function. Please start the predefined protocol support function again if you want to synchronize the connection destination.

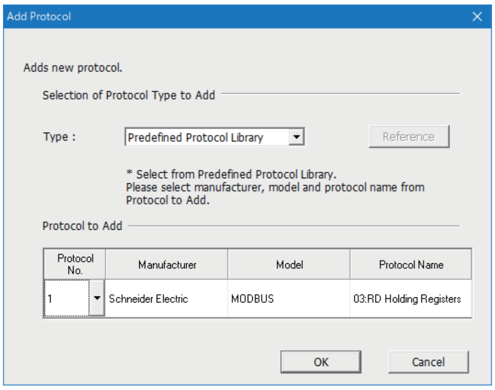
11. Select [File] ⇨ [New].


MELSOFT Series <Predefined Protocol Support Function-Serial Communication Module>
File
Edit
Online
Tool
Debugging Support Function
Window
New... Ctrl+N
Open... Ctrl+O
Close

12. Select [Edit] ⇨ [Add Protocol].


MELSOFT Series <Predefined Protocol Support Function-Serial Communication Module> - [Protocol Setting - Untitled]
File
Edit
Online
Tool
Debugging Support Function
Window
Add Protocol...
Change to Editable Protocol
Protocol Detailed Setting...
Protoc No.
Protocol Name
Communication Type
-> Ser
<- Rece

13. Select "Predefined Protocol Library" for "Type," and select a protocol to be used in "Protocol to Add."



Add Protocol

Adds new protocol.

Selection of Protocol Type to Add

Type : Predefined Protocol Library

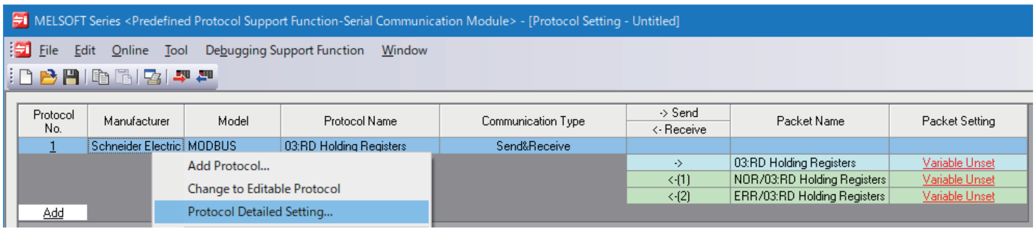
* Select from Predefined Protocol Library.
Please select manufacturer, model and protocol name from Protocol to Add.

Protocol to Add

Protocol No.	Manufacturer	Model	Protocol Name
1	Schneider Electric	MODBUS	03:RD Holding Registers

Protocol No.	Manufacturer	Model	Protocol name
1	Schneider Electric	MODBUS	03: RD Holding Registers

14. Right-click the added protocol, and select [Protocol Detailed Setting] from the shortcut menu.

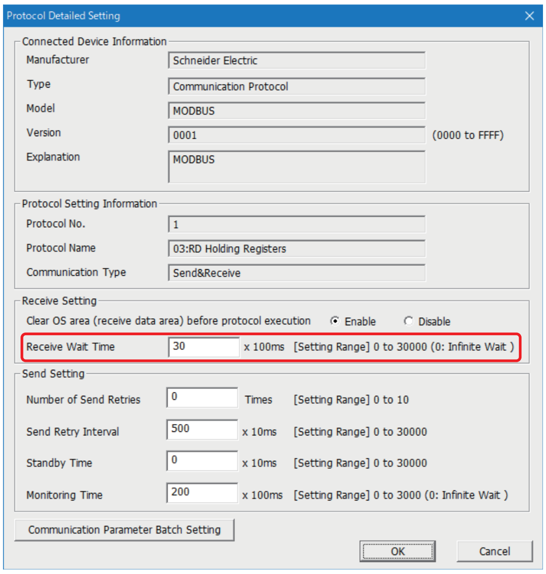


MELSOFT Series <Predefined Protocol Support Function-Serial Communication Module> - [Protocol Setting - Untitled]

Protocol No.	Manufacturer	Model	Protocol Name	Communication Type	-> Send <- Receive	Packet Name	Packet Setting
1	Schneider Electric	MODBUS	03:RD Holding Registers	Send&Receive			
					->	03:RD Holding Registers	Variable Unset
					<-(1)	NOR/03:RD Holding Registers	Variable Unset
					<-(2)	ERR/03:RD Holding Registers	Variable Unset

File
Edit
Online
Tool
Debugging Support Function
Window
Add Protocol...
Change to Editable Protocol
Protocol Detailed Setting...
Add

15. Set '30' for "Receive Wait Time."



Protocol Detailed Setting

Connected Device Information

Manufacturer: Schneider Electric

Type: Communication Protocol

Model: MODBUS

Version: 0001 (0000 to FFFF)

Explanation: MODBUS

Protocol Setting Information

Protocol No.: 1

Protocol Name: 03:RD Holding Registers

Communication Type: Send&Receive

Receive Setting

Clear OS area (receive data area) before protocol execution ● Enable ○ Disable

Receive Wait Time: 30 x 100ms [Setting Range] 0 to 30000 (0: Infinite Wait)

Send Setting

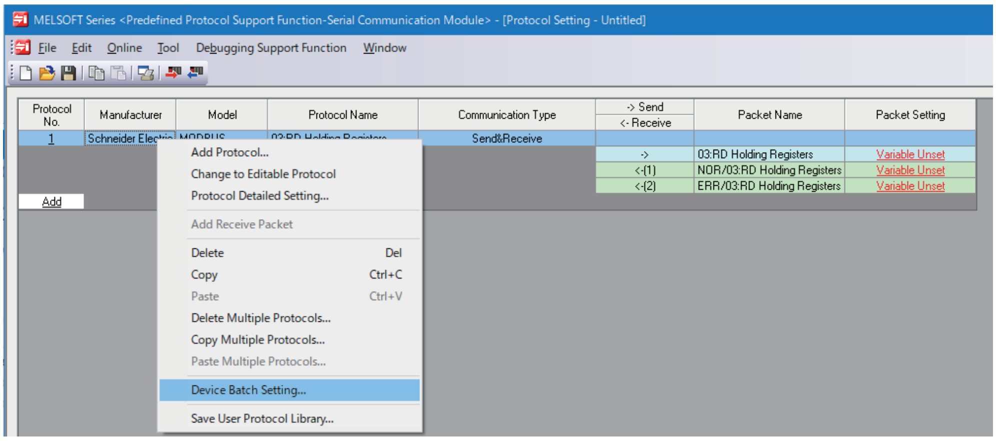
Number of Send Retries: 0 Times [Setting Range] 0 to 10

Send Retry Interval: 500 x 10ms [Setting Range] 0 to 30000

Standby Time: 0 x 10ms [Setting Range] 0 to 30000

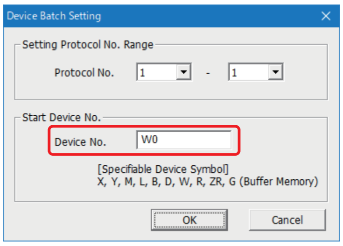
Monitoring Time: 200 x 100ms [Setting Range] 0 to 3000 (0: Infinite Wait)

16. Right-click the added protocol, and select [Device Batch Setting] from the shortcut menu.


MELSOFT Series <Predefined Protocol Support Function-Serial Communication Module> - [Protocol Setting - Untitled]
File Edit Online Tool Debugging Support Function Window
Add
Add Protocol...
Change to Editable Protocol
Protocol Detailed Setting...
Add Receive Packet
Delete Del
Copy Ctrl+C
Paste Ctrl+V
Delete Multiple Protocols...
Copy Multiple Protocols...
Paste Multiple Protocols...
Device Batch Setting...
Save User Protocol Library...

Protocol No.	Manufacturer	Model	Protocol Name	Communication Type	-> Send <- Receive	Packet Name	Packet Setting
1	Schneider Electric	MODBUS	03:RD Holding Registers	Send&Receive			
					->	03:RD Holding Registers	Variable Unset
					<-(1)	NOR/03:RD Holding Registers	Variable Unset
					<-(2)	ERR/03:RD Holding Registers	Variable Unset

17. Set 'W0' for "Device No."


Device Batch Setting

Setting Protocol No. Range

Protocol No. 1 - 1

Start Device No.

Device No. W0

[Specifiable Device Symbol]
X, Y, M, L, B, D, W, R, ZR, G (Buffer Memory)

18. Write the set protocol setting data to the module.

[Online] ⇨ [Write to Module]

19. Reset the CPU module or turn the power OFF and ON.

Precautions

- If '0' is set for "Receive Wait Time," the system is not switched because it continues waiting for a response from a slave station.
- If the CPU module is in the backup mode, protocol setting data cannot be written. Be sure to switch the CPU module to the separate mode before writing.
- Protocol setting data cannot be written to both systems at the same time. Write it to the control system and the standby system individually.
- The target memory in which protocol setting data is written must be the same for the control system and the standby system.

Setting for a master station (standby system)

Set a master station (standby system) with the same procedure as a master station (control system).

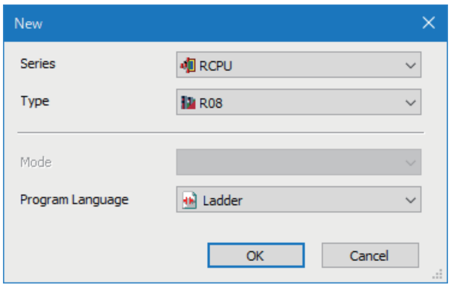
☞ Page 630 Setting for a master station (control system)

Setting for a slave station

The following shows the procedure for connecting to the CPU module on a slave station and setting parameters in an engineering tool.

1. Set the CPU module as follows:

[Project] ⇨ [New]

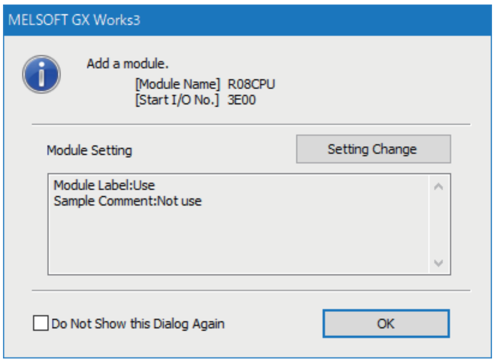


New

Series	RCPU
Type	R08
Mode	
Program Language	Ladder

2. Click the [Setting Change] button to set to use module labels.

3. Click the [OK] button to add module labels of the CPU module.



MELSOFT GX Works3

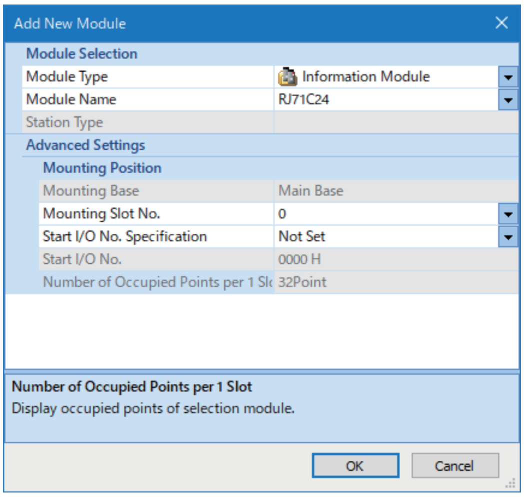
Add a module.
[Module Name] R08CPU
[Start I/O No.] 3E00

Module Setting

Module Label:Use
Sample Comment:Not use

4. Set the RJ71C24 as follows:

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ right-click ⇨ [Add New Module]

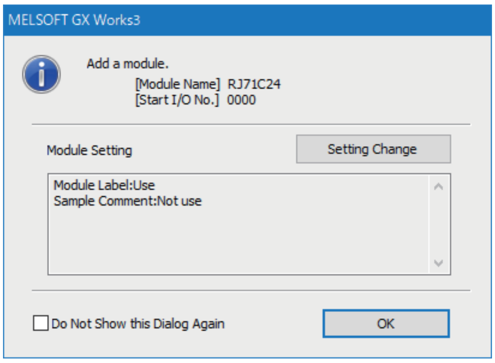


Add New Module

Module Selection	
Module Type	Information Module
Module Name	RJ71C24
Station Type	
Advanced Settings	
Mounting Position	
Mounting Base	Main Base
Mounting Slot No.	0
Start I/O No. Specification	Not Set
Start I/O No.	0000 H
Number of Occupied Points per 1 Slo	32Point

Number of Occupied Points per 1 Slot
Display occupied points of selection module.

5. Click the [OK] button to add module labels of the RJ71C24.



MELSOFT GX Works3

Add a module.
[Module Name] RJ71C24
[Start I/O No.] 0000

Module Setting

Module Label:Use
Sample Comment:Not use

6. Set the RS-485 (CH2) of the slave station as follows:

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter] ⇨ [Basic Settings] ⇨ [Various control specification]

Item			Description
Various control specification	Communication protocol setting		MODBUS slave (RTU)
	Communication speed setting*1		230400 bps
	Transmission setting	Operation setting	Independent
		Data bit*1	8
		Parity bit*1	None
		Odd/even parity*1	—
		Stop bit*1	1
		Sum check code*1	None
		Online change	Enable
		Setting change	Disable
	Station number setting		0
	MODBUS station number setting		1

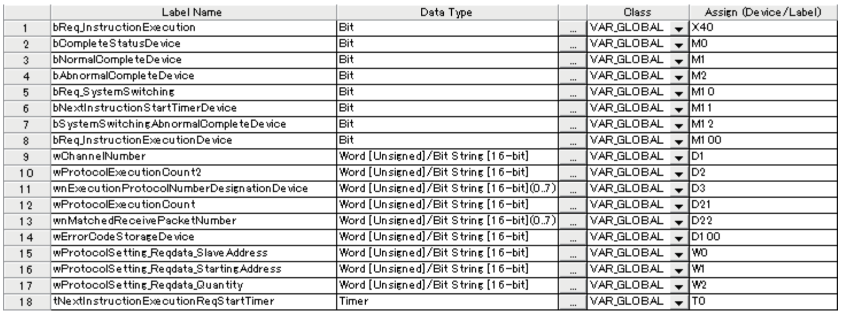
*1 Set the same value as the master station.

7. Write the set parameters to the CPU module on the slave station.

[Online] ⇨ [Write to PLC]

8. Reset the CPU module or turn the power OFF and ON.

Program example for a master station

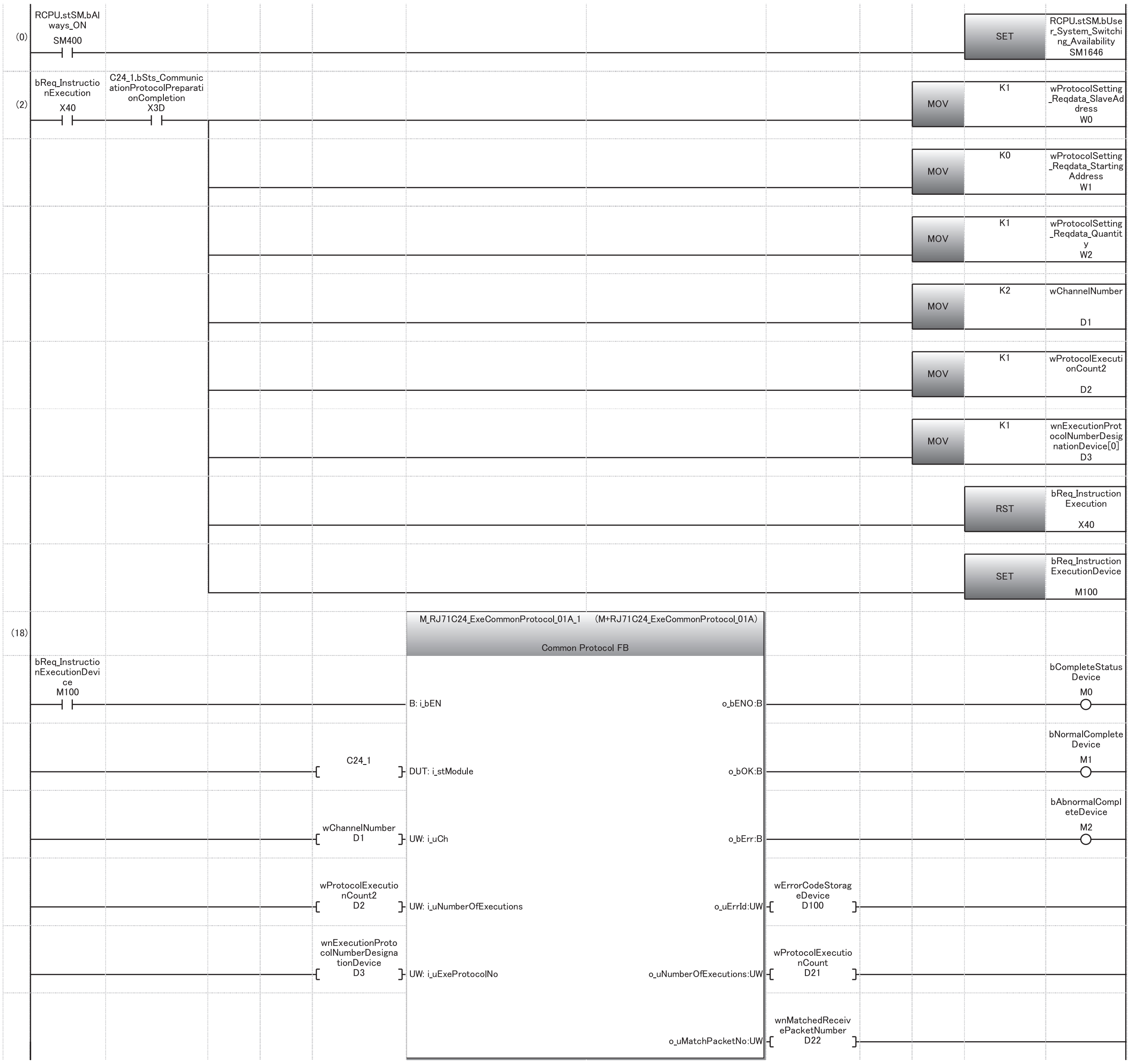
Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_1.bSts_CommunicationProtocolPreparationCompletion	Predefined protocol ready	X3D*1
	RCPU.stSM.bAlways_ON	Always ON	SM400
	RCPU.stSM.bUser_System_Switching_Availability	System switching by a user	SM1646
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_ExeCommonProtocol	Executes the protocol registered with GX Works3.	—

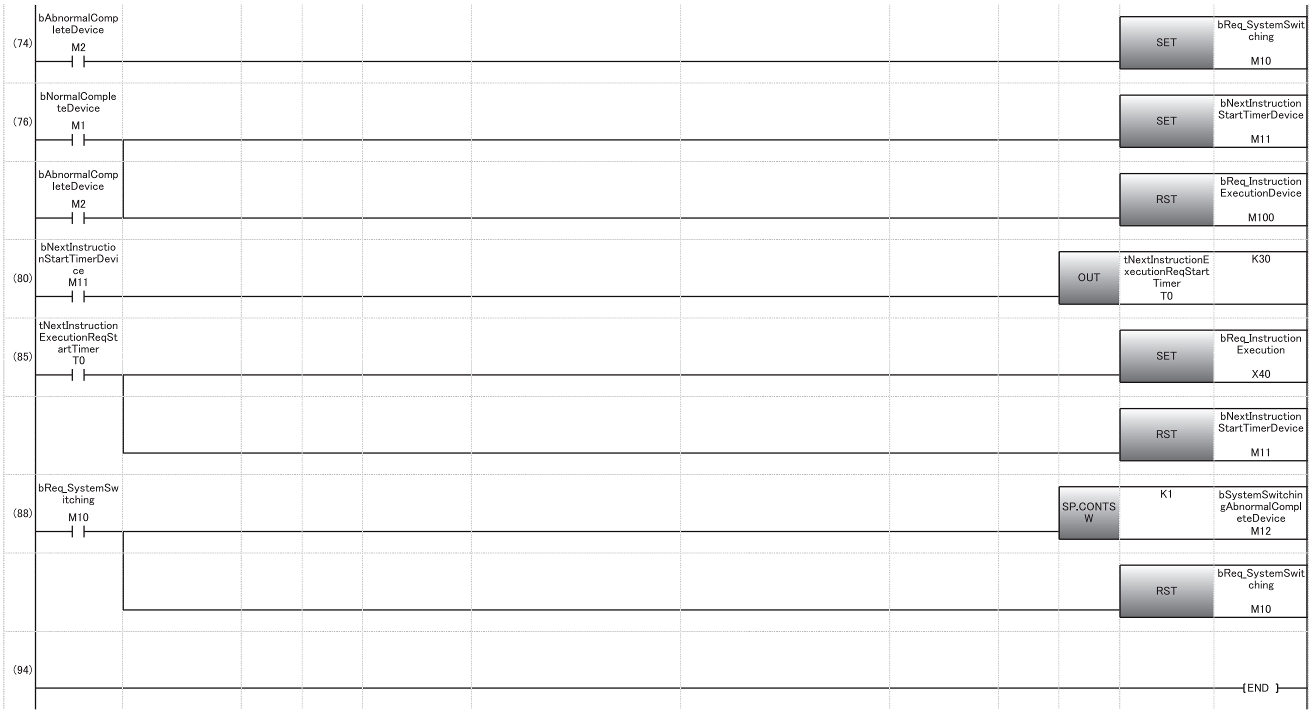
	Label Name	Data Type		Class	Assign (Device/Label)
1	bReqInstructionExecution	Bit	...	VAR_GLOBAL	X40
2	bCompleteStatusDevice	Bit	...	VAR_GLOBAL	M0
3	bNormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
4	bAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M2
5	bReq_SystemSwitching	Bit	...	VAR_GLOBAL	M10
6	bNextInstructionStartTimerDevice	Bit	...	VAR_GLOBAL	M11
7	bSystemSwitchingAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M12
8	bReqInstructionExecutionDevice	Bit	...	VAR_GLOBAL	M100
9	wChannelNumber	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
10	wProtocolExecutionCount2	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2
11	wnExecutionProtocolNumberDesignationDevice	Word [Unsigned]/Bit String [16-bit](0..7)	...	VAR_GLOBAL	D3
12	wProtocolExecutionCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D21
13	wnMatchedReceivePacketNumber	Word [Unsigned]/Bit String [16-bit](0..7)	...	VAR_GLOBAL	D22
14	wErrorCodeStorageDevice	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D100
15	wProtocolSetting_Reqdata_SlaveAddress	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W0
16	wProtocolSetting_Reqdata_StartingAddress	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W1
17	wProtocolSetting_Reqdata_Quantity	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W2
18	tNextInstructionExecutionReqStartTimer	Timer	...	VAR_GLOBAL	T0

*1 Do not specify it as a target for tracking transfer. (MELSEC iQ-R CPU Module User's Manual (Application))

For information on the module FB to be used, refer to the following manual.

MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
RCPU.stSM.bAlways_ON
SM400
SET
RCPU.stSM.bUser_System_Switching_Availability
SM1646
(2)
bReq_InstructionExecution
X40
C24_1.bSts_CommunicationProtocolPreparationCompletion
X3D
MOV
K1
wProtocolSetting_Reqdata_SlaveAddress
W0
MOV
K0
wProtocolSetting_Reqdata_StartingAddress
W1
MOV
K1
wProtocolSetting_Reqdata_Quantity
W2
MOV
K2
wChannelNumber
D1
MOV
K1
wProtocolExecutionCount2
D2
MOV
K1
wnExecutionProtocolNumberDesignationDevice[0]
D3
RST
bReq_InstructionExecution
X40
SET
bReq_InstructionExecutionDevice
M100
(18)
M_RJ71C24_ExeCommonProtocol_01A_1 (M+RJ71C24_ExeCommonProtocol_01A)
Common Protocol FB
bReq_InstructionExecutionDevice
M100
B: i_bEN
o_bENO:B
bCompleteStatusDevice
M0
C24_1
DUT: i_stModule
o_bOK:B
bNormalCompleteDevice
M1
wChannelNumber
D1
UW: i_uCh
o_bErr:B
bAbnormalCompleteDevice
M2
wProtocolExecutionCount2
D2
UW: i_uNumberOfExecutions
o_uErrId:UW
wErrorCodeStorageDevice
D100
wnExecutionProtocolNumberDesignationDevice
D3
UW: i_uExeProtocolNo
o_uNumberOfExecutions:UW
wProtocolExecutionCount
D21
o_uMatchPacketNo:UW
wnMatchedReceivePacketNumber
D22


(74)
bAbnormalCompleteDevice
M2
SET
bReq_SystemSwitching
M10
(76)
bNormalCompleteDevice
M1
SET
bNextInstructionStartTimerDevice
M11
bAbnormalCompleteDevice
M2
RST
bReq_InstructionExecutionDevice
M100
(80)
bNextInstructionStartTimerDevice
M11
OUT
tNextInstructionExecutionReqStartTimer
T0
K30
(85)
tNextInstructionExecutionReqStartTimer
T0
SET
bReq_InstructionExecution
X40
RST
bNextInstructionStartTimerDevice
M11
(88)
bReq_SystemSwitching
M10
SP.CONTSW
K1
bSystemSwitchingAbnormalCompleteDevice
M12
RST
bReq_SystemSwitching
M10
(94)
END

(0) Turns SM1646 ON to execute the SP.CONTSW instruction.

(2) Sets a value to the device assigned in the protocol setting.

- Sets '1' for the request destination slave station number.
- Sets the start holding register number and the number of read points so that MODBUS device (400001) will be read.

Sets a value for the input label of M+RJ71C24_ExeCommonProtocol.

- Sets '2' for the communication channel.
- Sets '1' for the number of consecutive protocol executions.
- Sets '1' for the execution protocol number specification.

Turns M+RJ71C24_ExeCommonProtocol execution request ON.

(18) Executes M+RJ71C24_ExeCommonProtocol.

(74) Turns the SP.CONTSW instruction execution request ON if an error occurs in completion.

