MITSUBISHI ELECTRIC

Programmable Controller

MELSEC iQ-R series

MELSEC iQ-R Flexible High-Speed I/O Control Module User's Manual (Application)

-RD40PD01

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 of the programmable controller system, refer to 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 "General Safety Requirements" in 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.

[Design Precautions]

⚠ WARNING

- 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.
- Do not write any data to the "system area" and "write prohibited 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 prohibited 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.
- Outputs may remain on or off due to a failure of a transistor for external output. Configure an external circuit for monitoring output signals that could cause a serious accident.
- When changing data and operating status of the running module from an external device such as a personal computer connected, configure an interlock circuit external to the programmable controller to ensure that the entire system always operates safely.
  In addition, before performing online operations, determine corrective actions to be taken between the external device and the module in case of a communication failure due to poor contact of cables.

[Design Precautions]

⚠ CAUTION

- 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 reset the CPU module while the settings are being written. Doing so will make the data in the flash ROM and SD memory card undefined. The values need to be set in the buffer memory and written to the flash ROM and SD memory card again. Doing so also 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.
- Do not install the control lines or communication cables together with the main circuit lines or power cables. Keep a distance of 150mm or more between them. Failure to do so may result in malfunction due to noise.

[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 Safety Guidelines (IB-0800525). 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.
- 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.
- 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 and an included extension connector protective cover to the unused extension cable connector before powering on the system for operation. Failure to do so may result in electric shock.
- When connecting a differential output terminal to a differential receiver of a drive unit, connect the high-speed output common terminal to the differential receiver common terminal of the drive unit. Failure to do so may cause the module to fail or malfunction because of the potential difference that occurs between the high-speed output common terminal and the differential receiver common terminal.

[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.
- 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.
- A protective film is attached to the top of the module to prevent foreign matter, such as wire chips, from entering the module during wiring. Do not remove the film during wiring. Remove it for heat dissipation before system operation.
- 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.
- Do not install the control lines or communication cables together with the main circuit lines or power cables. Keep a distance of 150mm or more between them. Failure to do so may result in malfunction due to noise.
- Ground the shielded cables on the encoder side (relay box) with a ground resistance of 100 ohms or less. Failure to do so may cause malfunction.

[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 and 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) 25cm or more 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
- 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.

[Startup and Maintenance Precautions]

⚠ CAUTION

- 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 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. Failure to do so may cause the module to fail or malfunction.
- When performing online operations of the running module from an external device such as a personal computer connected, read the relevant manuals carefully and ensure that the operation is safe before proceeding.
- Before changing any setting of the module, read the relevant manuals carefully, ensure the safety, and change the operating status of the CPU module to STOP.
  Especially when operating the module in the network system, ensure the safety thoroughly because controlled machines are likely to be moved inadvertently. Improper operation may damage machines or cause accidents.

[Operating 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 and SD memory card undefined. The values need to be set in the buffer memory and written to the flash ROM and SD memory card again. Doing so can cause malfunction or failure of the module.

[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 specifications, procedures before operation, wiring, and programming of the relevant products listed below.
Before using this product, please read this manual and the relevant manuals carefully and develop familiarity with the functions and performance of the MELSEC iQ-R series programmable controller to handle the product correctly.
When applying the program examples 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

Unless otherwise specified, this manual provides program examples in which the I/O numbers of X/Y0 to X/Y1F are assigned to the flexible high-speed I/O control module. Assign I/O numbers when applying the program examples to an actual system. For I/O number assignment, refer to the following.
MELSEC iQ-R Module Configuration Manual

Relevant product

RD40PD01

CONTENTS

RELEVANT MANUALS

Manual name [manual number]	Description	Available form
MELSEC iQ-R Flexible High-Speed I/O Control Module User's Manual (Application) [SH-081647ENG] (this manual)	Functions, parameter settings, I/O signals, buffer memory, and troubleshooting of the flexible high-speed I/O control module, and details on the configuration tool	Print book e-Manual PDF
MELSEC iQ-R Module Configuration Manual [SH-081262ENG]	Combinations of MELSEC iQ-R series modules to be used, common information on installations/wiring for configuring systems, and specifications of the power supply module, base unit, SD memory card, and battery	Print book e-Manual PDF
MELSEC iQ-R Flexible High-Speed I/O Control Module User's Manual (Startup) [SH-081645ENG]	System configuration, specifications, procedures before operation, wiring, and operation examples of the flexible high-speed I/O control module	Print book e-Manual PDF
MELSEC iQ-R Programming Manual (Module Dedicated Instructions) [SH-081976ENG]	Dedicated instructions for the intelligent function modules	e-Manual PDF
MELSEC iQ-R Flexible High-Speed I/O Control Module Function Block Reference [BCN-P5999-0684]	FBs of the flexible high-speed I/O control modules	e-Manual PDF
GX Works3 Operating Manual [SH-081215ENG]	System configuration, parameter settings, and online operations of GX Works3	e-Manual PDF

Point

e-Manual refers to the Mitsubishi 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.

Term	Description
Buffer memory	A memory in an intelligent function module for storing data (such as setting values and monitored values). For a CPU module, it refers to a memory for storing data (such as setting values and monitored values of the Ethernet function, data used for data communications of the multiple CPU system function).
Configuration tool	The abbreviation for the configuration tool for flexible high-speed I/O control modules
Engineering tool	A tool used for setting up programmable controllers, programming, debugging, and maintenance
External wiring	Wiring between a flexible high-speed I/O control module and external devices
Flash ROM	A non-volatile memory to which the hardware logic can be written. Although the number of writes to a flash ROM is limited, the written hardware logic is not deleted even after the power off and automatically read at the power-on.
Global label	A label that is valid for all the program data when multiple program data are created in the project. The global label has two types: a module specific label (module label), which is generated automatically by GX Works3, and an optional label, which can be created for any specified device.
Hardware logic	A control logic that users create graphically combining inputs, outputs, logical operation circuits, and counters with the configuration tool
Link	Connecting blocks or terminals on the configuration tool
Module label	A label that represents one of memory areas (I/O signals and buffer memory areas) specific to each module in a given character string. For the module used, GX Works3 automatically generates this label, which can be used as a global label.
Redundant system with redundant extension base unit	A redundant system that is configured using extension base unit(s)
Watchdog timer error	An error that occurs if the internal processing of the module is abnormal. Watchdog timer enables the module to monitor its own internal processing.

GENERIC TERMS AND ABBREVIATIONS

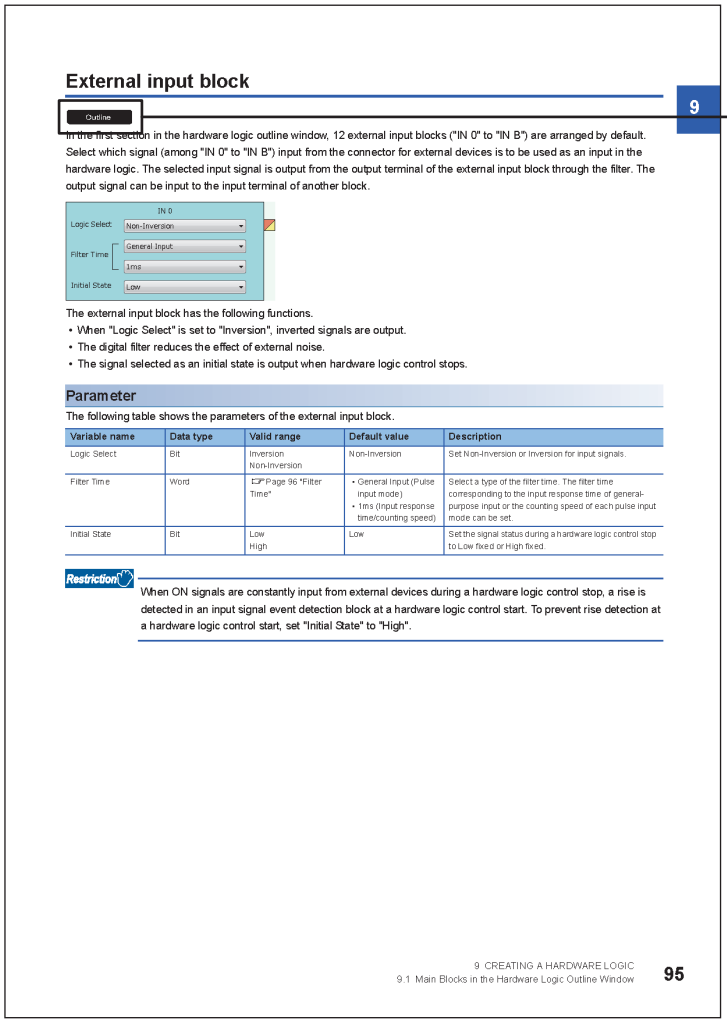
Unless otherwise specified, this manual uses the following generic terms and abbreviations.

Generic term/abbreviation	Description
Flexible high-speed I/O control module	The abbreviation for the MELSEC iQ-R series flexible high-speed I/O control module

MANUAL PAGE ORGANIZATION

Pages describing the hardware logic are organized as shown below.

The following illustration is for explanation purpose only, and should not be referred to as an actual documentation.



External input block

Outline

9

In the first section in the hardware logic outline window, 12 external input blocks ("IN 0" to "IN B") are arranged by default. Select which signal (among "IN 0" to "IN B") input from the connector for external devices is to be used as an input in the hardware logic. The selected input signal is output from the output terminal of the external input block through the filter. The output signal can be input to the input terminal of another block.

IN 0
Logic Select: Non-Inversion
Filter Time: General Input / 1ms
Initial State: Low

The external input block has the following functions.

- When "Logic Select" is set to "Inversion", inverted signals are output.
- The digital filter reduces the effect of external noise.
- The signal selected as an initial state is output when hardware logic control stops.

Parameter

The following table shows the parameters of the external input block.

Variable name	Data type	Valid range	Default value	Description
Logic Select	Bit	Inversion Non-Inversion	Non-Inversion	Set Non-Inversion or Inversion for input signals.
Filter Time	Word	Page 96 "Filter Time"	• General Input (Pulse input mode) • 1ms (Input response time/counting speed)	Select a type of the filter time. The filter time corresponding to the input response time of general-purpose input or the counting speed of each pulse input mode can be set.
Initial State	Bit	Low High	Low	Set the signal status during a hardware logic control stop to Low fixed or High fixed.

Restriction

When ON signals are constantly input from external devices during a hardware logic control stop, a rise is detected in an input signal event detection block at a hardware logic control start. To prevent rise detection at a hardware logic control start, set "Initial State" to "High".

9 CREATING A HARDWARE LOGIC
9.1 Main Blocks in the Hardware Logic Outline Window
95

An icon displayed here indicates the window where the terminal or block is used.

The meaning of each icon is as follows.

Icon	Description
Outline	The "Outline" icon indicates that the terminal or block is the one for the hardware logic outline window.
Multi function	The "Multi function" icon indicates that the terminal or block is the one for multi function counter block detail windows.

1 FUNCTIONS

This chapter describes the details on the functions that can be used in the flexible high-speed I/O control module and their setting methods.

1.1 Modes

The flexible high-speed I/O control module has the normal operation mode and inter-module synchronous mode. Change the mode according to the function used.

Normal operation mode

This mode is for performing the hardware logic control.

Inter-module synchronous mode

This mode is for synchronizing the control timings of various functions of multiple modules according to the inter-module synchronization cycle.

Mode transitions

The mode is determined depending on whether the flexible high-speed I/O control module is selected for a synchronization target module in GX Works3 after the power-on or the CPU module is reset.

[System Parameter] ⇨ "Inter-module Synchronization Setting" ⇨ "Inter-module Synchronization Setting" ⇨ "Select Inter-module Synchronization Target Module"

Select Inter-module Synchronization Target Module	Mode
"Do not Synchronize" (default value)	Normal operation mode
"Synchronize"	Inter-module synchronous mode

Update timings of the X/Y signals and Hardware logic area

The update timings of the X/Y signals and Hardware logic area (Un\G1000 to Un\G1099) differ depending on the mode. The following table lists the update timing in each mode.

Device		Update timing	
		Inter-module synchronous mode	Normal operation mode
X signal	X0 to XF	1ms	100μs
	X10 to X1F	When the inter-module synchronization cycle is started	
Y signal	Y0 to YF	1ms	
	Y10 to Y1F	Immediately after the completion of a synchronous interrupt program	
Hardware logic area (High speed area) (Un\G1000 to Un\G1029)	Control data	Immediately after the completion of a synchronous interrupt program	100μs
	Monitor data	1ms	
Hardware logic area (Low speed area) (Un\G1030 to Un\G1099)	Control data	1ms	1ms
	Monitor data		

1.2 Hardware Logic Control Function

Users can create the hardware logic to perform a desired control with the configuration tool.

For details, refer to the following.

☞ Page 144 CREATING A HARDWARE LOGIC

1.3 Continuous Logging Function

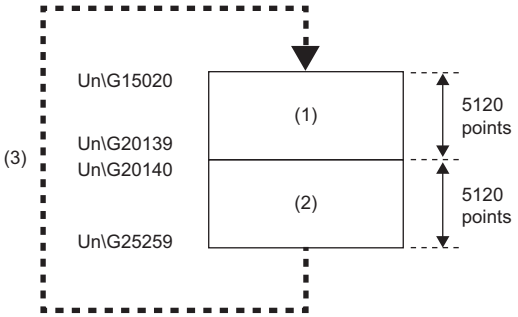
This function collects the status of the Output terminal of the external input blocks (IN 0 to IN B) continuously at a specified interval.

Collecting logging data

Logged data is stored in Continuous logging data storage area (Un\G15020 to Un\G25259).

■Storing data to the buffer memory

Logged data is temporarily saved in the module and stored in Continuous logging data storage area (Un\G15020 to Un\G25259), from the start address in increments of 64 points.

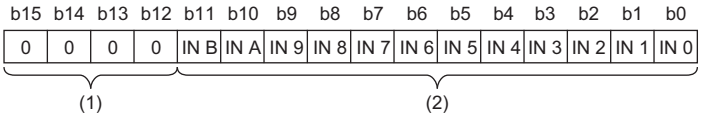


(1) Continuous logging data storage area (A side) (5120 points)

(2) Continuous logging data storage area (B side) (5120 points)

(3) After 5120 points of data are stored in the B side, the data in the A side is overwritten from its start address.

In logging data of 1 point, the High/Low state of the Output terminal of IN 0 to IN B is stored as follows.



(1) 0 is stored in b12 to b15.

(2) The High/Low state of each terminal is stored.

■Notification of logging data storage

- After logging data for 5120 points is stored in the logging data storage areas (A side), the value in 'Continuous logging data A side storage flag' (Un\G15012) is changed from Storage not completed (0) to Storage completed (1).
- After logging data for 5120 points is stored in the logging data storage areas (B side), the value in 'Continuous logging data B side storage flag' (Un\G15013) is changed from Storage not completed (0) to Storage completed (1).

■Transfer of logging data

Logging data can be transferred into the file register (ZR) of the CPU module with a function block (FB). Thus, continuous logging data for 10240 points or more can be collected. (☞ Page 24 Transfer of continuous logging data)

Starting continuous logging

1. Write the hardware logic to the module and start hardware logic control. When the hardware logic control starts, the value in 'Continuous logging status monitor' (Un\G15010) changes from Disabled (0) to Start request waiting (1).
2. Set the continuous logging cycle (1μs (0) to 1000μs (3)) in 'Continuous logging cycle setting' (Un\G15008).
3. Changing the value in 'Continuous logging start/stop request' (Un\G15009) from Stop request (0) to Start request (1) starts the continuous logging on the set continuous logging cycle. At this time, the value in 'Continuous logging status monitor' (Un\G15010) changes from Start request waiting (1) to In progress (2).

At the start of continuous logging, the values stored in the following buffer memory areas are initialized (cleared to 0).

- 'Continuous logging data A side storage flag' (Un\G15012)
- 'Continuous logging data B side storage flag' (Un\G15013)
- 'Continuous logging data points' (Un\G15014, Un\G15015)

■When a value out of range is set in Continuous logging cycle setting

When a value out of range is set in 'Continuous logging cycle setting' (Un\G15008) and the value in 'Continuous logging start/stop request' (Un\G15009) is changed from Stop request (0) to Start request (1), the continuous logging cycle setting range error (error code: 10D6H) occurs and the continuous logging does not start.

■Disabling continuous logging

When Disabled (0) is stored in 'Continuous logging status monitor' (Un\G15010) and the value in 'Continuous logging start/stop request' (Un\G15009) is changed from Stop request (0) to Start request (1), an error occurs and the continuous logging does not start.

When one of the following conditions is satisfied, Disabled (0) is stored in 'Continuous logging status monitor' (Un\G15010).

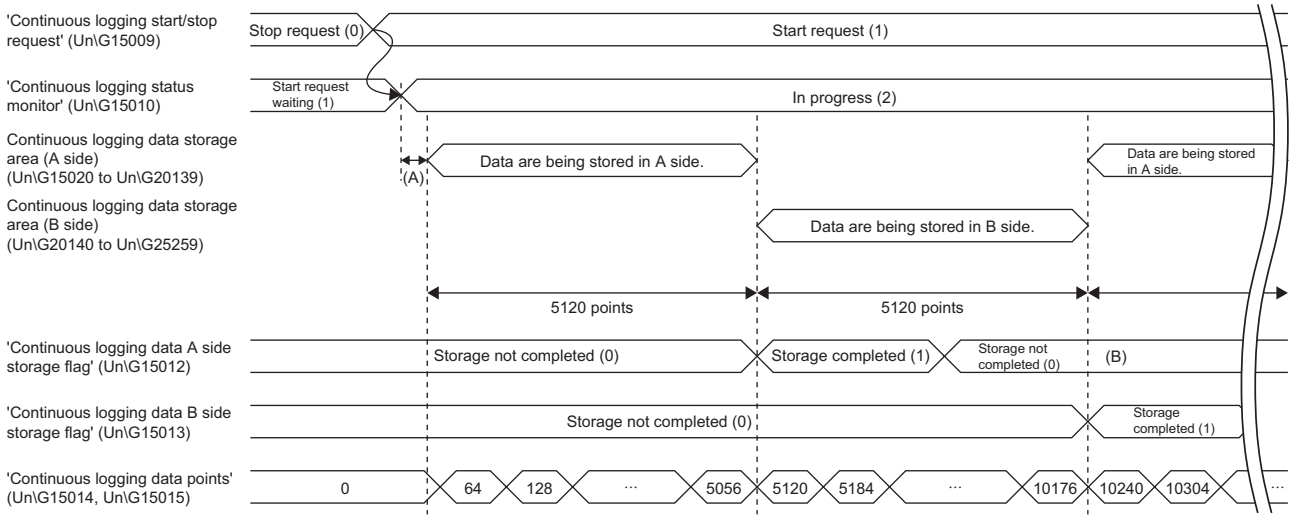
- During control of a hardware logic that uses Hardware logic area (High speed area) (Un\G1000 to Un\G1029)
- During control of a hardware logic that uses an SSI encoder block
- During simulation
- During logic analyzer function execution
- The flexible high-speed I/O control module is set as a target of inter-module synchronization.
- During hardware logic control stop
- A/D conversion value logging in progress

Operation of continuous logging

When logging data is stored in Continuous logging data storage area (Un\G15020 to Un\G25259), 'Continuous logging data points' (Un\G15014, Un\G15015) is refreshed for each data storage.

'Continuous logging data points' (Un\G15014, Un\G15015) indicates the number of data stored in Continuous logging data storage area (Un\G15020 to Un\G25259).

After data for 5120 points is stored in continuous logging data storage areas, the value in 'Continuous logging data A side storage flag' (Un\G15012) or 'Continuous logging data B side storage flag' (Un\G15013) is changed from Storage not completed (0) to Storage completed (1).



(A) The data is stored in Continuous logging data storage area (Un\G15020 to Un\G25259) "1024 × 'Continuous logging cycle monitor (μs)' (Un\G15011)" (μs) behind the actual continuous logging start timing.

(B) After the logging data is read, the program sets 0 in 'Continuous logging data A side storage flag' (Un\G15012) or 'Continuous logging data B side storage flag' (Un\G15013).

■Counting the continuous logging data points

After counting up to 3600000000, 'Continuous logging data points' (Un\G15014, Un\G15015) increases by 64 starting from 0 again.

Ex.

0→64→128→ to →3599999936→3600000000→64→ to

Point

When 1μs (0) is set in 'Continuous logging cycle setting' (Un\G15008), the maximum value 3600000000 of 'Continuous logging data points' (Un\G15014, Un\G15015) indicates that logging of 3600 seconds (1 hour) has been completed.

■When a trigger start request of the logic analyzer function is issued during execution of continuous logging

The trigger start error during continuous logging (error code: 10D5H) is output and the trigger start request is ignored. The continuous logging does not stop.

Stopping continuous logging

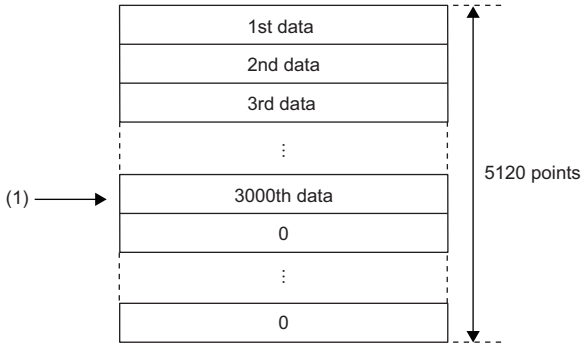
Changing the value in 'Continuous logging start/stop request' (Un\G15009) from Start request (1) to Stop request (0) stops continuous logging. After continuous logging is stopped, the data that was partially logged is stored in buffer memory areas and the value in 'Continuous logging data points' (Un\G15014, Un\G15015) is refreshed to the number of data points that was logged by the time the stop request was accepted. In addition, 0 is stored in the data areas later than the acceptance of the stop request.

Continuous logging stops in either of the following conditions as well.

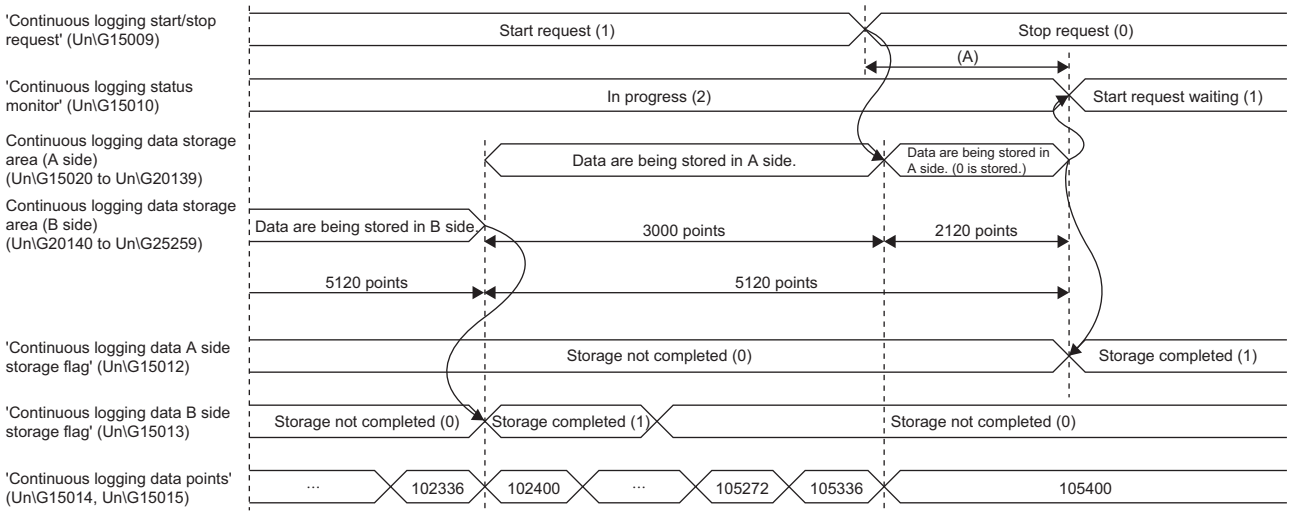
- When simulation is executed during execution of continuous logging
- When hardware logic control stops during execution of continuous logging

Ex.

A continuous logging stop request is accepted when data for 105400 points has been logged after continuous logging started.



(1) A stop request is accepted.



(A) Even though the stop request is issued, continuous logging does not stop immediately. Continuous logging stops when the data for 5120 points has been stored in the continuous logging data storage area in which data is being stored.

■Restarting continuous logging

To restart continuous logging after stopping it, perform the following operations.

1. Check that the value in 'Continuous logging status monitor' (Un\G15010) is Start request waiting (1).
2. Change the value in 'Continuous logging start/stop request' (Un\G15009) from Stop request (0) to Start request (1).
   When the value in 'Continuous logging status monitor' (Un\G15010) is In progress (2), the start request is ignored.

After continuous logging is restarted, values are stored in Continuous logging data storage area (A side) (Un\G15020 to Un\G20139) from the start address. When continuous logging is restarted, the following buffer memory areas are initialized (cleared to 0).

- 'Continuous logging data A side storage flag' (Un\G15012)
- 'Continuous logging data B side storage flag' (Un\G15013)
- 'Continuous logging data points' (Un\G15014, Un\G15015)

The values before the restart of continuous logging are held in Continuous logging data storage area (Un\G15020 to Un\G25259). Thus, before referring to the logging data, check that Storage completed (1) has been stored in 'Continuous logging data A side storage flag' (Un\G15012) again.

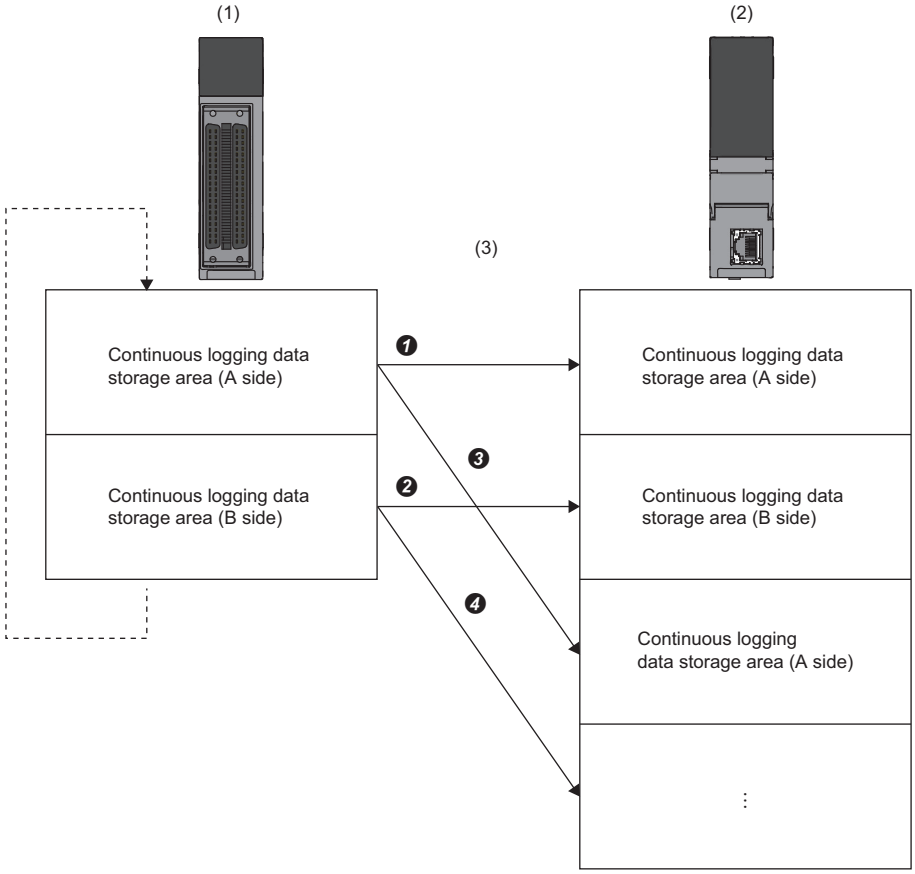
Transfer of continuous logging data

Logging data can be transferred into the file register (ZR) of the CPU module by using the following items.