(76) Starts the timer for next sending after completion of M+RJ71C24_ExeCommonProtocol.

(80) Measures for three seconds.

(85) Turns the next transmission request execution request ON after three seconds.

(88) Switches the system with the SP.CONTSW instruction if an error occurs in completion.

Program example for a slave station

Create a program for a slave station as necessary.

Considerations for a redundant master station (single line)

For a redundant master station (single line), a master station (standby system) also receives data sent by a master station (control system) and a slave station. This causes a buffer full error (7F6AH) in the master station (standby system). If a buffer full error occurs, the buffer area is cleared when the master station (standby system) switches to the new control system; therefore, the new control system operates normally after system switching.

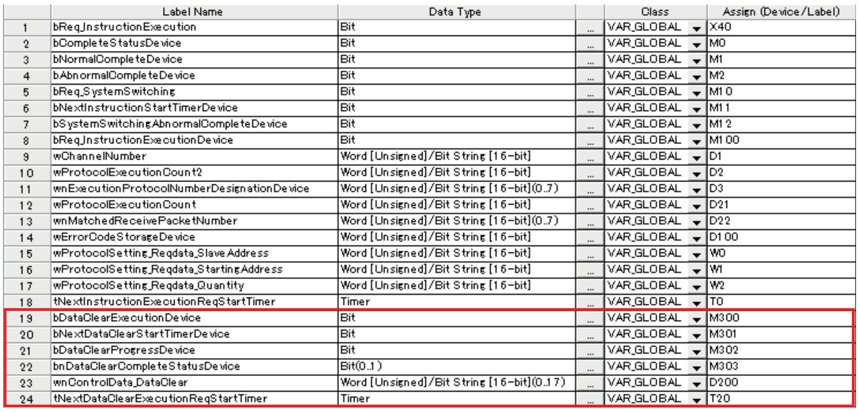
To avoid a buffer full error in a master station (standby system), a functional protocol 'receive data clear (protocol No.201)' of the predefined protocol function must be periodically executed in the master station (standby system).

In addition, add a sample program executed in the master station (standby system) and correct a sample program executed in the master station (control system).

Precautions

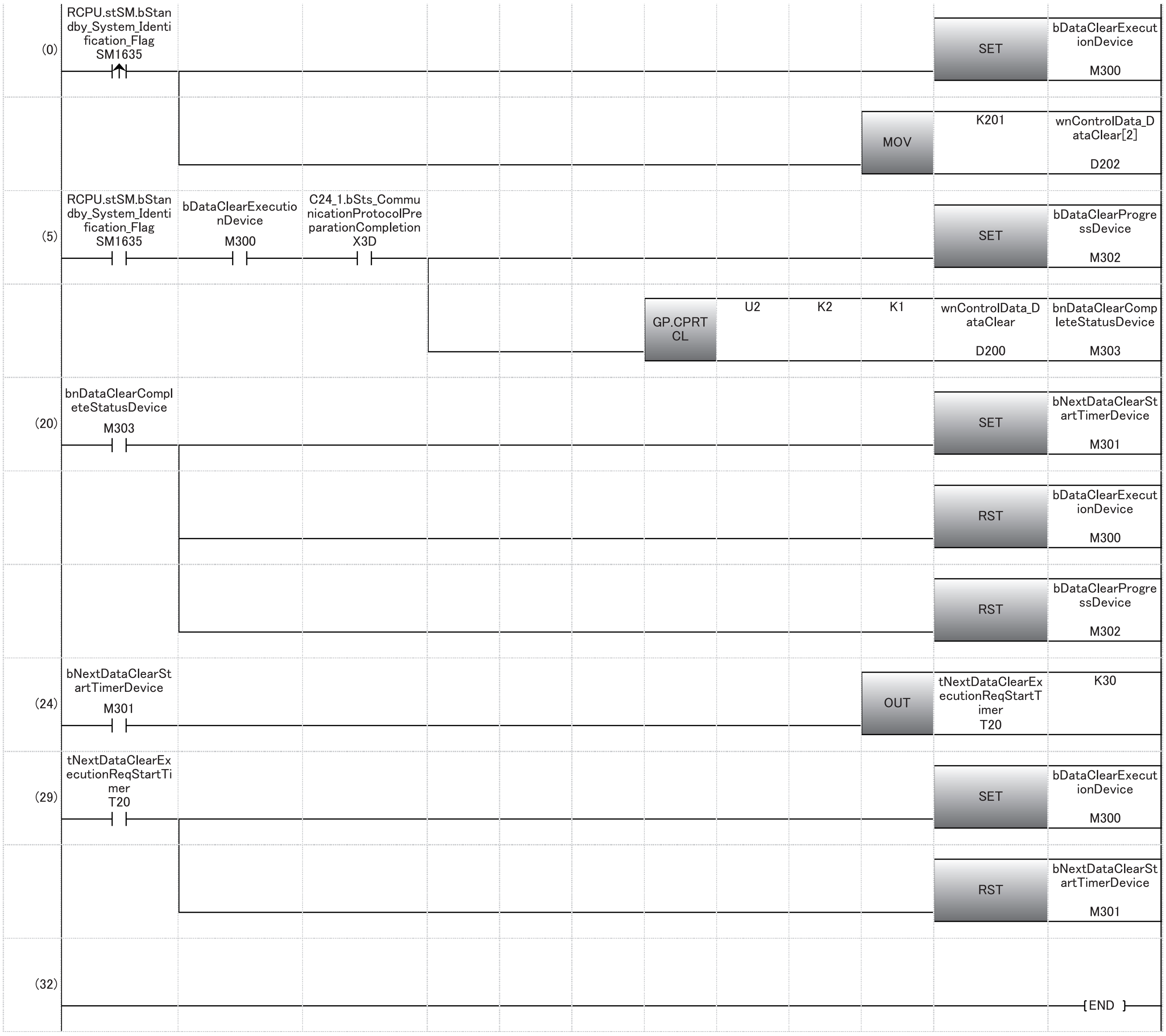
- Execute a program in both systems.
  For details, refer to the following manual.
  (📖MELSEC iQ-R CPU Module User's Manual (Application))
- Do not specify a device assigned to a module label or global label for a sample program as a target for tracking transfer.
  For details on tracking transfer, refer to the following manual.
  (📖MELSEC iQ-R CPU Module User's Manual (Application))

■Example of a program added for a master station (standby system)

Category	Label name	Description	Device
Module label	C24_1	Target module	—
	C24_1.bSts_CommunicationProtocolPreparationCompletion	Predefined protocol ready	X3D
	RCPU.stSM.bStandby_System_Identification_Flag	Standby system judgment flag	SM1635
Label to be defined	Define the global label as shown below.[*1] 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bReq_InstructionExecution	Bit	...	VAR_GLOBAL	X40
2	bCompleteStatusDevice	Bit	...	VAR_GLOBAL	M0
3	bNormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
4	bAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M2
5	bReq_SystemSwitching	Bit	...	VAR_GLOBAL	M10
6	bNextInstructionStartTimerDevice	Bit	...	VAR_GLOBAL	M11
7	bSystemSwitchingAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M12
8	bReq_InstructionExecutionDevice	Bit	...	VAR_GLOBAL	M100
9	wChannelNumber	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
10	wProtocolExecutionCount2	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2
11	wnExecutionProtocolNumberDesignationDevice	Word [Unsigned]/Bit String [16-bit](0..7)	...	VAR_GLOBAL	D3
12	wProtocolExecutionCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D21
13	wnMatchedReceivePacketNumber	Word [Unsigned]/Bit String [16-bit](0..7)	...	VAR_GLOBAL	D22
14	wErrorCodeStorageDevice	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D100
15	wProtocolSetting_Reqdata_SlaveAddress	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W0
16	wProtocolSetting_Reqdata_StartingAddress	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W1
17	wProtocolSetting_Reqdata_Quantity	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W2
18	tNextInstructionExecutionReqStartTimer	Timer	...	VAR_GLOBAL	T0
19	bDataClearExecutionDevice	Bit	...	VAR_GLOBAL	M300
20	bNextDataClearStartTimerDevice	Bit	...	VAR_GLOBAL	M301
21	bDataClearProgressDevice	Bit	...	VAR_GLOBAL	M302
22	bnDataClearCompleteStatusDevice	Bit(0..1)	...	VAR_GLOBAL	M303
23	wnControlData_DataClear	Word [Unsigned]/Bit String [16-bit](0..17)	...	VAR_GLOBAL	D200
24	tNextDataClearExecutionReqStartTimer	Timer	...	VAR_GLOBAL	T20

[*1] Add and define global labels as shown in the red frame.


(0)
RCPU.stSM.bStandby_System_Identification_Flag SM1635
SET
bDataClearExecutionDevice M300
MOV
K201
wnControlData_DataClear[2] D202
(5)
RCPU.stSM.bStandby_System_Identification_Flag SM1635
bDataClearExecutionDevice M300
C24_1.bSts_CommunicationProtocolPreparationCompletion X3D
SET
bDataClearProgressDevice M302
GP.CPRTCL
U2
K2
K1
wnControlData_DataClear D200
bnDataClearCompleteStatusDevice M303
(20)
bnDataClearCompleteStatusDevice M303
SET
bNextDataClearStartTimerDevice M301
RST
bDataClearExecutionDevice M300
RST
bDataClearProgressDevice M302
(24)
bNextDataClearStartTimerDevice M301
OUT
tNextDataClearExecutionReqStartTimer T20
K30
(29)
tNextDataClearExecutionReqStartTimer T20
SET
bDataClearExecutionDevice M300
RST
bNextDataClearStartTimerDevice M301
(32)
END

(0) Turns the GP.ECPRTCL instruction execution request ON.

- Sets functional protocol No.201 for control data 'execution protocol number specification (s+2)' of GP.ECPRTCL.

(5) Executes the GP.ECPRTCL instruction.

(24) Starts the timer to clear the next transmitted data after the GP.ECPRTCL instruction is completed.*1

(29) Turns the next GP.ECPRTCL instruction execution request ON.

*1 The timer value must be less than or equal to the transmission cycle from the control system MODBUS master station.

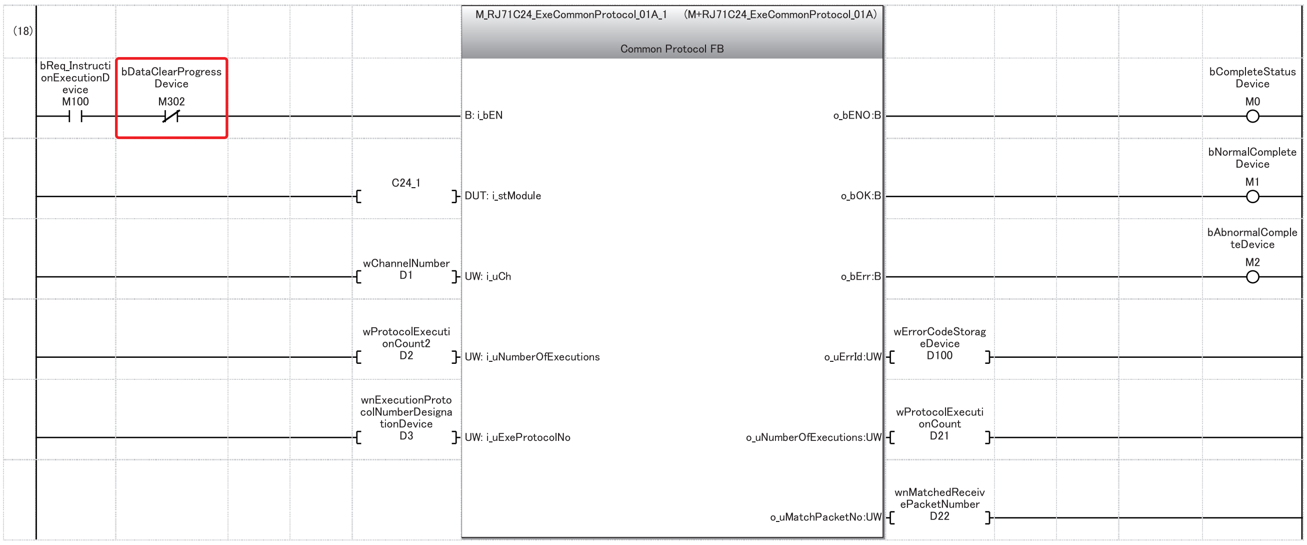
■Example of a program corrected for a master station (control system)

Correct the program shown in the following section.

☞ Page 636 Program example for a master station

- Part to be corrected

Add a close contact (M302) in the red frame.



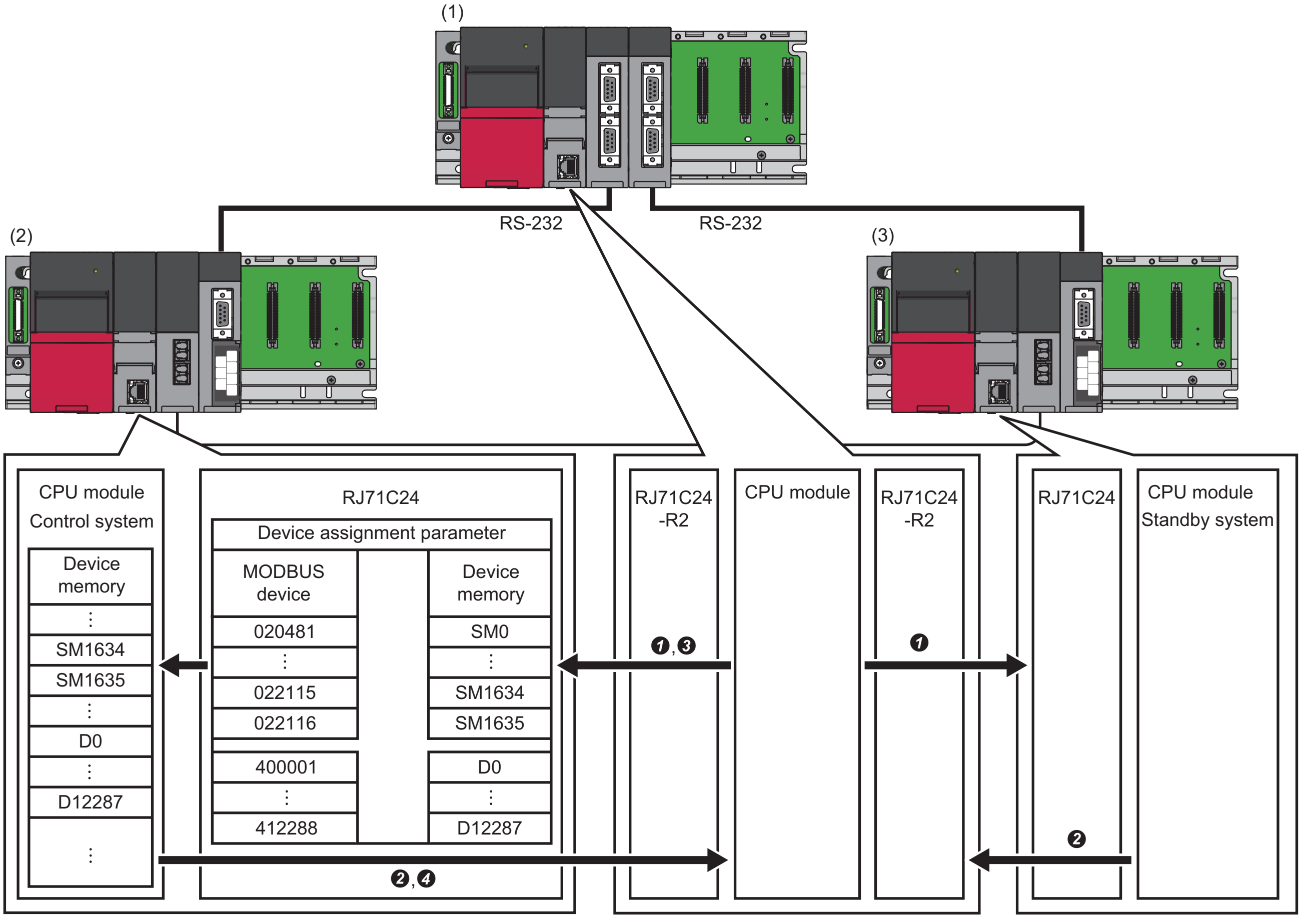
Communication examples for a redundant slave station (redundant line)

The following shows program examples to continue communications with the master station by switching a slave station (standby system) to the control system if an error occurs in a slave station (control system).

Overview

The following shows the operation overviews of a sample program used for a communication example.

■Normal operation



(1) Master station 1 and master station 2: RJ71C24-R2 (predefined protocol)

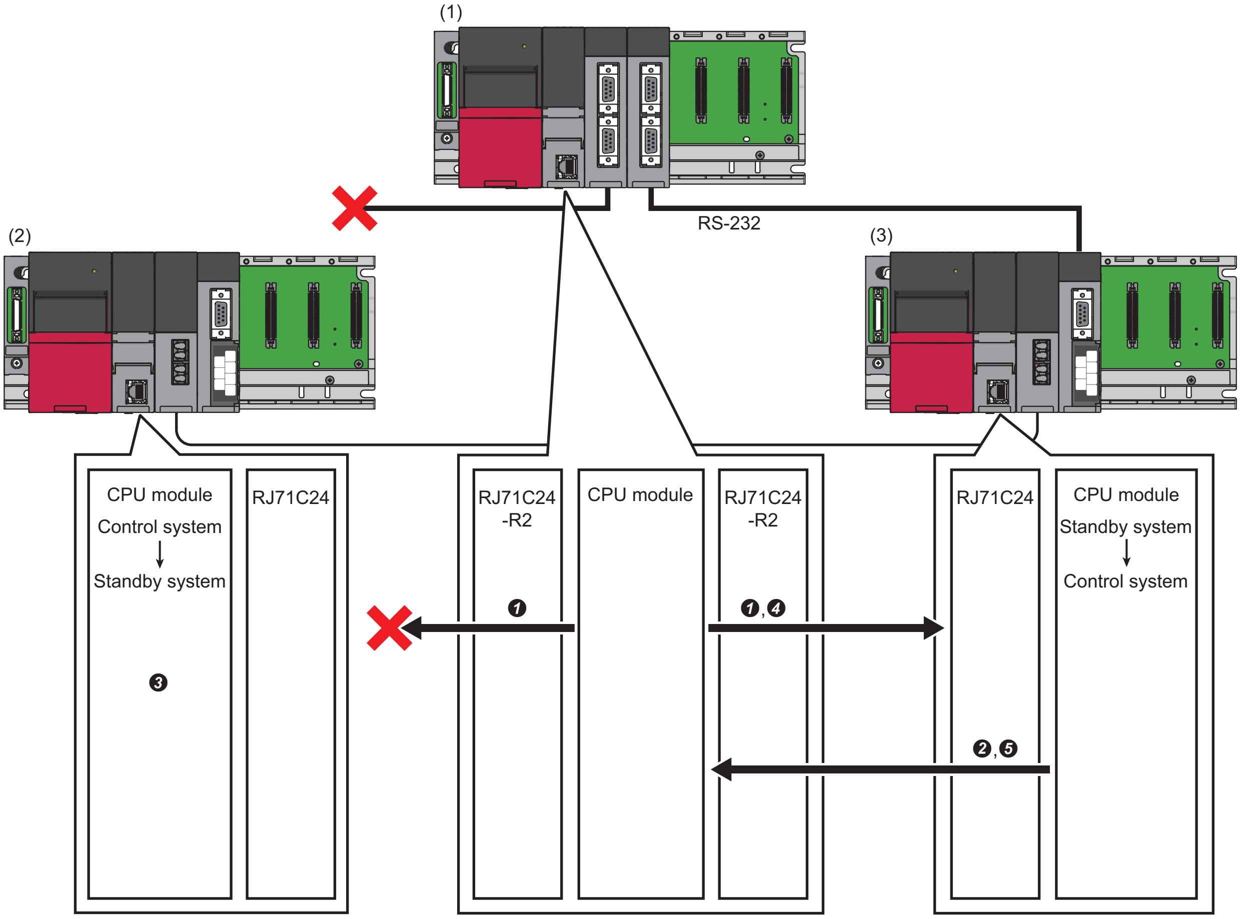
(2) Slave station (control system): RJ71C24 (MODBUS slave (RTU))

(3) Slave station (standby system): RJ71C24 (MODBUS slave (RTU))

No.	Description
❶	The master station sends the following request messages to the slave stations (control system and standby system) with the GP.CPRTCL instruction. • Reading two points from the coil 022115 with 'Read coils (FC:01)' (reading SM1634 and SM1635 used to determine the control system) • Writing a value for alive check to the holding register 412288 with 'Write multiple registers (FC:16)'
❷	Each of the slave stations (control system and standby system) processes the request from the master station according to the MODBUS assignment parameter and sends a response message.
❸	The master station determines the value acquired with 'Read coils' and sends a request message used for actual control to the slave station (control system).
❹	The slave station (control system) processes the request from the master station according to the MODBUS assignment parameter, and sends a response message.

The above operations are repeated every three seconds.

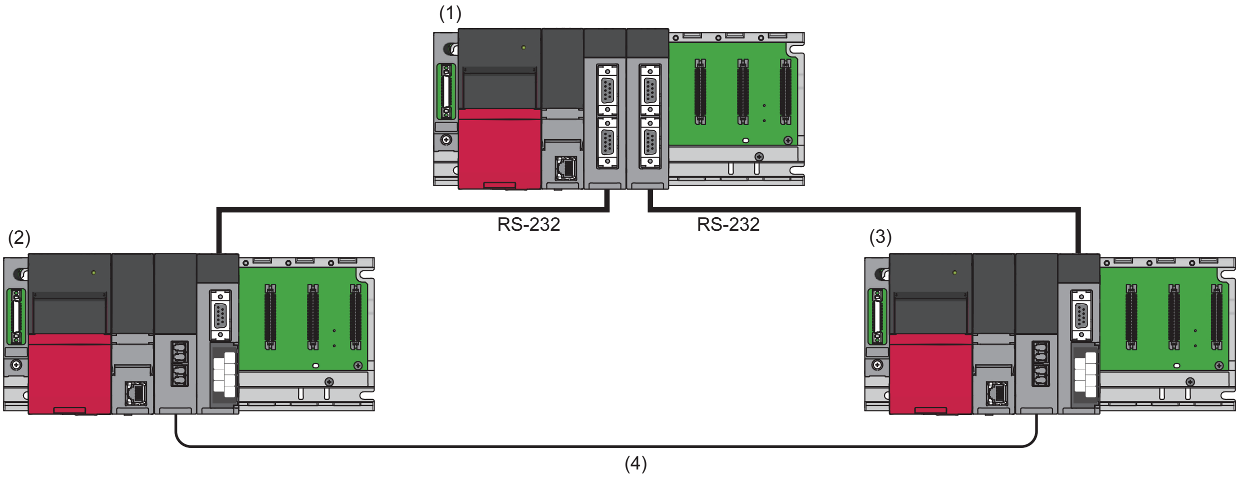
■When an error occurs



(1) Master station 1 and master station 2: RJ71C24-R2 (predefined protocol)
(2) Slave station (control system): RJ71C24 (MODBUS slave (RTU))
(3) Slave station (standby system): RJ71C24 (MODBUS slave (RTU))

No.	Description
❶	The master station sends a request message to each of the slave stations (control system and standby system) with the GP.CPRTCL instruction.
❷	The slave station (control system) cannot receive the request message, and only the slave station (standby system) sends a response message (data for alive check).
❸	The slave station (control system) switches the system with the SP.CONTSW instruction because data for alive check was not updated after 10 seconds.
❹	The master station determines the value acquired with 'Read coils' and sends a request message used for actual control to the slave station (standby system) that has switched to the new control system.
❺	The slave station (standby system) that has switched to the new control system processes the request from the master station according to the MODBUS assignment parameter, and sends a response message.

System configuration



(1) Master station 1 and master station 2: RJ71C24-R2 (predefined protocol)
(2) Slave station (control system): RJ71C24 (MODBUS slave (RTU))
(3) Slave station (standby system): RJ71C24 (MODBUS slave (RTU))
(4) Tracking cable

Type	Model	Type	Start I/O No.
Master station	RCPU	CPU module	3E00H
	RJ71C24-R2[*1]	Serial communication module	0000H
	RJ71C24-R2[*2]		0020H
Slave station	RnPCPU	Process CPU	3E00H
	R6RFM	Redundant function module	0000H
	RJ71C24	Serial communication module	0020H

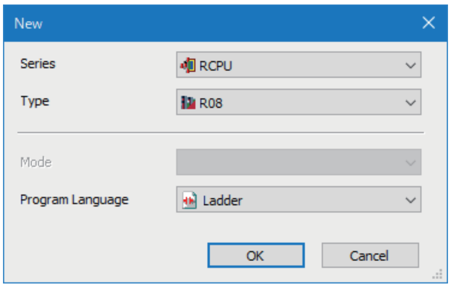
*1 Master station 1
*2 Master station 2

Setting for a master station

The following shows the procedure for connecting to the CPU module on a master station and setting parameters in an engineering tool.

1. Set the CPU module as follows:

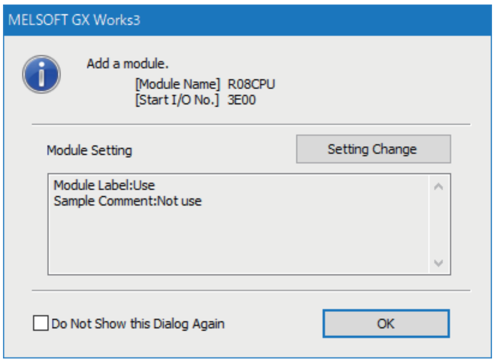
[Project] ⇨ [New]



Series: RCPU
Type: R08
Mode:
Program Language: Ladder

2. Click the [Setting Change] button to set to use module labels.

3. Click the [OK] button to add module labels of the CPU module.



MELSOFT GX Works3

Add a module.
[Module Name] R08CPU
[Start I/O No.] 3E00

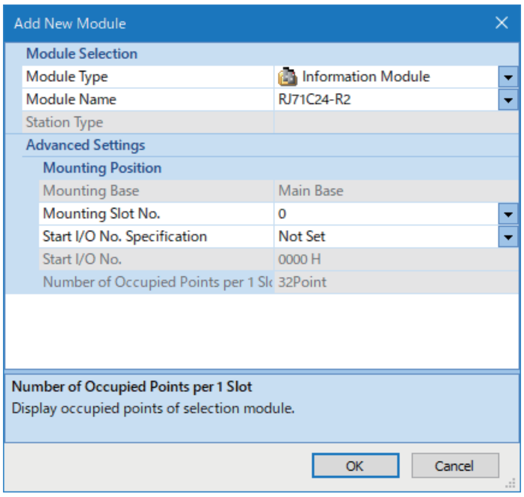
Module Setting

Module Label:Use
Sample Comment:Not use

□ Do Not Show this Dialog Again

4. Set the RJ71C24-R2 (master station 1) as follows:

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ right-click ⇨ [Add New Module]

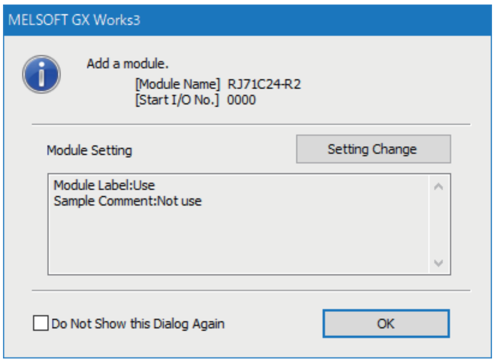


Add New Module

Module Selection	
Module Type	Information Module
Module Name	RJ71C24-R2
Station Type	
Advanced Settings	
Mounting Position	
Mounting Base	Main Base
Mounting Slot No.	0
Start I/O No. Specification	Not Set
Start I/O No.	0000 H
Number of Occupied Points per 1 Slc	32Point

Number of Occupied Points per 1 Slot
Display occupied points of selection module.

5. Click the [OK] button to add module labels of the RJ71C24-R2 (master station 1).



MELSOFT GX Works3

Add a module.
[Module Name] RJ71C24-R2
[Start I/O No.] 0000

Module Setting

Module Label:Use
Sample Comment:Not use

□ Do Not Show this Dialog Again

Set the RJ71C24-R2 (master station 2) with the same procedure.

6. Set the RS-232 (CH1) of master station 1 and master station 2 as follows:

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24-R2] ⇨ [Module Parameter] ⇨ [Basic Settings] ⇨ [Various control specification]

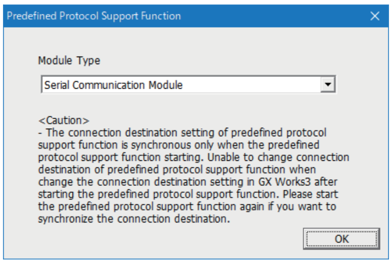
Item			Description
Various control specification	Communication protocol setting		Predefined protocol
	Communication speed setting		230400 bps
	Transmission setting	Operation setting	Independent
		Data bit	8
		Parity bit	None
		Odd/even parity	—
		Stop bit	1
		Sum check code	None
		Online change	Enable
		Setting change	Disable
	Station number setting		0
	MODBUS station number setting		—
	Transmission control setting	Transmission control	DC code control
		DC1/DC3 control	Control disabled
		DC2/DC4 control	Control disabled

7. Write the set parameters to the CPU module that controls the master stations.

[Online] ⇨ [Write to PLC]

8. Select [Tool] ⇨ [Predefined Protocol Support Function] in an engineering tool.

9. Select "Module Type" ⇨ "Serial Communication Module."



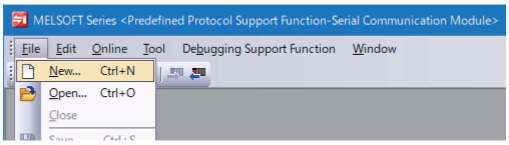
Predefined Protocol Support Function

Module Type

Serial Communication Module

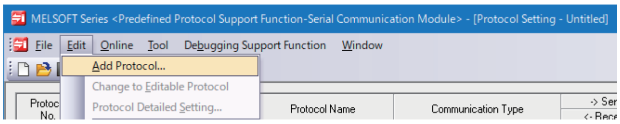
<Caution>
- The connection destination setting of predefined protocol support function is synchronous only when the predefined protocol support function starting. Unable to change connection destination of predefined protocol support function when change the connection destination setting in GX Works3 after starting the predefined protocol support function. Please start the predefined protocol support function again if you want to synchronize the connection destination.

10. Select [File] ⇨ [New].



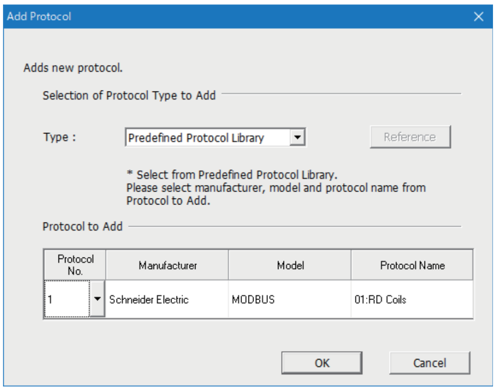
MELSOFT Series <Predefined Protocol Support Function-Serial Communication Module>
File Edit Online Tool Debugging Support Function Window
New... Ctrl+N
Open... Ctrl+O
Close

11. Select [Edit] ⇨ [Add Protocol].



MELSOFT Series <Predefined Protocol Support Function-Serial Communication Module> - [Protocol Setting - Untitled]
File Edit Online Tool Debugging Support Function Window
Add Protocol...
Change to Editable Protocol
Protocol Detailed Setting...
Protoc No.
Protocol Name
Communication Type
-> Ser
<- Rece

12. Select "Predefined Protocol Library" for "Type," and select a protocol to be used in "Protocol to Add."



Add Protocol

Adds new protocol.

Selection of Protocol Type to Add

Type : Predefined Protocol Library

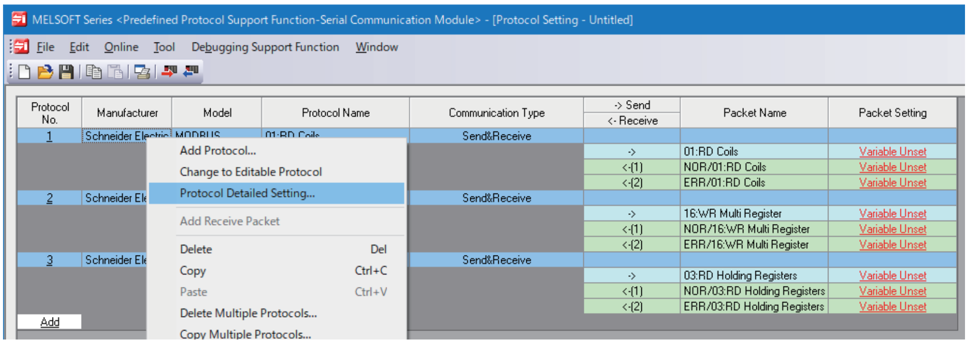
* Select from Predefined Protocol Library.
Please select manufacturer, model and protocol name from Protocol to Add.

Protocol to Add

Protocol No.	Manufacturer	Model	Protocol Name
1	Schneider Electric	MODBUS	01:RD Coils

Protocol No.	Manufacturer	Model	Protocol name	Purpose
1	Schneider Electric	MODBUS	01: RD Coils	Determination of the control system
2			16: WR Multi Register	Alive check
3			03: RD Holding Registers	Request data for control

13. Right-click the added protocol, and select [Protocol Detailed Setting] from the shortcut menu.



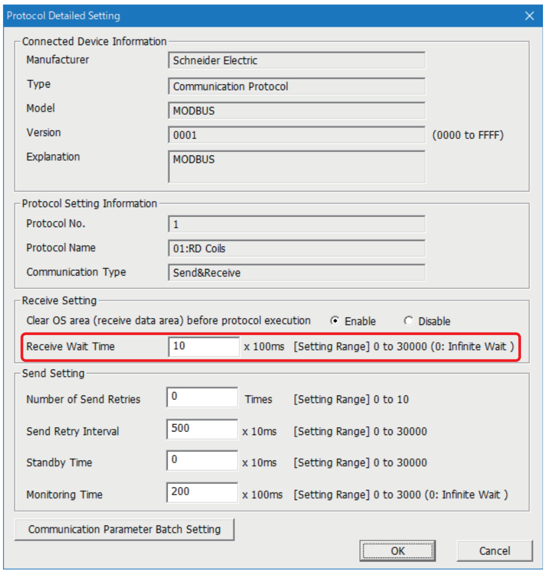
MELSOFT Series <Predefined Protocol Support Function-Serial Communication Module> - [Protocol Setting - Untitled]

Protocol No.	Manufacturer	Model	Protocol Name	Communication Type	-> Send <- Receive	Packet Name	Packet Setting
1	Schneider Electric	MODBUS	01:RD Coils	Send&Receive			
					->	01:RD Coils	Variable Unset
					<-(1)	NOR/01:RD Coils	Variable Unset
					<-(2)	ERR/01:RD Coils	Variable Unset
2	Schneider El			Send&Receive			
					->	16:WR Multi Register	Variable Unset
					<-(1)	NOR/16:WR Multi Register	Variable Unset
					<-(2)	ERR/16:WR Multi Register	Variable Unset
3	Schneider El			Send&Receive			
					->	03:RD Holding Registers	Variable Unset
					<-(1)	NOR/03:RD Holding Registers	Variable Unset
					<-(2)	ERR/03:RD Holding Registers	Variable Unset

File
Edit
Online
Tool
Debugging Support Function
Window
Add Protocol...
Change to Editable Protocol
Protocol Detailed Setting...
Add Receive Packet
Delete Del
Copy Ctrl+C
Paste Ctrl+V
Delete Multiple Protocols...
Copy Multiple Protocols...
Add

14. Set '10' for "Receive Wait Time."

Set the same item for protocol number 1 to 3.



Protocol Detailed Setting

Connected Device Information

- Manufacturer: Schneider Electric
- Type: Communication Protocol
- Model: MODBUS
- Version: 0001 (0000 to FFFF)
- Explanation: MODBUS

Protocol Setting Information

- Protocol No.: 1
- Protocol Name: 01:RD Coils
- Communication Type: Send&Receive

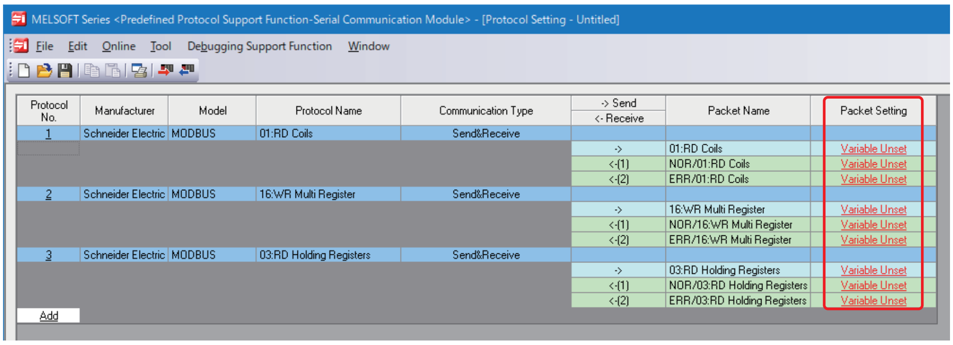
Receive Setting

- Clear OS area (receive data area) before protocol execution: ● Enable ○ Disable
- Receive Wait Time: 10 x 100ms [Setting Range] 0 to 30000 (0: Infinite Wait)

Send Setting

- Number of Send Retries: 0 Times [Setting Range] 0 to 10
- Send Retry Interval: 500 x 10ms [Setting Range] 0 to 30000
- Standby Time: 0 x 10ms [Setting Range] 0 to 30000
- Monitoring Time: 200 x 100ms [Setting Range] 0 to 3000 (0: Infinite Wait)

15. Select any cell of the added protocols in "Packet Setting."


MELSOFT Series <Predefined Protocol Support Function-Serial Communication Module> - [Protocol Setting - Untitled]
File
Edit
Online
Tool
Debugging Support Function
Window
Add

Protocol No.	Manufacturer	Model	Protocol Name	Communication Type	-> Send <- Receive	Packet Name	Packet Setting
1	Schneider Electric	MODBUS	01:RD Coils	Send&Receive			
					->	01:RD Coils	Variable Unset
					<-(1)	NOR/01:RD Coils	Variable Unset
					<-(2)	ERR/01:RD Coils	Variable Unset
2	Schneider Electric	MODBUS	16:WR Multi Register	Send&Receive			
					->	16:WR Multi Register	Variable Unset
					<-(1)	NOR/16:WR Multi Register	Variable Unset
					<-(2)	ERR/16:WR Multi Register	Variable Unset
3	Schneider Electric	MODBUS	03:RD Holding Registers	Send&Receive			
					->	03:RD Holding Registers	Variable Unset
					<-(1)	NOR/03:RD Holding Registers	Variable Unset
					<-(2)	ERR/03:RD Holding Registers	Variable Unset

Assign the following CPU devices to each protocol.

Protocol No.	Protocol name	Packet name	Element name	Device designation	
				Master station 1	**Master station 2**
1	01: RD Coils	01: RD Coils	Slave Address	[W0 - W0]	[W200 - W200]
			Starting Address	[W1 - W1]	[W201 - W201]
			Quantity of Coils	[W2 - W2]	[W202 - W202]
		NOR/01: RD Coils	Slave Address	[W3 - W3]	[W203 - W203]
			Coil Status	[W4], [W5 - W81]	[W204], [W205 - W281]
		ERR/01: RD Coils	Slave Address	[W82 - W82]	[W282 - W282]
			Exception Code	[W83 - W83]	[W283 - W283]
2	16: WR Multi Register	16: WR Multi Register	Slave Address	[W84 - W84]	[W284 - W284]
			Starting Address	[W85 - W85]	[W285 - W285]
			Quantity of Registers	[W86 - W86]	[W286 - W286]
			Registers Value	[W87], [W88 - W104]	[W287], [W288 - W304]
		NOR/16: WR Multi Register	Slave Address	[W105 - W105]	[W305 - W305]
			Starting Address	[W106 - W106]	[W306 - W306]
			Quantity of Registers	[W107 - W107]	[W307 - W307]
		ERR/16: WR Multi Register	Slave Address	[W108 - W108]	[W308 - W308]
			Exception Code	[W109 - W109]	[W309 - W309]
3	03: RD Holding Registers	03: RD Holding Registers	Slave Address	[W10A - W10A]	
			Starting Address	[W10B - W10B]	
			Quantity of Registers	[W10C - W10C]	
		NOR/03: RD Holding Registers	Slave Address	[W10D - W10D]	
			Register Value	[W10E], [W10F - W18B]	
		ERR/03: RD Holding Registers	Slave Address	[W18C - W18C]	
			Exception Code	[W18D - W18D]	

16. Write the set protocol setting data to the module.

[Online] ⇨ [Write to Module]

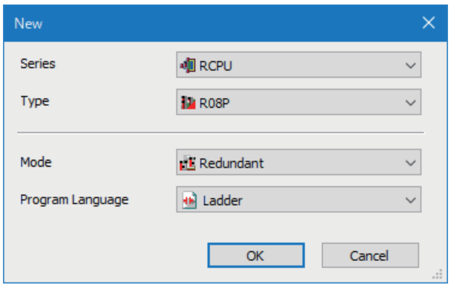
17. Reset the CPU module or turn the power OFF and ON.

Setting for a slave station (control system)

The following shows the procedure for connecting to the CPU module on a slave station and setting parameters in an engineering tool.

1. Set the CPU module as follows:

[Project] ⇨ [New]

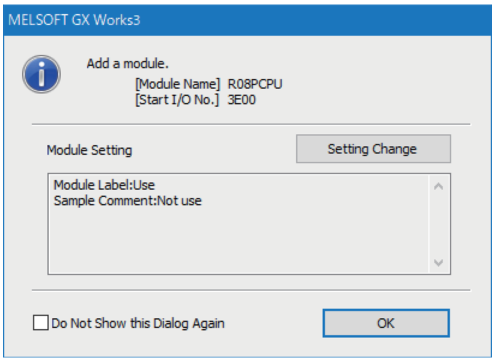


New

Series	RCPU
Type	R08P
Mode	Redundant
Program Language	Ladder

2. Click the [Setting Change] button to set to use module labels.

3. Click the [OK] button to add module labels of the CPU module.



MELSOFT GX Works3

Add a module.
[Module Name] R08PCPU
[Start I/O No.] 3E00

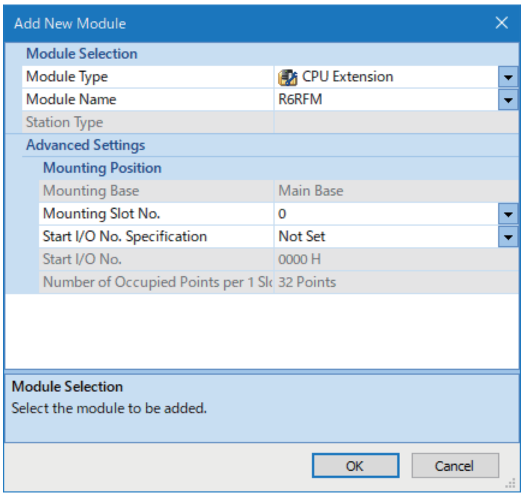
Module Setting

Module Label:Use
Sample Comment:Not use

□ Do Not Show this Dialog Again

4. Set the R6RFM as follows:

[Navigation window] ⇨ [Parameter] ⇨ [System Parameter] ⇨ [I/O Assignment] tab ⇨ [I/O Assignment Setting]



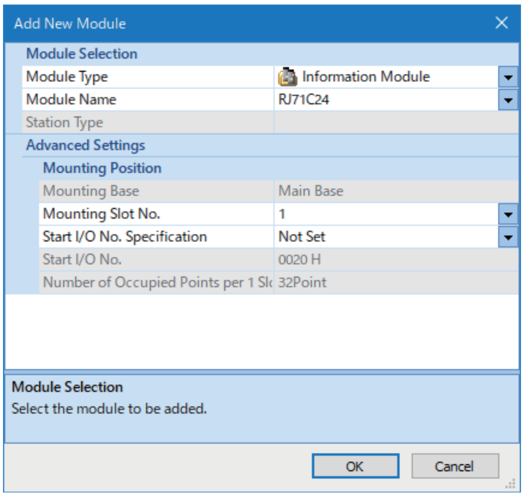
Add New Module

Module Selection	
Module Type	CPU Extension
Module Name	R6RFM
Station Type	
Advanced Settings	
Mounting Position	
Mounting Base	Main Base
Mounting Slot No.	0
Start I/O No. Specification	Not Set
Start I/O No.	0000 H
Number of Occupied Points per 1 Slc	32 Points

Module Selection
Select the module to be added.

5. Set the RJ71C24 as follows:

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ right-click ⇨ [Add New Module]

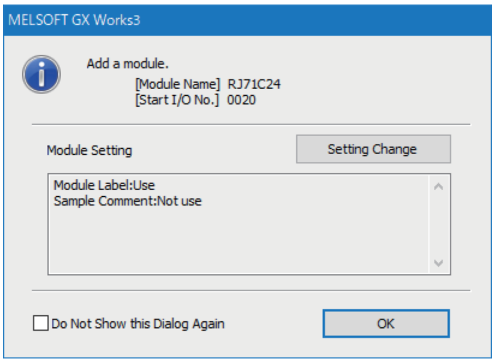


Add New Module

Module Selection	
Module Type	Information Module
Module Name	RJ71C24
Station Type	
Advanced Settings	
Mounting Position	
Mounting Base	Main Base
Mounting Slot No.	1
Start I/O No. Specification	Not Set
Start I/O No.	0020 H
Number of Occupied Points per 1 Sl	32Point

Module Selection
Select the module to be added.

6. Click the [OK] button to add module labels of the RJ71C24.



MELSOFT GX Works3

Add a module.
[Module Name] RJ71C24
[Start I/O No.] 0020

Module Setting

Module Label:Use
Sample Comment:Not use

7. Set the RS-232 (CH1) of each of the slave stations (control system and standby system) as follows:

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter] ⇨ [Basic Settings] ⇨ [Various control specification]

Item			Description
Various control specification	Communication protocol setting		MODBUS slave station (RTU)
	Communication speed setting[*1]		230400 bps
	Transmission setting	Operation setting	Independent
		Data bit[*1]	8
		Parity bit[*1]	None
		Odd/even parity[*1]	—
		Stop bit[*1]	1
		Sum check code[*1]	None
		Online change	Enable
		Setting change	Disable
	Station number setting		0
	MODBUS station number setting		1

[*1] Set the same value as the master station.

8. Write the set parameters to the CPU module on the slave station.

[Online] ⇨ [Write to PLC]

9. Reset the CPU module or turn the power OFF and ON.

Setting for a slave station (standby system)

Set a slave station (standby system) with the same procedure as a slave station (control system).

☞ Page 649 Setting for a slave station (control system)

Program examples for master stations

Point

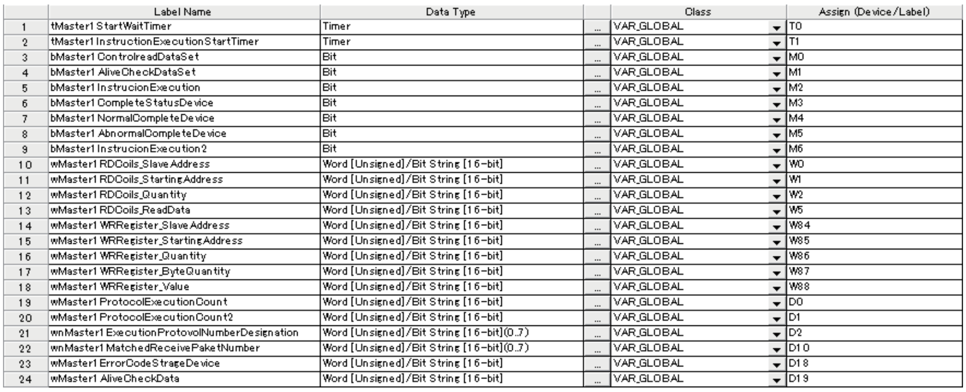
Create a sample program for a master station so that it behaves in the following order:

- Program for master station 1 → program for master station 2 → program for communication to a slave station of the control system

■Master station 1

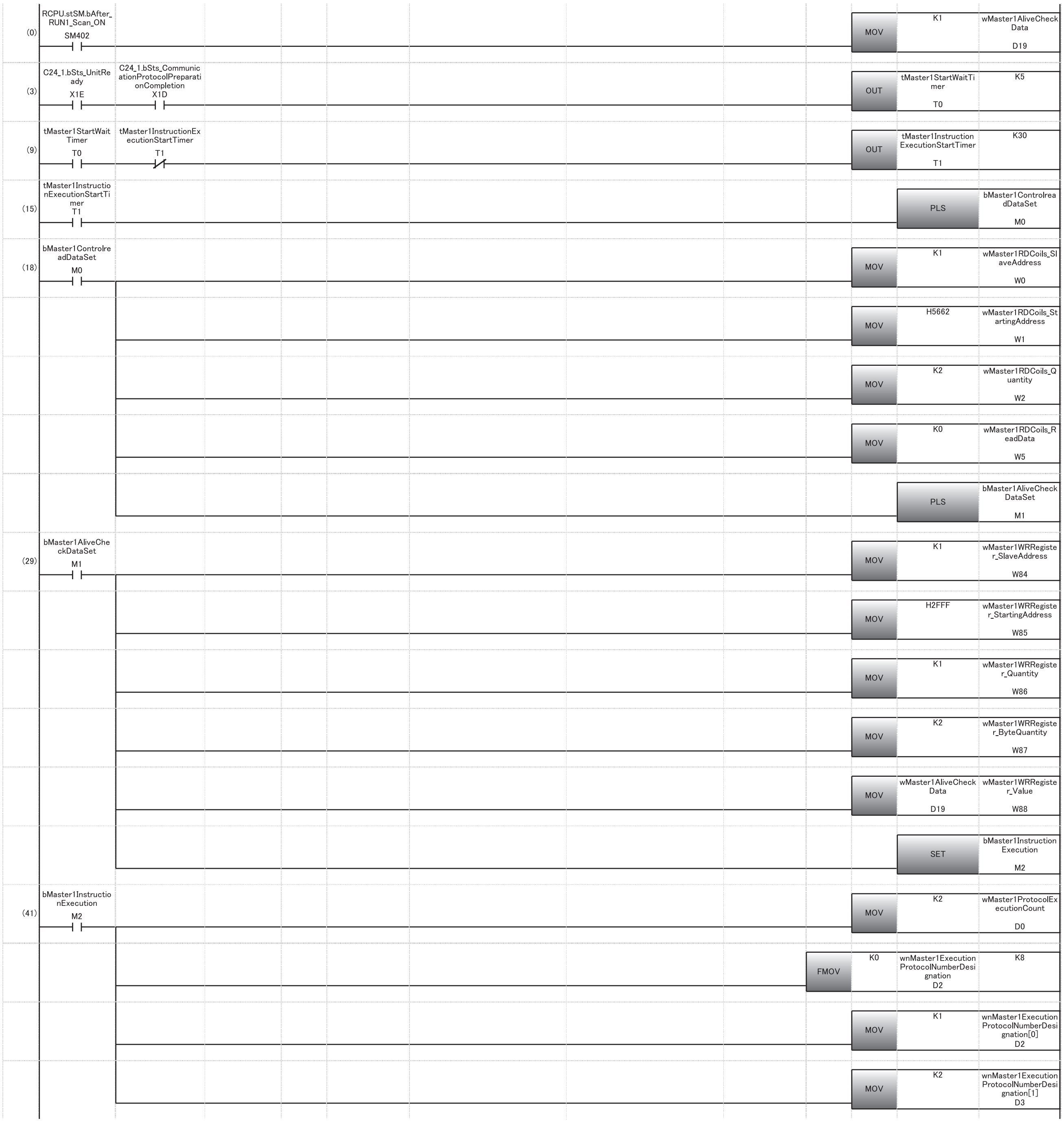
The following shows a program to control an RJ71C24-R2 the start I/O number of which is 0H.