- Function block for continuous logging data transfer (M+RD40PD01_ReadContinuousLogging)
- Interrupt function

Transfer of data with the function block

The function block for continuous logging data transfer continuously transfers logging data into the file register (ZR) of the CPU module in the storage order of logging data (A side → B side → A side ···). Set the number of data points to be transferred into the CPU module considering the capacity of the file register of the CPU module in use.



(1) Flexible high-speed I/O control module (buffer memory)

(2) CPU module (file register (ZR))

(3) The function block transfers data into the file register in the order of ❶→❷→❸→❹···.

Point

- For the file register capacity setting, refer to the MELSEC iQ-R CPU Module User's Manual (Application).
- For details on the function block for continuous logging data transfer, refer to the MELSEC iQ-R Flexible High-Speed I/O Control Module Function Block Reference.

■Program example to transfer data with the function block

Ex.

To start the continuous logging and copy logging data for 102400 points and register them in the file register (ZR0 to ZR102399) of the CPU module continuously

• Label settings

Classification	Label name	Description	Device
Module label	RD40PD01_1.uContinuousLoggingStartStopRequest_D	Continuous logging start/stop request	U0\G15009
Label to be defined	Define global labels as shown below:		

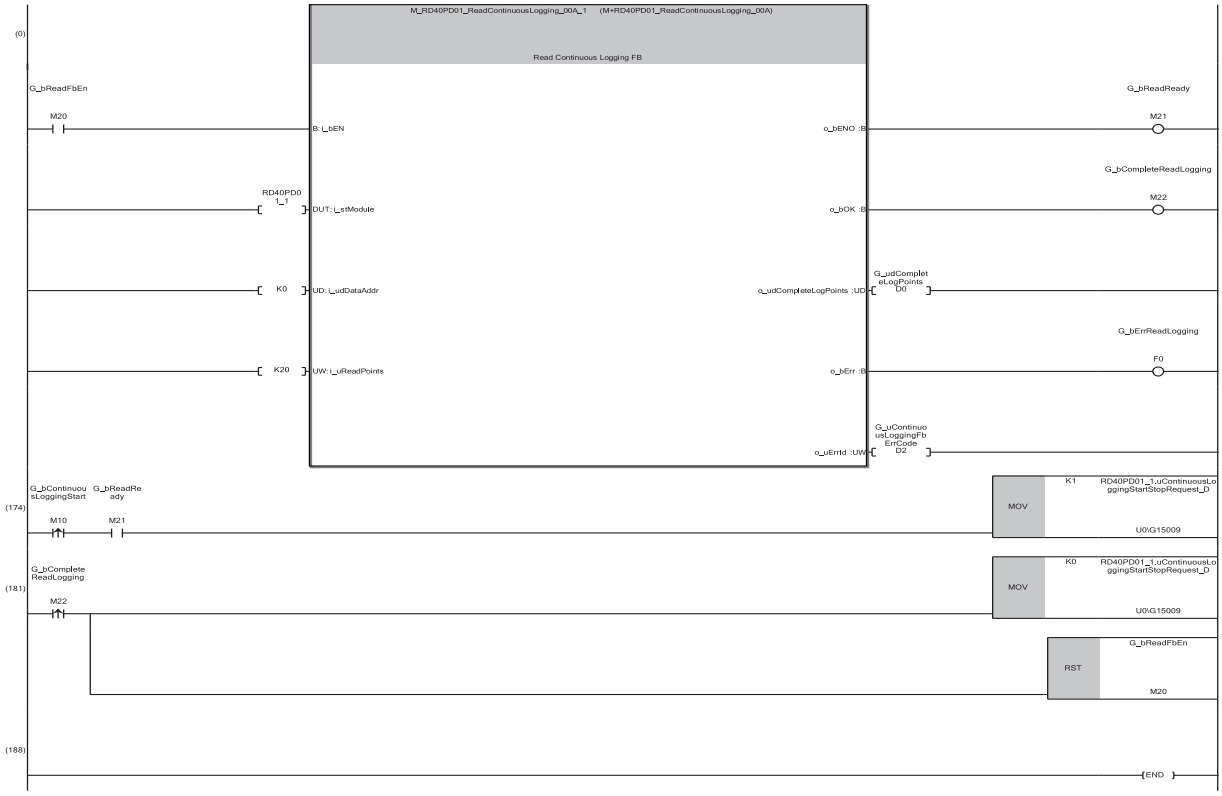
	Label Name	Data Type		Class	Assign (Device/Label)
1	G_bContinuousLoggingStart	Bit	...	VAR_GLOBAL	M10
2	G_bReadFbEn	Bit	...	VAR_GLOBAL	M20
3	G_bReadReady	Bit	...	VAR_GLOBAL	M21
4	G_bCompleteReadLogging	Bit	...	VAR_GLOBAL	M22
5	G_udCompleteLogPoints	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D0
6	G_bErrReadLogging	Bit	...	VAR_GLOBAL	F0
7	G_uContinuousLoggingFbErrCode	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D2

• The function block used

The following function block is used in this program example.

M+RD40PD01_ReadContinuousLogging

• Program example



(0) 'G_bReadFbEn' (M20) is turned on and the function block for continuous logging data transfer is started.
(190)After the FB is activated, turn on 'G_bContinuousLoggingStart' (M10) to start continuous logging.
Logging data are automatically stored in the file registers (ZR0 to ZR102399) of the CPU module by the FB.
(207)After the logging of 102400-point data is complete, stop the continuous logging and FB.
If data omission occurs during continuous logging read because of the processing time of other scan program or interrupt program, the FB turns on 'G_bErrReadLogging' (M22), and stops the read of logging data.

■Precautions for creating the program

- The function block for continuous logging data transfer reads out the data to the file register each time 5120 data points are stored in the flexible high-speed I/O control module. The approximate processing time of the function block is obtained by adding up the required time of following instruction operations used in the function block: 5.12 times of the FROM instruction (1000 points)*1, 7 times of the FROM instruction (1 point)*1, 3 times of the TO instruction*1. If the exact processing time is required, add up the processing time of all instructions used in the function block. For details on the processing time of each instruction used in the function block, refer to the MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks).

Item	Calculation formula
Processing time (as a guide) of the function block for continuous logging data transfer	= (Processing time of the FROM instruction (1000 points)) × (5.12 times) + (Processing time of the FROM instruction (1 point)) × (7 times) + (Processing time of the TO instruction (1 point)) × (3 times) = 55.6μs*1 × 5.12 + 4.3μs*1 × 7 + 2.7μs*1 × 3 = 322.872μs

- To prevent logging data omission, create the program so that the function block is executed once or more times in 3.9ms*2. (The processing time of this program must be 3.9ms*2 or shorter including the processing time of the function block.)

*1 Maximum processing time of each instruction is used for the calculation. (MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks))

*2 This value is for when the continuous logging is executed every 1μs cycle. When using the continuous logging cycle different from 1μs, calculate a reference value (ms) using the following calculation formula.
Reference value (ms) = (Continuous logging cycle (μs) × 5120 (points)) ÷ 1000 - 1.1 (ms)

This processing time is a value for the case when the program consists of only the function block for continuous logging data transfer. If other processing is added to the program, or another program is added, the processing time becomes longer. Please consider an actual application and configure the system.

Transfer of data with the interrupt function

The interrupt program is executed on the timing when the logging data for 5120 points are stored in the buffer memory. Thus, logging data can be transferred into the file register (ZR) without depending on the scan time.
For the interrupt program, do not use the function block for continuous logging data transfer because the function block does not operate properly by doing so.

■Program example to transfer data with the interrupt function

Ex.
To start the continuous logging and copy logging data for 102400 points and register them in the file register (ZR0 to ZR102399) of the CPU module continuously

• Label settings

Classification	Label name	Description	Device
Module label	RCPU.stSM.bAfter_RUN1_Scan_ON	ON for one scan after RUN	SM402
	RD40PD01_1.uContinuousLoggingStartStopRequest_D	Continuous logging start/stop request	Un\G15009
	RD40PD01_1.uContinuousLoggingASideStorageFlag_D	Continuous logging data A side storage flag	Un\G15012
	RD40PD01_1.uContinuousLoggingBSideStorageFlag_D	Continuous logging data B side storage flag	Un\G15013
Label to be defined	Define global labels as shown below: Although 'G_wSaveRegisterPlusIndex' operates without problem if it is used with the class of VAR_GLOBAL, using this label in the same way on GX Works3 of Ver.1.055H or later will cause a conversion error on the consistency check of global labels. Set the class to VAR_GLOBAL_RETAIN when using 'G_wSaveRegisterPlusIndex' on GX Works3 of Ver.1.055H or later.		

	Label Name	Data Type		Class	Assign (Device/Label)
1	G_udWriteStartPosition	Double Word [Signed]	...	VAR_GLOBAL	D10
2	G_udWritePosition	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D20
3	G_udSaveFileRegisterMaxValue	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D30
4	G_bErrorDetection	Bit	...	VAR_GLOBAL	F0
5	G_bContinuousLoggingStart	Bit	...	VAR_GLOBAL	M0
6	G_wLoggingReadSideA	Word [Signed]	...	VAR_GLOBAL	U0\G15020
7	G_wLoggingReadSideB	Word [Signed]	...	VAR_GLOBAL	U0\G20140
8	G_udWritePositionIndex	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	Z4
9	G_wSaveFileRegisterPlusIndex	Word [Signed]	...	VAR_GLOBAL_RETAIN	ZR0ZZ4

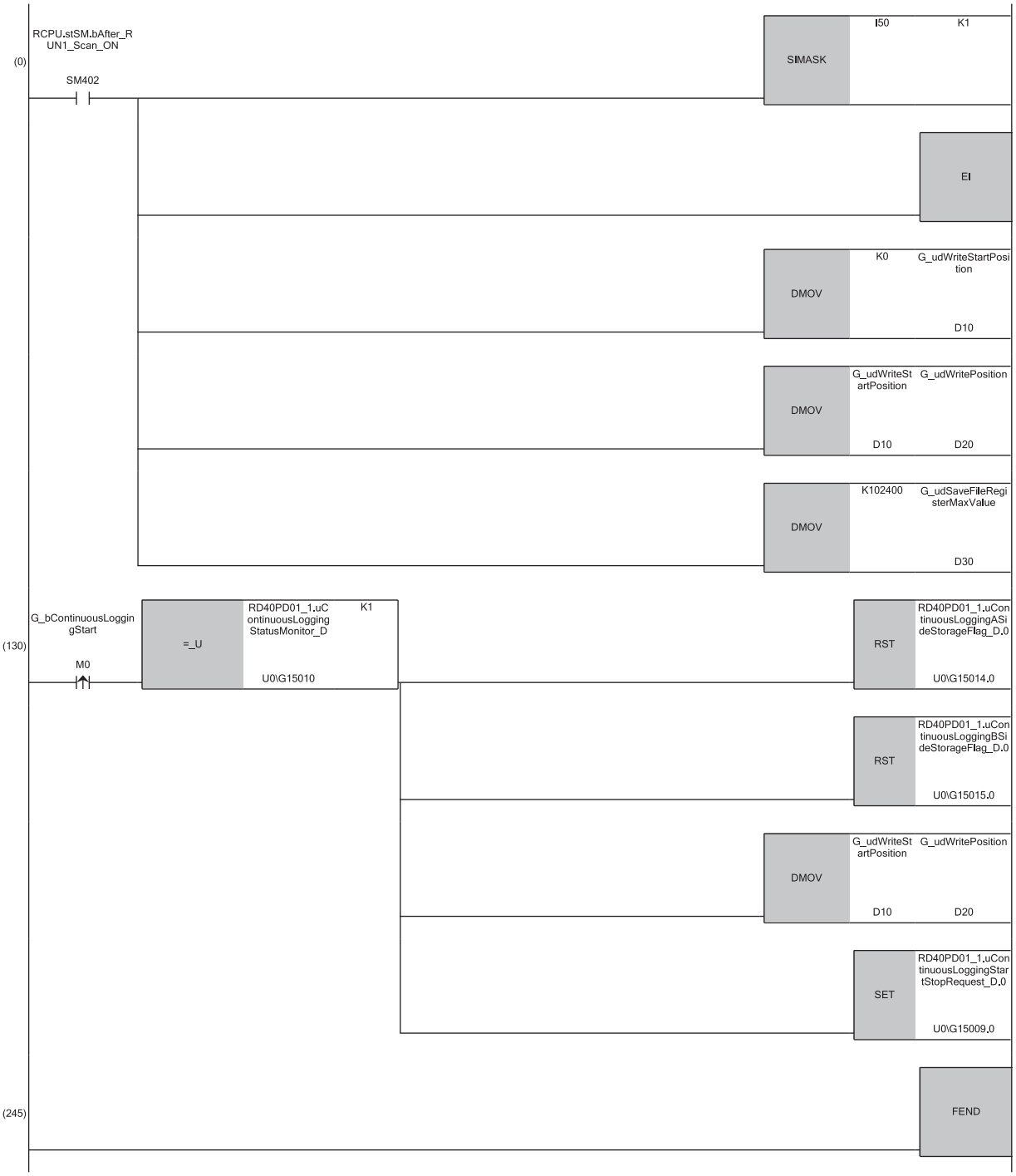
• Application setting

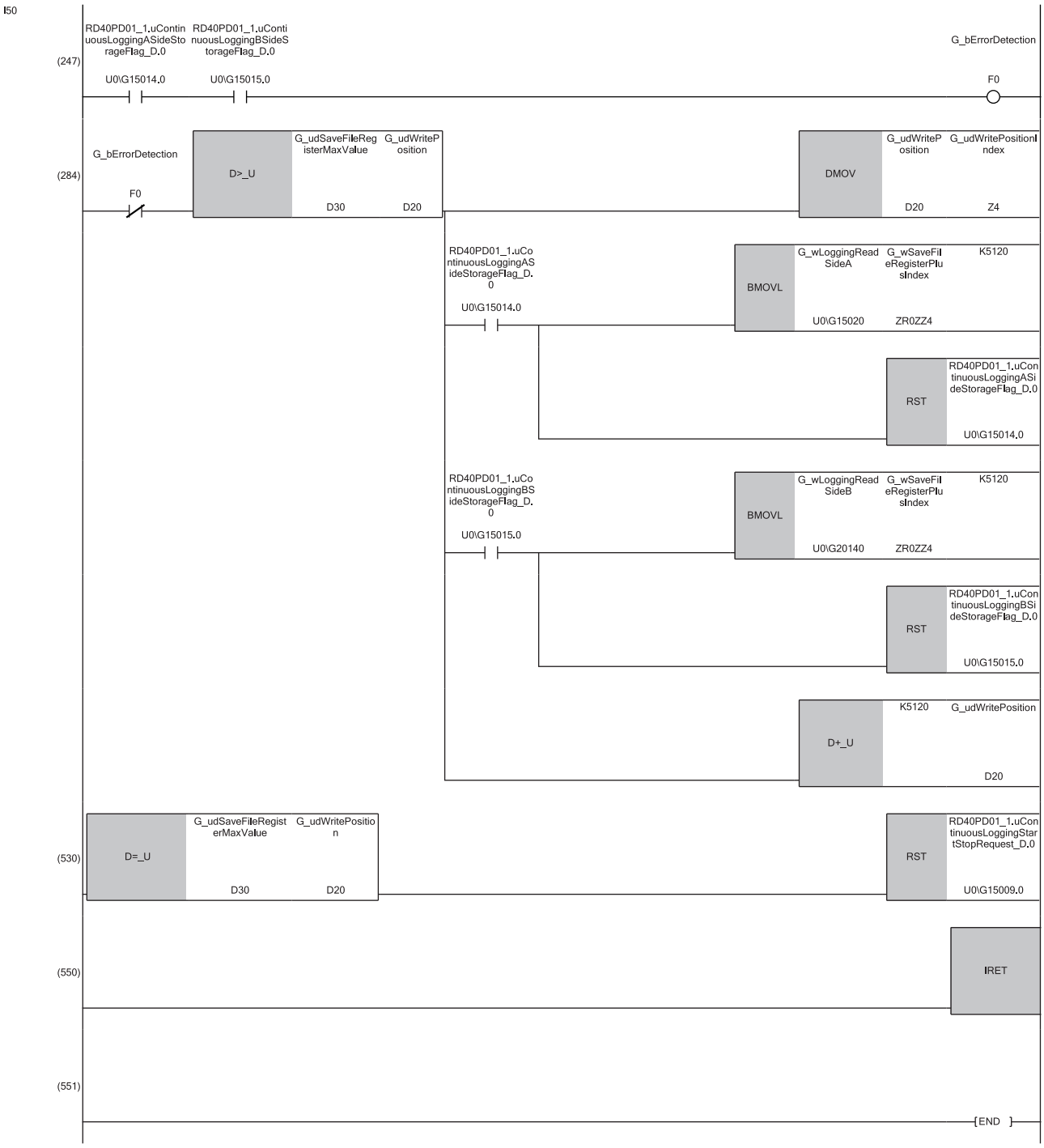
Item	Setting Value
Inter-module synchronization function	Hardware logic control function in accordance with the inter-module synchronization cycle.
Hardware logic control selection during synchronization error occurrence	Hardware logic control continue
Online module change	The module can be changed without the system being stopped.
Hardware logic control auto restoration executed/unexecuted	Enable
CC-Link IE Field Network head module installation setting	Set action when CC-Link IE Field Network head module is installed and using.
Hardware logic control at disconnection enable/disable	Disable
CPU error output mode setting	Clear
Continuous logging interrupt setting	Set wheter to send interruption requests when continuous logging data are stored.
Continuous logging data storage interrupt enable/disable	Enable

• Interrupt setting

No.	Condition target setting	Interruption pointer
1	SI device terminal 0(SI0)	
2	SI device terminal 1(SI1)	
3	SI device terminal 2(SI2)	
4	SI device terminal 3(SI3)	
5	SI device terminal 4(SI4)	
6	SI device terminal 5(SI5)	
7	SI device terminal 6(SI6)	
8	SI device terminal 7(SI7)	
9	Continuous logging data storage	I50

• Program example





(0) The settings are configured as follows by the contact turning on for one scan after RUN.
 • 'G_udWriteStartPosition' (D10) is set to 0 so that ZR0 is set as the start position of the save destination file register.
 • 'G_udWritePosition' (D20) is set to a value in 'G_udWriteStartPosition' (D10) so that the write position of the file register is initialized.
 • 'G_udSaveFileRegisterMaxValue' (D30) is set to 102400, which is the maximum number of storage data points of the save destination file register.
(130)Turn on 'G_bContinuousLoggingStart' (M0) at the desired timing to start the continuous logging.
(247)An interrupt occurs each time logging data of 5120 points are stored in a buffer memory area. Before logging data transfer, whether logging data omission has occurred is checked.
(284)The logging data are transferred to the file registers. The 5120 point logging data set are transferred per interrupt. The transfer is repeated until the data are transferred to ZR102399.
(530)When the transfer of 102400 point logging data sets (ZR0 to ZR102399) is complete, the continuous logging is stopped.
If data omission occurs during continuous logging read because of the processing time of other high-priority interrupt program, 'G_bErrorDetection' (F0) turns on at interrupt occurrence, and the read of logging data stops.

■Starting the continuous logging again

To start the continuous logging again after the completion of logging data transfer, have the execution interval (ms) described below or longer between them.

- Execution interval (ms) ≥ 5120 (points) × Continuous logging cycle monitor (μs) ÷ 1000 × 2

Ex.

When 'Continuous logging cycle monitor' (Un\G15011) is 1(μs)

- Execution interval (ms) ≥ 5120 (points) × 1(μs) ÷ 1000 × 2 = 10.24(ms)

■Precautions for creating the program

- Each time 5120 logging data points are stored in the flexible high-speed I/O control module, an interrupt program is executed and the data is transferred to the file register. The approximate instruction processing time of the interrupt program is obtained by adding up the processing time of instructions accessing the buffer memory areas in the interrupt program (equivalent to a value of "5.12 times of FROM instruction (1000 points)[*1] + 5 times of FROM instruction (1 point)[*1] + 1 time of TO instruction (1 point)[*1]") and the processing time of the IRET instruction. If the exact instruction processing time is required, add up the processing time of all instructions used in the interrupt program. For details on the processing time of each instruction used in the interrupt program, refer to the MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks).
- The approximate processing time of the interrupt program is obtained by adding both the interrupt overhead time and the refresh processing time (when the refresh processing is set) at the interrupt execution to the instruction processing time of the interrupt program. The interrupt overhead time and the refresh processing time at the interrupt execution vary depending on the parameter settings of the CPU module. For details, refer to MELSEC iQ-R CPU Module User's Manual (Application).
- The following table provides a calculation example for when the interrupt overhead time is 20.6μs and the refresh processing time at the interrupt execution is 0μs.

Item	Calculation formula
Instruction processing time (as a guide) of the interrupt program	= (Processing time of the FROM instruction (1000 points)) × (5.12 times) + (Processing time of the FROM instruction (1 point)) × (5 times) + (Processing time of the TO instruction (1 point)) × (1 time) + (Processing time of the IRET instruction) × (1 time) = 55.6μs[*1] × 5.12 + 4.3μs[*1] × 5 + 2.7μs[*1] × 1 + 2.2μs[*1] × 1 = 311.072μs
Processing time (as a guide) of the interrupt program	= (Instruction processing time (as a guide) of the interrupt program) + (Interrupt overhead time) + (Refresh processing time at the interrupt execution) = 311.072μs + 20.6μs + 0μs = 331.672μs

- To prevent logging data omission, create the program so that a series of processing of the interrupt program (from the start to the completion) takes place within 4.02ms[*2] after an interrupt factor (storage of 5120 continuous logging data points) occurs.

*1 This value indicates the maximum processing time of each instruction. (📖 MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks))

*2 This value is for when the continuous logging is executed every 1μs cycle. When using the continuous logging cycle different from 1μs, calculate a reference value (ms) using the following calculation formula.
Reference value (ms) = (Continuous logging cycle (μs) × 5120 (points)) ÷ 1000 - 1.1 (ms)

This processing time is a value for the case when the program consists of only the interrupt program which transfers continuous logging data. If other processing is added to the program, or another program is added, the processing time becomes longer. Please consider an actual application and configure the system.

Saving continuous logging data

The logging data stored in the file register (ZR) can be written into an SD memory card as a CSV file and checked with spreadsheet software.

Use the SP.FWRITE instruction to write the data into the SD memory card. For details on the SP.FWRITE instruction, refer to the following.

MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks)

Restrictions

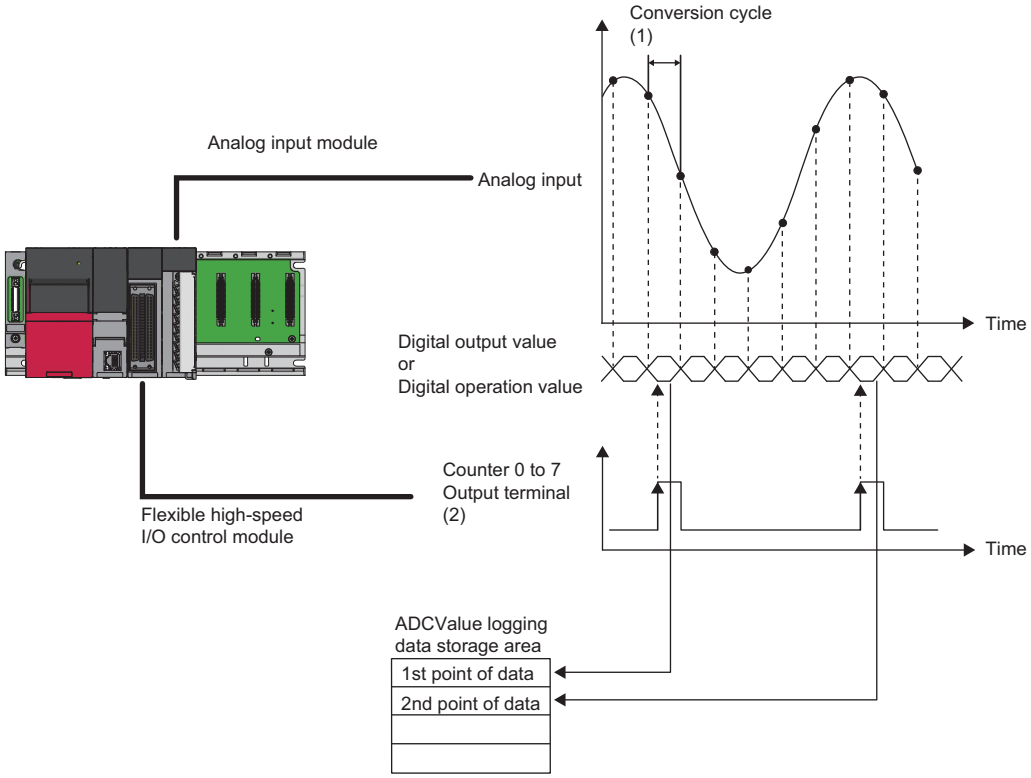
The maximum response delay time of interrupt during the execution of continuous logging is 200μs.

1.4 A/D Conversion Value Logging Function

This function logs A/D conversion values of analog input modules on the same base unit, based on the output timing of the multi function counter block.
Multiple data points can be read out at the logging data storage timing without using SI device terminals in the hardware logic or interrupt programs. This reduces the processing load on the CPU module.

Overview

The A/D conversion values of the analog input modules are logged by the flexible high-speed I/O control module. The target A/D conversion values are the digital output values or digital operation values stored in the buffer memory of the analog input modules.
The flexible high-speed I/O control module executes logging at the timing when the condition is satisfied (rise) for the output terminals of the multi function counter block.



(1) The conversion cycle changes according to the type of analog input module and the number of A/D conversion enabled channels.
(2) Specify the multi function counter block to be used in the module parameters.

This function specifies one analog input module, and specifies and uses up to two locations from the specified module's input channels. A/D conversion values can be logged under separate logging conditions for each specified channel.
The logging conditions of this function are divided into No.1 and No.2 and are set individually with module parameters. This function is described using the No.1 module parameter, buffer memory addresses, and module labels.

Usable analog input modules

The analog input modules that can log A/D conversion values using this function are listed below.

- R60AD4
- R60ADV8
- R60ADI8
- R60ADH4

Point

If the above analog input modules are operating in Q compatible mode, A/D conversion value logging cannot be performed.

Setting method

Set the following items to use this function.

1. Configure the system to meet the following conditions, and set up the module configuration diagram of the engineering tool.

- Use an analog input module that can use this function.
- Mount the flexible high-speed I/O control module and the analog input module on the same base unit.

2. Create the hardware logic.

Create the hardware logic to generate the timing to perform A/D conversion value logging. (☞ Page 34 Create the hardware logic.)

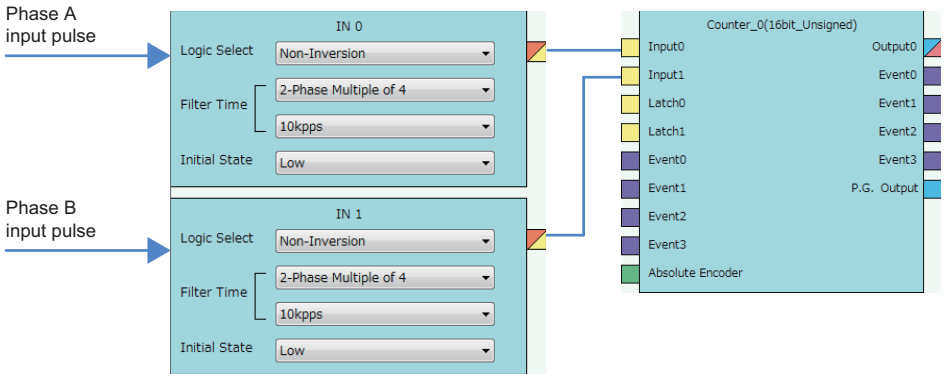
3. Set the module parameters.

Set the module parameters related to the A/D conversion value logging function of the flexible high-speed I/O control module. (☞ Page 35 Setting the module parameters)

Create the hardware logic.