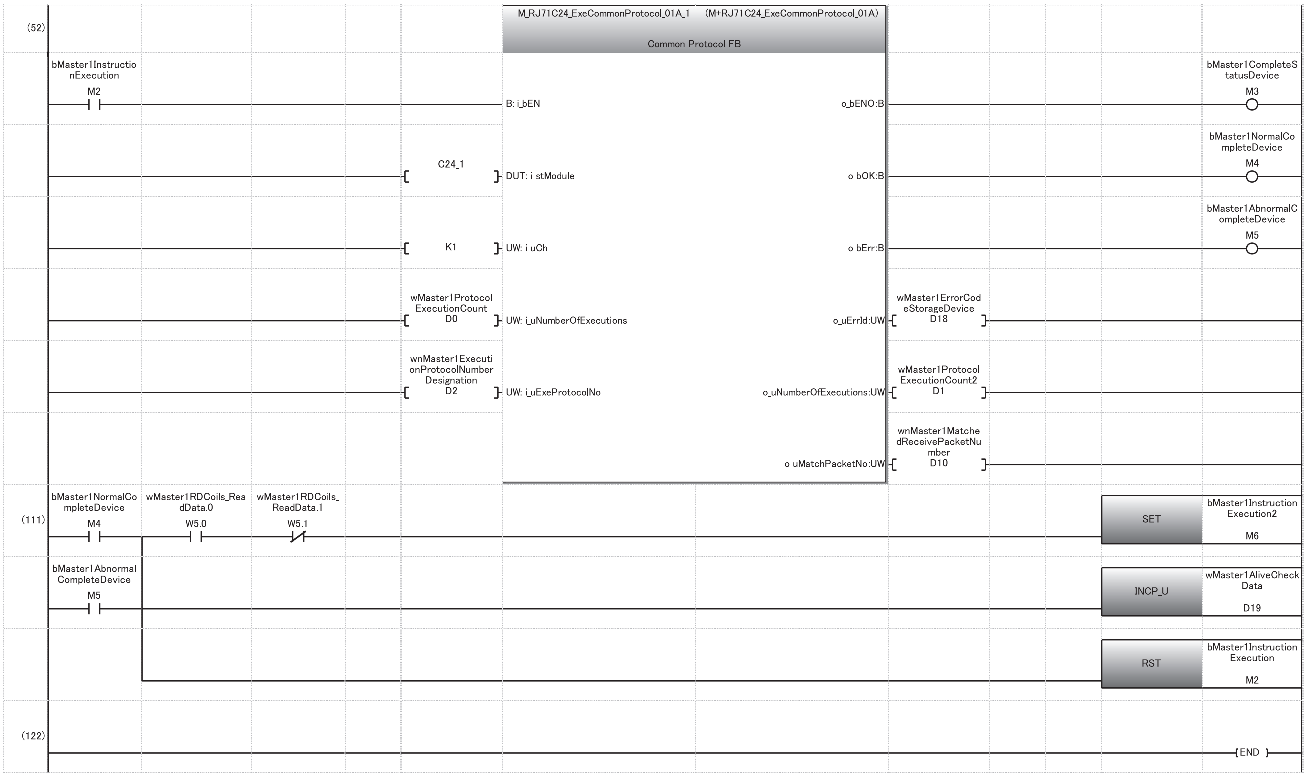
The control system can be determined and data for alive check can be updated.

<table>
<tr><th>Category</th><th>Label name/FB name</th><th>Description</th><th>Device</th></tr>
<tr><td>Module label</td><td>C24_1</td><td>Target module</td><td>—</td></tr>
<tr><td></td><td>C24_1.bSts_UnitReady</td><td>Module READY</td><td>X1E</td></tr>
<tr><td></td><td>C24_1.bSts_CommunicationProtocolPreparationCompletion</td><td>Predefined protocol ready</td><td>X1D</td></tr>
<tr><td></td><td>RCPU.stSM.bAfter_RUN1_Scan_ON</td><td>ON for one scan after RUN</td><td>SM402</td></tr>
<tr><td>Label to be defined</td><td colspan="3">Define the global label as shown below.

<table>
<tr><th></th><th>Label Name</th><th>Data Type</th><th></th><th>Class</th><th>Assign (Device/Label)</th></tr>
<tr><td>1</td><td>tMaster1 StartWaitTimer</td><td>Timer</td><td>...</td><td>VAR_GLOBAL</td><td>T0</td></tr>
<tr><td>2</td><td>tMaster1 InstructionExecutionStartTimer</td><td>Timer</td><td>...</td><td>VAR_GLOBAL</td><td>T1</td></tr>
<tr><td>3</td><td>bMaster1 ControlreadDataSet</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M0</td></tr>
<tr><td>4</td><td>bMaster1 AliveCheckDataSet</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M1</td></tr>
<tr><td>5</td><td>bMaster1 InstrucionExecution</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M2</td></tr>
<tr><td>6</td><td>bMaster1 CompleteStatusDevice</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M3</td></tr>
<tr><td>7</td><td>bMaster1 NormalCompleteDevice</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M4</td></tr>
<tr><td>8</td><td>bMaster1 AbnormalCompleteDevice</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M5</td></tr>
<tr><td>9</td><td>bMaster1 InstrucionExecution2</td><td>Bit</td><td>...</td><td>VAR_GLOBAL</td><td>M6</td></tr>
<tr><td>10</td><td>wMaster1 RDCoils_SlaveAddress</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>W0</td></tr>
<tr><td>11</td><td>wMaster1 RDCoils_StartingAddress</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>W1</td></tr>
<tr><td>12</td><td>wMaster1 RDCoils_Quantity</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>W2</td></tr>
<tr><td>13</td><td>wMaster1 RDCoils_ReadData</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>W5</td></tr>
<tr><td>14</td><td>wMaster1 WRRegister_SlaveAddress</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>W84</td></tr>
<tr><td>15</td><td>wMaster1 WRRegister_StartingAddress</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>W85</td></tr>
<tr><td>16</td><td>wMaster1 WRRegister_Quantity</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>W86</td></tr>
<tr><td>17</td><td>wMaster1 WRRegister_ByteQuantity</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>W87</td></tr>
<tr><td>18</td><td>wMaster1 WRRegister_Value</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>W88</td></tr>
<tr><td>19</td><td>wMaster1 ProtocolExecutionCount</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D0</td></tr>
<tr><td>20</td><td>wMaster1 ProtocolExecutionCount2</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D1</td></tr>
<tr><td>21</td><td>wnMaster1 ExecutionProtovolNumberDesignation</td><td>Word [Unsigned]/Bit String [16-bit](0..7)</td><td>...</td><td>VAR_GLOBAL</td><td>D2</td></tr>
<tr><td>22</td><td>wnMaster1 MatchedReceivePaketNumber</td><td>Word [Unsigned]/Bit String [16-bit](0..7)</td><td>...</td><td>VAR_GLOBAL</td><td>D10</td></tr>
<tr><td>23</td><td>wMaster1 ErrorCodeStrageDevice</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D18</td></tr>
<tr><td>24</td><td>wMaster1 AliveCheckData</td><td>Word [Unsigned]/Bit String [16-bit]</td><td>...</td><td>VAR_GLOBAL</td><td>D19</td></tr>
</table>
</td></tr>
<tr><td>FB</td><td>M+RJ71C24_ExeCommonProtocol</td><td>Executes the protocol registered with GX Works3.</td><td>—</td></tr>
</table>

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
RCPU.stSM.bAfter_RUN1_Scan_ON
SM402
MOV
K1
wMaster1AliveCheckData
D19
(3)
C24_1.bSts_UnitReady
X1E
C24_1.bSts_CommunicationProtocolPreparationCompletion
X1D
OUT
tMaster1StartWaitTimer
T0
K5
(9)
tMaster1StartWaitTimer
T0
tMaster1InstructionExecutionStartTimer
T1
OUT
tMaster1InstructionExecutionStartTimer
T1
K30
(15)
tMaster1InstructionExecutionStartTimer
T1
PLS
bMaster1ControlreadDataSet
M0
(18)
bMaster1ControlreadDataSet
M0
MOV
K1
wMaster1RDCoils_SlaveAddress
W0
MOV
H5662
wMaster1RDCoils_StartingAddress
W1
MOV
K2
wMaster1RDCoils_Quantity
W2
MOV
K0
wMaster1RDCoils_ReadData
W5
PLS
bMaster1AliveCheckDataSet
M1
(29)
bMaster1AliveCheckDataSet
M1
MOV
K1
wMaster1WRRegister_SlaveAddress
W84
MOV
H2FFF
wMaster1WRRegister_StartingAddress
W85
MOV
K1
wMaster1WRRegister_Quantity
W86
MOV
K2
wMaster1WRRegister_ByteQuantity
W87
MOV
wMaster1AliveCheckData
D19
wMaster1WRRegister_Value
W88
SET
bMaster1InstructionExecution
M2
(41)
bMaster1InstructionExecution
M2
MOV
K2
wMaster1ProtocolExecutionCount
D0
FMOV
K0
wnMaster1ExecutionProtocolNumberDesignation
D2
K8
MOV
K1
wnMaster1ExecutionProtocolNumberDesignation[0]
D2
MOV
K2
wnMaster1ExecutionProtocolNumberDesignation[1]
D3


(52)
M_RJ71C24_ExeCommonProtocol_01A_1
(M+RJ71C24_ExeCommonProtocol_01A)
Common Protocol FB
bMaster1InstructionExecution
M2
B: i_bEN
o_bENO:B
bMaster1CompleteStatusDevice
M3
C24_1
DUT: i_stModule
o_bOK:B
bMaster1NormalCompleteDevice
M4
K1
UW: i_uCh
o_bErr:B
bMaster1AbnormalCompleteDevice
M5
wMaster1ProtocolExecutionCount
D0
UW: i_uNumberOfExecutions
o_uErrId:UW
wMaster1ErrorCodeStorageDevice
D18
wnMaster1ExecutionProtocolNumberDesignation
D2
UW: i_uExeProtocolNo
o_uNumberOfExecutions:UW
wMaster1ProtocolExecutionCount2
D1
o_uMatchPacketNo:UW
wnMaster1MatchedReceivePacketNumber
D10
(111)
bMaster1NormalCompleteDevice
M4
wMaster1RDCoils_ReadData.0
W5.0
wMaster1RDCoils_ReadData.1
W5.1
SET
bMaster1InstructionExecution2
M6
bMaster1AbnormalCompleteDevice
M5
INCP_U
wMaster1AliveCheckData
D19
RST
bMaster1InstructionExecution
M2
(122)
END

(0) Sets an initial value for data for alive check.

(3) Waits for a slave station to start (500 ms).

(9) Reads special relays (SM1634 and SM1635), sends data for alive check, and performs communication with a slave station every three seconds.

(15) Reads special relays (SM1634 and SM1635) and turns a send execution trigger for data for alive check ON.

(18) Sets predefined protocol data to read special relays (SM1634 and SM1635).

- Sets '1' for the request destination slave station number.
- Sets the start coil number and the number of read points so that MODBUS devices (022115 and 022116) will be read.

(29) Sets predefined protocol data to send data for alive check.

- Sets '1' for the request destination slave station number.
- Sets the start register number, the number of write points, and the write size so that MODBUS device (412288) will be written.
- Sets data for alive check as writing data.

(41) Sets a value for the input label of M+RJ71C24_ExeCommonProtocol.

- Sets '2' for the number of consecutive protocol executions.
- Sets '1' and '2' for the execution protocol number specification.

(52) Executes M+RJ71C24_ExeCommonProtocol.

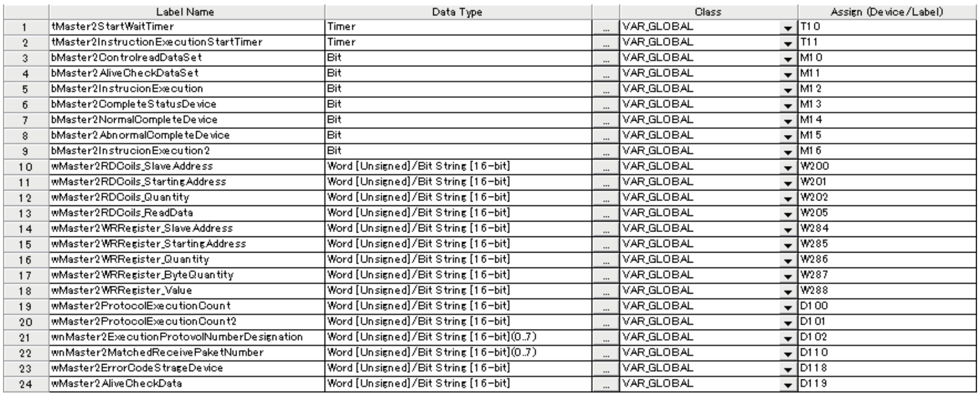
(111) Updates data for alive check after completion of M+RJ71C24_ExeCommonProtocol.

- Performs communication with a slave station if the control system is connected. (☞ Page 658 Communication to a slave station of the control system)

■Master station 2

The following shows a program to control an RJ71C24-R2 the start I/O number of which is 20H.

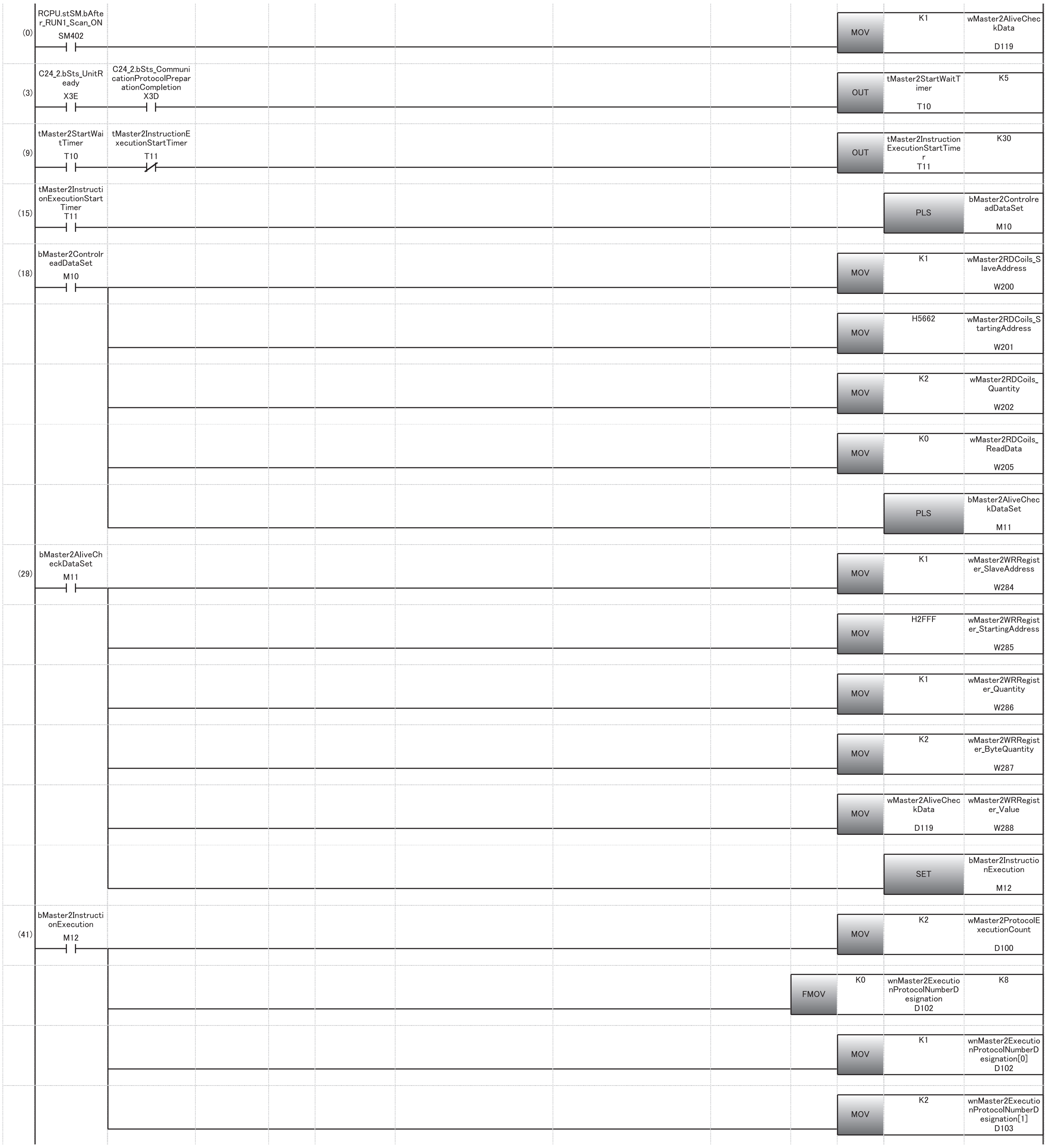
The control system can be determined and data for alive check can be updated.

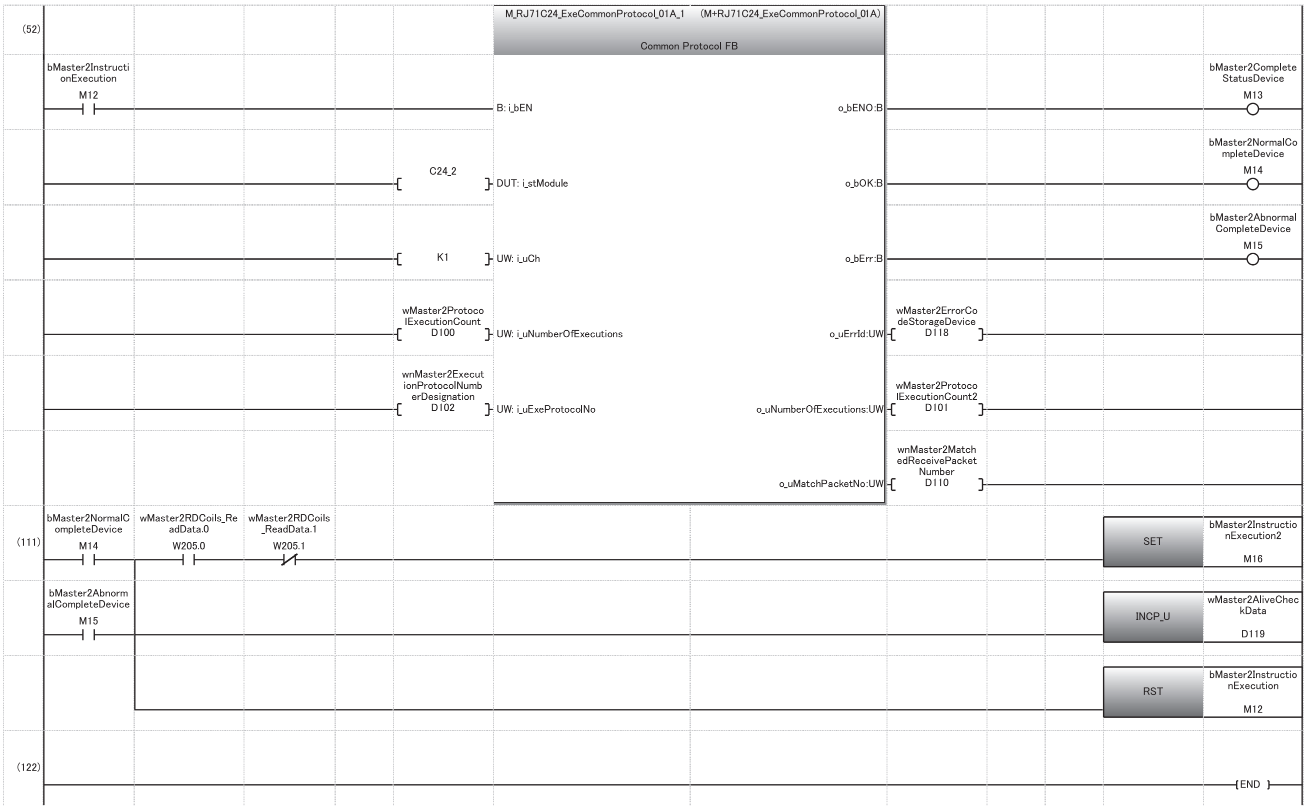
Category	Label name/FB name	Description	Device
Module label	C24_2	Target module	—
	C24_2.bSts_UnitReady	Module READY	X3E
	C24_2.bSts_CommunicationProtocolPreparationCompletion	Predefined protocol ready	X3D
	RCPU.stSM.bAfter_RUN1_Scan_ON	ON for one scan after RUN	SM402
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_ExeCommonProtocol	Executes the protocol registered with GX Works3.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	tMaster2StartWaitTimer	Timer	...	VAR_GLOBAL	T10
2	tMaster2InstructionExecutionStartTimer	Timer	...	VAR_GLOBAL	T11
3	bMaster2ControlreadDataSet	Bit	...	VAR_GLOBAL	M10
4	bMaster2AliveCheckDataSet	Bit	...	VAR_GLOBAL	M11
5	bMaster2InstrucionExecution	Bit	...	VAR_GLOBAL	M12
6	bMaster2CompleteStatusDevice	Bit	...	VAR_GLOBAL	M13
7	bMaster2NormalCompleteDevice	Bit	...	VAR_GLOBAL	M14
8	bMaster2AbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M15
9	bMaster2InstrucionExecution2	Bit	...	VAR_GLOBAL	M16
10	wMaster2RDCoils_SlaveAddress	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W200
11	wMaster2RDCoils_StartingAddress	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W201
12	wMaster2RDCoils_Quantity	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W202
13	wMaster2RDCoils_ReadData	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W205
14	wMaster2WRRegister_SlaveAddress	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W284
15	wMaster2WRRegister_StartingAddress	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W285
16	wMaster2WRRegister_Quantity	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W286
17	wMaster2WRRegister_ByteQuantity	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W287
18	wMaster2WRRegister_Value	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W288
19	wMaster2ProtocolExecutionCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D100
20	wMaster2ProtocolExecutionCount2	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D101
21	wnMaster2ExecutionProtovolNumberDesignation	Word [Unsigned]/Bit String [16-bit](0..7)	...	VAR_GLOBAL	D102
22	wnMaster2MatchedReceivePaketNumber	Word [Unsigned]/Bit String [16-bit](0..7)	...	VAR_GLOBAL	D110
23	wMaster2ErrorCodeStrageDevice	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D118
24	wMaster2AliveCheckData	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D119

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference


(0)
RCPU.stSM.bAfter_RUN1_Scan_ON
SM402
MOV
K1
wMaster2AliveCheckData
D119
(3)
C24_2.bSts_UnitReady
X3E
C24_2.bSts_CommunicationProtocolPreparationCompletion
X3D
OUT
tMaster2StartWaitTimer
T10
K5
(9)
tMaster2StartWaitTimer
T10
tMaster2InstructionExecutionStartTimer
T11
OUT
tMaster2InstructionExecutionStartTimer
T11
K30
(15)
tMaster2InstructionExecutionStartTimer
T11
PLS
bMaster2ControlreadDataSet
M10
(18)
bMaster2ControlreadDataSet
M10
MOV
K1
wMaster2RDCoils_SlaveAddress
W200
MOV
H5662
wMaster2RDCoils_StartingAddress
W201
MOV
K2
wMaster2RDCoils_Quantity
W202
MOV
K0
wMaster2RDCoils_ReadData
W205
PLS
bMaster2AliveCheckDataSet
M11
(29)
bMaster2AliveCheckDataSet
M11
MOV
K1
wMaster2WRRegister_SlaveAddress
W284
MOV
H2FFF
wMaster2WRRegister_StartingAddress
W285
MOV
K1
wMaster2WRRegister_Quantity
W286
MOV
K2
wMaster2WRRegister_ByteQuantity
W287
MOV
wMaster2AliveCheckData
D119
wMaster2WRRegister_Value
W288
SET
bMaster2InstructionExecution
M12
(41)
bMaster2InstructionExecution
M12
MOV
K2
wMaster2ProtocolExecutionCount
D100
FMOV
K0
wnMaster2ExecutionProtocolNumberDesignation
D102
K8
MOV
K1
wnMaster2ExecutionProtocolNumberDesignation[0]
D102
MOV
K2
wnMaster2ExecutionProtocolNumberDesignation[1]
D103


(52)
M_RJ71C24_ExeCommonProtocol_01A_1 (M+RJ71C24_ExeCommonProtocol_01A)
Common Protocol FB
bMaster2InstructionExecution
M12
B: i_bEN
o_bENO:B
bMaster2CompleteStatusDevice
M13
C24_2
DUT: i_stModule
o_bOK:B
bMaster2NormalCompleteDevice
M14
K1
UW: i_uCh
o_bErr:B
bMaster2AbnormalCompleteDevice
M15
wMaster2ProtocolExecutionCount
D100
UW: i_uNumberOfExecutions
o_uErrId:UW
wMaster2ErrorCodeStorageDevice
D118
wnMaster2ExecutionProtocolNumberDesignation
D102
UW: i_uExeProtocolNo
o_uNumberOfExecutions:UW
wMaster2ProtocolExecutionCount2
D101
o_uMatchPacketNo:UW
wnMaster2MatchedReceivePacketNumber
D110
(111)
bMaster2NormalCompleteDevice
M14
wMaster2RDCoils_ReadData.0
W205.0
wMaster2RDCoils_ReadData.1
W205.1
SET
bMaster2InstructionExecution2
M16
bMaster2AbnormalCompleteDevice
M15
INCP_U
wMaster2AliveCheckData
D119
RST
bMaster2InstructionExecution
M12
(122)
END

(0) Sets an initial value for data for alive check.