Create the timing for logging A/D conversion values by using the multi function counter block. The explanation of this function uses the wiring for when logging A/D conversion values at the timing of 2-phase multiple of 4 + pulse that is input to IN 0, IN 1 as shown below.

■Hardware logic outline window

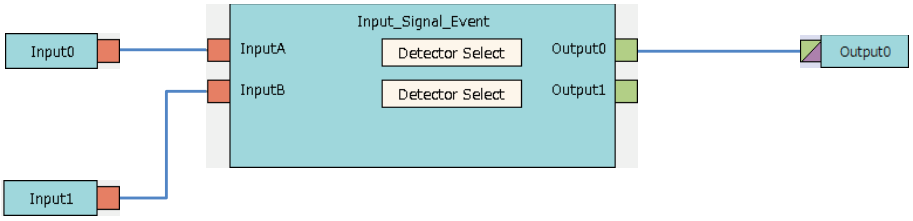


Block	Variable name	Setting value	Description
IN 0	Logic Select	Non-Inversion	Default value
	Filter Time	2-Phase Multiple of 4	Set this variable to 2-phase multiple of 4.
		10kpps	Set according to the specifications of the input pulse.
	Initial State	Low	Default value
IN 1	Same as IN 0		

Point

- The multi function counter block to be used can be either 16 bit (signed/unsigned) or 32 bit (signed/unsigned).
- The A/D conversion value logging function can be used even with settings of external input blocks other than the above. The timing of logging can be changed by changing the settings of the external input block according to the usage environment.

■Multi function counter block detail window (multifunction counter block_0)



Block	Variable name	Setting value	Description
Input 0	—	—	—
Input 1	—	—	—
Input_Signal_Event	Detector Select	2-Phase Multiple of 4+	• Set Detector Select (upper side) to 2-phase multiple of 4+. • Setting is not required for Detector Select (lower side).
Output 0	—	—	—

Point

- Set Detector Select according to the specifications of the device (such as encoder) to be connected to the external input terminal.
- The A/D conversion value logging function can be used even with settings of input signal events other than the above. The timing of logging can be changed by changing the settings of the input signal events according to the usage environment.

Setting the module parameters

The module parameters of the flexible high-speed I/O control module and the analog input module are shown below. Set the module parameters shown in this section.

■Flexible high-speed I/O control module

The following table shows the module parameters of the flexible high-speed I/O control module.

Item	Description	Setting range
Analog input module mounting slot number setting	Set the slot number of the analog input module on the same base unit to be used for the A/D conversion value logging function.	Slot 0 to slot 11
No.1 ADCValue logging enable/disable	Set whether to enable or disable the A/D conversion value logging function.	Disable, enable
No.1 ADCValue logging counter setting	Set the multi function counter block used for timing detection for logging A/D conversion values.	Counter_0 to Counter_7
No.1 ADCValue logging channel setting	Set the channel number of the analog input module for A/D conversion value logging.	CH1 to CH8
No.1 ADCValue logging target setting	Set the logging target data of the analog input module for A/D conversion value logging.	Digital output value, Digital operation value
No.1 ADCValue logging area switching method setting	Set the switching method for the ADCValue logging data storage area (A side/B side).	Automatic switching, Trigger switching
No.1 ADCValue logging area switching condition setting	When "Trigger switching" is set for "ADCValue logging area switching method setting", set the switching condition for the ADCValue logging data storage area.	Disable external input, Enable external input
No.1 ADCValue logging area switching external input setting	When "Enable external input" is set for "ADCValue logging area switching condition setting", set the external input block to be used for switching.	IN 0 to IN B

■Analog input module

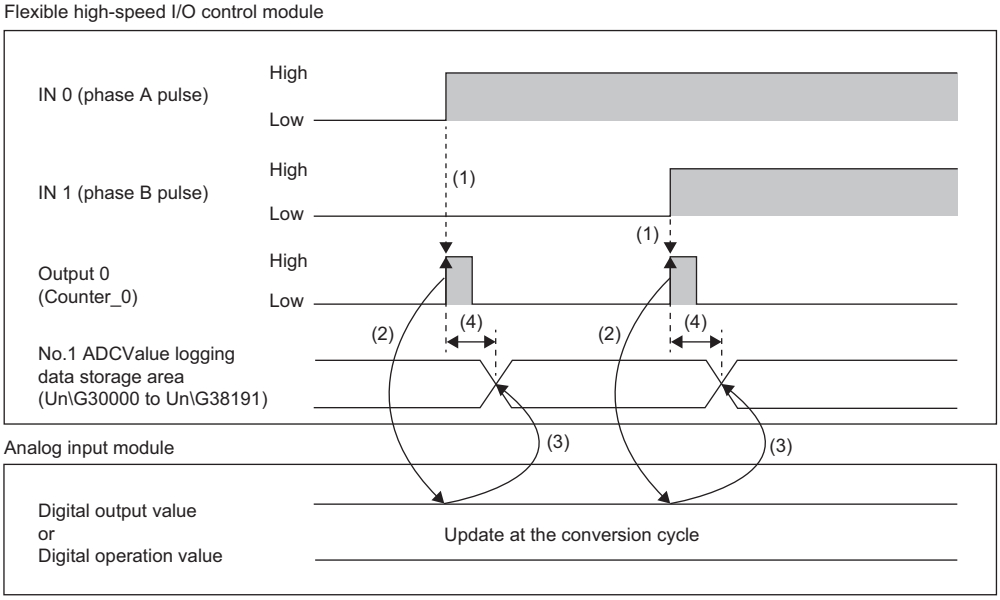
Set "A/D conversion enable/disable setting" for analog input module logging to "A/D conversion enable".

A/D conversion value logging operation

The following shows the operation of this function.

A/D conversion value logging start

A/D conversion value logging starts when 'Hardware logic control start request' (Y4) is turned on. As soon as hardware logic control starts, Enabled (1) is stored in 'No.1 ADCValue logging enable/disable monitor' (Un\G170).
When A/D conversion value logging starts, A/D conversion values are logged whenever the output condition (rise) of Counter 0 is satisfied.



(1) The output condition of Counter 0 is satisfied by the 2-phase multiple of 4+ pulse, and output 0 rises.
(2) The rise of output 0 (Counter 0) is the A/D conversion value acquisition timing.
(3) The acquired A/D conversion value is stored in No.1 ADCValue logging data storage area (Un\G30000 to Un\G38191).
(4) The response time takes up to 10μs from the rise of output 0 (Counter 0) until logging to No.1 ADCValue logging data storage area (Un\G30000 to Un\G38191). The response delay time differs depending on the number for 'No.☐ ADCValue logging enable/disable' (Un\G150, Un\G160) that is set to Enable (1). (The response delay time is 5μs maximum when the number is 1, and 10μs maximum when the number is 2.)

A/D conversion value logging will not start if the following items are set.

- An empty slot number or the slot number of a module that is not the target analog input module is set for 'Analog input module mounting slot number setting' (Un\G140). In this case, an analog input module mounting slot number setting error (error code: 10E1H) occurs.
- A value out of the setting range is set for 'No.1 ADCValue logging counter setting' (Un\G151). In this case, a No.1 ADCValue logging counter setting error (error code: 1101H) occurs.
- A value out of the setting range is set for 'No.1 ADCValue logging channel setting' (Un\G152). In this case, a No.1 ADCValue logging channel setting error (error code: 1111H) occurs.

Point

- By setting the counter timer block and comparison block of the multi function counter block to control the output conditions, the acquisition timing of A/D conversion values can be adjusted.
- Refer to the following as a wiring example of controlling the acquisition timing of A/D conversion values with the internal counter.

☞ Page 235 A/D conversion value logging

Operation during A/D conversion value logging

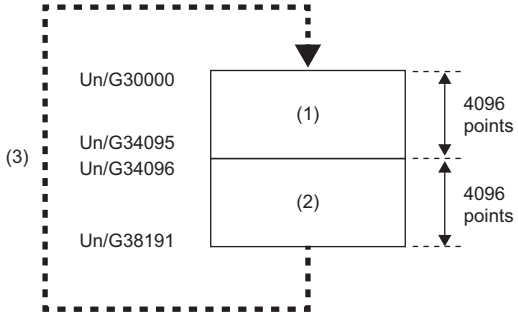
The A/D conversion values acquired at the logging timing of A/D conversion value logging are stored in the ADCValue logging data storage area for each No. The ADCValue logging data storage area is divided into two sides, A side and B side.

Item	Buffer memory address
No.1 ADCValue logging data storage area (A side)	30000 to 34095
No.1 ADCValue logging data storage area (B side)	34096 to 38191

Set the ADCValue logging data storage area switching method with 'No.1 ADCValue logging area switching method setting' (Un\G154). The switching methods are automatic switching and trigger switching.

■Automatic switching

This method keeps storing the logging data by switching between the A side and B side every time 4096 points are stored. This logging method is the best when acquiring logging data at multiple points, such as when inputting cyclic pulses to the flexible high-speed I/O control module, or when generating cyclic pulses with hardware logic.



(1) After 4096 logging data points are stored in No.1 ADCValue logging data storage area (A side) (Un\G30000 to Un\G34095), the value of 'No.1 ADCValue logging data A side storage flag' (Un\G191) changes from Storage not completed (0) to Storage completed (1). The storage location of logging data is switched to No.1 ADCValue logging data storage area (B side) (Un\G34096 to Un\G38191).

(2) After 4096 logging data points are stored in No.1 ADCValue logging data storage area (B side) (Un\G34096 to Un\G38191), the value of 'No.1 ADCValue logging data B side storage flag' (Un\G192) changes from Storage not completed (0) to Storage completed (1).

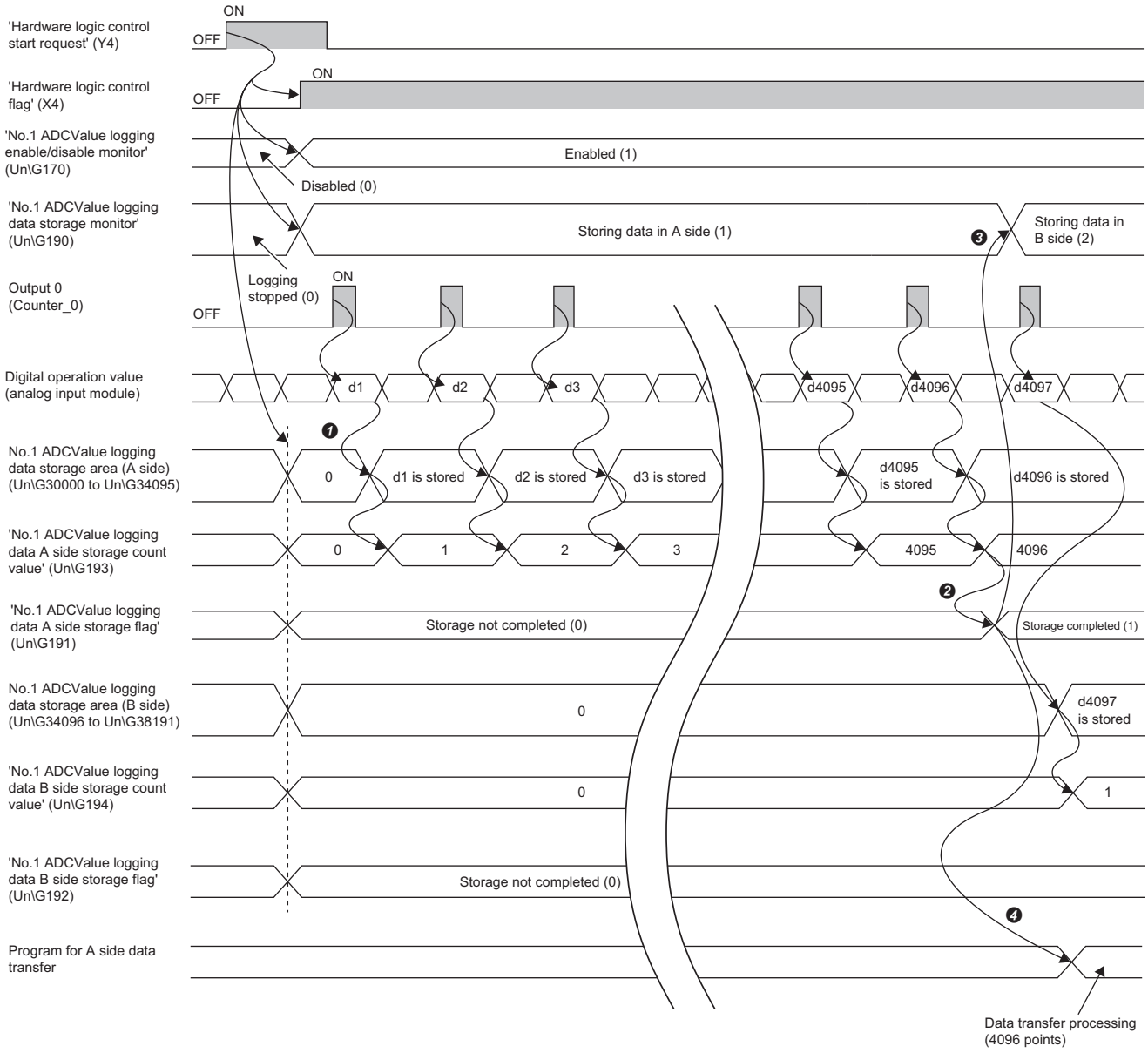
(3) After 4096 points are stored in No.1 ADCValue logging data storage area (B side) (Un\G34096 to Un\G38191), data is overwritten in sequence from the start address of No.1 ADCValue logging data storage area (A side) (Un\G30000 to Un\G34095).

Point

After logging data storage in the ADCValue logging data storage area is completed, save the data in the file register of the CPU module. After the data has been saved, set the flag of the logging side (either 'No.1 ADCValue logging data A side storage flag' (Un\G191) or 'No.1 ADCValue logging data B side storage flag' (Un\G192)) for which the save was completed to Storage not completed (0).

Create a program so that after the above operation, logging data is saved when confirming that Storage completed (1) has been stored again in each flag. (☞ Page 47 Automatic switching program example)

• Operation from the start of logging to the start of storage on B side



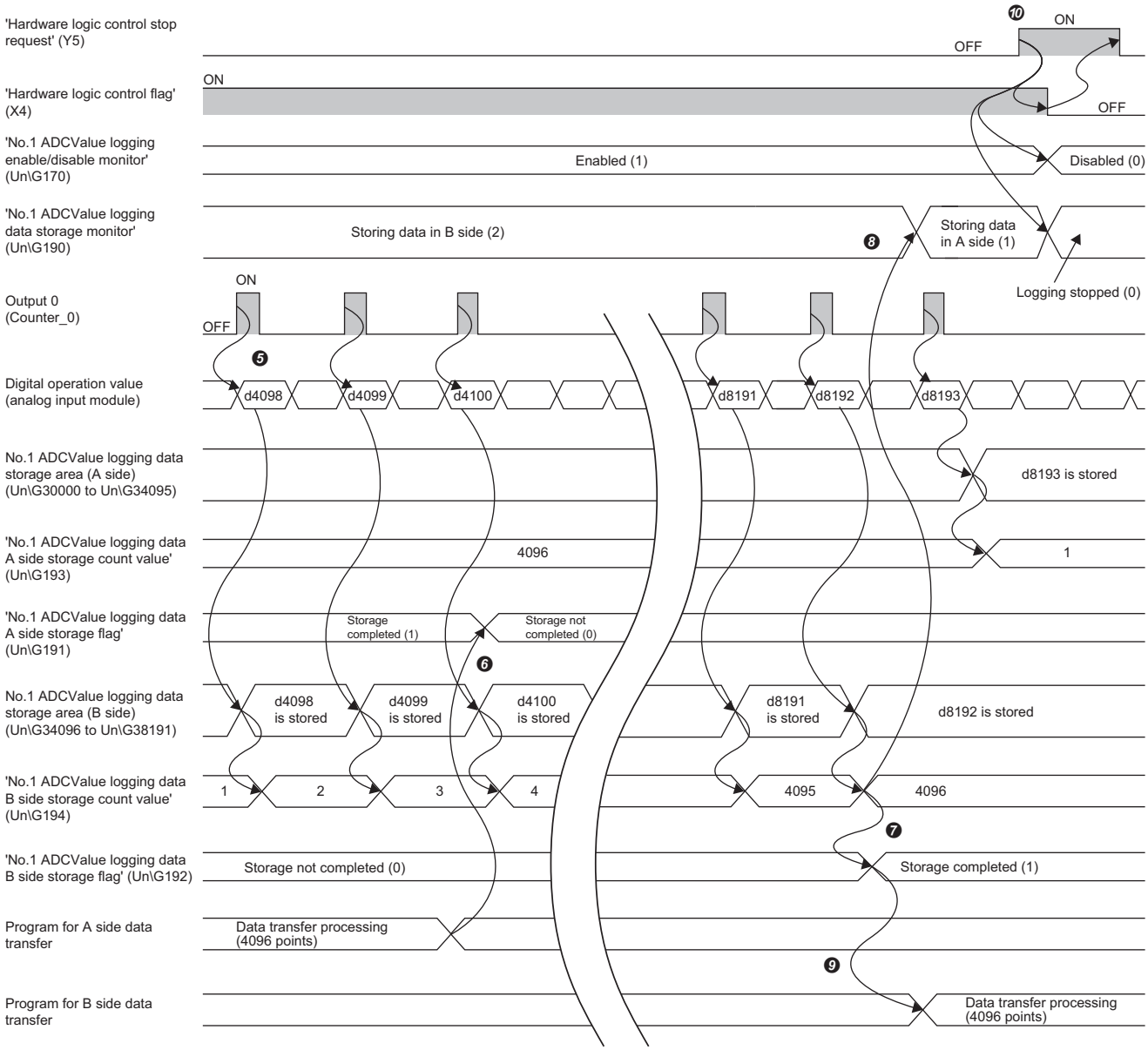
❶ The A/D conversion values are stored in sequence from Un\G30000 of No.1 ADCValue logging data storage area (A side) (Un\G30000 to Un\G34095). Every time A/D conversion value logging is executed, the number of logging points is added to 'No.1 ADCValue logging data A side storage count value' (Un\G193).

❷ After A/D conversion value logging of 4096 points is completed, Storage completed (1) is stored in 'No.1 ADCValue logging data A side storage flag' (Un\G191). From the 4097th point onward, the data is stored in No.1 ADCValue logging data storage area (B side) (Un\G34096 to Un\G38191).

❸ When A/D conversion value logging to B side starts, B side storage in progress (2) is stored in 'No.1 ADCValue logging data storage monitor' (Un\G190), and switching of storage location to the B side is reported.

❹ Execute the program that saves the logging data in the file register of the CPU module by using Storage completed (1) being stored in 'No.1 ADCValue logging data A side storage flag' (Un\G191) as an interlock condition.

• Operation from B side storage in progress to A/D conversion logging stop



❺ The A/D conversion values are stored in sequence from Un\G34096 of No.1 ADCValue logging data storage area (B side) (Un\G34096 to Un\G38191). Every time A/D conversion value logging is executed, the number of logging points is added to 'No.1 ADCValue logging data B side storage count value' (Un\G194).

❻ During A/D conversion logging to B side, execute the program that stores Storage not completed (0) in 'No.1 ADCValue logging data A side storage flag' (Un\G191).

❼ After A/D conversion value logging of 4096 points is completed, Storage completed (1) is stored in 'No.1 ADCValue logging data B side storage flag' (Un\G192). From the 4097th point onward, the data are stored in No.1 ADCValue logging data storage area (A side) (Un\G30000 to Un\G34095).

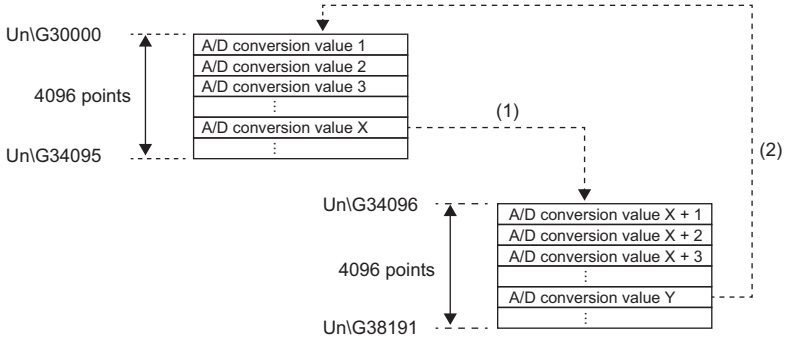
❽ When A/D conversion value logging to A side starts, A side storage in progress (1) is stored in 'No.1 ADCValue logging data storage monitor' (Un\G190), and switching of storage location to the A side is reported.

❾ Execute the program that saves the logging data in the file register of the CPU module by using Storage completed (1) being stored in 'No.1 ADCValue logging data B side storage flag' (Un\G192) as an interlock condition.

❿ To stop A/D conversion value logging, turn off 'Hardware logic control start request' (Y4). By stopping A/D conversion value logging, the number of logging points until conversion stop can be checked.

■Trigger switching

This storage method switches between ADCValue logging data storage area A side and B side as independent logging areas. Switching the ADCValue logging data storage area at specific trigger or pulse modulation timing is the best logging method when acquiring detailed logging data.



(1) When the trigger condition is satisfied during A/D conversion value logging to No.1 ADCValue logging data storage area (A side) (Un\G30000 to Un\G34095), storage switches to No.1 ADCValue logging data storage area (B side) (Un\G34096 to Un\G38191). At this time, the value of 'ADCValue logging data A side storage flag' (Un\G191) changes from Storage not completed (0) to Storage completed (1).

(2) When the trigger condition is satisfied during A/D conversion value logging to No.1 ADCValue logging data storage area (B side) (Un\G34096 to Un\G38191), storage switches to No.1 ADCValue logging data storage area (A side) (Un\G30000 to Un\G34095). At this time, the value of the 'No.1 ADCValue logging data B side storage flag' (Un\G192) changes from Storage not completed (0) to Storage completed (1).

The following can be used as triggers for switching the ADCValue logging data storage area.

Switching trigger	How to use
'No.1 ADCValue logging area switching request' (YD)	The rising edge of 'No.1 ADCValue logging area switching request' (YD) can be used as a trigger.
External input (IN 0 to IN B)	The rising edge of the external input (IN 0 to IN B) can be used as a trigger by performing the following settings. • Set 'No.1 ADCValue logging area switching condition setting' (Un\G155) to Enable external input (1). • Set the number of external inputs (IN 0 to IN B) to be used as trigger conditions in 'No.1 ADCValue logging area switching external input setting' (Un\G156).

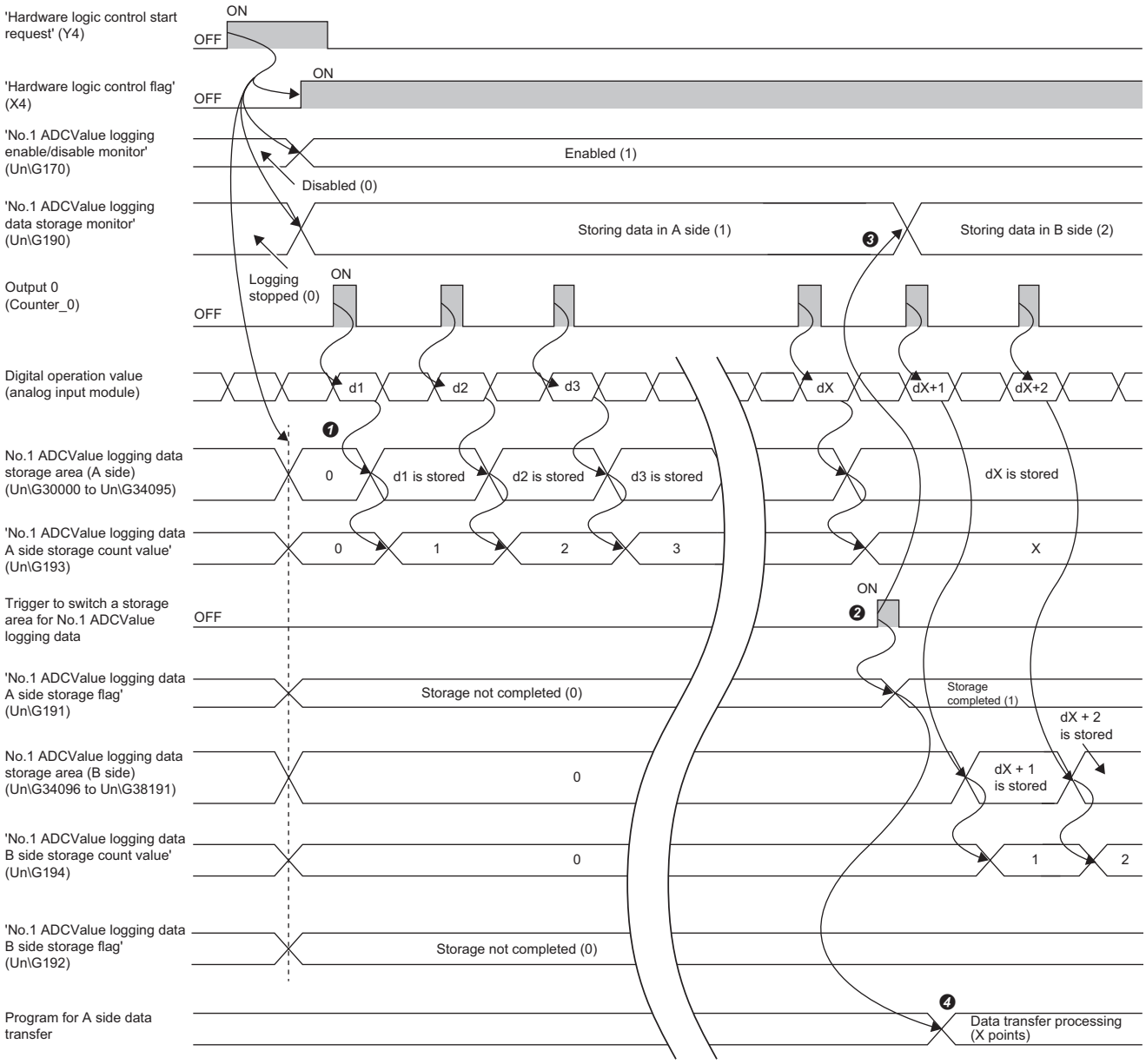
Point

Create a program that, after checking 'No.1 ADCValue logging data A side storage flag' (Un\G191) and 'No.1 ADCValue logging data B side storage flag' (Un\G192) and identifying the side where the logging data is being stored, acquires the stored data count value from the start address of the ADCValue logging data storage area. (☞ Page 53 Trigger switching program example)

Precautions

Switch the ADCValue logging data storage area by the time the number of logging points reaches 4096 points. A/D conversion values exceeding 4096 points will not be logged. Use the automatic switching method when logging A/D conversion values continue to 4097 points or more.