(3) Waits for a slave station to start (500 ms).

(9) Reads special relays (SM1634 and SM1635), sends data for alive check, and performs communication with a slave station every three seconds.

(15) Reads special relays (SM1634 and SM1635) and turns a send execution trigger for data for alive check ON.

(18) Sets predefined protocol data to read special relays (SM1634 and SM1635).
- Sets '1' for the request destination slave station number.
- Sets the coil number and the number of read points so that MODBUS devices (022115 and 022116) will be read.

(29) Sets predefined protocol data to send data for alive check.
- Sets '1' for the request destination slave station number.
- Sets the start register number, the number of write points, and the write size so that MODBUS device (412288) will be written.
- Sets data for alive check as writing data.

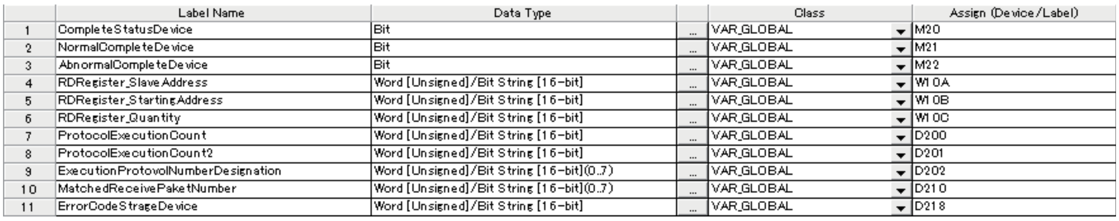
(41) Sets a value for the input label of M+RJ71C24_ExeCommonProtocol.
- Sets '2' for the number of consecutive protocol executions.
- Sets '1' and '2' for the execution protocol number specification.

(52) Executes M+RJ71C24_ExeCommonProtocol.

(111) Updates data for alive check after completion of M+RJ71C24_ExeCommonProtocol.
- Performs communication with a slave station if the control system is connected. (☞ Page 658 Communication to a slave station of the control system)

■Communication to a slave station of the control system

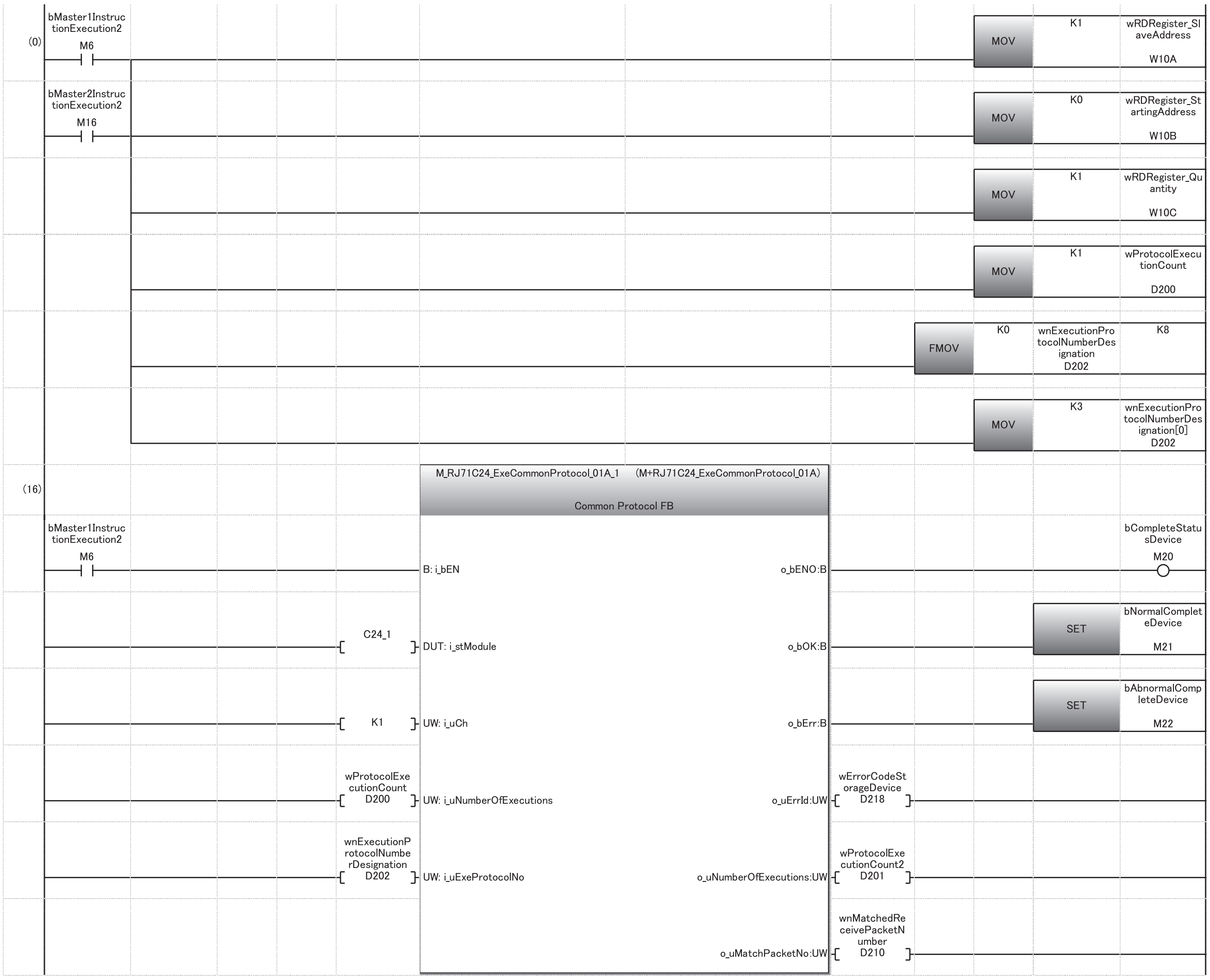
Communication is performed in a master station connected to a slave station of the control system.

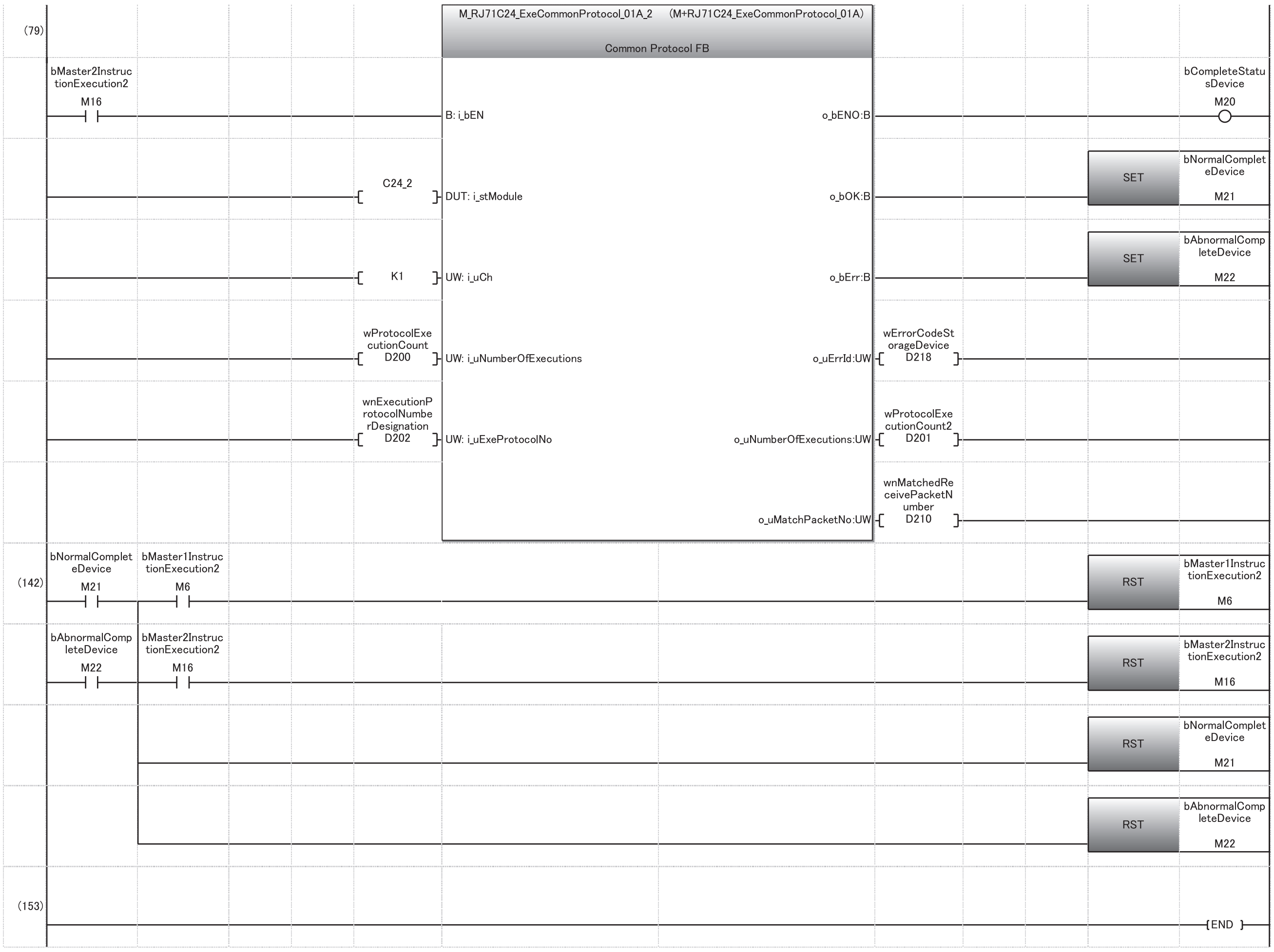
Category	Label name/FB name	Description	Device
Module label	C24_1	Target module	—
	C24_2		
Label to be defined	Define the global label as shown below. 		
FB	M+RJ71C24_ExeCommonProtocol	Executes the protocol registered with GX Works3.	—

	Label Name	Data Type		Class	Assign (Device/Label)
1	CompleteStatusDevice	Bit	...	VAR_GLOBAL	M20
2	NormalCompleteDevice	Bit	...	VAR_GLOBAL	M21
3	AbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M22
4	RDRegister_SlaveAddress	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W10A
5	RDRegister_StartingAddress	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W10B
6	RDRegister_Quantity	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	W10C
7	ProtocolExecutionCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D200
8	ProtocolExecutionCount2	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D201
9	ExecutionProtovolNumberDesignation	Word [Unsigned]/Bit String [16-bit](0..7)	...	VAR_GLOBAL	D202
10	MatchedReceivePaketNumber	Word [Unsigned]/Bit String [16-bit](0..7)	...	VAR_GLOBAL	D210
11	ErrorCodeStrageDevice	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D218

For information on the module FB to be used, refer to the following manual.

📖MELSEC iQ-R Serial Communication Module Function Block Reference





(0) Sets protocol data for communication to a slave station.

- Sets '1' for the request destination slave station number.
- Sets the start register number and the number of read points so that MODBUS device (400001) will be read.

Sets a value for the input label of M+RJ71C24_ExeCommonProtocol.

- Sets '1' for the number of consecutive protocol executions.
- Sets '3' for the execution protocol number specification.

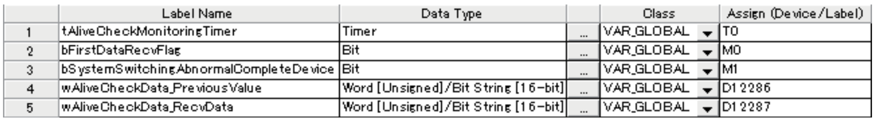
(16) Performs communication in master station 1.

(79) Performs communication in master station 2.

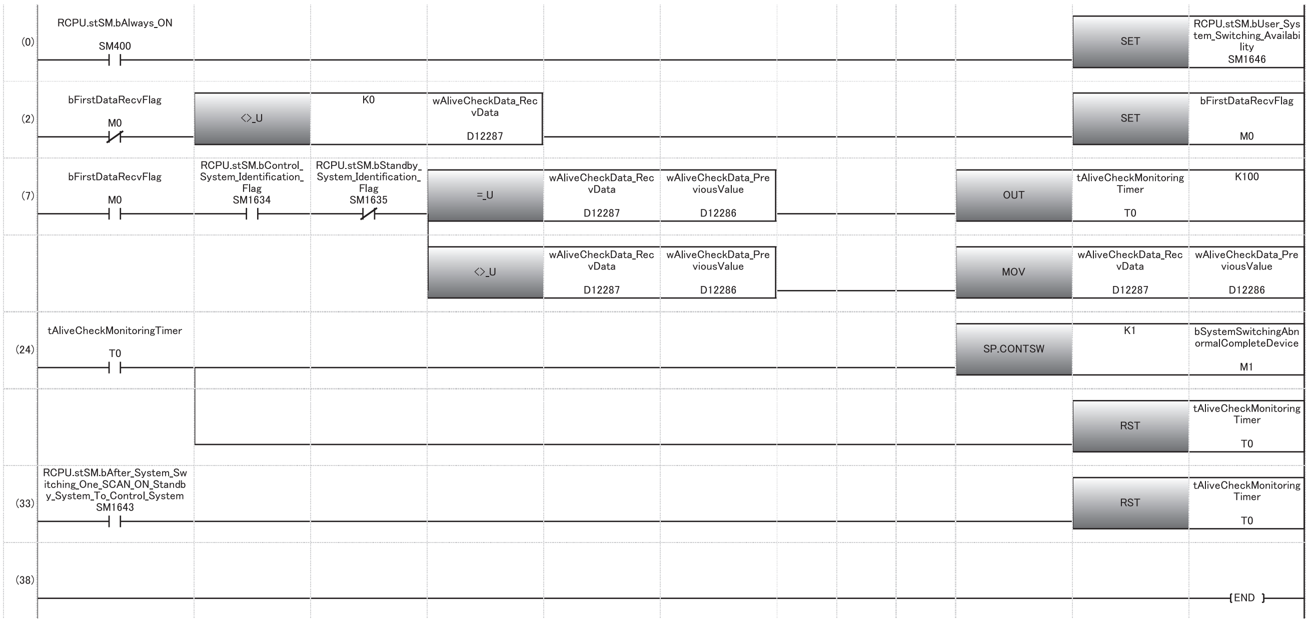
(142) Performs processing after communication is completed.

Program example for a slave station

The control system can be determined and data for alive check can be updated.

Category	Label name	Description	Device
Module label	RCPU.stSM.bAlways_ON	Always ON	SM400
	RCPU.stSM.bUser_System_Switching_Availability	System switching by a user	SM1646
	RCPU.stSM.bControl_System_Identification_Flag	Control system judgment flag	SM1634
	RCPU.stSM.bStandby_System_Identification_Flag	Standby system judgment flag	SM1635
	RCPU.stSM.bAfter_System_Switching_One_SCAN_ON_Standby_System_To_Control_System	ON for one scan after system switching (Standby system to control system)	SM1643
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	tAliveCheckMonitoringTimer	Timer	...	VAR_GLOBAL	T0
2	bFirstDataRecvFlag	Bit	...	VAR_GLOBAL	M0
3	bSystemSwitchingAbnormalCompleteDevice	Bit	...	VAR_GLOBAL	M1
4	wAliveCheckData_PreviousValue	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D12286
5	wAliveCheckData_RecvData	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D12287


(0)
RCPU.stSM.bAlways_ON
SM400
SET
RCPU.stSM.bUser_System_Switching_Availability
SM1646
(2)
bFirstDataRecvFlag
M0
<>_U
K0
wAliveCheckData_RecvData
D12287
SET
bFirstDataRecvFlag
M0
(7)
bFirstDataRecvFlag
M0
RCPU.stSM.bControl_System_Identification_Flag
SM1634
RCPU.stSM.bStandby_System_Identification_Flag
SM1635
=_U
wAliveCheckData_RecvData
D12287
wAliveCheckData_PreviousValue
D12286
OUT
tAliveCheckMonitoringTimer
T0
K100
<>_U
wAliveCheckData_RecvData
D12287
wAliveCheckData_PreviousValue
D12286
MOV
wAliveCheckData_RecvData
D12287
wAliveCheckData_PreviousValue
D12286
(24)
tAliveCheckMonitoringTimer
T0
SP.CONTSW
K1
bSystemSwitchingAbnormalCompleteDevice
M1
RST
tAliveCheckMonitoringTimer
T0
(33)
RCPU.stSM.bAfter_System_Switching_One_SCAN_ON_Standby_System_To_Control_System
SM1643
RST
tAliveCheckMonitoringTimer
T0
(38)
END

(1) Turns SM1646 ON to execute the SP.CONTSW instruction.

(2) Turns M0 ON to enable system switching after receiving the first data for alive check from the master station.

(7) Monitors whether data for alive check can be received from the master station (for 10 seconds).

(24) Switches the system if no data can be received for 10 seconds.

(33) Resets the monitoring timer if the system is switched by a cause other than a MODBUS communication error.

Appendix 9 Using the Module in the Redundant System with Redundant Extension Base Unit

This section explains examples when using a C24 in a redundant system with a redundant extension base unit.

Restrictions

The following shows the restricted functions and specifications of when using a C24 in a redundant system with a redundant extension base unit.

Available modules

Use the following modules in a redundant system.

Model	Type	Firmware version
RJ71C24	Serial communication module	'14' or later
RJ71C24-R2		
RJ71C24-R4		

Restricted functions

The following functions have some restrictions applied when using a C24 in a redundant system with a redundant extension base unit.

The other functions can be used without restrictions.

Function	Restriction	Reference
Data communication using the MC protocol	When executing a command that manages files, specify the I/O number other than those of the control system (03D0H) and standby system (03D1H).	MELSEC Communication Protocol Reference Manual
MELSOFT connection	Specifying a connection destination has some restrictions. When specifying "Not specified", the operation is performed in the same way as when specifying "Control System". Operations involving file writing (such as 'Write to PLC') cannot be performed. To perform operations involving file writing, connect an engineering tool via a module on the main base unit.	GX Works3 Operating Manual

■Considerations

- If the system is switched during the MELSOFT connection, MC protocol communications, or communications using dedicated instructions from other stations, a communication timeout may occur.
- If the system is switched during the simple CPU communications, a CPU response monitoring timer error (7A02H) may occur. Try again as necessary.
- If the system is switched during communications using MODBUS, a CPU response monitoring timer timeout (7380H) may occur. Try again as necessary.

Dedicated instructions

Dedicated instructions cannot be used.

Mount a serial communication module on the main base unit to use dedicated instructions.

Point

For alternative dedicated instructions in the program using I/O signals, refer to the following section.

☞ Page 662 Alternative program examples of dedicated instructions

Module FBs

Module FBs cannot be used.

Module parameter

- The module extended parameter and predefined protocol setting must be written to the C24, not to the CPU module.
- Data reception with an interrupt program cannot be used. Do not set the item in [Module Parameter] ⇨ [Interrupt Settings].

Alternative program examples of dedicated instructions

The following shows the programs using I/O signals to alternate dedicated instructions.

Availability of alternatives using I/O signals

Dedicated instructions which can be replaced with the program using I/O signals are as follows:

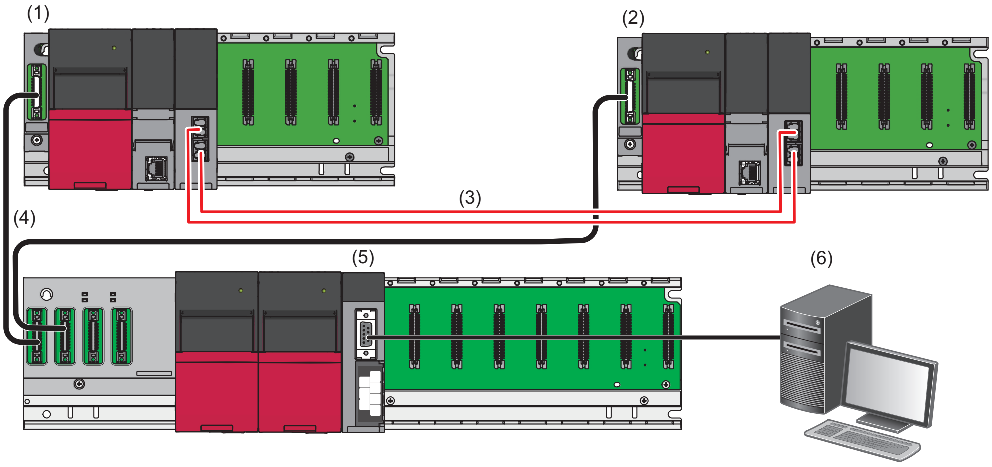
○: Available, —: Not available, △: Available with restriction

Dedicated instruction		Processing	Alternative	Reference
ONDEMAND	Sending data using the on-demand function	Sends data using the on-demand function of MC protocol.	○	Page 665 Alternative program example of the ONDEMAND instruction
CPRTCL	Executing the protocols registered for the predefined protocol support function	Executes the protocol registered using the engineering tool.	○	Page 668 Alternative program example of the CPRTCL instruction
OUTPUT	Sending data using the nonprocedural protocol	Sends data in user-specified message format through communications using the nonprocedural protocol.	○	Page 670 Alternative program example of the OUTPUT instruction
INPUT	Receiving data using the nonprocedural protocol	Receives data in user-specified message format through communications using the nonprocedural protocol.	○	Page 672 Alternative program example of the INPUT instruction
BIDOUT	Sending data using the bidirectional protocol	Sends the specified amount of data through communications using the bidirectional protocol.	○	Page 674 Alternative program example of the BIDOUT instruction
BIDIN	Receiving data using the bidirectional protocol	Receives data through communications using the bidirectional protocol.	○	Page 674 Alternative program example of the BIDIN instruction
SPBUSY	Reading the data send/receive status	Reads the status of data sent/received by using the serial communication instruction.	—	—
BUFRCVS	Receiving data using an interrupt program	Receives data using an interrupt program through communications under control of the nonprocedural protocol or bidirectional protocol.	—	—
PRR	Sending data using user frames	Sends data with user frames according to the specification in the user frame specification area for sending, through communication with the nonprocedural protocol.	○	Page 675 Alternative program example of the PRR instruction
CSET	Clearing receive data	Clears receive data using the nonprocedural protocol.	△[*1]	Page 678 Alternative program example of the CSET instruction (receive data clear)
	Registering/canceling the programmable controller CPU monitoring	Registers or cancels programmable controller CPU monitoring to use the programmable controller CPU monitoring function.	—	—
	Initial setting	Sets the send/receive data quantity unit (word/byte) and data send/receive area.	○	Page 680 Alternative program example of the CSET instruction (initial setting)
PUTE	Registering user frames	Registers user frames.	○	Page 682 Alternative program example of the PUTE instruction
GETE	Reading user frames	Reads user frames.	○	Page 684 Alternative program example of the GETE instruction
UINI	Switching the mode	Switches the serial communication module mode, transmission specifications, and self-station number.	△[*2]	Page 685 Alternative program example of the UINI instruction

[*1] Transmission processing must be stopped when clearing receive data.

[*2] Station number or operating status (independent/linked) cannot be changed.

System configuration example



(1) Control system
(2) Standby system
(3) Tracking cable
(4) Extension cable
(5) RJ71C24 (start I/O number: 0100H)
(6) External device

Connect the master station of the control system and that of the standby system to a redundant extension base unit with extension cables.

Parameter settings of C24 mounted on a redundant extension base unit

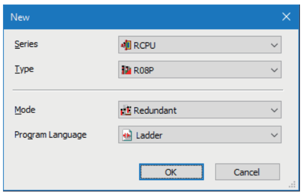
The following shows the parameter settings required to be set in an engineering tool.

Point

In this example, default values are used for parameters that are not shown below.
Change the module parameter according to the specifications of the target device.

1. Select "R08P" for "Type" and "Redundant" for "Mode" to set the CPU module.

[Project] ⇨ [New]

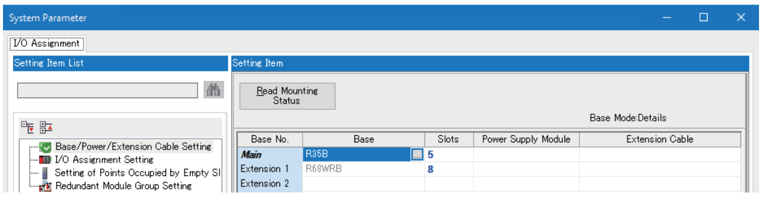


New

Series: RCPU
Type: R08P
Mode: Redundant
Program Language: Ladder

2. Select the model name of a base unit in the system parameter.

[Navigation window] ⇨ [Parameter] ⇨ [System Parameter] ⇨ [Base/Power/Extension Cable Setting]



System Parameter

I/O Assignment
Setting Item List
Setting Item
Read Mounting Status
Base/Power/Extension Cable Setting
I/O Assignment Setting
Setting of Points Occupied by Empty Sl
Redundant Module Group Setting

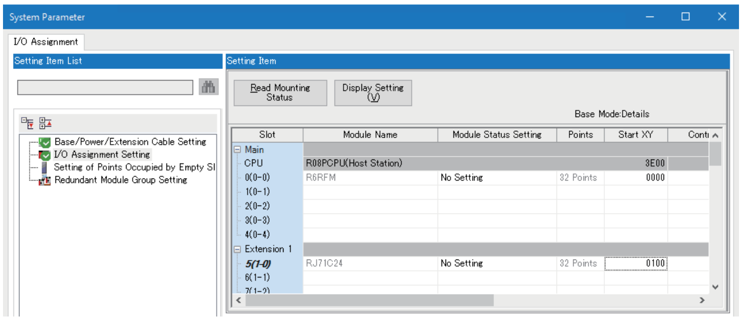
Base Mode:Details

Base No.	Base	Slots	Power Supply Module	Extension Cable
Main	R35B	5		
Extension 1	R68WRB	8		
Extension 2				

Item		Description	
		Base Type	Base Module Name
Base/Power/Extension Cable Setting	Main	Main Base Unit	R35B
	Extension 1	Extension Base Unit	R68WRB

3. Set the I/O assignment setting as follows:

[Navigation window] ⇨ [Parameter] ⇨ [System Parameter] ⇨ [I/O Assignment Setting]

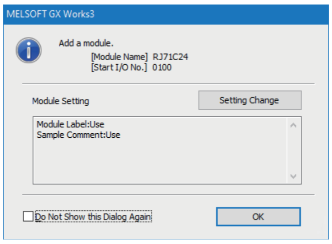


Base Mode:Details

Slot	Module Name	Module Status Setting	Points	Start XY	Conti
Main					
CPU	R08PCPU(Host Station)			3E00	
0(0-0)	R6RFM	No Setting	32 Points	0000	
1(0-1)					
2(0-2)					
3(0-3)					
4(0-4)					
Extension 1					
5(1-0)	RJ71C24	No Setting	32 Points	0100	
6(1-1)					
7(1-2)					

Item		Description	
		Module Name	**Start XY**
I/O Assignment Setting	Main	R08PCPU (Host Station)	3E00
		R6RFM	0000
	Extension 1	RJ71C24	0100

4. Click the [OK] button to add module labels of the RJ71C24.



Add a module.
[Module Name] RJ71C24
[Start I/O No.] 0100

Module Setting

Module Label:Use
Sample Comment:Use

Point

"Signal Flow Memory Tracking Setting" must be set to "Transfer" (default) in [CPU Parameter] ⇨ [Redundant System Settings] ⇨ [Tracking Setting] ⇨ [Signal Flow Memory Tracking Setting] due to some instructions used in a program, such as PLS instruction.