• Operation from the start of logging to trigger switching (B side storage start)



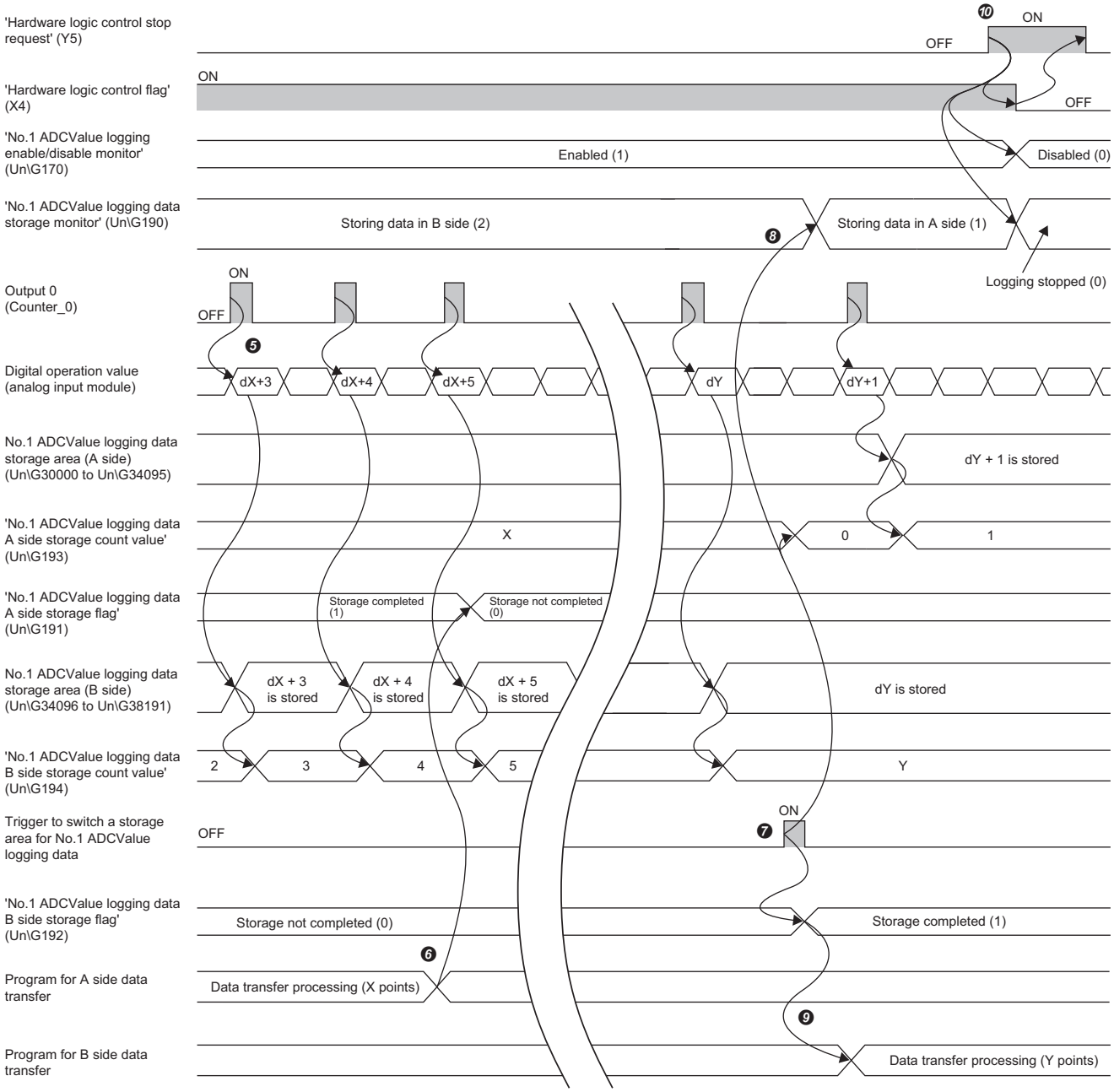
❶ The A/D conversion values are stored in sequence from Un\G30000 of No.1 ADCValue logging data storage area (A side) (Un\G30000 to Un\G34095). Every time A/D conversion value logging is executed, the number of logging points is added to 'No.1 ADCValue logging data A side storage count value' (Un\G193).

❷ When the switching trigger for the ADCValue logging data storage area rises, Storage completed (1) is stored in 'No.1 ADCValue logging data A side storage flag' (Un\G191). From the next point (X + 1 point) after the switching trigger rises and onward, the data is stored in the No.1 ADCValue logging data storage area (B side) (Un\G34096 to Un\G38191).

❸ When A/D conversion value logging to B side starts, B side storage in progress (2) is stored in 'No.1 ADCValue logging data storage monitor' (Un\G190), and switching of storage location to the B side is reported.

❹ Execute the program that saves the logging data in the file register of the CPU module by using Storage completed (1) being stored in 'No.1 ADCValue logging data A side storage flag' (Un\G191) as an interlock condition.

• Operation from B side storage to trigger switching (A side storage start), and until stop



❺ The A/D conversion values are stored in sequence from Un\G34096 of No.1 ADCValue logging data storage area (B side) (Un\G34096 to Un\G38191). Every time A/D conversion value logging is executed, the number of logging points is added to 'No.1 ADCValue logging data B side storage count value' (Un\G194).

❻ During A/D conversion logging to B side, execute the program that stores Storage not completed (0) in 'No.1 ADCValue logging data A side storage flag' (Un\G191).

❼ When the switching trigger for the ADCValue logging data storage area rises, Storage completed (1) is stored in 'No.1 ADCValue logging data B side storage flag' (Un\G192). From the next point (Y + 1 point) after the switching trigger rises and onward, the data is stored in the No.1 ADCValue logging data storage area (A side) (Un\G30000 to Un\G34095). At this time, the stored value in 'No.1 ADCValue logging data A side storage count value' (Un\G193) is reset to 0.

❽ When A/D conversion value logging to A side starts, A side storage in progress (1) is stored in 'No.1 ADCValue logging data storage monitor' (Un\G190), and switching of storage location to the A side is reported.

❾ Execute the program that saves the logging data in the file register of the CPU module by using Storage completed (1) being stored in 'No.1 ADCValue logging data B side storage flag' (Un\G192) as an interlock condition.

❿ To stop A/D conversion value logging, turn off 'Hardware logic control start request' (Y4). By stopping A/D conversion value logging, the number of logging points until conversion stop can be checked.

■Stored data count value reset

'No.1 ADCValue logging data A side storage count value' (Un\G193) and 'No.1 ADCValue logging data B side storage count value' (Un\G194) are reset at the following timings.

ADCValue logging data storage area switching method	Reset description
Common	■Reset timing • At hardware logic control start • When 'No.1 ADCValue logging reset request' (Un\G195) is set from No request (0) to Reset request (1) ■Reset target All stored data count values are reset to 0.
Automatic switching	■Reset timing When the storage location of logging data is switched and A/D conversion value logging of the 1st point is executed ■Reset target The stored data count value of the switching destination is reset to 1.
Trigger switching	■Reset timing When the trigger condition for the storage location of logging data is satisfied ■Reset target The stored data count value of the switching destination is reset to 0.

■A/D conversion value logging function setting confirmation

The setting details of this function can be checked in the following buffer memory areas.

Buffer memory	Reference
'Analog input module mounting slot number setting monitor' (Un\G145)	Page 275 Analog input module mounting slot number setting monitor
'No.1 ADCValue logging enable/disable monitor' (Un\G170)	Page 279 No.☐ ADCValue logging enable/disable monitor
'No.1 ADCValue logging counter setting monitor' (Un\G171)	Page 279 No.☐ ADCValue logging counter setting monitor
'No.1 ADCValue logging channel setting monitor' (Un\G172)	Page 280 No.☐ ADCValue logging channel setting monitor
'No.1 ADCValue logging target setting monitor' (Un\G173)	Page 280 No.☐ ADCValue logging target setting monitor
'No.1 ADCValue logging area switching method setting monitor' (Un\G174)	Page 281 No.☐ ADCValue logging area switching method setting monitor
'No.1 ADCValue logging area switching condition setting monitor' (Un\G175)	Page 281 No.☐ ADCValue logging area switching condition setting monitor
'No.1 ADCValue logging area switching external input setting monitor' (Un\G176)	Page 282 No.☐ ADCValue logging area switching external input setting monitor

In the following states, 0 is stored regardless of the settings in the above buffer memory areas.

- Disable (0) is set for 'No.1 ADCValue logging enable/disable' (Un\G150).
- Hardware logic control is stopped and 'Hardware logic control flag' (X4) was turned off.

A/D conversion value logging stop

At the same time as hardware logic control stop, A/D conversion value logging operation stops.

If the simulation function is executed during hardware logic control, or if a stop error occurs in the control CPU, A/D conversion value logging automatically stops.

☞ Page 123 Simulation function

For details on the hardware logic control stop conditions, refer to the following.

☞ Page 259 Hardware logic control flag

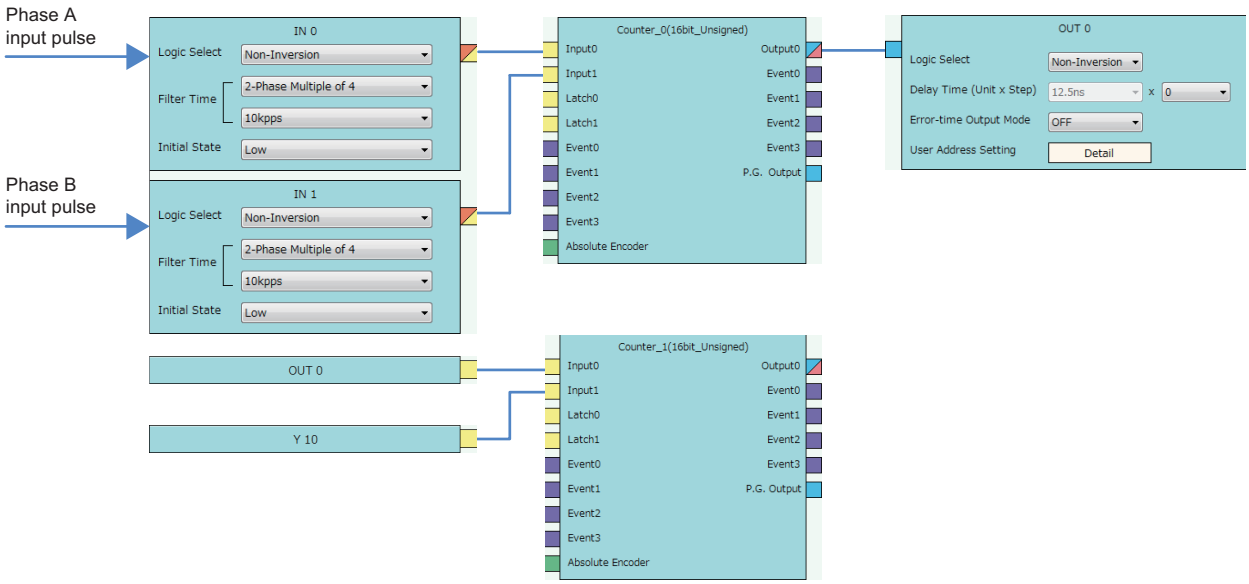
■When stopping A/D conversion value logging without stopping hardware logic control

To stop A/D conversion value logging without stopping hardware logic control, create the hardware logic as shown below. A/D conversion value logging can be controlled by turning on and off the Y device terminals (Y10 to Y1F).

Ex.

When using Y device terminal Y10

• Hardware logic outline window

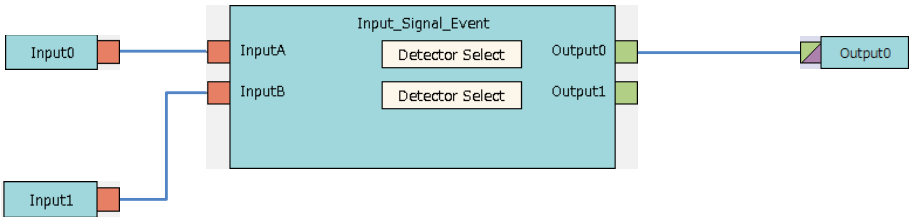


• Counter block detail window (multi function counter block_0)

Set the same content as shown below.

☞ Page 34 Multi function counter block detail window (multifunction counter block_0)

• Counter block detail window (multi function counter block_1)



Block	Variable name	Setting value	Description
Input 0	—	—	—
Input 1	—	—	—
Input_Signal_Event	Detector Select	1/2-phase multiple of 1(CW/CCW)+	• For Detector Select (upper side), set phase A input pulse High to Low and phase B input pulse Low as the detection conditions. • Setting is not required for Detector Select (lower side).
Output 0	—	—	—

After creating the above hardware logic, set 'No.1 ADCValue logging counter setting' (Un\G151) to multi function counter 1 (1). Because the A/D conversion value logging condition is satisfied when Y10 is OFF, and the condition is not satisfied while Y10 is ON, A/D conversion value logging can be controlled.

A/D conversion value logging restart

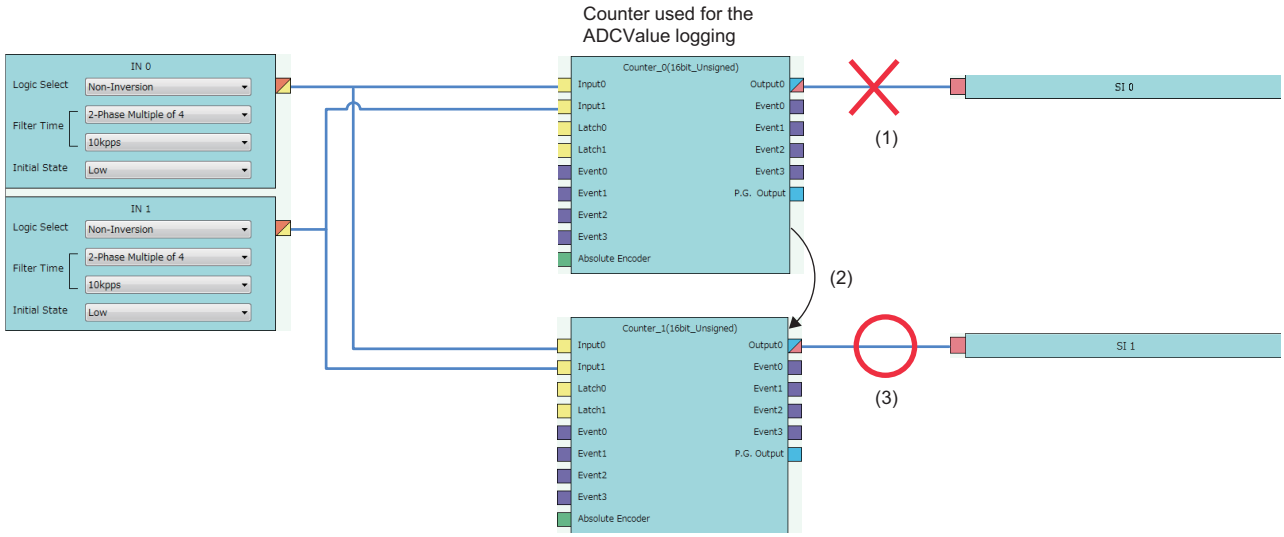
After A/D conversion value logging stop, A/D conversion value logging can be restarted by turning on 'Hardware logic control start request' (Y4). At this time, the following values are reset to 0, and storage of A/D conversion values restarts from the first point of the No.1 ADCValue logging data storage area (A side).

- 'No.1 ADCValue logging data A side storage count value' (Un\G193)
- 'No.1 ADCValue logging data B side storage count value' (Un\G194)
- No.1 ADCValue logging data storage area (Un\G30000 to Un\G38191)

Restrictions

■Wiring to the SI device terminal of the multi function counter block used to generate A/D conversion value acquisition timing

Even if wiring is performed to the SI device terminal block from the output of the multi function counter block set to 'No.1 ADCValue logging counter setting' (Un\G151), interrupts will not be output to the CPU module. To generate interrupts for the CPU module in synchronization with the A/D conversion value acquisition timing, perform the same hardware logic settings and wiring to another multi function counter block, and connect it to an SI device terminal.



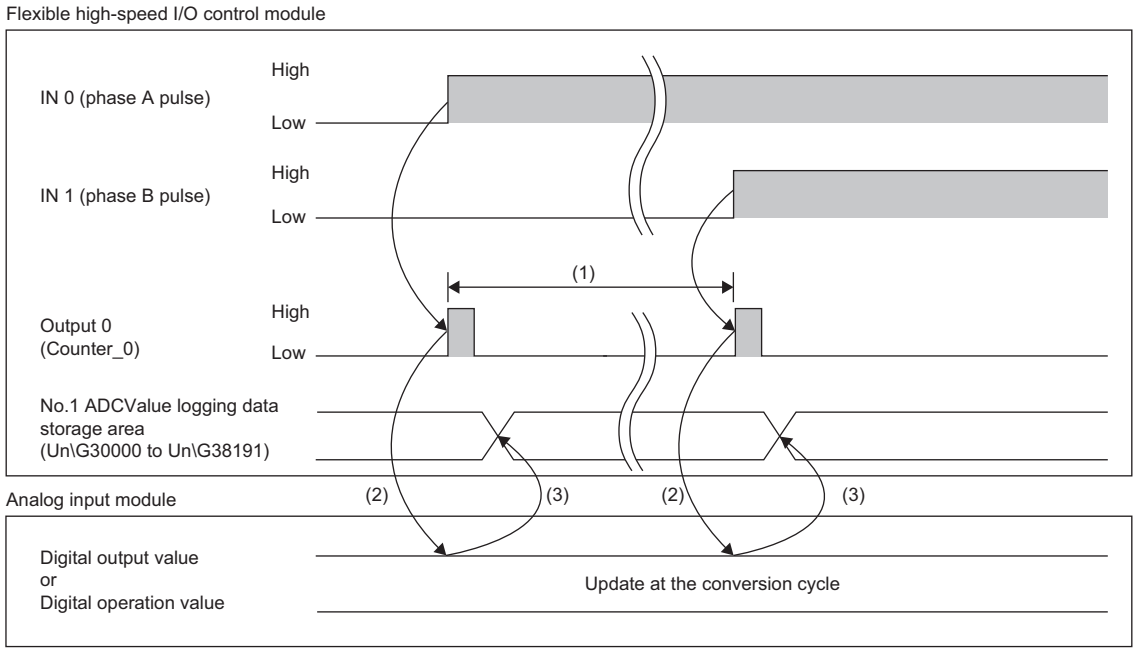
(1) No interrupts are generated, even if the hardware logic is wired.
(2) Set a multi function counter for A/D conversion value logging and set the same parameters to another multi function counter.
(3) Interrupts will be generated at the same timing as the output timing of the multi function counter used for A/D conversion value logging.

■A/D conversion value acquisition timing interval

Set the timing of A/D conversion value acquisition to enact at the intervals indicated below.

Condition	Required timing
Either No.1 ADCValue logging enable/disable or No.2 ADCValue logging enable/disable is set to Enable (1).	20μs or more
Both No.1 ADCValue logging enable/disable and No.2 ADCValue logging enable/disable are set to Enable (1).	50μs or more

If the output condition is satisfied at a timing shorter than the above intervals, the A/D conversion values may not be logged.



(1) A/D conversion value acquisition timing interval
(2) A/D conversion value acquisition timing
(3) Logging of acquired A/D conversion values

■Processing order of No.1 A/D conversion value logging and No.2 A/D conversion value logging

If the logging timings of No.1 A/D conversion value logging and No.2 A/D conversion value logging overlap, processing is performed first from No.1 A/D conversion value logging, then No.2 A/D conversion value logging. Therefore, the timing of the A/D conversion values acquired by No.1 A/D conversion value logging and the timing of the A/D conversion values acquired by No.2 A/D conversion value logging are not completely synchronized.

■Combination with other functions

- Continuous logging cannot be used during A/D conversion value logging. If continuous logging is used, a continuous logging start error (error code: 10D7H) will occur during A/D conversion value logging execution.
- When the flexible high-speed I/O control module is set as the synchronization target module, the A/D conversion value logging function cannot be used. When the A/D conversion value logging function is set to enable while set to the synchronization target module, and 'Hardware logic control start request' (Y4) is turned on, an A/D conversion value logging function start error (error code: 10E0H) will occur during inter-module synchronization. In this case, A/D conversion value logging does not start.

■Online module change of analog input module

While using the A/D conversion value logging function, do not perform online module change of the analog input module that is the logging target. An analog input module communication error (error code: 10E2H) will occur if the A/D conversion value acquisition timing is satisfied during online module change.

Program examples

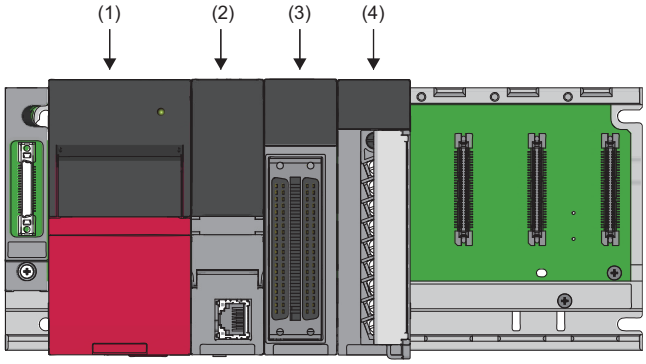
The following shows examples of programs for saving the logging data that is stored in the No.1 ADCValue logging data storage area (Un\G30000 to Un\G38191) to the file register of the CPU module. Program examples are shown for each ADCValue logging data storage area switching method.

Automatic switching program example

The following shows a program example for when the switching method is automatic switching. Hardware logic control (A/D conversion value logging) is started, and digital operation values are logged from the analog input module in slot No.1 (CH1). The acquired logging data for 102400 points (ZR0 to ZR102399) is saved continuously to the file register of the CPU module.

■System configuration

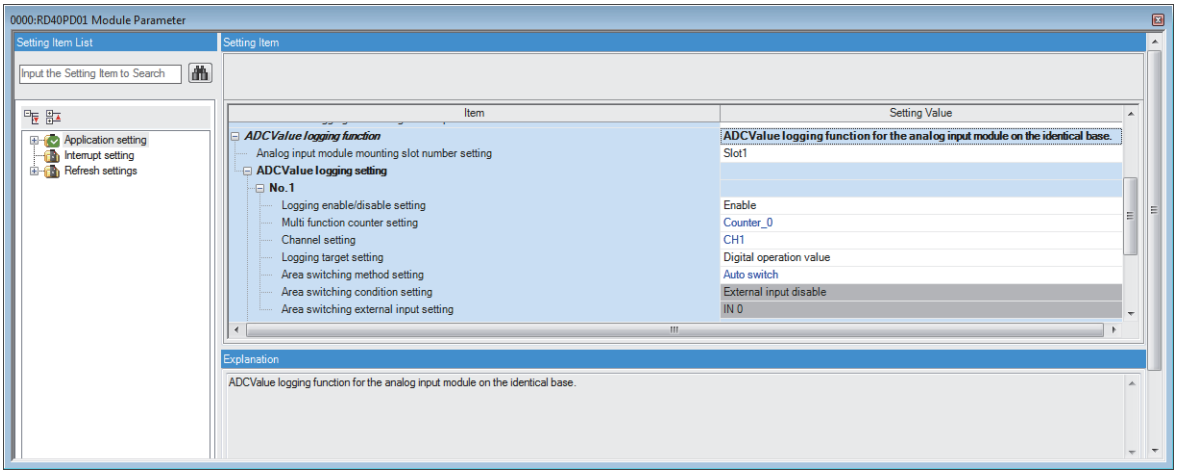
A system configuration example is shown below.



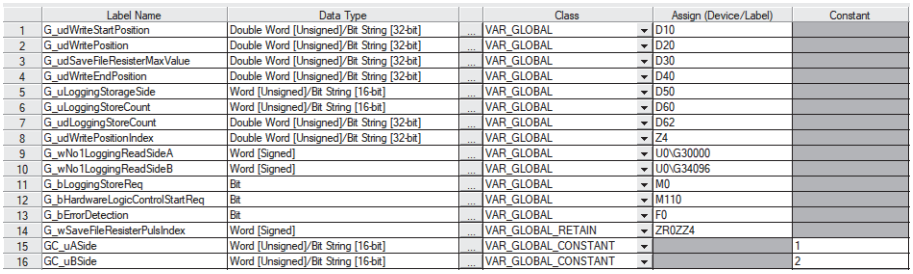
(1) R61P
(2) R04CPU
(3) RD40PD01
(4) R60ADH4

■Parameter settings

Set the module parameter of the flexible high-speed I/O control module as follows in GX Works3.