Alternative program example of the ONDEMAND instruction

The following shows the program example to send data using the on-demand function of MC protocol.

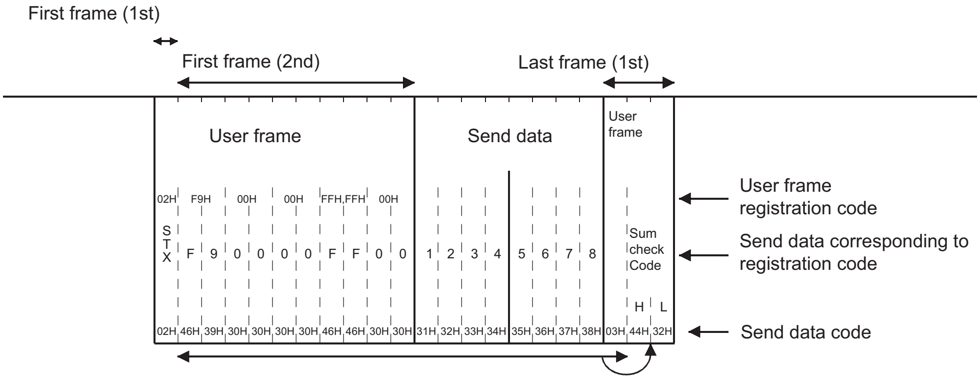
■Overview

An operation overview of sample program is as follows.

The data "12345678" is sent by using the following user frames.

User frame	User frame No.	Registered code	Registered data content
Start frame (1st)	02H(2)	02H	Data code from STX to self-station number matched to QnA compatible 3C frame, format 3
Start frame (2nd)	3EBH(1003)	F9H, 00H, 00H, FFH, FFH, 00H	
Final frame (1st)	401H(1025)	03H, FFH, F1H	Data code of ETX, sum check code equivalent to QnA compatible 3C frame, format 3

The send data is as follows:



■Parameter settings

Set parameters in an engineering tool as follows:

1. Set the following parameters.

☞ Page 663 Parameter settings of C24 mounted on a redundant extension base unit

2. Set the following items for CH1 in the module parameter.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter]

Item				Description
Basic settings	Various control specification	Communication protocol setting		MC protocol (any of "Format 1" to "Format 4")
		Communication speed setting		Select other than "Automatically set" according to the communication destination.
Application settings	User frame specification	On demand user frame specification	Start frame No. designation (1st)	2H(2)
			Final frame No. designation (1st)	401H(1025)
			Start frame No. designation (2nd)	3EBH(1003)
			Final frame No. designation (2nd)	0 (No specification)

3. Set the module extended parameters.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Extended Parameter]

Item			Description
User frame contents	User frame	Registration frame No.1003	F90000FFFF00
		Registration frame No.1025	03FFF1 (Control code: "ETX", Special code: "FFF1")

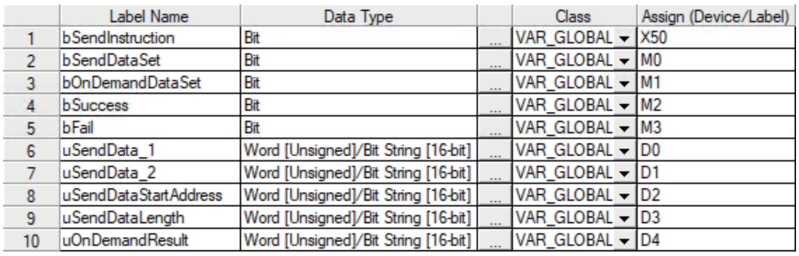
4. Write the set parameters to the module.

[Online] ⇨ [Write to PLC]

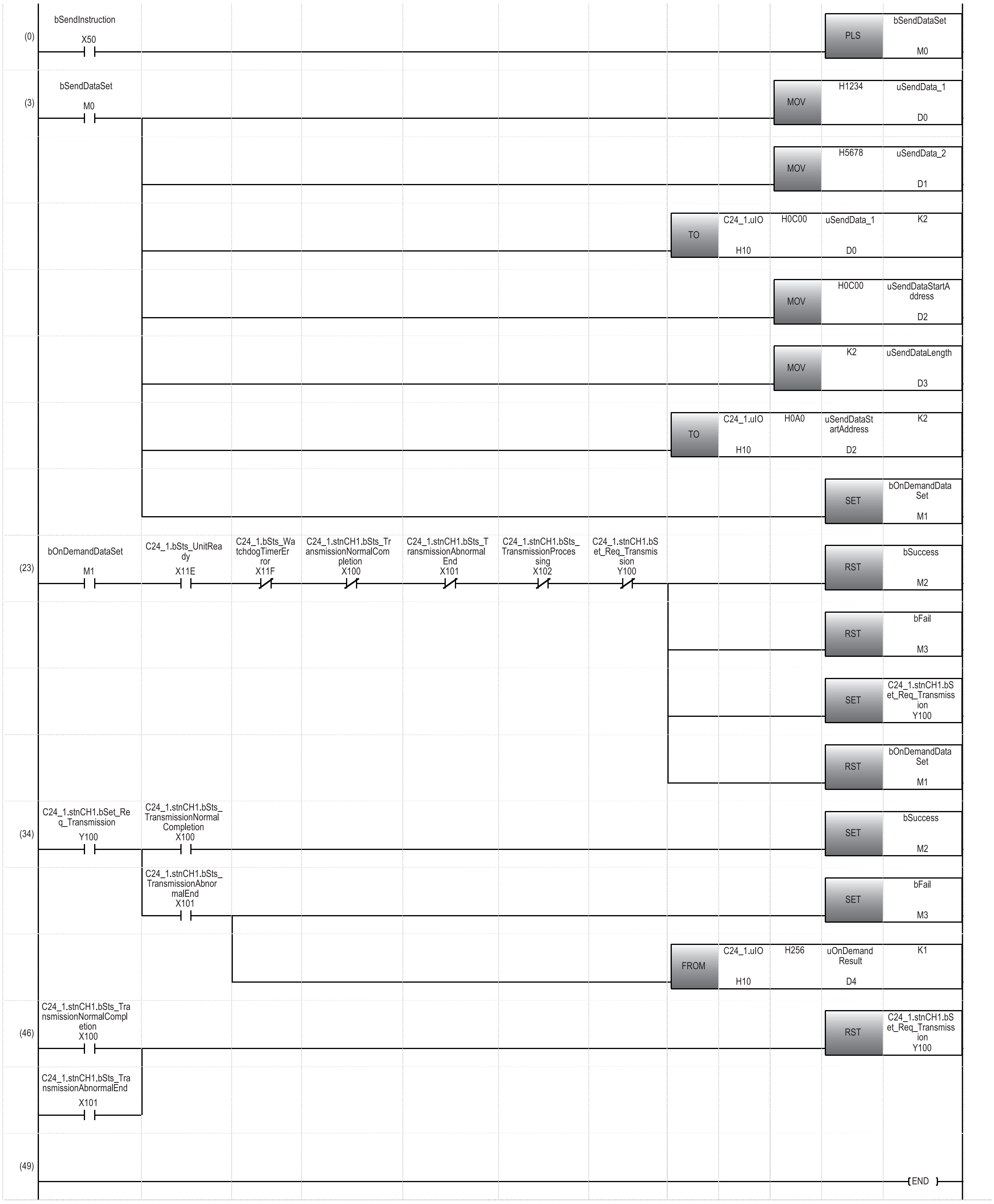
- When writing module extended parameters, an engineering tool must be connected to the control system.
- Specify "Intelligent Function Module" as the writing destination of module extended parameters.

5. Reset the CPU module or turn the power OFF and ON.

■Program example

Category	Label name	Description	Device
Module label	C24_1.uIO	Start I/O No.	—
	C24_1.stnCH1.bSts_TransmissionNormalCompletion	CH1 Transmission normal completion	X100
	C24_1.stnCH1.bSts_TransmissionAbnormalEnd	CH1 Transmission abnormal completion	X101
	C24_1.stnCH1.bSts_TransmissionProcessing	CH1 Transmission processing	X102
	C24_1.bSts_UnitReady	Module READY	X11E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X11F
	C24_1.stnCH1.bSet_Req_Transmission	CH1 Transmission request	Y100
	C24_1.bSet_Req_ErrorInitialization	Error initialization request	Y10E
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bSendInstruction	Bit	...	VAR_GLOBAL	X50
2	bSendDataSet	Bit	...	VAR_GLOBAL	M0
3	bOnDemandDataSet	Bit	...	VAR_GLOBAL	M1
4	bSuccess	Bit	...	VAR_GLOBAL	M2
5	bFail	Bit	...	VAR_GLOBAL	M3
6	uSendData_1	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
7	uSendData_2	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
8	uSendDataStartAddress	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2
9	uSendDataLength	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D3
10	uOnDemandResult	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D4



(0) Executes the program when 'bSendInstruction' turns ON.

(3) Writes send data to the C24 buffer memory C00H (user setting area).
Writes the buffer memory address C00H and 2 (send data quantity) to the C24 buffer memory A0H (for designation of on-demand function).

(23) Turns 'CH1 Transmission request' ON.

(34) Turns 'bSuccess' ON when 'CH1 Transmission normal completion' turns ON.
Turns 'bFail' ON and reads the C24 buffer memory 256H (CH1 on-demand execution result) when 'CH1 Transmission abnormal completion' turns ON.

(46) Turns 'CH1 Transmission request' OFF.

Alternative program example of the CPRTCL instruction

The following shows the program example to execute the protocol registered in an engineering tool.

■Overview

An operation overview of sample program is as follows.

A request message of the command 0401 is sent by using the MC protocol (4C frame, format 4) and data from the target device D100 is read.

The read data is stored in D6.

■Parameter settings

Set parameters in an engineering tool as follows:

1. Set the following parameters.

☞ Page 663 Parameter settings of C24 mounted on a redundant extension base unit

2. Set the following items for CH1 in the module parameter.

🖱 [Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter]

Item			Description
Basic settings	Various control specification	Communication protocol setting	Predefined protocol
		Communication speed setting	Select other than "Automatically set" according to the communication destination.

3. Write the set parameters to the module.

🖱 [Online] ⇨ [Write to PLC]

4. Start the predefined protocol support function from the [Tool] menu of engineering tool.

🖱 [Tool] ⇨ [Predefined Protocol Support Function]

5. Select a target module of the protocol setting data.

6. Create new protocol setting data.

🖱 [File]⇨[New]

7. Add the following protocol in the protocol setting screen of predefined protocol support function.

Protocol No.	Manufacturer	Model	Protocol Name
1	MITSUBISHI ELECTRIC	MC Protocol 4C Frame (Format 4)	0401:Batch RD

8. Right-click the added protocol, and select "Device Batch Setting".

9. Enter 'D0' in "Device No." in the "Device Batch Setting" screen.

10. Write the created protocol setting data to C24.

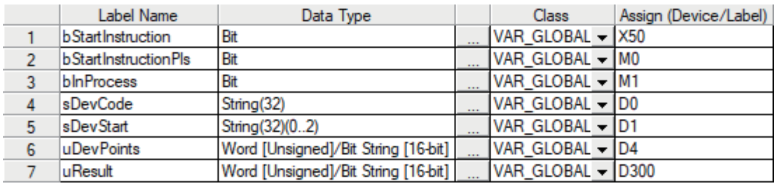
🖱 [Online] ⇨ [Write to Module]

Specify "Intelligent Function Module" for "Target Memory".

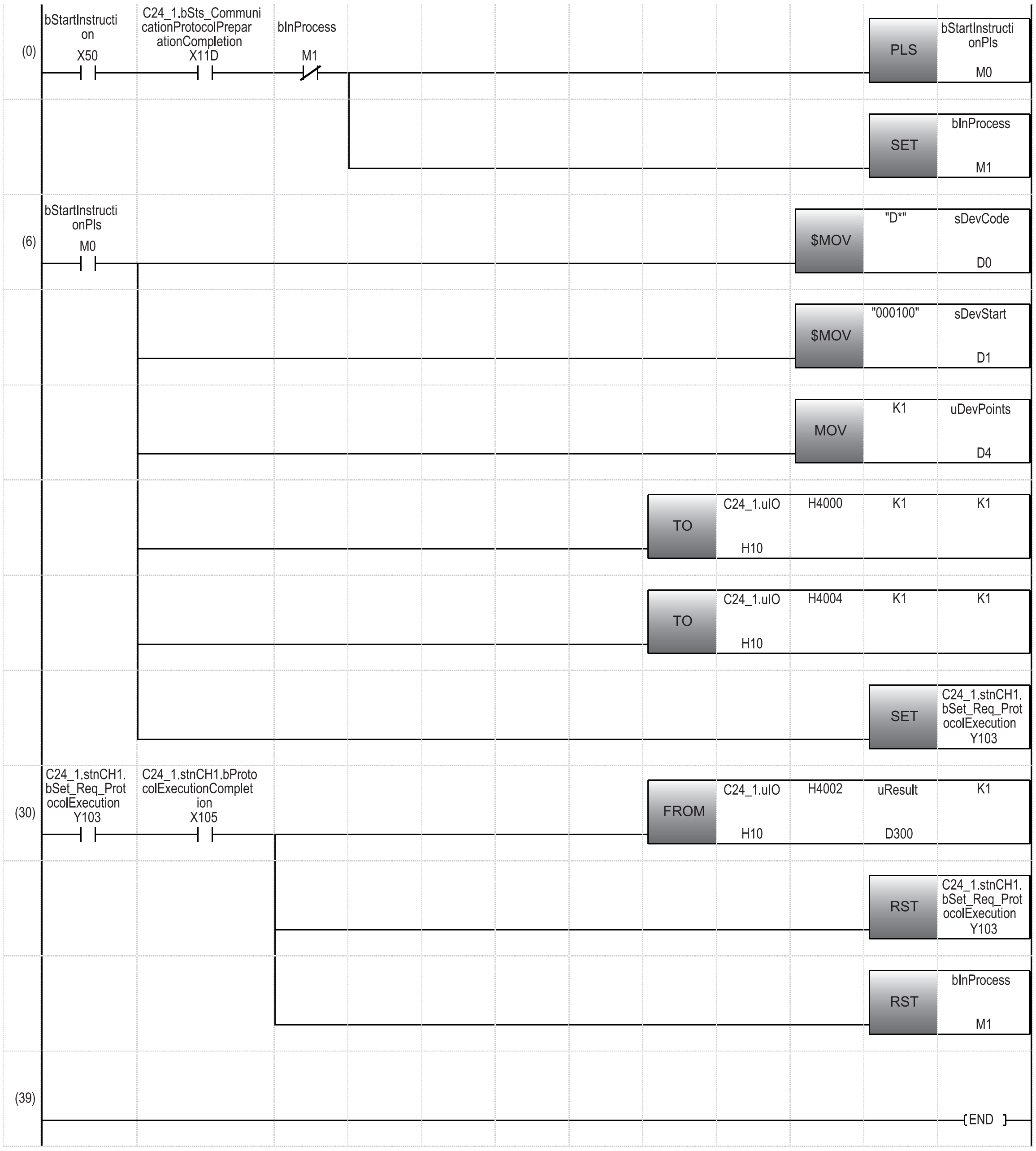
If the CPU module is in the backup mode, protocol setting data cannot be written. Be sure to switch the CPU module to the separate mode before writing.

11. Reset the CPU module or turn the power OFF and ON.

■Program example

Category	Label name	Description	Device
Module label	C24_1.uIO	Start I/O No.	—
	C24_1.bSts_CommunicationProtocolPreparationCompletion	Predefined protocol ready	X11D
	C24_1.stnCH1.bProtocolExecutionCompletion	CH1 Protocol execution completion	X105
	C24_1.stnCH1.bSet_Req_ProtocolExecution	CH1 protocol execution request	Y103
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bStartInstruction	Bit	...	VAR_GLOBAL ▾	X50
2	bStartInstructionPls	Bit	...	VAR_GLOBAL ▾	M0
3	bInProcess	Bit	...	VAR_GLOBAL ▾	M1
4	sDevCode	String(32)	...	VAR_GLOBAL ▾	D0
5	sDevStart	String(32)(0..2)	...	VAR_GLOBAL ▾	D1
6	uDevPoints	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL ▾	D4
7	uResult	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL ▾	D300


(0)
bStartInstruction X50
C24_1.bSts_CommunicationProtocolPreparationCompletion X11D
bInProcess M1
PLS bStartInstructionPls M0
SET bInProcess M1
(6)
bStartInstructionPls M0
$MOV "D*" sDevCode D0
$MOV "000100" sDevStart D1
MOV K1 uDevPoints D4
TO C24_1.uIO H10 H4000 K1 K1
TO C24_1.uIO H10 H4004 K1 K1
SET C24_1.stnCH1.bSet_Req_ProtocolExecution Y103
(30)
C24_1.stnCH1.bSet_Req_ProtocolExecution Y103
C24_1.stnCH1.bProtocolExecutionCompletion X105
FROM C24_1.uIO H10 H4002 uResult D300 K1
RST C24_1.stnCH1.bSet_Req_ProtocolExecution Y103
RST bInProcess M1
(39)
END

(0) Executes the program when 'bStartInstruction' turns ON.

(6) Sets the device code, start device number, and number of device points to be read.
Writes '1' to the C24 buffer memory 4000H (CH1 number of consecutive protocol executions).
Writes '1' to the C24 buffer memory 4004H (CH1 protocol number).
Turns 'CH1 Protocol execution request' ON.

(30) Reads the C24 buffer memory 4002H (CH1 data transmission result) when 'Protocol execution completion' turns ON.
Turns 'CH1 Protocol execution request' OFF.

Alternative program example of the OUTPUT instruction

The following shows the program to send data in user-specified message format through communications using the nonprocedural protocol.

■Overview

An operation overview of sample program is as follows.
The data stored in D11 or later is sent by using the nonprocedural protocol.

■Parameter settings

Set parameters in an engineering tool as follows:

1. Set the following parameters.

☞ Page 663 Parameter settings of C24 mounted on a redundant extension base unit

2. Set the following items for CH1 in the module parameter.

🖱 [Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter]

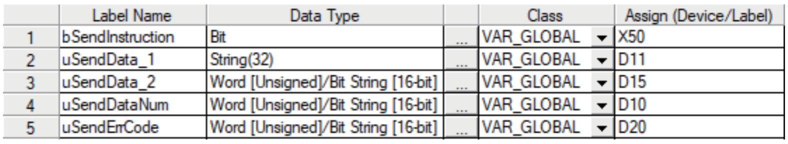
Item			Description
Basic settings	Various control specification	Communication protocol setting	Nonprocedural protocol
		Communication speed setting	Select other than "Automatically set" according to the communication destination.

3. Write the set parameters to the module.

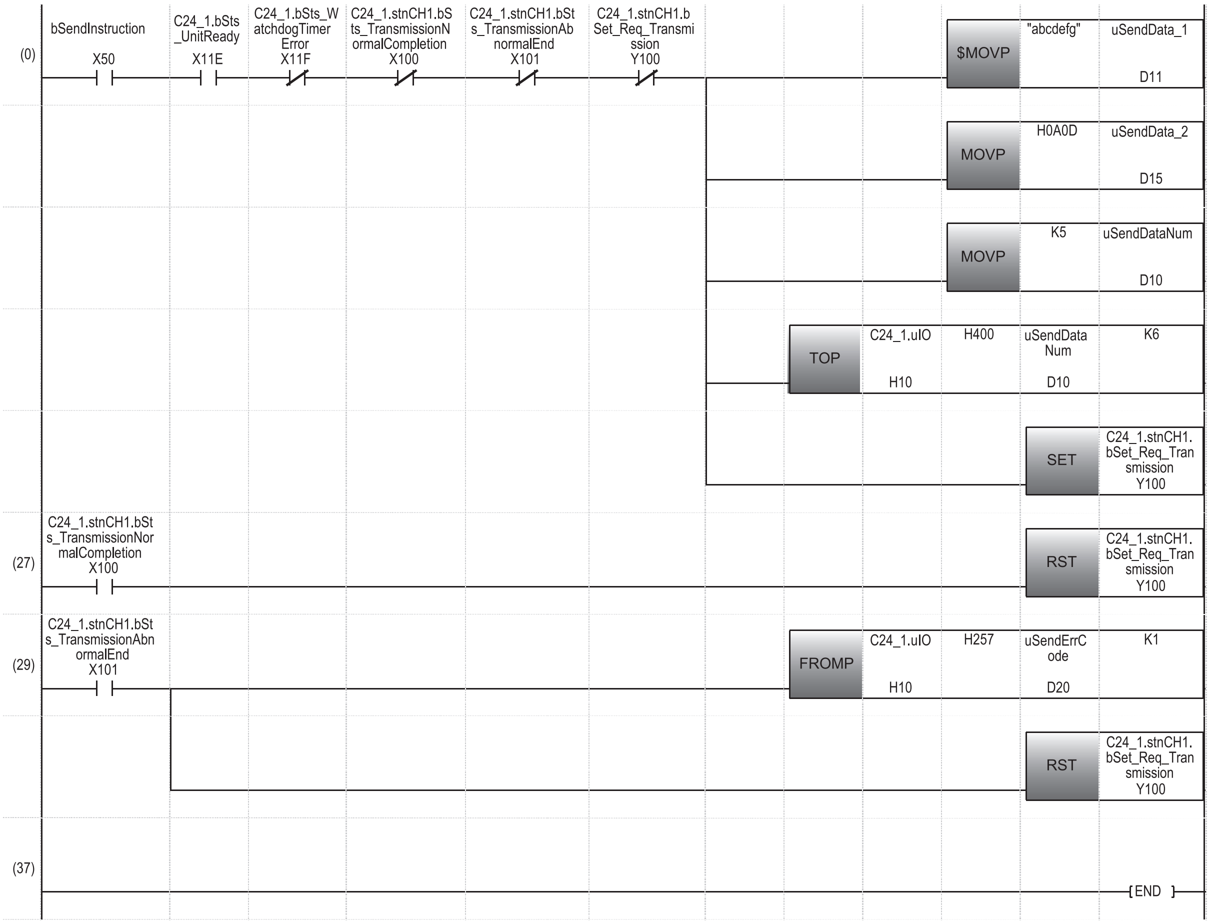
🖱 [Online] ⇨ [Write to PLC]

4. Reset the CPU module or turn the power OFF and ON.

■Program example

Category	Label name	Description	Device
Module label	C24_1.uIO	Start I/O No.	—
	C24_1.stnCH1.bSts_TransmissionNormalCompletion	CH1 Transmission normal completion	X100
	C24_1.stnCH1.bSts_TransmissionAbnormalEnd	CH1 Transmission abnormal completion	X101
	C24_1.bSts_UnitReady	Module READY	X11E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X11F
	C24_1.stnCH1.bSet_Req_Transmission	CH1 Transmission request	Y100
	C24_1.bSet_Req_ErrorInitialization	Error initialization request	Y10E
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bSendInstruction	Bit	...	VAR_GLOBAL	X50
2	uSendData_1	String(32)	...	VAR_GLOBAL	D11
3	uSendData_2	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D15
4	uSendDataNum	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D10
5	uSendErrCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D20


(0)
bSendInstruction X50
C24_1.bSts_UnitReady X11E
C24_1.bSts_WatchdogTimerError X11F
C24_1.stnCH1.bSts_TransmissionNormalCompletion X100
C24_1.stnCH1.bSts_TransmissionAbnormalEnd X101
C24_1.stnCH1.bSet_Req_Transmission Y100
$MOVP "abcdefg" uSendData_1 D11
MOVP H0A0D uSendData_2 D15
MOVP K5 uSendDataNum D10
TOP C24_1.uIO H10 H400 uSendDataNum D10 K6
SET C24_1.stnCH1.bSet_Req_Transmission Y100
(27)
C24_1.stnCH1.bSts_TransmissionNormalCompletion X100
RST C24_1.stnCH1.bSet_Req_Transmission Y100
(29)
C24_1.stnCH1.bSts_TransmissionAbnormalEnd X101
FROMP C24_1.uIO H10 H257 uSendErrCode D20 K1
RST C24_1.stnCH1.bSet_Req_Transmission Y100
(37)
END

(0) Sets the send data, end code, and send data quantity.
Writes the above items to the C24 buffer memory 400H (send/receive area).
Turns 'CH1 Transmission request' ON.

(27) Turns 'CH1 Transmission request' OFF when 'CH1 Transmission normal completion' turns ON.

(29) Reads the C24 buffer memory 257H (CH1 data transmission result) when 'CH1 Transmission abnormal completion' turns ON.
Turns 'CH1 Transmission request' OFF.

Alternative program example of the INPUT instruction

The following shows the program example to receive data in user-specified message format through communications using the nonprocedural protocol.

■Overview

An operation overview of sample program is as follows.

The data received from a target device is stored in D1 or later by using the nonprocedural protocol.

■Parameter settings

Set parameters in an engineering tool as follows:

1. Set the following parameters.

☞ Page 663 Parameter settings of C24 mounted on a redundant extension base unit

2. Set the following items for CH1 in the module parameter.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter]

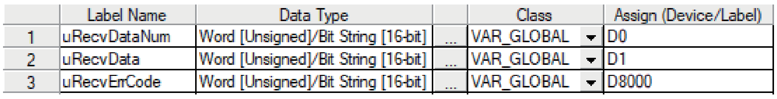
Item			Description
Basic settings	Various control specification	Communication protocol setting	Nonprocedural protocol
		Communication speed setting	Select other than "Automatically set" according to the communication destination.

3. Write the set parameters to the module.

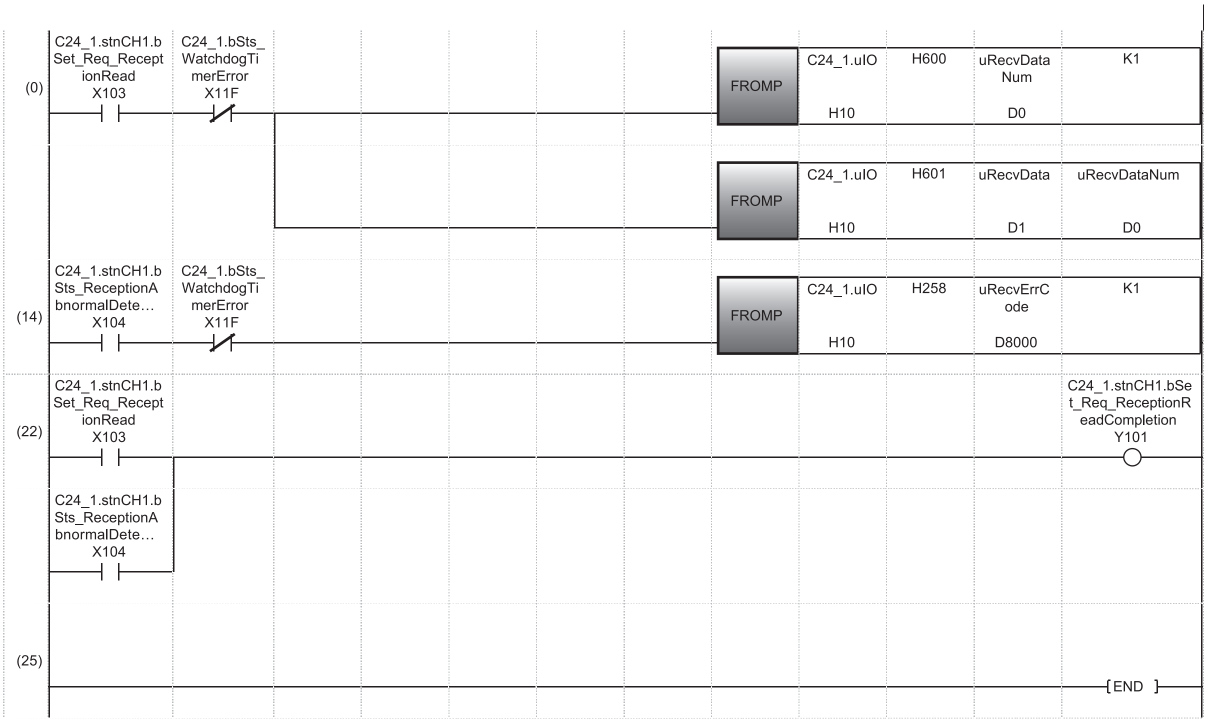
[Online] ⇨ [Write to PLC]

4. Reset the CPU module or turn the power OFF and ON.

■Program example

Category	Label name	Description	Device
Module label	C24_1.uIO	Start I/O No.	—
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X103
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection	CH1 Reception abnormal detection	X104
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X11F
	C24_1.stnCH1.bSet_Req_ReceptionReadCompletion	CH1 Reception data read completion	Y101
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	uRecvDataNum	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
2	uRecvData	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
3	uRecvErrCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D8000


(0)
C24_1.stnCH1.bSet_Req_ReceptionRead X103
C24_1.bSts_WatchdogTimerError X11F
FROMP C24_1.uIO H10 H600 uRecvDataNum D0 K1
FROMP C24_1.uIO H10 H601 uRecvData D1 uRecvDataNum D0
(14)
C24_1.stnCH1.bSts_ReceptionAbnormalDete… X104
C24_1.bSts_WatchdogTimerError X11F
FROMP C24_1.uIO H10 H258 uRecvErrCode D8000 K1
(22)
C24_1.stnCH1.bSet_Req_ReceptionRead X103
C24_1.stnCH1.bSts_ReceptionAbnormalDete… X104
C24_1.stnCH1.bSet_Req_ReceptionReadCompletion Y101
(25)
END

(0) Reads the C24 buffer memory 600H (CH1 receive data quantity) and 601H (CH1 receive data) when data is received.

(14) Reads the C24 buffer memory 258H (CH1 data reception result) when abnormal reception is detected.

(22) Turns 'CH1 Reception data read completion' ON when data is received or abnormal reception is detected.

Alternative program example of the BIDOUT instruction

The following shows the program example to send data for the designated data count using the bidirectional protocol.

■Overview

An operation overview of sample program is as follows.

The data stored in D11 or later is sent by using the bidirectional protocol.

■Parameter settings

Set parameters in an engineering tool as follows:

1. Set the following parameters.

☞ Page 663 Parameter settings of C24 mounted on a redundant extension base unit

2. Set the following items for CH1 in the module parameter.

🖱 [Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter]

Item			Description
Basic settings	Various control specification	Communication protocol setting	Bidirectional protocol
		Communication speed setting	Select other than "Automatically set" according to the communication destination.

3. Write the set parameters to the module.

🖱 [Online] ⇨ [Write to PLC]

4. Reset the CPU module or turn the power OFF and ON.

■Program example

☞ Page 671 Program example

Alternative program example of the BIDIN instruction

The following shows the program example to receive data using the bidirectional protocol.

■Overview

An operation overview of sample program is as follows.

The data received from a target device is stored in D1 or later by using the bidirectional protocol.

■Parameter settings

Set parameters in an engineering tool as follows:

1. Set the following parameters.

☞ Page 663 Parameter settings of C24 mounted on a redundant extension base unit

2. Set the following items for CH1 in the module parameter.

🖱 [Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter]

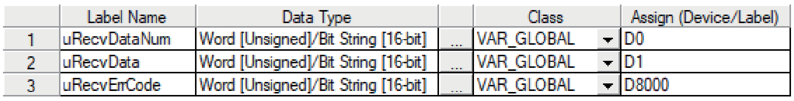
Item			Description
Basic settings	Various control specification	Communication protocol setting	Bidirectional protocol
		Communication speed setting	Select other than "Automatically set" according to the communication destination.

3. Write the set parameters to the module.

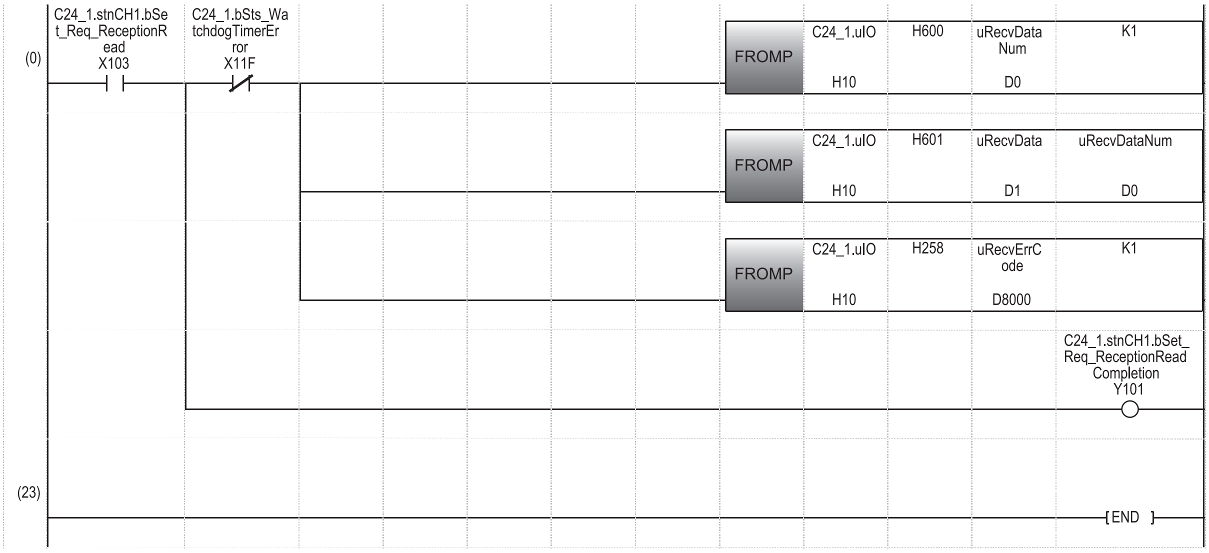
🖱 [Online] ⇨ [Write to PLC]

4. Reset the CPU module or turn the power OFF and ON.

■Program example

Category	Label name	Description	Device
Module label	C24_1.uIO	Start I/O No.	—
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X103
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X11F
	C24_1.stnCH1.bSet_Req_ReceptionReadCompletion	CH1 Reception data read completion	Y101
	C24_1.bSet_Req_ErrorInitialization	Error initialization request	Y10E
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	uRecvDataNum	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
2	uRecvData	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
3	uRecvErrCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D8000



(0) Reads the C24 buffer memory 600H (CH1 receive data quantity), 601H (CH1 receive data), and 258H (CH1 data reception result) when data is received. Turns 'CH1 Reception data read completion' ON.

Alternative program example of the PRR instruction

The following shows the program example to send data with a user frame using the nonprocedural protocol (according to the contents defined in the send user frame designation area).

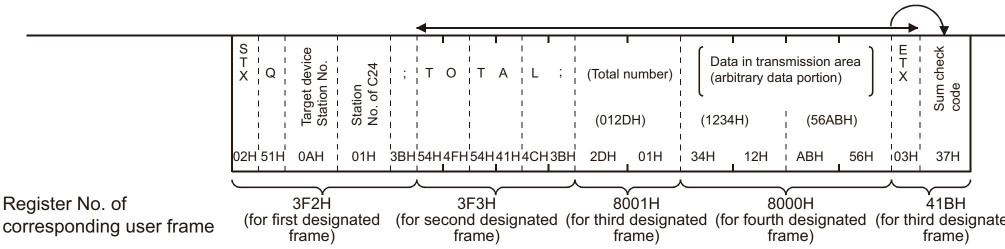
■Overview

An operation overview of sample program is as follows.

Data is sent by using the following user frames.

User frame	User frame No.	Registered code	Registered data content
Send frame No. designation (1st)	3F2H(1010)	02H, 51H, 0AH, FFH, 01H, 3BH	[STX], "Q", Station No. of target device, Self-station No., ";"
Send frame No. designation (2nd)	3F3H(1011)	54H, 4FH, 54H, 41H, 4CH, 3BH	"TOTAL;"
Send frame No. designation (3rd)	8001H(32769)	2DH, 01H	Arbitrary data (021DH)
Send frame No. designation (4th)	8000H(32768)	34H, 12H, ABH, 56H	Arbitrary data in send area (1234H, 56ABH)
Send frame No. designation (5th)	41BH(1051)	03H, FFH, F0H	Data code of [ETX], sum check code

The send data is as follows.



STX
Q
Target device Station No.
Station No. of C24
;
T
O
T
A
L
;
(Total number)
(012DH)
Data in transmission area (arbitrary data portion)
(1234H)
(56ABH)
ETX
Sum check code
02H
51H
0AH
01H
3BH
54H
4FH
54H
41H
4CH
3BH
2DH
01H
34H
12H
ABH
56H
03H
37H
Register No. of corresponding user frame
3F2H (for first designated frame)
3F3H (for second designated frame)
8001H (for third designated frame)
8000H (for fourth designated frame)
41BH (for third designated frame)

* Four bytes of data in the transmission area are transmitted as is.
(Byte units in the left diagram.)

■Parameter settings

Set parameters in an engineering tool as follows:

1. Set the following parameters.

☞ Page 663 Parameter settings of C24 mounted on a redundant extension base unit

2. Set the following items for CH1 in the module parameter.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter]

Item				Description
Basic settings	Various control specification	Communication protocol setting		Nonprocedural protocol
		Communication speed setting		Select other than "Automatically set" according to the communication destination.
		Transmission setting	Data bit	8
Application settings	User frame specification	Send user frame designation	Output start pointer designation	1
			Output quantity designation	5
			Send frame No. designation (1st)	3F2H(1010)
			Send frame No. designation (2nd)	3F3H(1011)
			Send frame No. designation (3rd)	8001H(32769)
			Send frame No. designation (4th)	8000H(32768)
			Send frame No. designation (5th)	41BH(1051)

3. Set the module extended parameters.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Extended Parameter]

Item			Description
User frame contents	User frame	Registration frame No.1010	02510AFF013B
		Registration frame No.1011	544F54414C3B
		Registration frame No.1051	03FFF0 (Control code: "ETX", Special code: "FFF0")

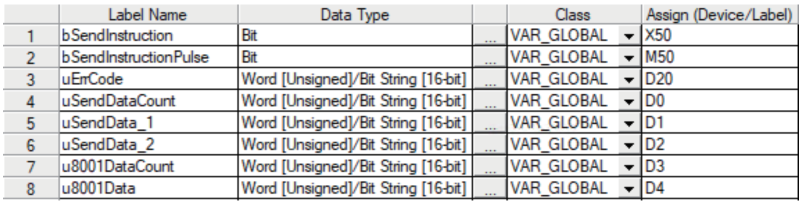
4. Write the set parameters to the module.

[Online] ⇨ [Write to PLC]

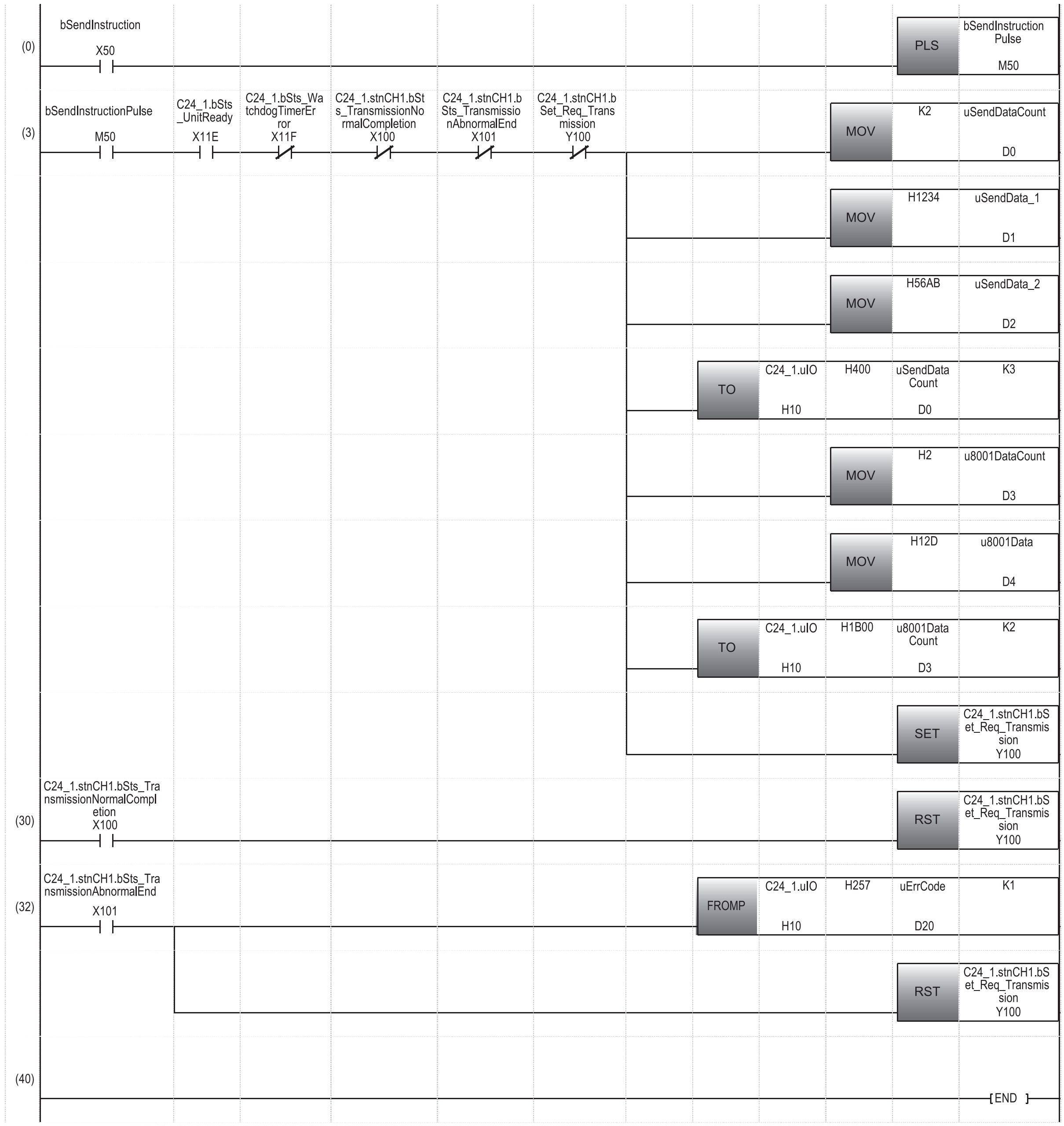
- When writing module extended parameters, an engineering tool must be connected to the control system.
- Specify "Intelligent Function Module" as the writing destination of module extended parameters.

5. Reset the CPU module or turn the power OFF and ON.

■Program example

Category	Label name	Description	Device
Module label	C24_1.uIO	Start I/O No.	—
	C24_1.stnCH1.bSts_TransmissionNormalCompletion	CH1 Transmission normal completion	X100
	C24_1.stnCH1.bSts_TransmissionAbnormalEnd	CH1 Transmission abnormal completion	X101
	C24_1.bSts_UnitReady	Module READY	X11E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X11F
	C24_1.stnCH1.bSet_Req_Transmission	CH1 Transmission request	Y100
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bSendInstruction	Bit	...	VAR_GLOBAL	X50
2	bSendInstructionPulse	Bit	...	VAR_GLOBAL	M50
3	uErrCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D20
4	uSendDataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
5	uSendData_1	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
6	uSendData_2	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2
7	u8001DataCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D3
8	u8001Data	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D4



(0) Executes the program when 'bSendInstruction' turns ON.

(3) Writes send data to the C24 buffer memory 400H (send/receive area).
Writes the user frame No.8001H to the C24 buffer memory 1B00H (for user registration).
Turns 'CH1 Transmission request' ON.

(30) Turns 'CH1 Transmission request' OFF when 'Transmission normal completion' turns ON.

(32) Reads the C24 buffer memory 257H (data transmission result) when 'Transmission abnormal completion' turns ON.
Turns 'CH1 Transmission request' OFF.

Alternative program example of the CSET instruction (receive data clear)

The following shows the program example to clear receive data using the nonprocedural protocol.

■Overview

An operation overview of sample program is as follows.

Receive data is cleared for the OS area of the C24 by using the nonprocedural protocol.

Point

- Do not execute a clear request of receive data during data transmission. Otherwise, the data transmission is canceled.
- Resume data communication with a target device after the receive data clear processing with C24 is completed.

■Parameter settings

Set parameters in an engineering tool as follows:

1. Set the following parameters.

☞ Page 663 Parameter settings of C24 mounted on a redundant extension base unit

2. Set the following items for CH1 in the module parameter.

🖱 [Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Parameter]

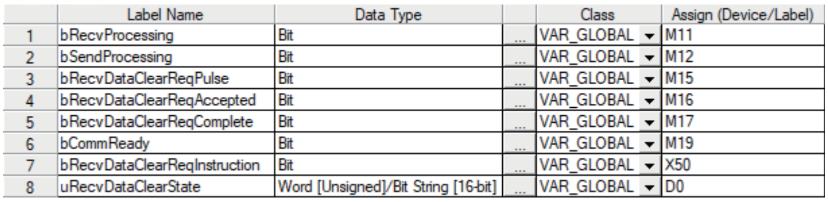
Item			Description
Basic settings	Various control specification	Communication protocol setting	Nonprocedural protocol
		Communication speed setting	Select other than "Automatically set" according to the communication destination.

3. Write the set parameters to the module.

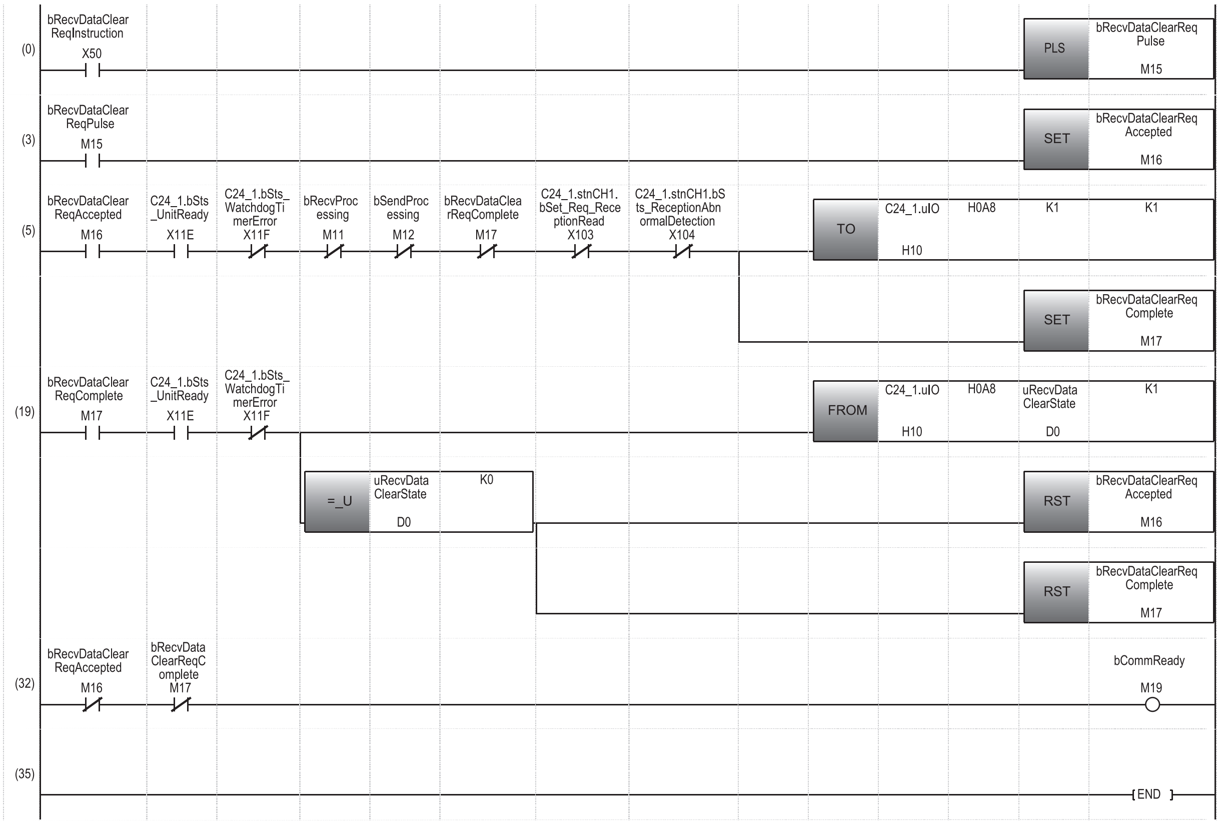
🖱 [Online] ⇨ [Write to PLC]

4. Reset the CPU module or turn the power OFF and ON.

■Program example

Category	Label name	Description	Device
Module label	C24_1.uIO	Start I/O No.	—
	C24_1.stnCH1.bSet_Req_ReceptionRead	CH1 Reception data read request	X103
	C24_1.stnCH1.bSts_ReceptionAbnormalDetection	CH1 Reception abnormal detection	X104
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bRecvProcessing	Bit	...	VAR_GLOBAL	M11
2	bSendProcessing	Bit	...	VAR_GLOBAL	M12
3	bRecvDataClearReqPulse	Bit	...	VAR_GLOBAL	M15
4	bRecvDataClearReqAccepted	Bit	...	VAR_GLOBAL	M16
5	bRecvDataClearReqComplete	Bit	...	VAR_GLOBAL	M17
6	bCommReady	Bit	...	VAR_GLOBAL	M19
7	bRecvDataClearReqInstruction	Bit	...	VAR_GLOBAL	X50
8	uRecvDataClearState	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0


(0)
bRecvDataClear ReqInstruction X50
PLS
bRecvDataClearReq Pulse M15
(3)
bRecvDataClear ReqPulse M15
SET
bRecvDataClearReq Accepted M16
(5)
bRecvDataClear ReqAccepted M16
C24_1.bSts _UnitReady X11E
C24_1.bSts_ WatchdogTi merError X11F
bRecvProc essing M11
bSendProc essing M12
bRecvDataClea rReqComplete M17
C24_1.stnCH1. bSet_Req_Rece ptionRead X103
C24_1.stnCH1.bS ts_ReceptionAbn ormalDetection X104
TO
C24_1.uIO H10
H0A8
K1
K1
SET
bRecvDataClearReq Complete M17
(19)
bRecvDataClear ReqComplete M17
C24_1.bSts _UnitReady X11E
C24_1.bSts_ WatchdogTi merError X11F
FROM
C24_1.uIO H10
H0A8
uRecvData ClearState D0
K1
=_U
uRecvData ClearState D0
K0
RST
bRecvDataClearReq Accepted M16
RST
bRecvDataClearReq Complete M17
(32)
bRecvDataClear ReqAccepted M16
bRecvData ClearReqC omplete M17
bCommReady M19
(35)
END

(0) Executes the program when 'X50' turns ON.