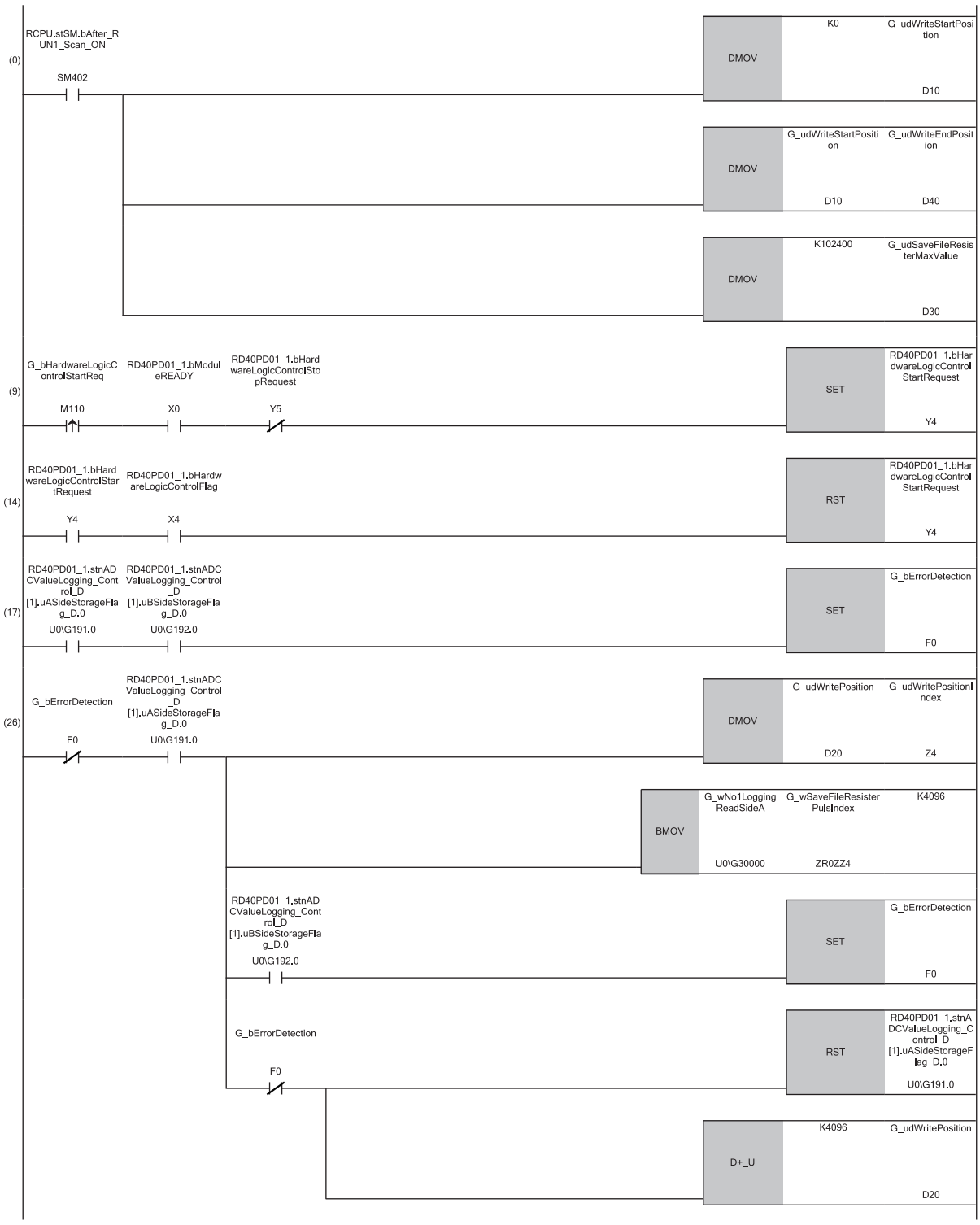


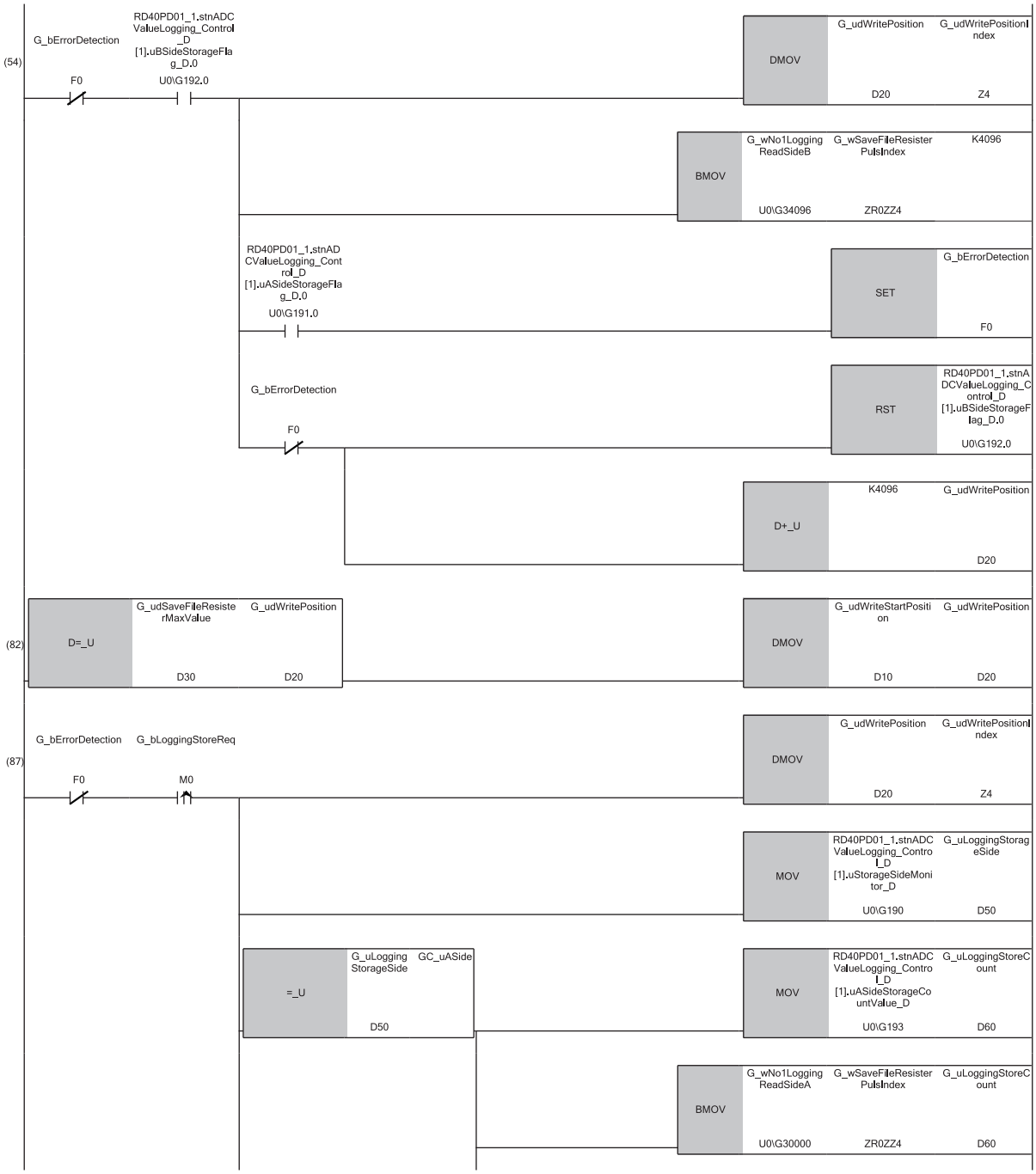
■Label settings

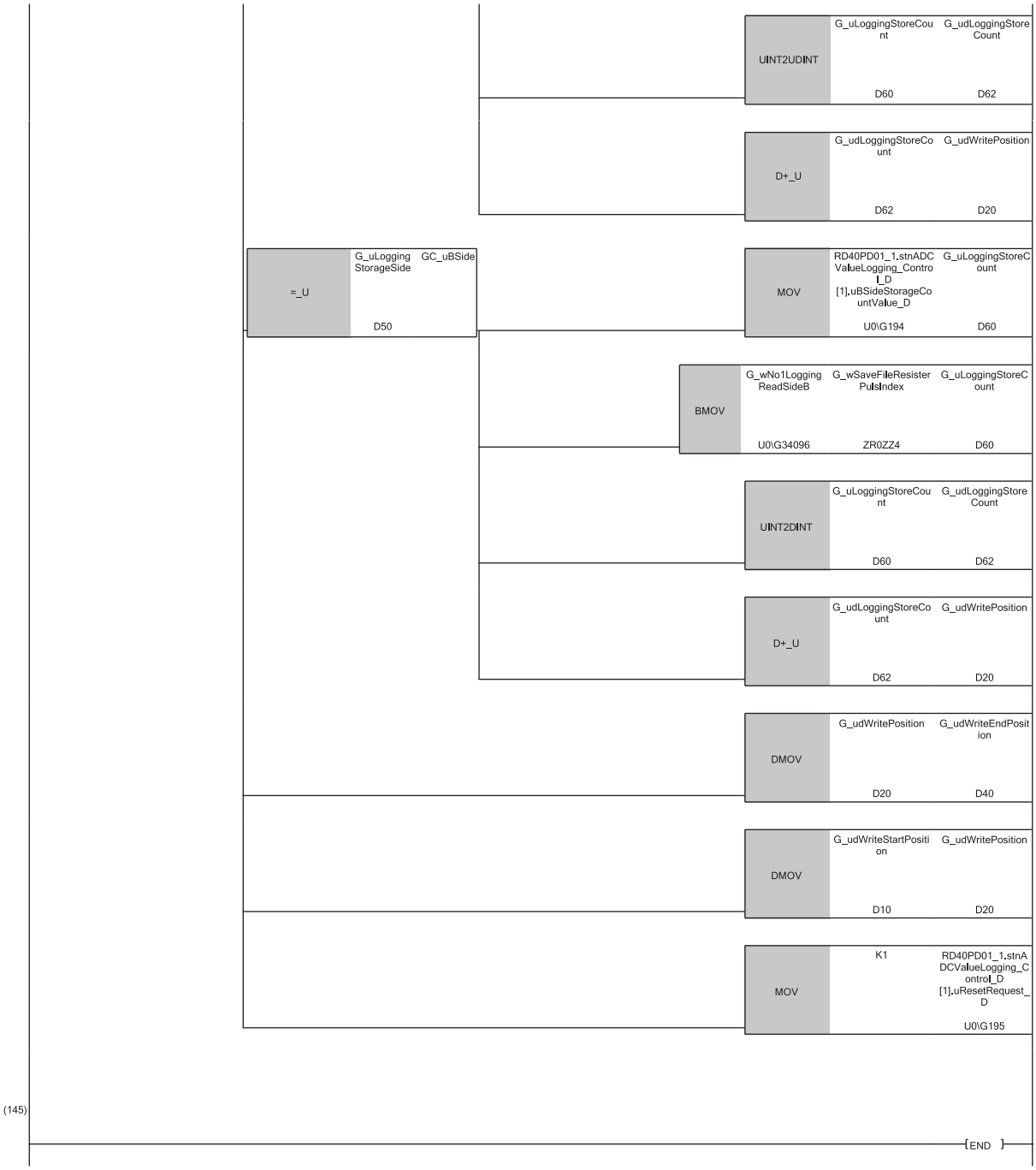
Classification	Label name	Description	Device
Module label	RCPU.stSM.bAfter_RUN1_Scan_ON	ON for one scan after RUN	SM402
	RD40PD01_1.bModuleREADY	Module READY	X0
	RD40PD01_1.bHardwareLogicControlFlag	Hardware logic control flag	X4
	RD40PD01_1.bHardwareLogicControlStartRequest	Hardware logic control start request	Y4
	RD40PD01_1.stnADCValueLogging_Control_D[1].uStorageSideMonitor_D	No.1 ADCValue logging data storage monitor	U0\G190
	RD40PD01_1.stnADCValueLogging_Control_D[1].uASideStorageFlag_D.0	No.1 ADCValue logging data A side storage flag	U0\G191.0
	RD40PD01_1.stnADCValueLogging_Control_D[1].uBSideStorageFlag_D.0	No.1 ADCValue logging data B side storage flag	U0\G192.0
	RD40PD01_1.stnADCValueLogging_Control_D[1].uASideStorageCountValue_D	No.1 ADCValue logging data A side storage count value	U0\G193
	RD40PD01_1.stnADCValueLogging_Control_D[1].uBSideStorageCountValue_D	No.1 ADCValue logging data B side storage count value	U0\G194
	RD40PD01_1.stnADCValueLogging_Control_D[1].uResetRequest_D	No.1 ADCValue logging reset request	U0\G195
Label to be defined	Define global labels as shown below:  Although 'G_wSaveRegisterPlusIndex' operates without problem if it is used with the class of VAR_GLOBAL, using this label in the same way on GX Works3 of Ver.1.055H or later will cause a conversion error on the consistency check of global labels. Set the class to VAR_GLOBAL_RETAIN when using 'G_wSaveRegisterPlusIndex' on GX Works3 of Ver.1.055H or later.		

	Label Name	Data Type		Class	Assign (Device/Label)	Constant
1	G_udWriteStartPosition	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D10	
2	G_udWritePosition	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D20	
3	G_udSaveFileResisterMaxValue	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D30	
4	G_udWriteEndPosition	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D40	
5	G_uLoggingStorageSide	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D50	
6	G_uLoggingStoreCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D60	
7	G_udLoggingStoreCount	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D62	
8	G_udWritePositionIndex	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	Z4	
9	G_wNo1LoggingReadSideA	Word [Signed]	...	VAR_GLOBAL	U0\G30000	
10	G_wNo1LoggingReadSideB	Word [Signed]	...	VAR_GLOBAL	U0\G34096	
11	G_bLoggingStoreReq	Bit	...	VAR_GLOBAL	M0	
12	G_bHardwareLogicControlStartReq	Bit	...	VAR_GLOBAL	M110	
13	G_bErrorDetection	Bit	...	VAR_GLOBAL	F0	
14	G_wSaveFileResisterPulsIndex	Word [Signed]	...	VAR_GLOBAL_RETAIN	ZR0ZZ4	
15	GC_uASide	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL_CONSTANT		1
16	GC_uBSide	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL_CONSTANT		2

■Program


(0)
RCPU.stSM.bAfter_RUN1_Scan_ON
SM402
DMOV
K0
G_udWriteStartPosition
D10
DMOV
G_udWriteStartPosition
G_udWriteEndPosition
D10
D40
DMOV
K102400
G_udSaveFileResisterMaxValue
D30
(9)
G_bHardwareLogicControlStartReq
RD40PD01_1.bModuleREADY
RD40PD01_1.bHardwareLogicControlStopRequest
M110
X0
Y5
SET
RD40PD01_1.bHardwareLogicControlStartRequest
Y4
(14)
RD40PD01_1.bHardwareLogicControlStartRequest
RD40PD01_1.bHardwareLogicControlFlag
Y4
X4
RST
RD40PD01_1.bHardwareLogicControlStartRequest
Y4
(17)
RD40PD01_1.stnADCValueLogging_Control_D[1].uASideStorageFlag_D.0
RD40PD01_1.stnADCValueLogging_Control_D[1].uBSideStorageFlag_D.0
U0\G191.0
U0\G192.0
SET
G_bErrorDetection
F0
(26)
G_bErrorDetection
RD40PD01_1.stnADCValueLogging_Control_D[1].uASideStorageFlag_D.0
F0
U0\G191.0
DMOV
G_udWritePosition
G_udWritePositionIndex
D20
Z4
BMOV
G_wNo1LoggingReadSideA
G_wSaveFileResisterPulsIndex
K4096
U0\G30000
ZR0ZZ4
RD40PD01_1.stnADCValueLogging_Control_D[1].uBSideStorageFlag_D.0
U0\G192.0
SET
G_bErrorDetection
F0
G_bErrorDetection
F0
RST
RD40PD01_1.stnADCValueLogging_Control_D[1].uASideStorageFlag_D.0
U0\G191.0
D+_U
K4096
G_udWritePosition
D20


(54)
G_bErrorDetection
F0
RD40PD01_1.stnADC ValueLogging_Control _D [1].uBSideStorageFla g_D.0
U0\G192.0
DMOV
G_udWritePosition
D20
G_udWritePositionI ndex
Z4
BMOV
G_wNo1Logging ReadSideB
U0\G34096
G_wSaveFileResister PulsIndex
ZR0ZZ4
K4096
RD40PD01_1.stnAD CValueLogging_Cont rol_D [1].uASideStorageFla g_D.0
U0\G191.0
SET
G_bErrorDetection
F0
G_bErrorDetection
F0
RST
RD40PD01_1.stnA DCValueLogging_C ontrol_D [1].uBSideStorageF lag_D.0
U0\G192.0
D+_U
K4096
G_udWritePosition
D20
(82)
D=_U
G_udSaveFileResiste rMaxValue
D30
G_udWritePosition
D20
DMOV
G_udWriteStartPositi on
D10
G_udWritePosition
D20
(87)
G_bErrorDetection
F0
G_bLoggingStoreReq
M0
DMOV
G_udWritePosition
D20
G_udWritePositionI ndex
Z4
MOV
RD40PD01_1.stnADC ValueLogging_Contro l_D [1].uStorageSideMoni tor_D
U0\G190
G_uLoggingStorag eSide
D50
=_U
G_uLogging StorageSide
D50
GC_uASide
MOV
RD40PD01_1.stnADC ValueLogging_Contro l_D [1].uASideStorageCo untValue_D
U0\G193
G_uLoggingStoreC ount
D60
BMOV
G_wNo1Logging ReadSideA
U0\G30000
G_wSaveFileResister PulsIndex
ZR0ZZ4
G_uLoggingStoreC ount
D60


UINT2UDINT
G_uLoggingStoreCount
G_udLoggingStoreCount
D60
D62
D+_U
G_udLoggingStoreCount
G_udWritePosition
D62
D20
=_U
G_uLoggingStorageSide
GC_uBSide
D50
MOV
RD40PD01_1.stnADCValueLogging_Control_D[1].uBSideStorageCountValue_D
G_uLoggingStoreCount
U0\G194
D60
BMOV
G_wNo1LoggingReadSideB
G_wSaveFileResisterPulsIndex
G_uLoggingStoreCount
U0\G34096
ZR0ZZ4
D60
UINT2DINT
G_uLoggingStoreCount
G_udLoggingStoreCount
D60
D62
D+_U
G_udLoggingStoreCount
G_udWritePosition
D62
D20
DMOV
G_udWritePosition
G_udWriteEndPosition
D20
D40
DMOV
G_udWriteStartPosition
G_udWritePosition
D10
D20
MOV
K1
RD40PD01_1.stnADCValueLogging_ControlD[1].uResetRequest_D
U0\G195
(145)
END

(0) The settings are configured as follows by the contact turning on for one scan after RUN.
 • 'G_udWriteStartPosition' (D10) is set to 0 so that ZR0 is set as the start position of the save destination file register.
 • G_udWriteendPosition (D20) is set to a value in 'G_udWriteStartPosition' (D10) so that the write position of the file register is initialized.
 • 'G_udSaveFileRegisterMaxValue' (D30) is set to 102400, which is the maximum number of storage points of the save destination file register.
(9) Turn on 'G_bHardwareLogicControlStartReq' (M110) to start hardware logic control (A/D conversion value logging).
(17)Depending on the processing time of other high-priority interrupt programs, 'G_bErrDetection' (F0) turns on and stops the transfer of logging data when 'No.1 ADCValue logging data A side storage flag' (Un\G191) and 'No.1 ADCValue logging data B side storage flag' (Un\G192) are both turned on (logging data omission has occurred).
(26)When 'No.1 ADCValue logging data A side storage flag' (Un\G191) is turned on, the values of No.1 ADCValue logging data storage area (A side) (Un\G30000 to Un\G34095) are transferred to the file register.
(54)When 'No.1 ADCValue logging data B side storage flag' (Un\G192) is turned on, the values of No.1 ADCValue logging data storage area (B side) (Un\G34096 to Un\G38191) are transferred to the file register.
(82)When data gets stored up to ZR102399, storage of logging data is repeated from ZR0.
(87)If untransferred logging data remains in the ADCValue logging data storage area due to external pulse stop, turn on 'G_bLoggingStoreReq' (M0) to transfer the remaining logging data. After the transfer of logging data is completed, the logging stop position of the file register is stored in 'G_udWriteEndPosition' (D40). Also, the file register storage position is returned to the start position (ZR0), and the A/D conversion value logging status is reset.

Precautions

According to the encoder cycle used for logging timing generation, and so on, configure the program so that the program that transfers data to the file register is executed within the minimum value of logging time for 4096 points. This will prevent logging data omission. If the program cannot be executed in time due to other processing, register it to a fixed-interval program.

Ex.

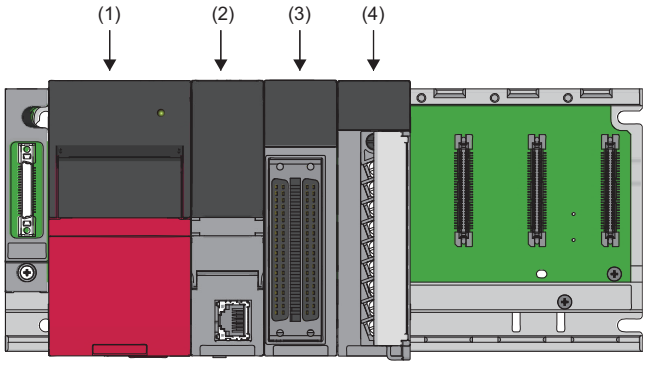
When logging A/D conversion values according to the pulse input from an external encoder at the cycle of 20μs, use a processing time within 81.92ms (20μs × 4096 (point)) as a guide, and configure a program that transfers the data to the file register.

Trigger switching program example

The following shows a program example for when the switching method is trigger switching. Hardware logic control (A/D conversion value logging) is started, and digital operation values are logged from the analog input module in slot No.1 (CH1). When trigger switching occurs, data is saved to the file register of the CPU module as appropriate. Because the number of logging points at the time of trigger switching varies depending on the timing, the total number of logging points is in the range of 102400 to 106495 points (ZR0 to ZR106494).

■System configuration

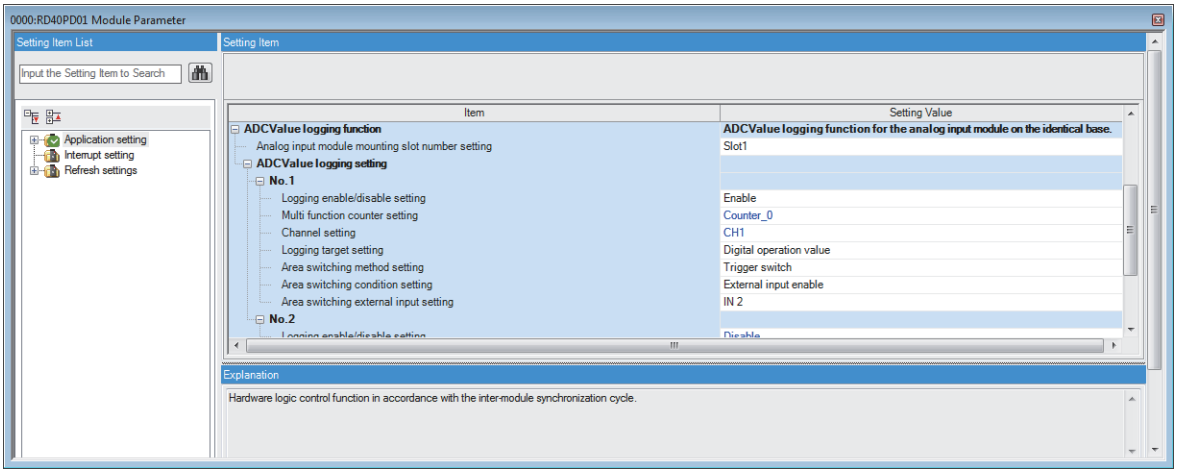
A system configuration example is shown below.



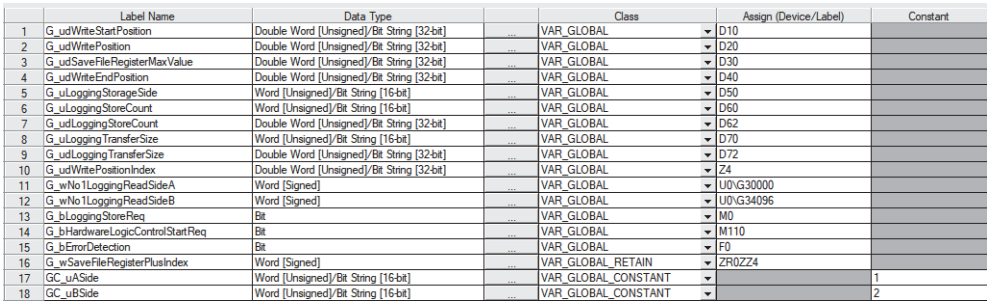
(1) R61P
(2) R04CPU
(3) RD40PD01
(4) R60ADH4

■Parameter settings

Set the module parameter of the flexible high-speed I/O control module as follows in GX Works3.

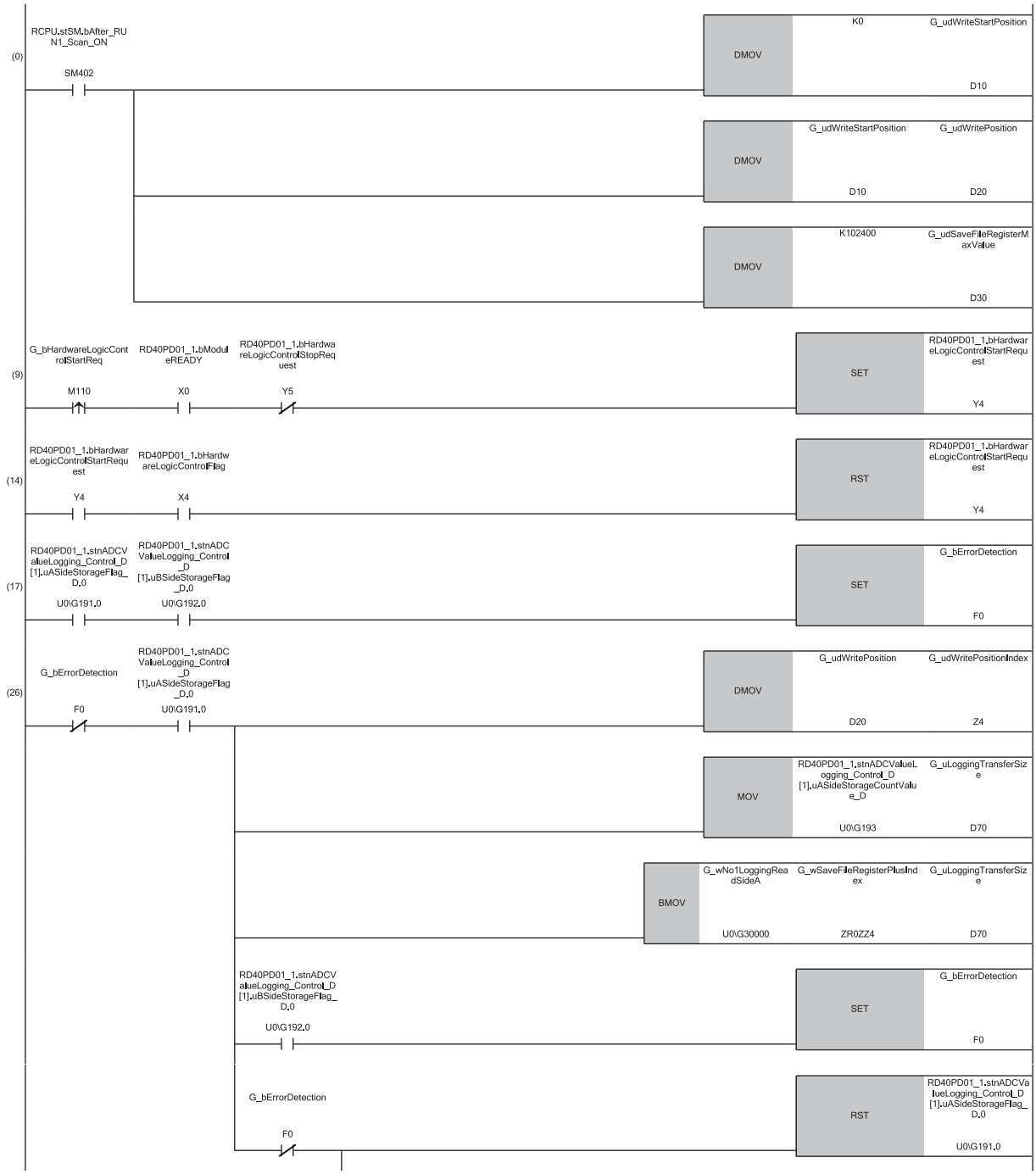


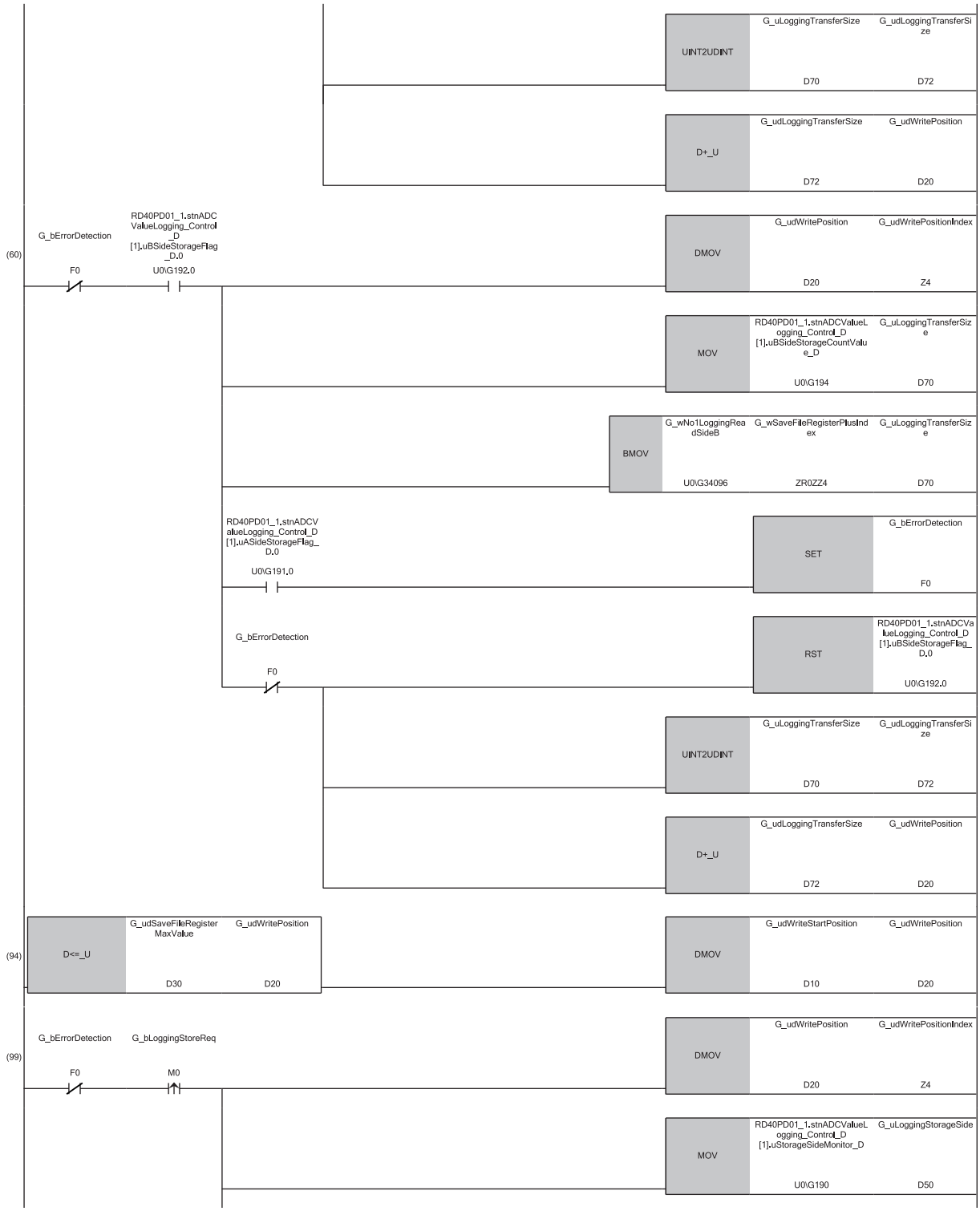
■Label settings

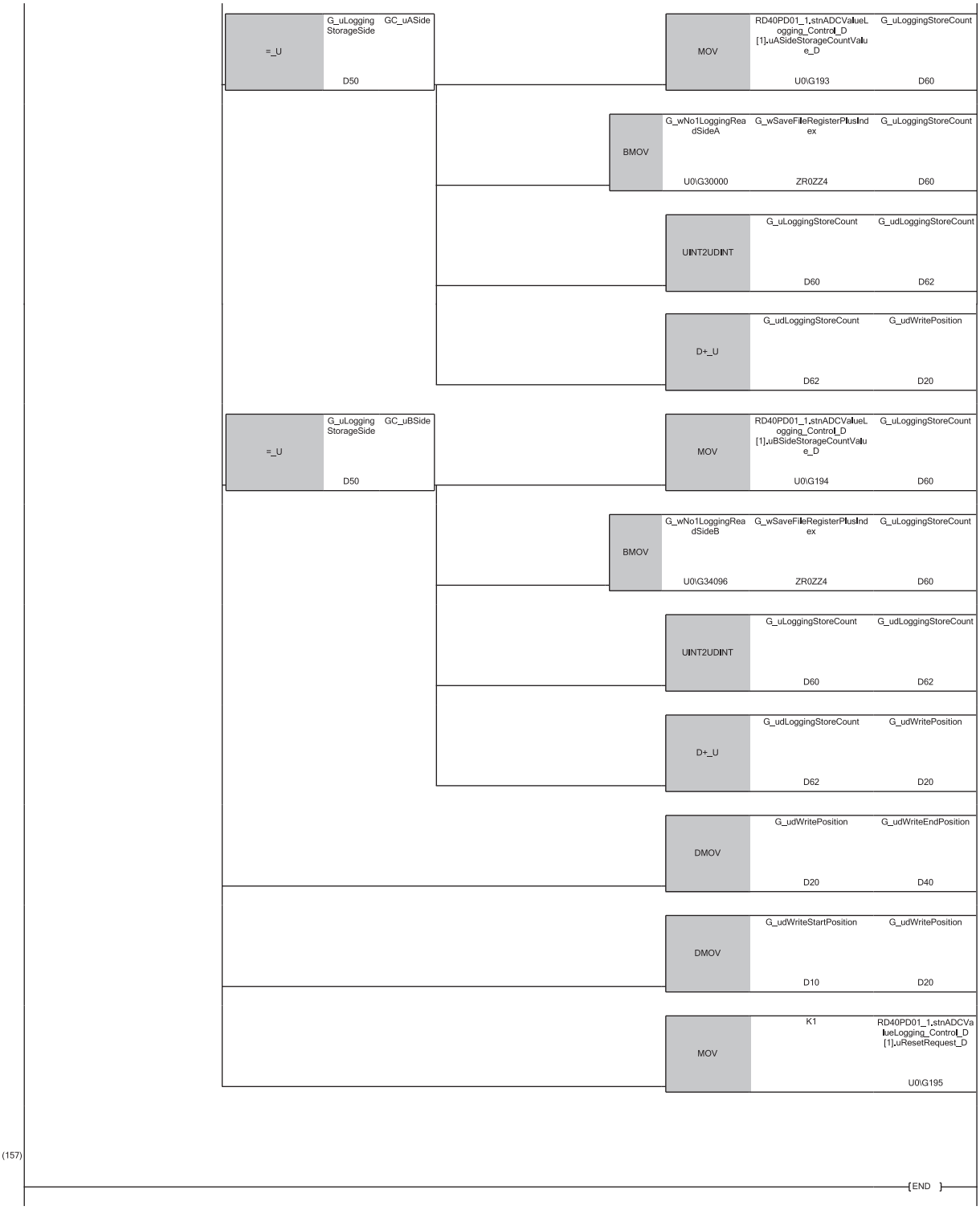
Classification	Label name	Description	Device
Module label	RCPU.stSM.bAfter_RUN1_Scan_ON	ON for one scan after RUN	SM402
	RD40PD01_1.bModuleREADY	Module READY	X0
	RD40PD01_1.bHardwareLogicControlFlag	Hardware logic control flag	X4
	RD40PD01_1.bHardwareLogicControlStartRequest	Hardware logic control start request	Y4
	RD40PD01_1.stnADCValueLogging_Control_D[1].uStorageSideMonitor_D	No.1 ADCValue logging data storage monitor	U0\G190
	RD40PD01_1.stnADCValueLogging_Control_D[1].uASideStorageFlag_D.0	No.1 ADCValue logging data A side storage flag	U0\G191.0
	RD40PD01_1.stnADCValueLogging_Control_D[1].uBSideStorageFlag_D.0	No.1 ADCValue logging data B side storage flag	U0\G192.0
	RD40PD01_1.stnADCValueLogging_Control_D[1].uASideStorageCountValue_D	No.1 ADCValue logging data A side storage count value	U0\G193
	RD40PD01_1.stnADCValueLogging_Control_D[1].uBSideStorageCountValue_D	No.1 ADCValue logging data B side storage count value	U0\G194
	RD40PD01_1.stnADCValueLogging_Control_D[1].uResetRequest_D	No.1 ADCValue logging reset request	U0\G195
Label to be defined	Define global labels as shown below:  Although 'G_wSaveRegisterPlusIndex' operates without problem if it is used with the class of VAR_GLOBAL, using this label in the same way on GX Works3 of Ver.1.055H or later will cause a conversion error on the consistency check of global labels. Set the class to VAR_GLOBAL_RETAIN when using 'G_wSaveRegisterPlusIndex' on GX Works3 of Ver.1.055H or later.		

	Label Name	Data Type		Class	Assign (Device/Label)	Constant
1	G_udWriteStartPosition	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D10	
2	G_udWritePosition	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D20	
3	G_udSaveFileRegisterMaxValue	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D30	
4	G_udWriteEndPosition	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D40	
5	G_uLoggingStorageSide	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D50	
6	G_uLoggingStoreCount	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D60	
7	G_udLoggingStoreCount	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D62	
8	G_uLoggingTransferSize	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D70	
9	G_udLoggingTransferSize	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D72	
10	G_udWritePositionIndex	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	Z4	
11	G_wNo1LoggingReadSideA	Word [Signed]	...	VAR_GLOBAL	U0\G30000	
12	G_wNo1LoggingReadSideB	Word [Signed]	...	VAR_GLOBAL	U0\G34096	
13	G_bLoggingStoreReq	Bit	...	VAR_GLOBAL	M0	
14	G_bHardwareLogicControlStartReq	Bit	...	VAR_GLOBAL	M110	
15	G_bErrorDetection	Bit	...	VAR_GLOBAL	F0	
16	G_wSaveFileRegisterPlusIndex	Word [Signed]	...	VAR_GLOBAL_RETAIN	ZR0ZZ4	
17	GC_uASide	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL_CONSTANT		1
18	GC_uBSide	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL_CONSTANT		2

■Program


RCPU.stSM.bAfter_RUN1_Scan_ON
(0)
SM402
DMOV
K0
G_udWriteStartPosition
D10
DMOV
G_udWriteStartPosition
D10
G_udWritePosition
D20
DMOV
K102400
G_udSaveFileRegisterMaxValue
D30
G_bHardwareLogicControlStartReq
RD40PD01_1.bModuleREADY
RD40PD01_1.bHardwareLogicControlStopRequest
(9)
M110
X0
Y5
SET
RD40PD01_1.bHardwareLogicControlStartRequest
Y4
RD40PD01_1.bHardwareLogicControlStartRequest
RD40PD01_1.bHardwareLogicControlFlag
(14)
Y4
X4
RST
RD40PD01_1.bHardwareLogicControlStartRequest
Y4
RD40PD01_1.stnADCValueLogging_Control_D[1].uASideStorageFlag_D.0
RD40PD01_1.stnADCValueLogging_Control_D[1].uBSideStorageFlag_D.0
(17)
U0\G191.0
U0\G192.0
SET
G_bErrorDetection
F0
G_bErrorDetection
RD40PD01_1.stnADCValueLogging_Control_D[1].uASideStorageFlag_D.0
(26)
F0
U0\G191.0
DMOV
G_udWritePosition
D20
G_udWritePositionIndex
Z4
MOV
RD40PD01_1.stnADCValueLogging_Control_D[1].uASideStorageCountValue_D
U0\G193
G_uLoggingTransferSize
D70
BMOV
G_wNo1LoggingReadSideA
U0\G30000
G_wSaveFileRegisterPlusIndex
ZR0ZZ4
G_uLoggingTransferSize
D70
RD40PD01_1.stnADCValueLogging_Control_D[1].uBSideStorageFlag_D.0
U0\G192.0
SET
G_bErrorDetection
F0
G_bErrorDetection
F0
RST
RD40PD01_1.stnADCValueLogging_Control_D[1].uASideStorageFlag_D.0
U0\G191.0


UINT2UDINT
G_uLoggingTransferSize
D70
G_udLoggingTransferSize
D72
D+_U
G_udLoggingTransferSize
D72
G_udWritePosition
D20
(60)
G_bErrorDetection
F0
RD40PD01_1.stnADCValueLogging_Control_D[1].uBSideStorageFlag_D.0
U0\G192.0
DMOV
G_udWritePosition
D20
G_udWritePositionIndex
Z4
MOV
RD40PD01_1.stnADCValueLogging_Control_D[1].uBSideStorageCountValue_D
U0\G194
G_uLoggingTransferSize
D70
BMOV
G_wNo1LoggingReadSideB
U0\G34096
G_wSaveFileRegisterPlusIndex
ZR0ZZ4
G_uLoggingTransferSize
D70
RD40PD01_1.stnADCValueLogging_Control_D[1].uASideStorageFlag_D.0
U0\G191.0
SET
G_bErrorDetection
F0
G_bErrorDetection
F0
RST
RD40PD01_1.stnADCValueLogging_Control_D[1].uBSideStorageFlag_D.0
U0\G192.0
UINT2UDINT
G_uLoggingTransferSize
D70
G_udLoggingTransferSize
D72
D+_U
G_udLoggingTransferSize
D72
G_udWritePosition
D20
(94)
D<=_U
G_udSaveFileRegisterMaxValue
D30
G_udWritePosition
D20
DMOV
G_udWriteStartPosition
D10
G_udWritePosition
D20
(99)
G_bErrorDetection
F0
G_bLoggingStoreReq
M0
DMOV
G_udWritePosition
D20
G_udWritePositionIndex
Z4
MOV
RD40PD01_1.stnADCValueLogging_Control_D[1].uStorageSideMonitor_D
U0\G190
G_uLoggingStorageSide
D50


=_U
G_uLogging StorageSide
GC_uASide
D50
MOV
RD40PD01_1.stnADCValueLogging_ControlI_D[1].uASideStorageCountValue_D
G_uLoggingStoreCount
U0\G193
D60
BMOV
G_wNo1LoggingReadSideA
G_wSaveFileRegisterPlusIndex
G_uLoggingStoreCount
U0\G30000
ZR0ZZ4
D60
UINT2UDINT
G_uLoggingStoreCount
G_udLoggingStoreCount
D60
D62
D+_U
G_udLoggingStoreCount
G_udWritePosition
D62
D20
=_U
G_uLogging StorageSide
GC_uBSide
D50
MOV
RD40PD01_1.stnADCValueLogging_ControlI_D[1].uBSideStorageCountValue_D
G_uLoggingStoreCount
U0\G194
D60
BMOV
G_wNo1LoggingReadSideB
G_wSaveFileRegisterPlusIndex
G_uLoggingStoreCount
U0\G34096
ZR0ZZ4
D60
UINT2UDINT
G_uLoggingStoreCount
G_udLoggingStoreCount
D60
D62
D+_U
G_udLoggingStoreCount
G_udWritePosition
D62
D20
DMOV
G_udWritePosition
G_udWriteEndPosition
D20
D40
DMOV
G_udWriteStartPosition
G_udWritePosition
D10
D20
MOV
K1
RD40PD01_1.stnADCValueLogging_ControlI_D[1].uResetRequest_D
U0\G195
(157)
{END }

(0) The settings are configured as follows by the contact turning on for one scan after RUN.
 • 'G_udWriteStartPosition' (D10) is set to 0 so that ZR0 is set as the start position of the save destination file register.
 • G_udWriteendPosition (D20) is set to a value in 'G_udWriteStartPosition' (D10) so that the write position of the file register is initialized.
 • 'G_udSaveFileRegisterMaxValue' (D30) is set to 102400, which is the maximum number of storage points of the save destination file register.
(9) Turn on 'G_bHardwareLogicControlStartReq' (M110) to start hardware logic control (A/D conversion value logging).
(26)When 'No.1 ADCValue logging data A side storage flag' (Un\G191) is turned on, the values of No.1 ADCValue logging data storage area (A side) (Un\G30000 to Un\G34095) are transferred to the file register.
(60)When 'No.1 ADCValue logging data B side storage flag' (Un\G192) is turned on, the values of No.1 ADCValue logging data storage area (B side) (Un\G34096 to Un\G38191) are transferred to the file register.
(94)If a number of logging data exceeding ZR102399 is stored, storage of logging data is repeated from ZR0. Depending on the file register write position and the transfer size of the logging data, data may be stored up to ZR106494.
(99)If untransferred logging data remains in the ADCValue logging data storage area due to external pulse stop, turn on 'G_bLoggingStoreReq' (M0) to transfer the remaining logging data. After the transfer of logging data is completed, the logging stop position of the file register is stored in 'G_udWriteEndPosition' (D40). Also, the file register storage position is returned to the start position (ZR0), and the A/D conversion value logging status is reset.

Precautions

To prevent logging data omission, configure the program so that the program that transfers data to the file register is executed before the next trigger switching is executed.

1.5 Interrupt Function

This function notifies the CPU module that either of following conditions is satisfied as an interrupt request.

- A signal is input to the SI device terminal of the hardware logic.
- The logging data for 5120 points is stored in Continuous logging data storage area (Un\G15020 to Un\G25259).

For the flexible high-speed I/O control module, the maximum number of interrupt pointers available is 9 per module.

Point

An interrupt program execution has a priority and interrupts the other processing in progress. Therefore, if interrupt programs are created in advance, processing for when a match of multi function counter block is detected in the hardware logic, fail-safe processing, file register transfer processing for logging data, and other processing can be executed immediately.

Operation

- When an input signal event is detected or a comparison condition is satisfied, a signal is input from the "Output" terminal of a multi function counter block to an SI device terminal. The interrupt program corresponding to the SI device terminal to which the signal has been input is executed.
- After logging data for 5120 points has been stored by the continuous logging function, the interrupt program is executed.

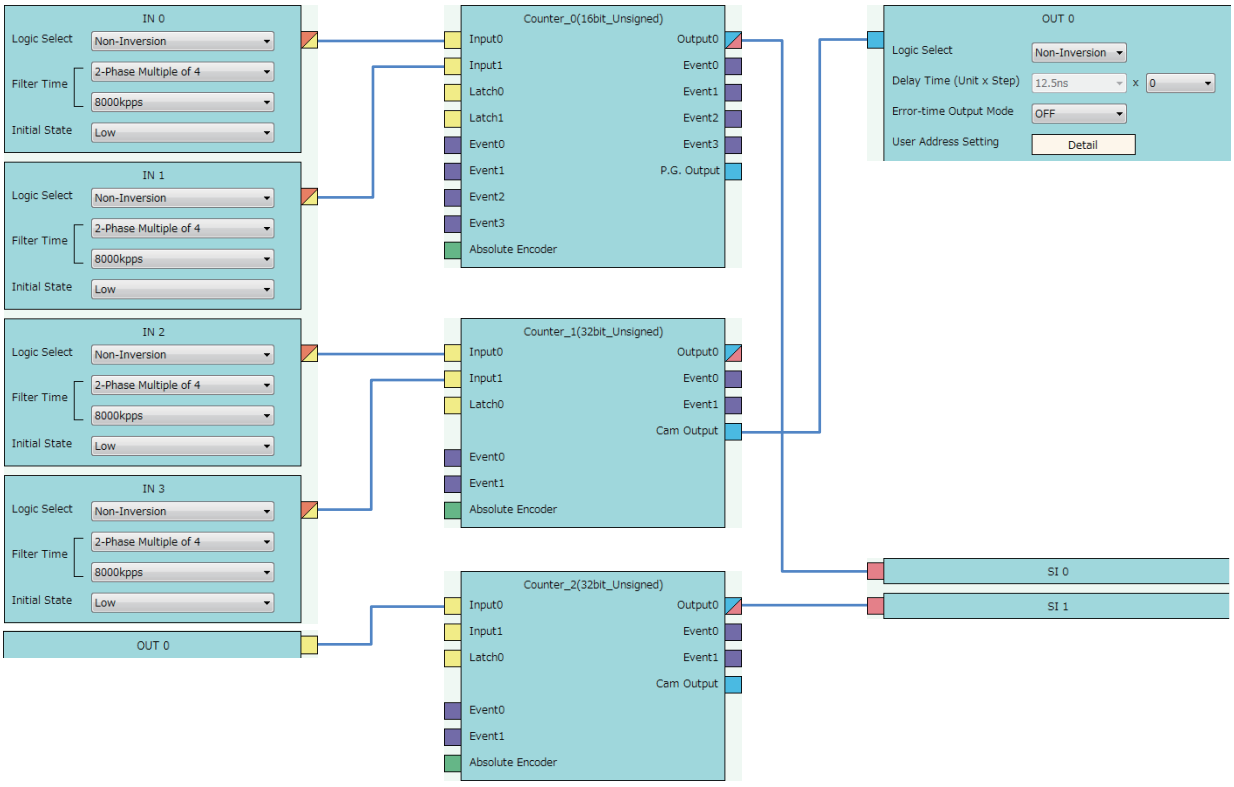
Setting method

The following describes the setting method for when a signal is input to the SI device terminal of the hardware logic.
To use the interrupt function, the hardware logic and [Interrupt setting] of [Module Parameter] are required to be set in advance.

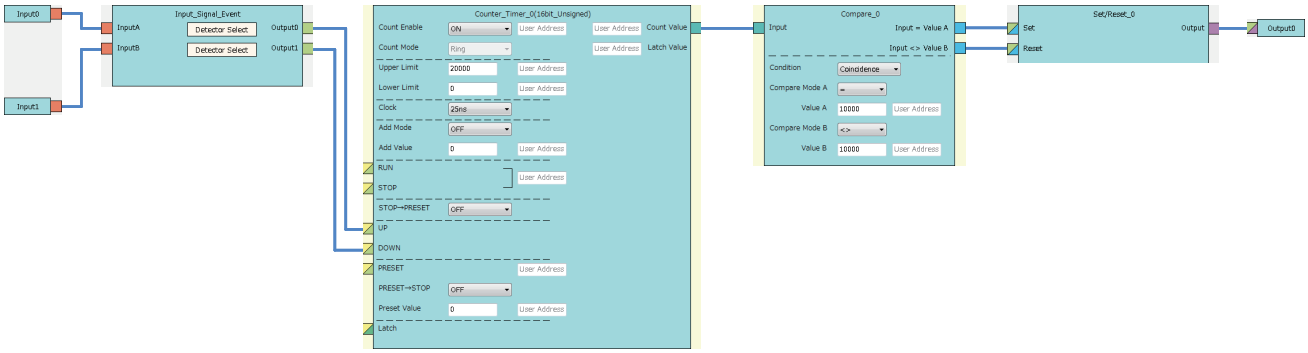
■Setting the hardware logic

The following shows link examples of the hardware logic.

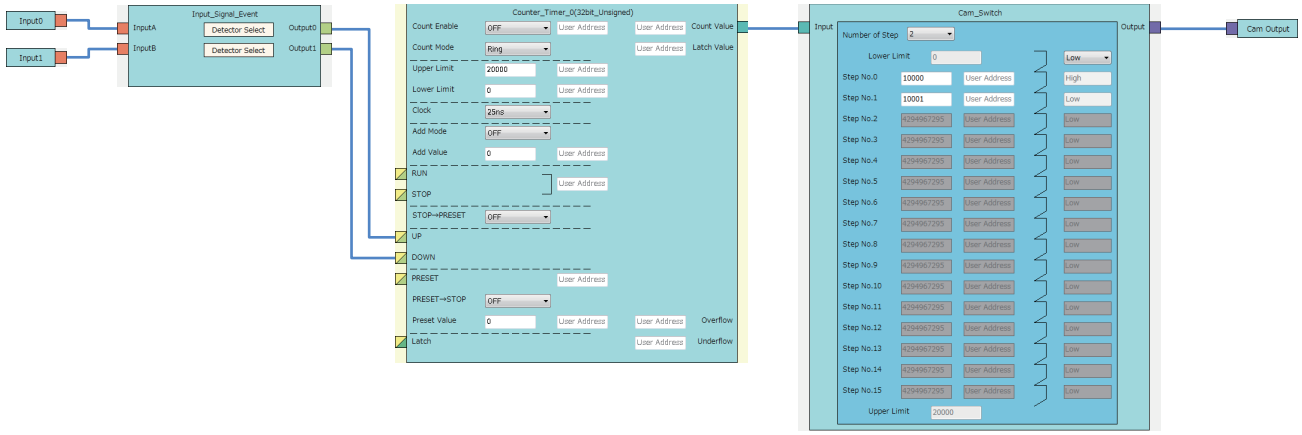
- Hardware logic outline window



• Multi function counter block detail window (Counter_0(16bit_Unsigned))



• Multi function counter block detail window (Counter_1(32bit_Unsigned))

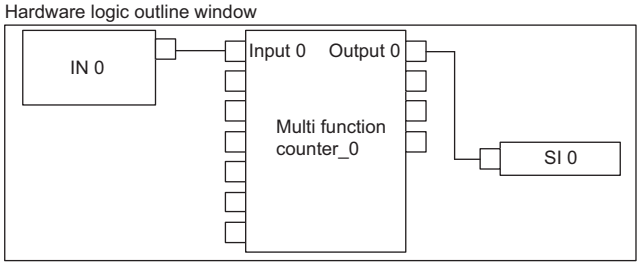


• Multi function counter block detail window (Counter_2(32bit_Unsigned))

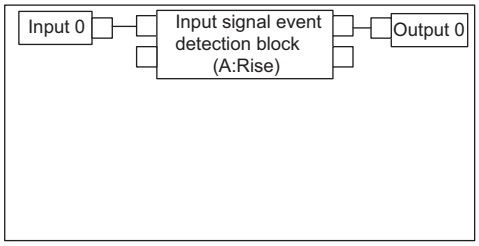
Point

• An interrupt request to the CPU module can also be output using a detection in the input signal event detection block as a trigger. The following figure shows a link example in the Hardware logic outline window and the Multi function counter block detail window.

Hardware logic outline window



Multi function counter block detail window



• For the linking of the multi function counter block detail window (Counter_0(16bit_Unsigned)), a value other than 0 is recommended for "Compare Value". If 0 is set in "Compare Value", an interrupt request is output to the CPU module when the operation of the hardware logic starts (when the count value starts to be counted from 0).

■"Interrupt setting" of "Module Parameter"

To use the interrupt function, set "Condition target setting" and "Interrupt pointer" with the engineering tool. After completing the settings, write the project to enable the settings.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ Module model name ⇨ [Module Parameter] ⇨ [Interrupt setting]

Item	Description
Condition target setting	The SI device terminal 0 (SI 0) to SI device terminal 7 (SI 7) and Continuous logging data storage are assigned to the interrupt number 1 to 9.
Interrupt pointer	Specify the number of an interrupt pointer that is initiated at the detection of an interrupt factor. For details on the interrupt pointers, refer to the following. 📖 MELSEC iQ-R CPU Module User's Manual (Application)

Point

When the same factor occurs again after the execution of an interrupt program, an interrupt request is output to the CPU module to start an interrupt program.

Precautions

- When "Interrupt pointer" in "Interrupt setting" is not set, even though an SI device terminal in the hardware logic turns to High or logging data for 5120 points is stored by the continuous logging function, an interrupt request to the CPU module is not output.
- To issue an interrupt request to the CPU module when continuous logging data for 5120 points is stored, set "Enable" in "Continuous logging data storage interrupt enable/disable" of "Application setting" in addition to setting "Interrupt setting".
- When multiple interrupt factors occur at the same time, the interrupt programs are executed in order of the priority of the interrupt pointers. For the priority of the interrupt pointers, refer to the following.

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

- When multiple interrupt factors occur at the same time, the CPU module executes multiple interrupt programs at the same time. When the time until the completion of all interrupt programs is long, the scan monitoring function of the CPU module judges that the programs do not complete successfully, and a CPU module error may occur. When a CPU error occurs, refer to the following.

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

Restrictions

■Cycle of interrupt request to CPU module

An interrupt request to the CPU module is output every 50μs. If the same interrupt factor (SI) occurs several times within 50μs, the second and later factors (SI) are ignored. If the interrupt function is used under any of the following conditions, an interrupt request to the CPU module is issued once every 100μs or less.

- When the flexible high-speed I/O control module is in the inter-module synchronous mode and "User Address" of the input terminals or parameters is assigned to Hardware logic area (High speed area) (Un\G1000 to Un\G1029)
- When the A/D conversion value logging function is used

■Interrupt response delay time

When the continuous logging function is used, the maximum response delay time of interrupt is 200μs.

Setting example

Ex.

When 'General command 0' (Y10) is turned on in the interrupt program (I50)

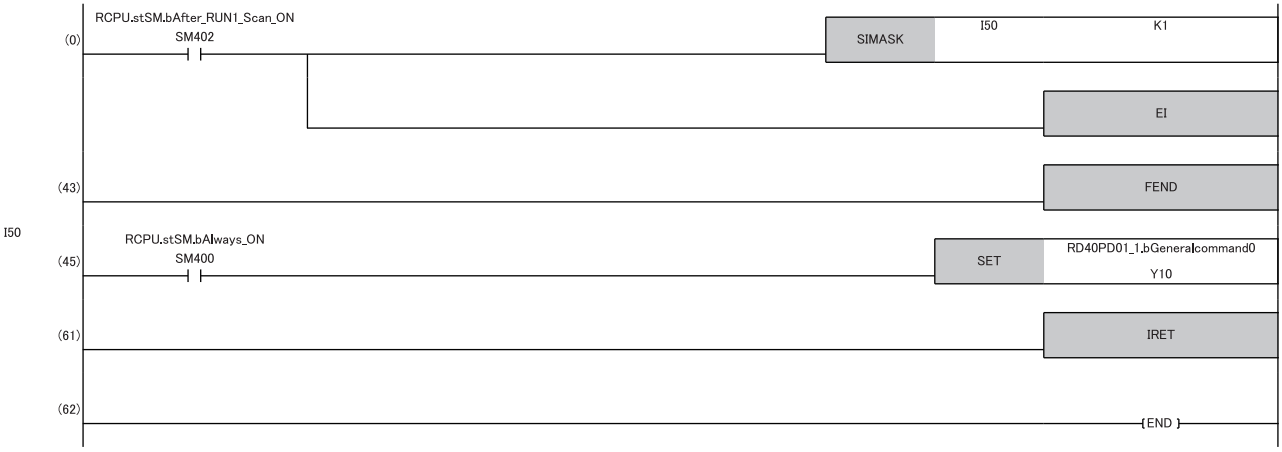
- Parameter setting

Set "Interrupt setting" of the module parameter as follows.

No.	Condition target setting	Interrupt pointer
1	SI device terminal 0 (SI0)	I50

- Label settings

Classification	Label name	Description	Device
Module label	RCPU.stSM.bAlways_ON	Always ON	SM400
	RCPU.stSM.bAfter_RUN1_Scan_ON	ON for one scan after RUN	SM402
	RD40PD01_1.bGeneralcommand0	General command 0	Y10



(0) Enable only the interrupt pointer I50.
(45)Turn on 'General command 0' (Y10).

1.6 Inter-Module Synchronization Function

This function synchronizes the control timings of multiple flexible high-speed I/O control modules.