(5) Writes '1' (requested) to the C24 buffer memory 0A8H (receive data clear request) when 'data clear request acknowledgement flag' turns ON.

(19) Reads the C24 buffer memory 0A8H (receive data clear status) when 'data clear request completion flag' turns ON.
Turns 'data clear request completion flag' OFF when the receive data clear status is '0' (not requested).

(32) When 'data clear request acknowledgement flag' and 'data clear request completion flag' are turned OFF, data communication becomes available.

Alternative program example of the CSET instruction (initial setting)

The following shows the program example to set the send/receive data quantity unit (word/byte) and data send/receive area.

■Overview

An operation overview of sample program is as follows.

The following items for CH1 are set.

Item	Description
Send/receive data quantity unit (word/byte)	Word unit
On-demand function buffer memory start address	400H
On-demand function buffer memory size	10H
Sending area start address for nonprocedural/bidirectional protocols	C00H
Nonprocedural/bidirectional protocols sending area size	100H
Nonprocedural/bidirectional protocols receiving area start address	D00H
Nonprocedural/bidirectional protocols receiving area size	300H

■Parameter settings

Set parameters in an engineering tool as follows:

1. Set the following parameters.

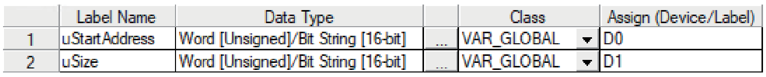
☞ Page 663 Parameter settings of C24 mounted on a redundant extension base unit

2. Write the set parameters to the module.

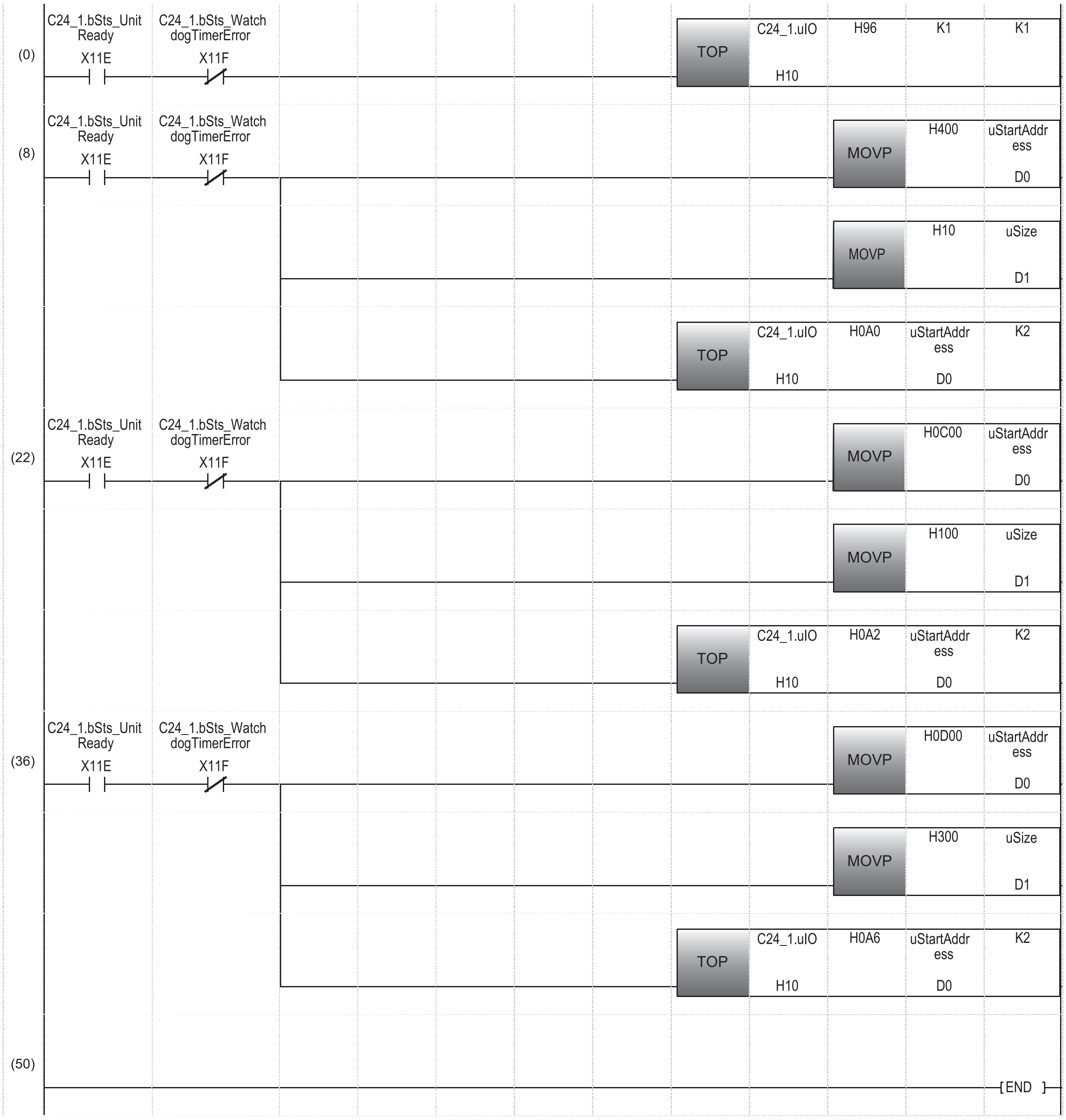
🖱 [Online] ⇨ [Write to PLC]

3. Reset the CPU module or turn the power OFF and ON.

■Program example

Category	Label name	Description	Device
Module label	C24_1.uIO	Start I/O No.	—
	C24_1.bSts_UnitReady	Module READY	X11E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X11F
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	uStartAddress	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
2	uSize	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1


(0)
C24_1.bSts_UnitReady X11E
C24_1.bSts_WatchdogTimerError X11F
TOP C24_1.uIO H10 H96 K1 K1
(8)
C24_1.bSts_UnitReady X11E
C24_1.bSts_WatchdogTimerError X11F
MOVP H400 uStartAddress D0
MOVP H10 uSize D1
TOP C24_1.uIO H10 H0A0 uStartAddress D0 K2
(22)
C24_1.bSts_UnitReady X11E
C24_1.bSts_WatchdogTimerError X11F
MOVP H0C00 uStartAddress D0
MOVP H100 uSize D1
TOP C24_1.uIO H10 H0A2 uStartAddress D0 K2
(36)
C24_1.bSts_UnitReady X11E
C24_1.bSts_WatchdogTimerError X11F
MOVP H0D00 uStartAddress D0
MOVP H300 uSize D1
TOP C24_1.uIO H10 H0A6 uStartAddress D0 K2
(50)
END

(0) Writes '1' (word unit) to the C24 buffer memory 96H (CH1 word/byte units designation).

(8) Writes 400H to the C24 buffer memory A0H (CH1 on-demand function buffer memory start address designation) and 10H to A1H (CH1 on-demand function data length designation).

(22) Writes C00H to the C24 buffer memory A2H (CH1 send buffer memory start address designation) and 100H to A3H (CH1 send buffer memory length designation).

(36) Writes D00H to the C24 buffer memory A6H (CH1 receiving buffer memory start address designation) and 300H to A7H (CH1 receive buffer memory length designation).

Alternative program example of the PUTE instruction

The following shows the program example to register a user frame.

■Overview

An operation overview of sample program is as follows.

F9H, 00H, 00H, FFH, FFH, and 00H are registered in the user frame No.3EBH (1003).

■Parameter settings

Set parameters in an engineering tool as follows:

1. Set the following parameters.

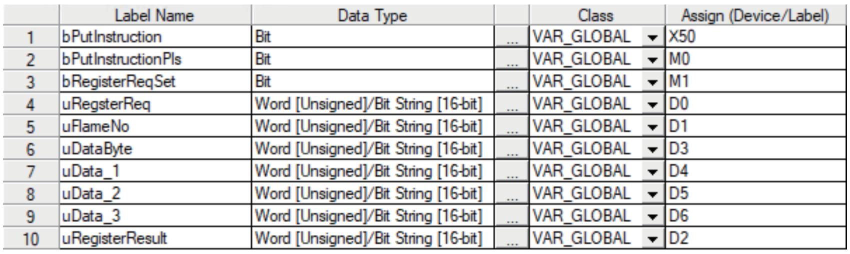
☞ Page 663 Parameter settings of C24 mounted on a redundant extension base unit

2. Write the set parameters to the module.

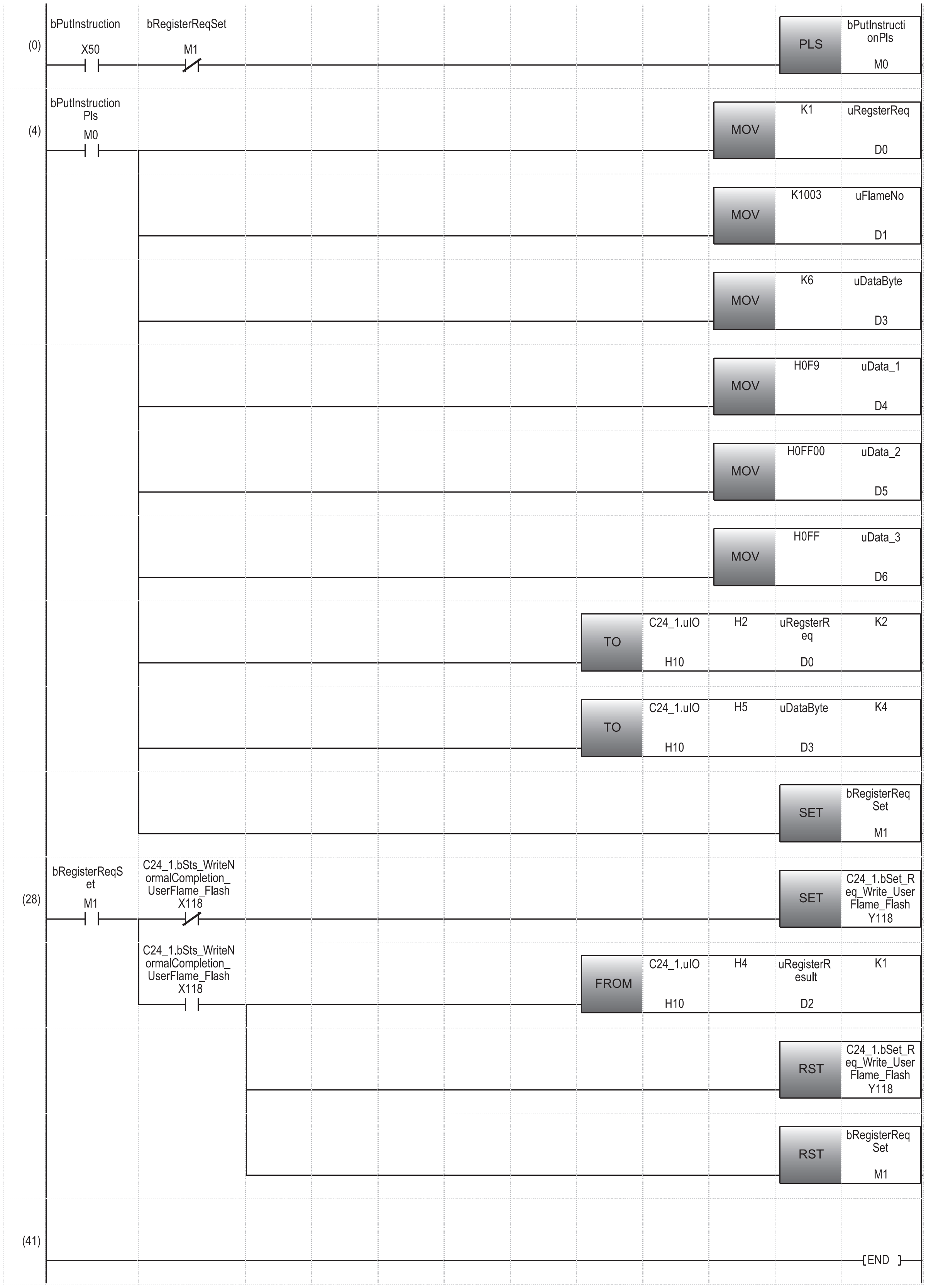
[Online] ⇨ [Write to PLC]

3. Reset the CPU module or turn the power OFF and ON.

■Program example

Category	Label name	Description	Device
Module label	C24_1.uIO	Start I/O No.	—
	C24_1.bSts_WriteNormalCompletion_UserFlame_Flash	User frame write completion to flash ROM	X118
	C24_1.bSet_Req_Write_UserFlame_Flash	User frame write request to flash ROM	Y118
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bPutInstruction	Bit	...	VAR_GLOBAL	X50
2	bPutInstructionPls	Bit	...	VAR_GLOBAL	M0
3	bRegisterReqSet	Bit	...	VAR_GLOBAL	M1
4	uRegsterReq	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
5	uFlameNo	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
6	uDataByte	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D3
7	uData_1	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D4
8	uData_2	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D5
9	uData_3	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D6
10	uRegisterResult	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2



(0) Executes the program when 'X50' turns ON.

(4) Writes to the C24 buffer memory 2H (register/read/delete instructions), 3H (frame No. direction), 5H (number of registered data bytes), and 6H (user frame).

(28) Turns 'User frame write request to flash ROM' ON.
Reads the C24 buffer memory 4H (register/read/delete results) and turns 'User frame write request to flash ROM' OFF at the completion of writing.

Alternative program example of the GETE instruction

The following shows the program example to read a user frame.

■Overview

The user frame No.3EBH (1003) is read.

■Parameter settings

Set parameters in an engineering tool as follows:

1. Set the following parameters.

☞ Page 663 Parameter settings of C24 mounted on a redundant extension base unit

2. Set the module extended parameters.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71C24] ⇨ [Module Extended Parameter]

Item			Description
User frame contents	User frame	Registration frame No.1003	F90000FFFF00

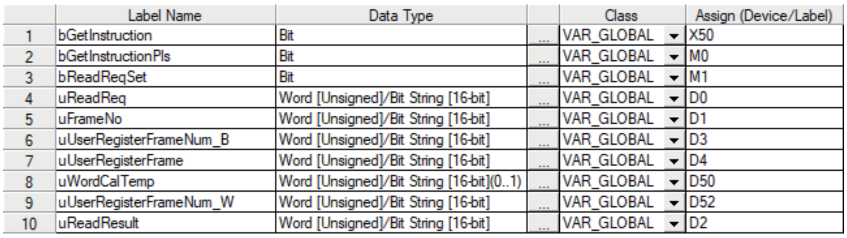
3. Write the set parameters to the module.

[Online] ⇨ [Write to PLC]

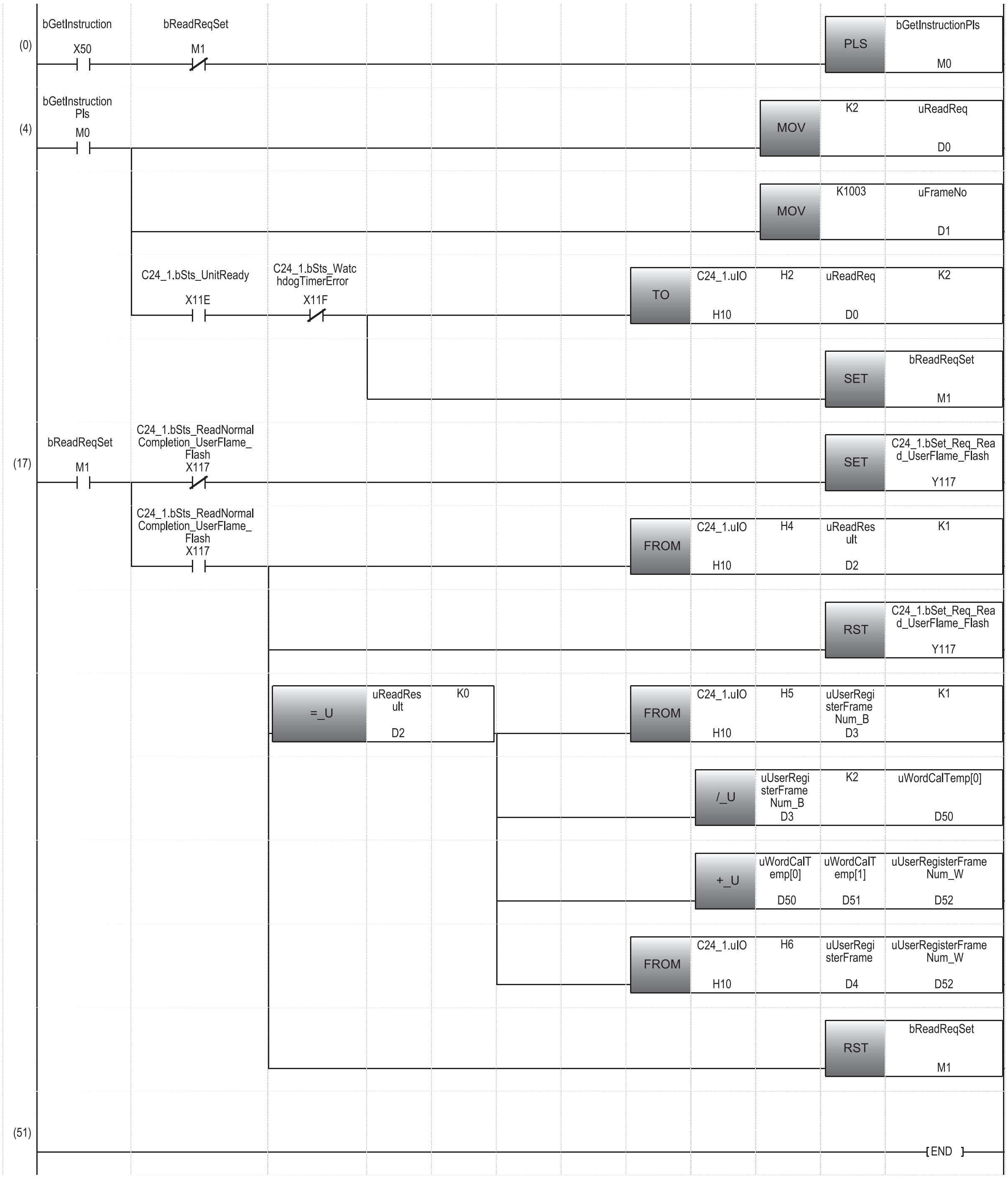
- When writing module extended parameters, an engineering tool must be connected to the control system.
- Specify "Intelligent Function Module" as the writing destination of module extended parameters.

4. Reset the CPU module or turn the power OFF and ON.

■Program example

Category	Label name	Description	Device
Module label	C24_1.uIO	Start I/O No.	—
	C24_1.bSts_ReadNormalCompletion_UserFlame_Flash	User frame read completion from flash ROM	X117
	C24_1.bSts_UnitReady	Module READY	X11E
	C24_1.bSts_WatchdogTimerError	Watchdog timer error	X11F
	C24_1.bSet_Req_Read_UserFlame_Flash	User frame read request from flash ROM	Y117
Label to be defined	Define the global label as shown below. 		

	Label Name	Data Type		Class	Assign (Device/Label)
1	bGetInstruction	Bit	...	VAR_GLOBAL	X50
2	bGetInstructionPls	Bit	...	VAR_GLOBAL	M0
3	bReadReqSet	Bit	...	VAR_GLOBAL	M1
4	uReadReq	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D0
5	uFrameNo	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1
6	uUserRegisterFrameNum_B	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D3
7	uUserRegisterFrame	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D4
8	uWordCalTemp	Word [Unsigned]/Bit String [16-bit](0..1)	...	VAR_GLOBAL	D50
9	uUserRegisterFrameNum_W	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D52
10	uReadResult	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2



(0) Executes the program when 'X50' turns ON.

(4) Writes to the C24 buffer memory 2H (register/read/delete instructions) and 3H (frame No. direction).

(17) Turns 'User frame read request to flash ROM' ON.
Reads the C24 buffer memory 4H (register/read/delete results) and turns 'User frame read request from flash ROM' OFF at the completion of reading.
Reads the C24 buffer memory 5H (registration data byte count designation) if the read result is '0' (normal completion).
Calculates the number of words of registration data for FROM instruction.
Reads the C24 buffer memory 6H (user frame).

Alternative program example of the UINI instruction

For the program examples of switching serial communication module mode and transmission specifications, refer to the following section.

☞ Page 415 Mode Switching Method by Program

Appendix 10 Added and Changed Functions

This section shows the added and changed functions of C24.

Added/changed contents	Software version of an engineering tool	Firmware version	Reference
A redundant system is supported for remote head modules.	—	'07' or later	—
The MODBUS slave function is supported.	—	'11' or later	Page 112 MODBUS Slave Function
The following function is supported by the predefined protocol function: • Redundant system	—	'13' or later	—
The following functions are supported by the MODBUS slave function: • Redundant system • Interlink operation • Mode switching using the dedicated instruction (UINI) • Mode switching using the MC protocol			Page 112 MODBUS Slave Function Page 409 SWITCHING THE MODE AFTER STARTING Page 580 Dedicated Instructions
The following function is supported by the MODBUS slave function: • Access target specification for a CC-Link IE Field Network remote head module	'1.065T' or later		Page 112 MODBUS Slave Function
The simple CPU communication function is supported. • Supported model: RJ71C24[*1], RJ71C24-R2 • Communication destination device: MELSEC-A/AnS (CPU COM)			Page 118 DATA COMMUNICATION IN SIMPLE CPU COMMUNICATION
A redundant extension base unit is supported.	'1.070Y' or later	'14' or later	Page 661 Using the Module in the Redundant System with Redundant Extension Base Unit
The simple CPU communication function is supported. • Supported model: RJ71C24(CH2), RJ71C24-R4	'1.075D' or later	'15' or later	Page 118 DATA COMMUNICATION IN SIMPLE CPU COMMUNICATION
The following communication destination devices are supported by the simple CPU communication function: • MODBUS (RTU)-compatible device • MODBUS (ASCII)-compatible device			

[*1] Only CH1 (RS-232) is supported.

INDEX

REVISIONS

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

Revision date	*Manual number	Description
June 2014	SH(NA)-081251ENG-A	First edition
September 2014	SH(NA)-081251ENG-B	Partial correction
June 2015	SH(NA)-081251ENG-C	■Added or modified parts Section 7.1, Section 26.2, Section 27.4
May 2016	SH(NA)-081251ENG-D	■Added or modified parts TERMS, Appendix 7, Appendix 8
December 2016	SH(NA)-081251ENG-E	■Added or modified parts SAFETY PRECAUTIONS, Section 3.1, Section 27.4
October 2019	SH(NA)-081251ENG-F	■Added function MODBUS slave function ■Added or modified parts SAFETY PRECAUTIONS, RELEVANT MANUALS, TERMS, Chapter 1, Chapter 5, Chapter 12, Chapter 20, Section 20.1, Section 20.2, Section 20.3, Section 22.2, Chapter 23, Section 27.2, Section 28.1, Section 28.3, Section 28.4, Section 28.5, Section 28.6, Appendix 3, Appendix 4, Appendix 7, Appendix 8
May 2020	SH(NA)-081251ENG-G	■Added function Simple CPU communication function ■Added or modified parts Chapter 5, Section 5.1, Section 5.2, Section 5.3, Chapter 6, Chapter 21, Section 21.2, Section 23.2, Section 28.2, Section 28.3, Section 29.1, Section 29.2, Section 29.3, Section 29.4, Appendix 3, Appendix 4, Appendix 7, Appendix 8, Appendix 9
October 2020	SH(NA)-081251ENG-H	■Added or modified parts SAFETY PRECAUTIONS, CONDITIONS OF USE FOR THE PRODUCT, TERMS, Section 3.1, Section 3.2, Section 4.1, Section 4.2, Section 5.5, Section 5.6, Section 9.5, Section 10.5, Section 16.4, Section 17.3, Section 17.5, Section 18.4, Section 18.6, Section 19.4, Section 19.6, Chapter 24, Section 27.4, Section 27.6, Section 29.4, Appendix 8, Appendix 9, Appendix 10
April 2021	SH(NA)-081251ENG-I	■Added or modified parts Section 5.1, Chapter 6, Section 28.3, Section 29.3, Appendix 10
May 2022	SH(NA)-081251ENG-J	■Added or modified parts SAFETY PRECAUTIONS, Section 3.1, Section 3.2, Section 3.3, Section 4.1, Section 4.2, Section 4.4, Section 9.3, Section 10.5, Appendix 2, Appendix 10
August 2022	SH(NA)-081251ENG-K	■Added or modified parts Section 6.5, COPYRIGHTS

Japanese manual number: SH-081249-K

This manual confers no industrial property rights or any rights of any other kind, nor does it confer any patent licenses. Mitsubishi Electric Corporation cannot be held responsible for any problems involving industrial property rights which may occur as a result of using the contents noted in this manual.

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 to:

(1) Damages caused by any cause found not to be the responsibility of Mitsubishi.

(2) Loss in opportunity, lost profits incurred to the user by Failures of Mitsubishi products.

(3) Special damages and secondary damages whether foreseeable or not, compensation for accidents, and compensation for damages to products other than Mitsubishi products.

(4) 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.

TRADEMARKS

The company names, system names and product names mentioned in this manual are either registered trademarks or trademarks of their respective companies.

In some cases, trademark symbols such as '™' or '®' are not specified in this manual.

COPYRIGHTS

This Product uses the Source Code of μT-Kernel under μT-License granted by the TRON Forum (www.tron.org).

SH(NA)-081251ENG-K(2208)KWIX

MODEL: R-C24-U-OU-E

MODEL CODE: 13JX21

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.

Specifications subject to change without notice.