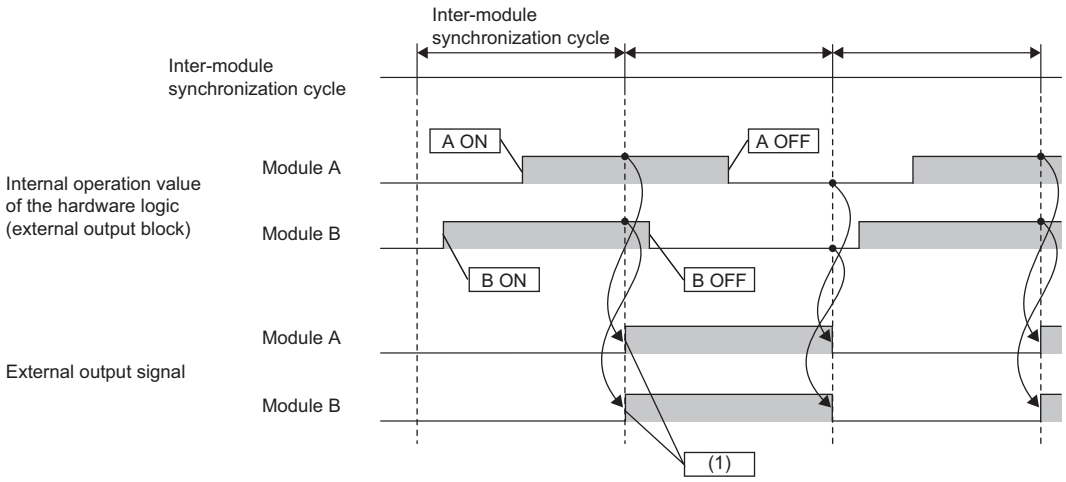
Overview

The following table lists the targets that can be synchronized by the inter-module synchronization function.

Target	Timing	Overview
Synchronous output of external output signals	Inter-module synchronization signal	The High/Low states of the external output terminals are changed at a time.
Latch of the count values and the states of external input signals		• The latch count values of multi function counter blocks are refreshed to the buffer memory areas. • The High/Low states of external input signals are latched and refreshed to the X signals.
External output of the inter-module synchronization signal	Always	The inter-module synchronization signal is output from the external output terminals.

■Synchronous output of external output signals

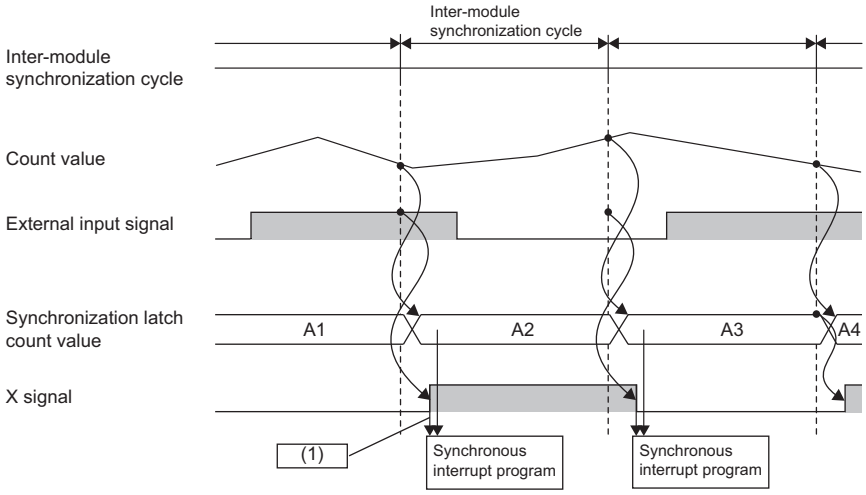
The High/Low states of the external output signals are changed at a time in synchronization with the inter-module synchronization signal. Synchronous output synchronizes the control timings of external devices connected to multiple flexible high-speed I/O control modules.



A ON: The ON condition for the module A is satisfied.
A OFF: The OFF condition for the module A is satisfied.
B ON: The ON condition for the module B is satisfied.
B OFF: The OFF condition for the module B is satisfied.
(1) The ON timings of the external output signals are synchronized at the inter-module synchronization cycle.

■Latch of the count values and the states of external input signals

The count values of all the multi function counter blocks and the High/Low states of external input signals are latched in synchronization with the inter-module synchronization signal. After the latch count values are stored in the buffer memory areas and the external input signals are stored in the X signals, the latch count values and the High/Low states of the external input signals at the latest synchronization cycle can be referred to in the inter-module synchronous interrupt program.
When the inter-module synchronization function is used, basically perform the control using the inter-module synchronous interrupt program (I44).



(1) The count values and external input signals at the latest synchronization cycle that have been obtained at the same time are referred to.

■External output of the inter-module synchronization signal

When an external output block is linked from the inter-module synchronization signal input terminal, the inter-module synchronization signal can be output from the external output terminals. Using this output signal allows extending the synchronous control to external devices that do not have the inter-module synchronization function.

Setting method

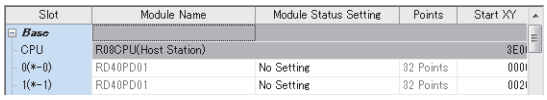
■System parameter setting

To use the inter-module synchronization function, set the inter-module synchronization function in "System Parameter" and set the flexible high-speed I/O control module for an inter-module synchronization target module.

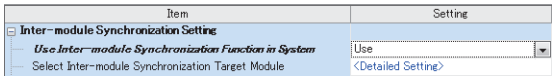
The following describes an example of the setting procedure.

[Navigation window] ⇨ [Parameter] ⇨ [System Parameter] ⇨ [Inter-module Synchronization Setting]

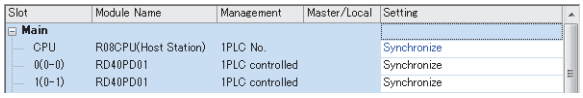
"I/O Assignment" window



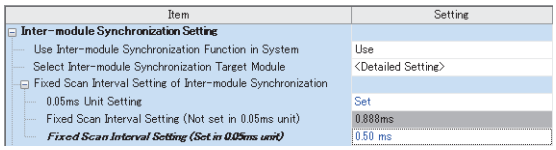
"Inter-module Synchronization Setting" window



"Select the Synchronous Target Module" window



"Inter-module Synchronization Setting" window



1. Set the I/O assignment according to the system used.
2. Select "Use" for "Use Inter-module Synchronization Function in System".
3. Double-click "Detailed Setting" of "Select Inter-module Synchronization Target Module".
4. Set the flexible high-speed I/O control module to "Synchronize".
5. Set the inter-module synchronization cycle in "Fixed Scan Interval Setting of Inter-module Synchronization".
6. Write the system parameter into the CPU module and change the state of the CPU module from RESET to RUN. When the inter-module synchronization function is enabled, the inter-module synchronization status (during synchronization or synchronization suspended) can be monitored in 'Synchronization status monitor' (Un\G8173).

■Setting an interrupt program

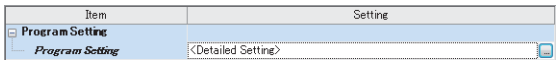
Set an interrupt program that refers to Synchronization latch count value (Un\G700 to Un\G715) latched in synchronization with the inter-module synchronization cycle.

The interrupt program indicates a program from the interrupt pointer (I) to the IRET instruction and is executed at a cycle set by users. The flexible high-speed I/O control module refreshes Synchronization latch count value (Un\G700 to Un\G715) when the interrupt program starts to operate.

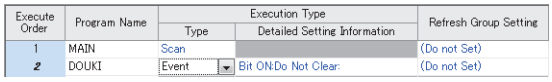
The following describes the setting method of an interrupt program.

[Navigation window] ⇨ [Parameter] ⇨ CPU module model name ⇨ [CPU Parameter] ⇨ [Program Setting]

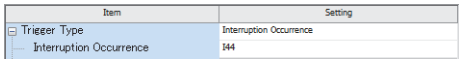
"CPU Parameter" window



"Program Setting" window



"Event Execution Type Detailed Setting" window



1. Double-click "Detailed Setting" of "Program Setting".
2. Set the name of an interrupt program for "Program Name".
3. Set "Type" of "Execution Type" to "Event".
4. Double-click "Detailed Setting Information" of "Execution Type".
5. Set "Trigger Type" to "Interruption Occurrence".
6. Set "I44" for "Interruption Occurrence".
7. Create an interrupt program.

The interrupt program is executed at the inter-module synchronization cycle set by users. For details on the interrupt program, refer to the following.

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

Point

When the flexible high-speed I/O control module is set for a synchronization target, the I/O refresh is executed before and after the synchronous interrupt program. Thus, when the I/O refresh needs to be reflected to multiple modules at the same time, do not use the direct access (DY). When the direct access is used, the I/O refresh cannot be reflected to the modules at the same time but is reflected to each module at the execution of an instruction.

■Setting and link of the hardware logic

The following table lists the setting and link of the hardware logic required to use the inter-module synchronization function.

Synchronization target	Setting and link of the hardware logic
Synchronous output of external output signals	Set "SYNC" for "Delay Time(Unit)" of an external output block. When "SYNC" is not set, the external output signals are output according to the setting value of "Delay Time" in the external output block.
External output of the inter-module synchronization signal	Link an external output block from the inter-module synchronization signal input terminal. If an external output block is not linked from the inter-module synchronization signal input terminal through a logical operation block, the inter-module synchronization signal cannot be output from the external output terminals.

Operation

■Timing to start the inter-module synchronization cycle

• Refreshing the synchronization latch count values and the states of external input signals

The count values at the timing to start the inter-module synchronization cycle are latched to Synchronization latch count value (Un\G700 to Un\G715). At the same time, the states of external input signals are refreshed to 'IN 0 to IN B' (X10 to X1B). Synchronization latch count value (Un\G700 to Un\G715) and the X signals are read to the CPU module at a time by the input refresh processing of the CPU module. The latch count values and the states of the external input signals obtained at the same time among multiple modules can be used in an inter-module synchronous interrupt program.

Synchronization latch count value (Un\G700 to Un\G715) is not refreshed while the hardware logic control is stopped. When "Select Inter-module Synchronization Target Module" is set to "Do not Synchronize" in GX Works3, Synchronization latch count value (Un\G700 to Un\G715) is fixed to 0.

• Synchronous output of the external output terminals

When "SYNC" is set for "Delay Time(Unit)" of an external output block, the High/Low states of the external output block is determined by the operation result of the hardware logic control during the last inter-module synchronization cycle. The states are output from the external output terminals at a time in synchronization with the inter-module synchronization signal. The states are held until the next inter-module synchronization cycle.

Using synchronous output of the external output terminals synchronizes the timings to issue commands to external devices connected to multiple modules.

When "Select Inter-module Synchronization Target Module" is set to "Do not Synchronize" in GX Works3 and "SYNC" is set for "Delay Time(Unit)" of an external output block, or the state of the CPU module is changed to STOP, output of the external output terminals does not change because output of the inter-module synchronization signal stops.

■Refreshing General command and User Address from an inter-module synchronous interrupt program

The operation result of an inter-module synchronous interrupt program is sent to the flexible high-speed I/O control module by the output refresh processing from the CPU module.

After the completion of the output refresh, 'General command 0 to General command F' (Y10 to Y1F) and User Address (control data) that is assigned to Hardware logic area (High speed area) (Un\G1000 to Un\G1029) are immediately reflected to the hardware logic control. The following table lists User Address (control data) to be reflected.

User Address	Data type	Reflection	Reflection timing for the hardware logic control
Hardware logic area (High speed area) (Un\G1000 to Un\G1029)	Control	Performed	The values are refreshed at every inter-module synchronization cycle. (The values at the completion of an inter-module synchronous interrupt program are refreshed.)
	Monitor	Not performed	The values are refreshed every 1ms cycle.
Hardware logic area (Low speed area) (Un\G1030 to Un\G1099)	Control or monitor	Not performed	

■Internal actions of the hardware logic control

When "Select Inter-module Synchronization Target Module" is set to "Synchronize" in GX Works3, the following operations are performed.

- Synchronization latch count value (Un\G700 to Un\G715) is refreshed only at the timing to start the inter-module synchronization cycle.
- The High/Low input to the external input terminals or internal operations (such as a count operation) using the pulse input are continuously performed at the internal operation cycle of the module.
- The refresh processing of the data in Hardware logic area (Un\G1000 to Un\G1099) assigned to User Address is continuously performed at the internal operation cycle of the module.

Note that when data other than Synchronization latch count value (Un\G700 to Un\G715) is referred to in an inter-module synchronous interrupt program, the above operations are performed.

■Accuracy of the inter-module synchronization

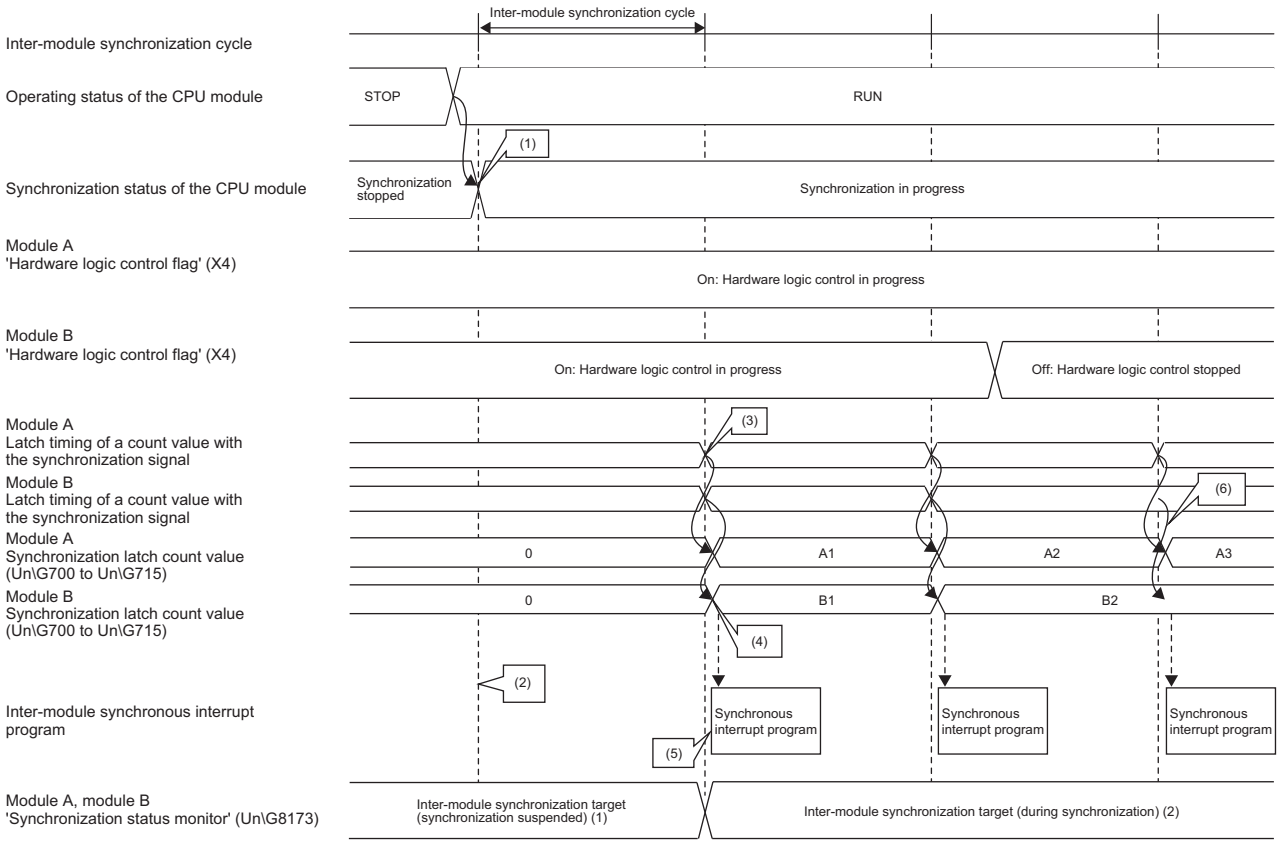
Accuracy	Description
Inter-module synchronization accuracy	The detection accuracy of the timing to start the inter-module synchronization cycle of this module (inter-module synchronization accuracy) differs depending on the system configuration (mounting position of the module). For details, refer to the following. 📖 MELSEC iQ-R Inter-Module Synchronization Function Reference Manual
Synchronization accuracy of the external output signals	When "SYNC" is set for "Delay Time(Unit)" of an external output block, the input status to the external output block is reflected to the external output within 1μs (output response time) from the timing to start the inter-module synchronization cycle. Thus, the external output timing differs among multiple flexible high-speed I/O control modules by the time of 1μs (output response time) plus the inter-module synchronization accuracy.

Operation examples

■Basic operations

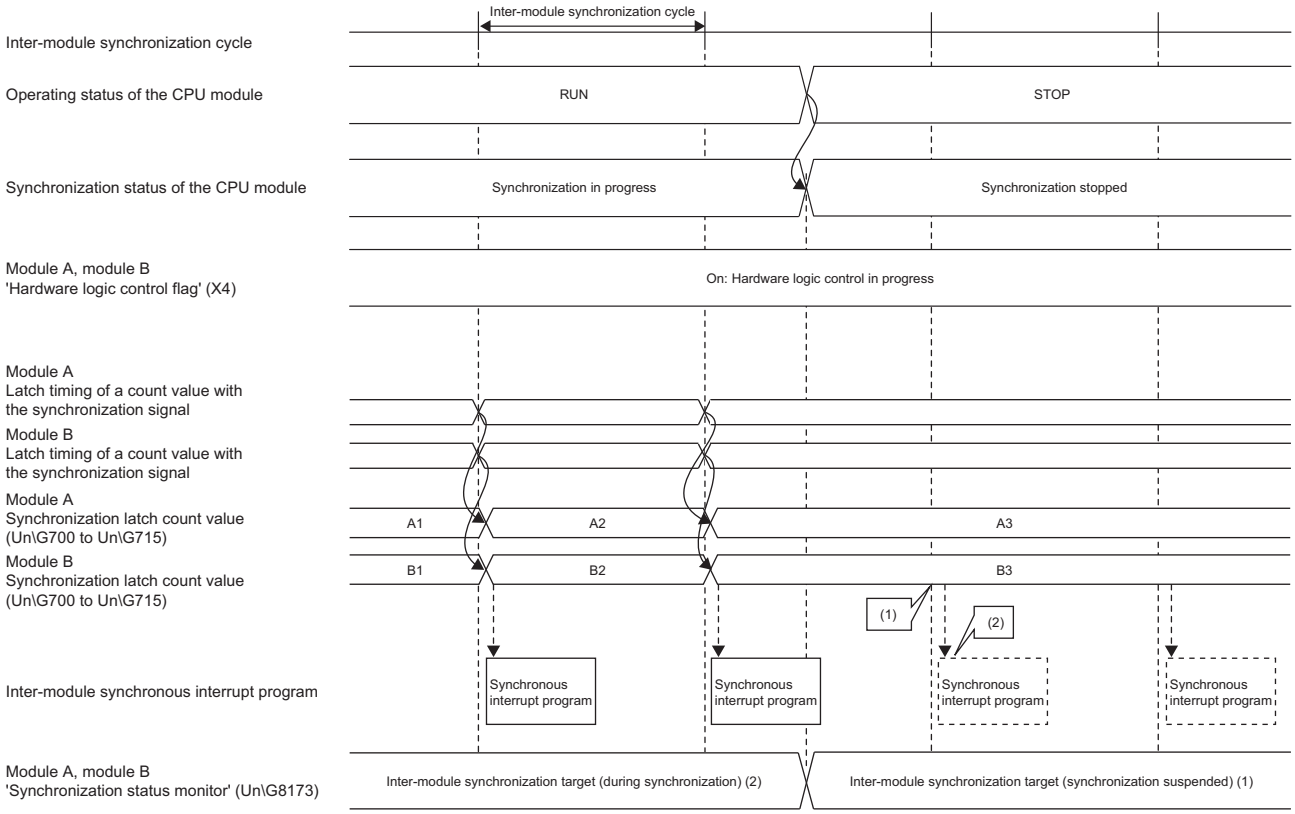
The following shows the basic operations when two flexible high-speed I/O control modules are in the synchronous operating status.

- Changes in the synchronous operations when the state of the CPU module is set from STOP to RUN (The flexible high-speed I/O control modules store the latest latch count values in Synchronization latch count value (Un\G700 to Un\G715) from the timing to start the inter-module synchronization cycle until the execution of the synchronous interrupt program.)



(1) When the state of the CPU module is set from STOP to RUN, the synchronization is started at the next timing to start the inter-module synchronization cycle.
(2) At the first inter-module synchronization cycle after the state of the CPU module is set to RUN, the synchronous interrupt program is not executed.
(3) When the CPU module is in the synchronization status, the count values are latched at the timing to start the inter-module synchronization cycle.
(4) The values synchronized among multiple modules are stored in Synchronization latch count value (Un\G700 to Un\G715).
(5) The program is executed from the next inter-module synchronization cycle.
(6) While the hardware logic control is stopped, Synchronization latch count value (Un\G700 to Un\G715) is not changed.

• Changes in the synchronous operations when the state of the CPU module is set from RUN to STOP



(1) When the CPU module is in the STOP state, Synchronization latch count value (Un\G700 to Un\G715) is not refreshed.
(2) When the CPU module is in the STOP state, the synchronous interrupt program is not executed.

Faulty operations during synchronization

Faulty operations during synchronization have the following causes.

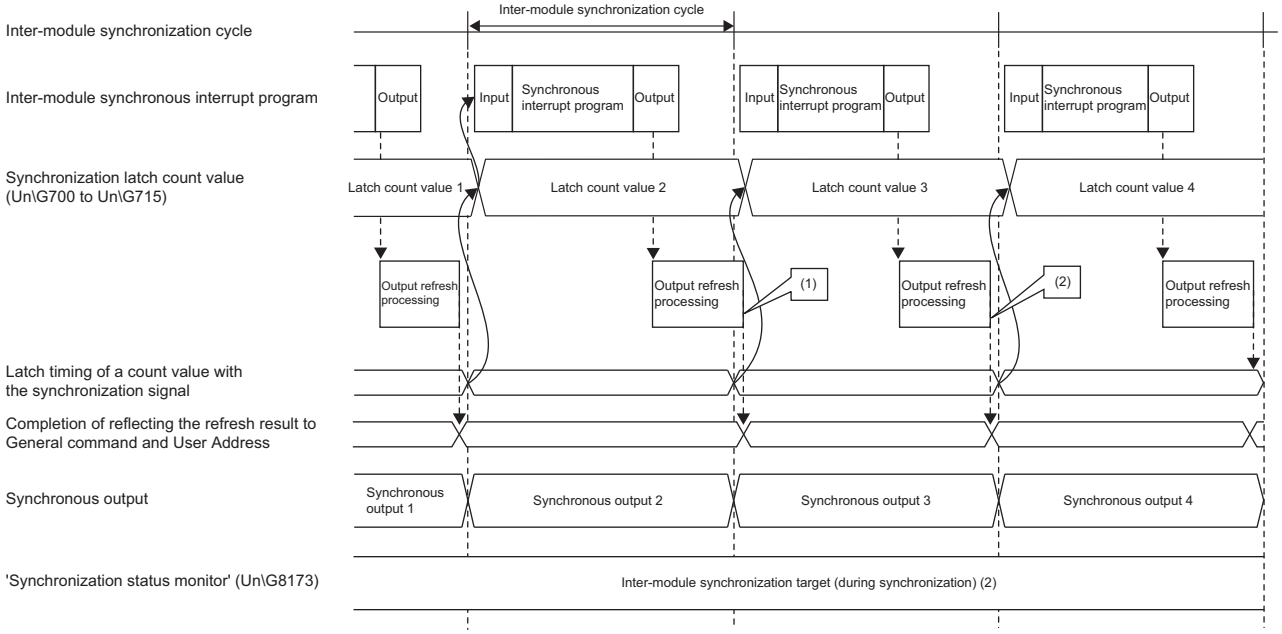
- Cycle crossing over (☞ Page 71 Operations at a cycle crossing over)
- Cycle skip (☞ Page 73 Operations at a cycle skip)
- Synchronization loss (☞ Page 74 Operations at a synchronization loss)

■Operations at a cycle crossing over

The "cycle crossing over" is a phenomenon that the refresh processing of General command and User Address to the modules are performed crossing over multiple inter-module synchronization cycles due to the operation timing of a synchronous interrupt program. When a cycle crossing over occurs, an inter-module synchronization cycle crossing over error (error code: 2600H) occurs.

For the operations of the flexible high-speed I/O control module when this error occurs, whether to continue or stop the hardware logic control can be selected in "Hardware logic control selection during synchronization error occurrence" of the module parameter.

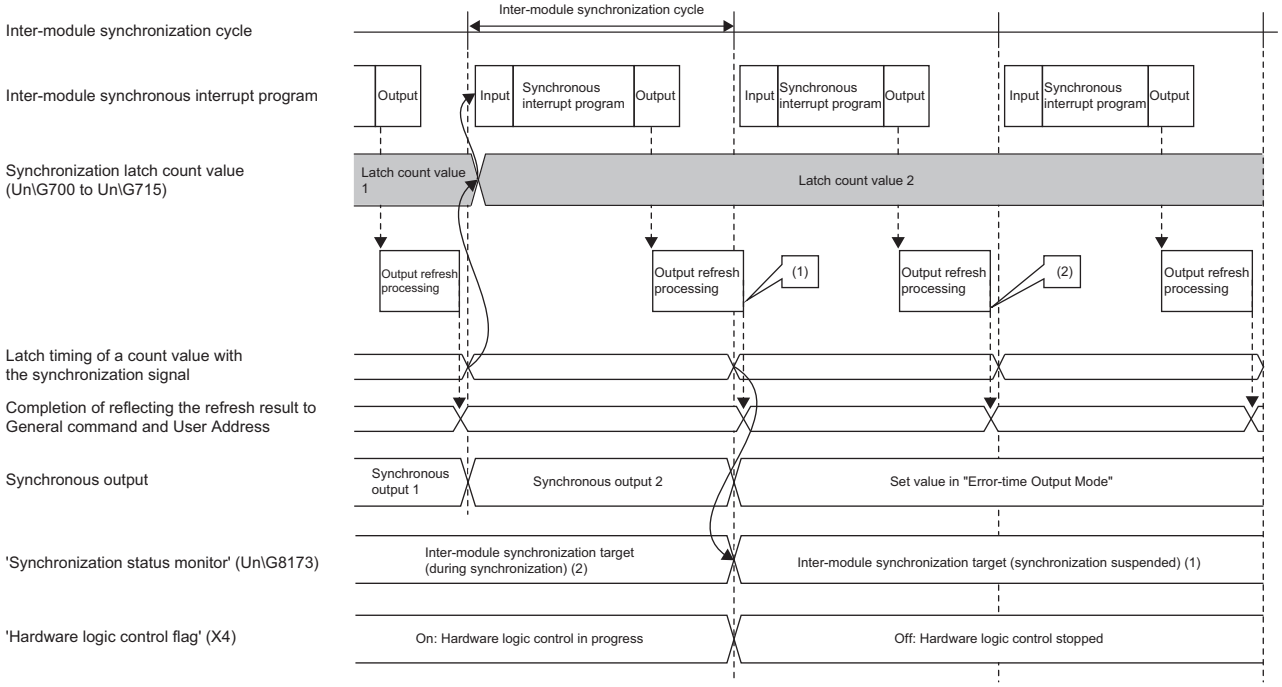
- The following shows the operations of the flexible high-speed I/O control module when "Hardware logic control continue" is set for "Hardware logic control selection during synchronization error occurrence" and a cycle crossing over occurs.



(1) A cycle crossing over occurs because the output refresh processing is performed crossing over the inter-module synchronization cycles. Even when a cycle crossing over occurs, Synchronization latch count value (Un\G700 to Un\G715) is refreshed because the latch timing of the count values is not changed. However, the values of 'General command 0 to General command F' (Y10 to Y1F) and User Address sent to the module by the output refresh processing may not be reflected to Synchronization latch count value (Un\G700 to Un\G715).

(2) When the output refresh processing is completed in an inter-module synchronization cycle even though the cycle crossing over has occurred, the values of 'General command 0 to General command F' (Y10 to Y1F) and User Address in the synchronous interrupt program are reflected to Synchronization latch count value (Un\G700 to Un\G715) at the next inter-module synchronization cycle.

- The following shows the operations of the flexible high-speed I/O control module when "Hardware logic control stop" is set for "Hardware logic control selection during synchronization error occurrence" and a cycle crossing over occurs.



(1) A cycle crossing over occurs because the output refresh processing is performed crossing over the inter-module synchronization cycles. When a cycle crossing over occurs, the hardware logic control is stopped.
(2) Even after the cycle crossing over has occurred, the values of 'General command 0 to General command F' (Y10 to Y1F) and User Address in the synchronous interrupt program are continuously reflected to the module.

When a cycle crossing over occurs, Synchronization latch count value (Un\G700 to Un\G715) and synchronous output to which only some of 'General command 0 to General command F' (Y10 to Y1F) and User Address (high speed area) have been refreshed may be output.

A cycle crossing over occurs under the following condition.

- Inter-module synchronization cycle < Synchronous interrupt program processing time + (Reflection time for General command and User Address (control data))

The following table lists the two possible causes of a cycle crossing over.

Cause	Action
The inter-module synchronization cycle is shorter than the time taken for reflecting General command and User Address (control data).	Set a longer inter-module synchronization cycle, or reduce the number of assignment points to Hardware logic area (High speed area) (Un\G1000 to Un\G1029) of User Address (control data). • Reflection of General command: 5μs • Reflection of User Address: 0.42μs per word
The run time of the inter-module synchronous interrupt program is long.	Reduce the number of steps of the synchronous interrupt program and shorten the run time of the synchronous interrupt program.

Ex.

When all the areas (30 words) of Hardware logic area (High speed area) (Un\G1000 to Un\G1029) are assigned to the control data, the processing time is as follows: 5(μs) + 0.42(μs) × 30 (words) = 17.6(μs).

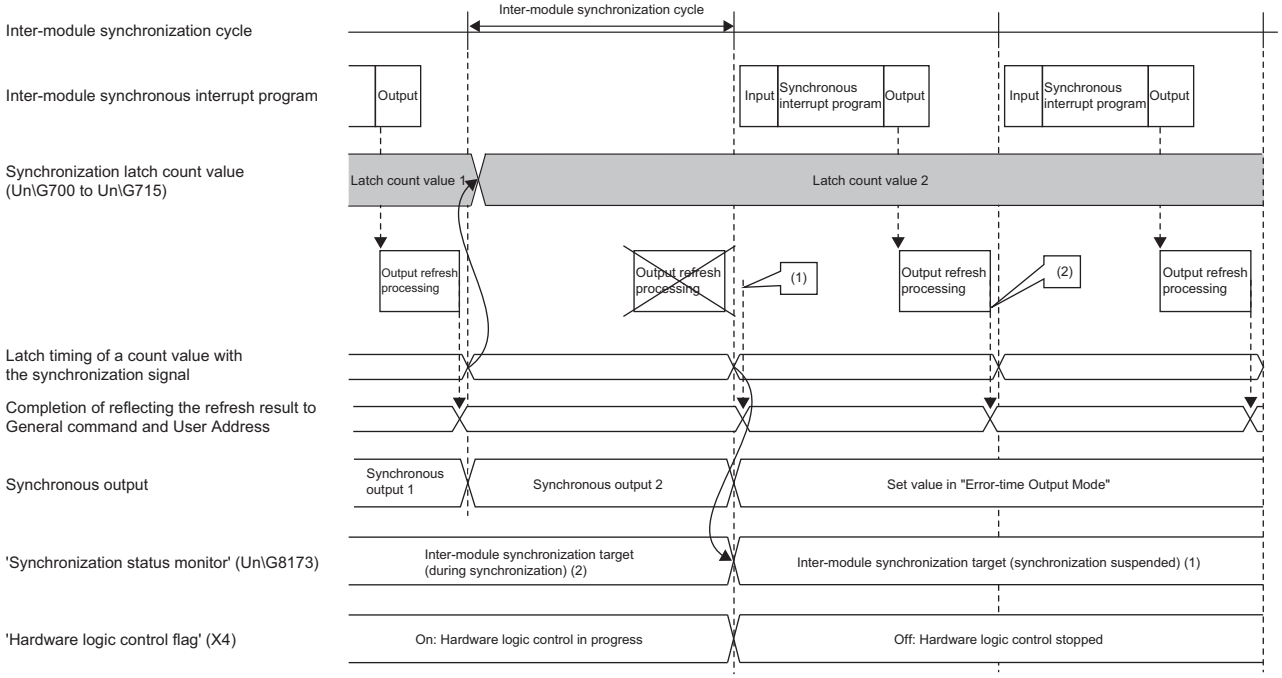
When 10 words of Hardware logic area (High speed area) (Un\G1000 to Un\G1029) are assigned to the control data, the processing time is as follows: 5(μs) + 0.42(μs) × 10 (words) = 9.2(μs).

As described above, when the number of assignment points of Hardware logic area (High speed area) (Un\G1000 to Un\G1029) are reduced from 30 points to 10 points, the processing time can be shortened by 8.4(μs).

■Operations at a cycle skip

The "cycle skip" is a phenomenon that the refresh processing of General command and User Address to the modules is not performed during an inter-module synchronization cycle due to the operation timing of a synchronous interrupt program. When a cycle skip occurs, an inter-module synchronization cycle skip error (error code: 2601H) occurs.

For the operations of the flexible high-speed I/O control module when this error occurs, the control of hardware logic stops regardless of the setting of "Hardware logic control selection during synchronization error occurrence" of the module parameter.



(1) A cycle skip occurs because the output refresh processing is not performed in the inter-module synchronization cycle. When a cycle skip occurs, the hardware logic control is stopped.

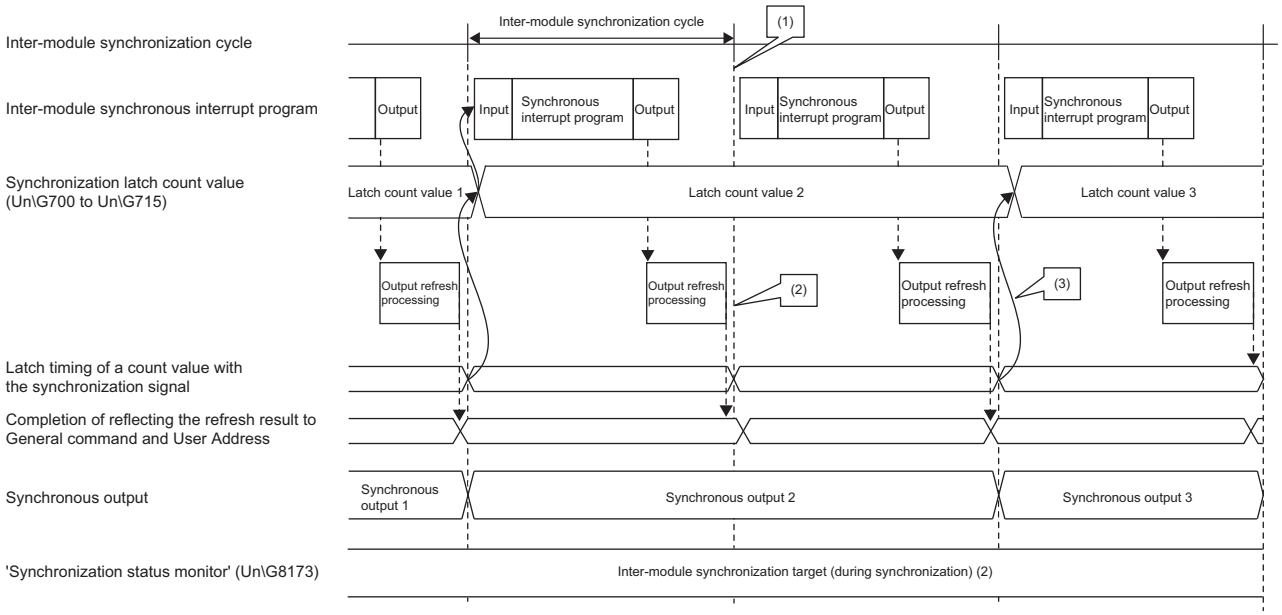
(2) When the output refresh processing is performed at the inter-module synchronization cycle after the cycle skip, the values of 'General command 0 to General command F' (Y10 to Y1F) and User Address in the synchronous interrupt program are reflected to the module.

■Operations at a synchronization loss

The "synchronization loss" is a phenomenon that the flexible high-speed I/O control module cannot receive an inter-module synchronization signal at the proper cycle. When a synchronization loss occurs, an inter-module synchronization signal error (error code: 2610H) occurs.

For the operations of the flexible high-speed I/O control module when this error occurs, whether to continue or stop the hardware logic control can be selected in "Hardware logic control selection during synchronization error occurrence" of the module parameter.

- The following shows the operations of the flexible high-speed I/O control module when "Hardware logic control continue" is set for "Hardware logic control selection during synchronization error occurrence" and a synchronization loss occurs.

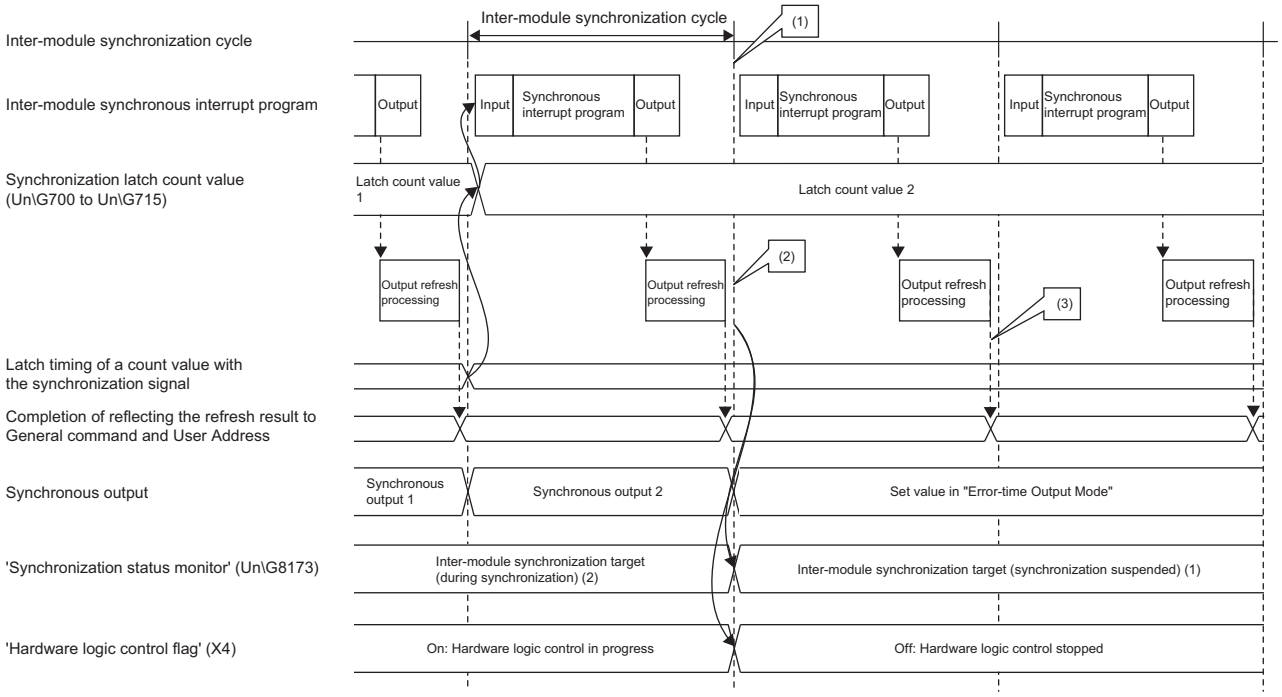


(1) A synchronization loss occurs because the flexible high-speed I/O control module failed to receive an inter-module synchronization signal.

(2) If the flexible high-speed I/O control module fails to receive the inter-module synchronization signal at the proper cycle, an inter-module synchronization signal error (error code: 2610H) occurs, and Synchronization latch count value (Un\G700 to Un\G715) and the synchronous output are not refreshed.

(3) When the inter-module synchronization signal is properly input to the flexible high-speed I/O control module after the synchronization loss, Synchronization latch count value (Un\G700 to Un\G715) and the synchronous output are refreshed.

- The following shows the operations of the flexible high-speed I/O control module when "Hardware logic control stop" is set for "Hardware logic control selection during synchronization error occurrence" and a synchronization loss occurs.



(1) A synchronization loss occurs because the flexible high-speed I/O control module failed to receive an inter-module synchronization signal.
(2) If the flexible high-speed I/O control module fails to receive the inter-module synchronization signal at the proper cycle, an inter-module synchronization signal error (error code: 2610H) occurs, and Synchronization latch count value (Un\G700 to Un\G715) and the synchronous output are not refreshed. The control of hardware logic stops.
(3) Even after the synchronization loss has occurred, the values of 'General command 0 to General command F' (Y10 to Y1F) and User Address in the synchronous interrupt program are continuously reflected to the module.

Point

When a synchronization loss occurs, the module may be affected by noise. Check and adjust the cable wiring and the installation environment of the programmable controllers, and restart the system. If the above error occurs again even after the adjustment, the possible cause is a failure of the module. Please consult your local Mitsubishi representative.

Monitor

The inter-module synchronization status can be monitored when the inter-module synchronization function is used.
For details, refer to the following.
☞ Page 289 Synchronization status monitor

Precautions

- The refresh timings of the X/Y signals and User Address (high speed area) differ depending on whether the inter-module synchronization function is used or not. (☞ Page 18 Update timings of the X/Y signals and Hardware logic area)
- When "Select Inter-module Synchronization Target Module" is set to "Do not Synchronize" in GX Works3, the inter-module synchronization signal is not input. Since the inter-module synchronization signal is not input, the High/Low states of external output blocks for which "SYNC" is set for "Delay Time(Unit)" do not change.

1.7 Error History Function

Up to 16 errors that occurred in the flexible high-speed I/O control module are stored in the buffer memory areas as an error history.

Operation

The error code and error time of each error are stored in Error history No.1 (Un\G8010 to Un\G8019) and sequentially thereafter.

The error time is stored as shown below.

Ex.

The following shows the case of Error history No. 1.

	b15 to b8	b7 to b0
Un\G8010	Error code	
Un\G8011	First two digits of the year	Last two digits of the year
Un\G8012	Month	Day
Un\G8013	Hour	Minute
Un\G8014	Second	Day of the week
Un\G8015	Millisecond (higher-order digits)	Millisecond (lower-order digits)
Un\G8016 to Un\G8019	System area	

Item	Description	Storage example*1
Error code	The error code is stored.	1020H
First two digits of the year/last two digits of the year	Stored as a BCD code.	2016H
Month/day		1031H
Hour/minute		1234H
Second		56H
Day of the week	For each day of the week, one of the following values is stored as a BCD code. Sunday: 0H, Monday: 1H, Tuesday: 2H, Wednesday: 3H, Thursday: 4H, Friday: 5H, Saturday: 6H	1H
Millisecond (upper)/millisecond (lower)	Stored as a BCD code.	0789H
System area	—	—

*1 Value of when a reset error during hardware logic control (error code: 1020H) occurred at 12:34:56.789 on Monday, October 31st, 2016

Clearing the error history

The error history can be cleared with one of the following methods.

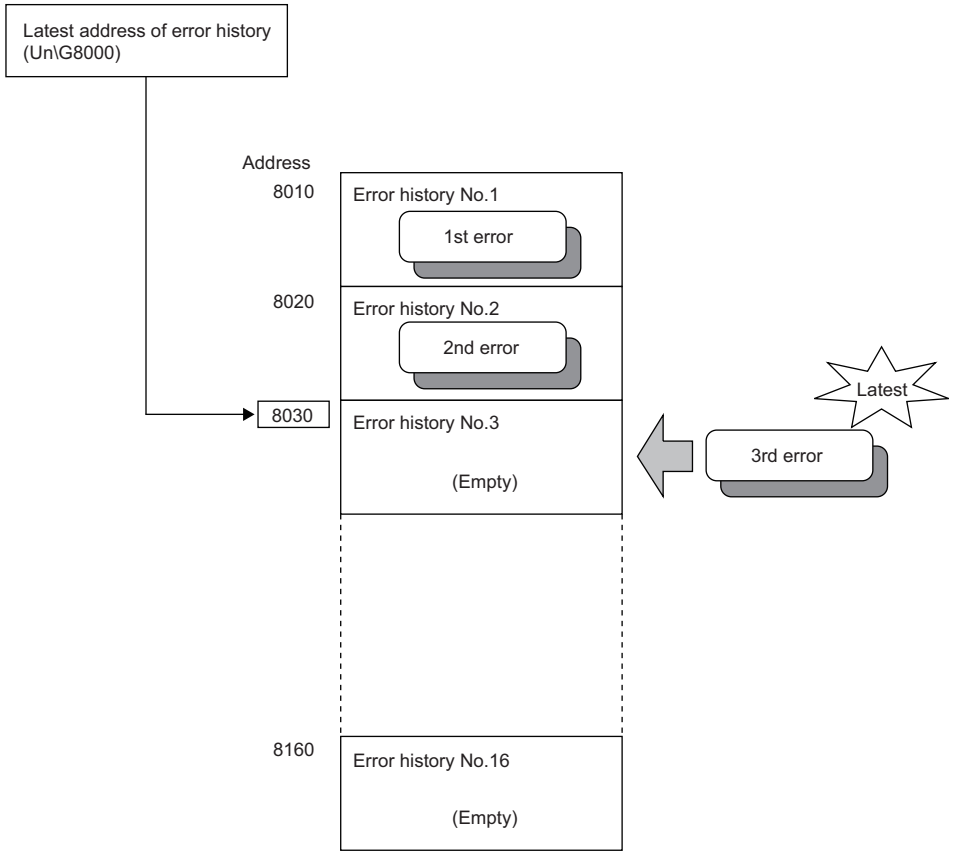
- Turning off the power
- Resetting the CPU module
- Setting 'Clear setting of error history' (Un\G8002) to Clear the history. (1) and turning on and off Error clear request (YF)

Checking the error history

The start address of the error history where the latest error has been stored can be checked with Latest address of error history (Un\G8000).

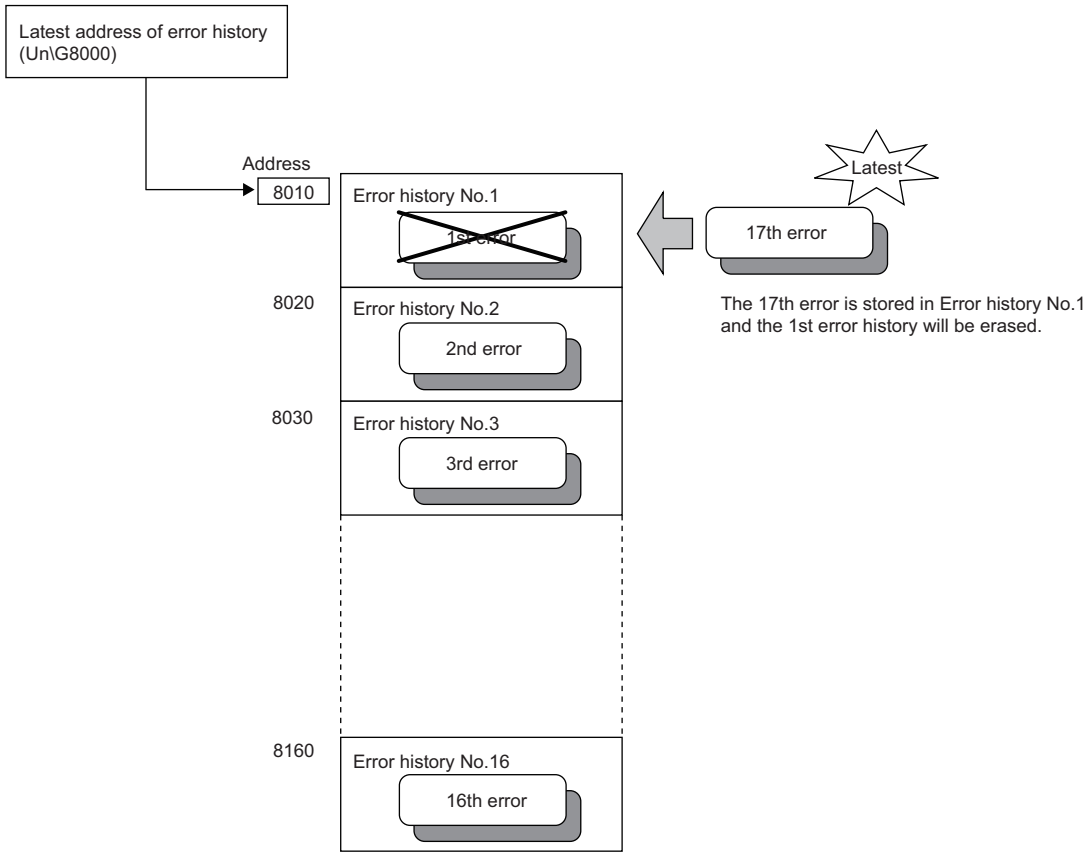
Ex.

The following shows the case in which the third error has occurred. The third error is stored in Error history No. 3 and 8030 (start address of Error history No. 3) is stored in Latest address of error history (Un\G8000).



Ex.

The following shows the case in which the 17th error has occurred. The 17th error is stored in Error history No. 1 and Latest address of error history (Un\G8000) is overwritten with 8010 (start address of Error history No.1).



Point

If the storage areas of the error history become full, the value in Error history No. 1 (Un\G8010 to Un\G8019) is overwritten and sequentially thereafter to keep registering errors. (The error histories before the overwriting are deleted.)

1.8 Event History Function

The errors generated or operations executed in the flexible high-speed I/O control module are collected as event information in the CPU module.

The CPU module collects the event information that occurred in the flexible high-speed I/O control module and keeps them in the data memory inside of the CPU module or an SD memory card.

The event information collected by the CPU module can be displayed on an engineering tool to check the occurrence history in a time series.

Event type	Classification	Description
System	Error	An error detected by the self-diagnostics in each module
	Warning	A warning (alarm) detected in each module
	Information	Operation by the normal detection of the system that is not classified as Error or Warning, or operation performed automatically by the system
Security	Warning	Operation that is judged as unauthorized access to each module
	Information	Operation that is hard to be judged as the success of unlocking passwords or unauthorized access
Operation	Warning	Deleting (data clear) operations that may change the action. (These operations are not judged as errors by the self-diagnostics.)
	Information	Operations such as clearing errors performed by users to change the system operation or configuration

Setting method

The event history function can be set from the event history setting window of the engineering tool. For the setting method, refer to the following.

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

Displaying event history

Access the menu window of the engineering tool. For details on the operating procedure and how to view the contents, refer to the following.

📖 GX Works3 Operating Manual

List of event history data

The following table lists the events that occur in the flexible high-speed I/O control module when the event type is operation.

Event code	Event class	Event name	Event detail	Additional information
20100	Information	Error clear	"Error clear" has been executed.	—
24000	Information	Write to Module (execution memory)	"Write to Module (execution memory)" has been executed.	—
24001	Information	Write to Module (execution + flash ROM)	"Write to Module (execution + flash ROM)" has been executed.	Cumulative number of writes to a flash ROM
24002	Information	Hardware logic control start	The hardware logic control has been started.	A request source is added from any of the following items. • [Hardware logic control start] of the configuration tool • 'Hardware logic control start request' (Y4)
24003	Information	Hardware logic control stop	The hardware logic control has been stopped.	A request source is added from any of the following items. • [Hardware logic control stop] of the configuration tool • 'Hardware logic control stop request' (Y5) • 'Hardware logic control stop signal at disconnection' (Y6)
24010	Information	Write to trigger condition setting of logic analyzer (execution memory)	"Write to trigger condition setting of logic analyzer (execution memory)" has been executed.	—
24011	Information	Write of trigger condition setting of logic analyzer (execution + flash ROM)	"Write of trigger condition setting of logic analyzer (execution + flash ROM)" has been executed.	Cumulative number of trigger setting writes
24020	Information	Simulation execution	Simulation has been executed.	—
2A000	Warning	Operating condition settings batch-reset	Execution memory setting has been reset with the data saved in the flash ROM.	—
2A010	Warning	Delete of trigger condition setting of logic analyzer	The trigger condition setting of logic analyzer has been deleted.	—

1.9 Backing up and Restoring the Hardware Logic

The flexible high-speed I/O control module can back up and restore the hardware logic written in the flash ROM.

- Back up: Creates a module-specific backup parameter and saves the hardware logic written in the flash ROM.
- Restoration: Writes the information backed up in the CPU module to this module.

In the event that the flexible high-speed I/O control module fails and needs to be replaced, the hardware logic written in the flash ROM of the failed flexible high-speed I/O control module can be restored onto the new flexible high-speed I/O control module.

Only when the model where the hardware logic is to be backed up and the model where the hardware logic is to be restored are the same, the hardware logic can be backed up and restored.

Each procedure differs depending on whether a module-specific backup parameter is used or not.

When the module-specific backup parameter is used

Hardware logic is automatically restored when the failed module is replaced with a new one using the online module change.

For details on the online module change, refer to the following.

MELSEC iQ-R Online Module Change Manual

Details of the module-specific backup parameter

A module-specific backup parameter is a file created in an SD memory card or the data memory of the control CPU. The contents of the parameter are the hardware logic written in the flash ROM of the flexible high-speed I/O control module.

The file name of a module-specific backup parameter is determined as follows based on the start I/O number of the flexible high-speed I/O control module.

UBPmmmnn.BPR

- mmm indicates a value calculated by dividing the module I/O number by 10H (3 digits in hexadecimal).
- nn indicates a consecutive number of the module-specific backup parameters for each module (fixed to 00).

Creating and updating a module-specific backup parameter

A module-specific backup parameter is created or updated when the hardware logic written in the flash ROM of the flexible high-speed I/O control module is updated.

Timing when backup data is created or updated	Description
When "Write to Module (execution + flash ROM)" is executed with the configuration tool	A module-specific backup parameter is created or updated when writing hardware logic to the flash ROM is completed.
When writing the trigger setting to a flash ROM is executed with the configuration tool	
When deleting the trigger setting from a flash ROM is executed with the configuration tool	A module-specific backup parameter is updated when deleting hardware logic from the flash ROM is completed.

When no module-specific backup parameter exists in the data memory of the control CPU and a module-specific backup parameter needs to be created with the current setting, write the hardware logic to the flexible high-speed I/O control module with the configuration tool. A module-specific backup parameter is created with the current setting of the flash ROM.

■Disabling the security

When creating or updating a module-specific backup parameter, execute "Write to Module (execution + flash ROM)" with the security disabled.

- When "Write to Module (execution + flash ROM)" is executed with the security enabled, a module-specific backup parameter is not created and updated.
- Even if the trigger setting is written to or deleted from the flash ROM after the execution of "Write to Module (execution + flash ROM)" with the security enabled, a module-specific backup parameter is not created and updated.

■Precautions

If the creation of a module-specific backup parameter fails because the data memory of the control CPU does not have sufficient free space or the module-specific backup parameter is being used, a module-specific backup parameter creation error (error code: 17E1H) occurs.

Reading of module-specific backup parameters

To read a module-specific backup parameter and restore hardware logic, set "Hardware logic control auto restoration executed/unexecuted" of the module parameter to "Enable" in advance.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ Module model name ⇨ [Module Parameter] ⇨ [Application setting] ⇨ [Online module change]

■Read timing

Module-specific backup parameters are read when a new module is mounted and recognized after the online module change. When the programmable controller is powered off and the module is replaced with a new one, module-specific backup parameters are not read.

■Precautions

When the module-specific backup parameter for the target slot does not exist in an SD memory card or the data memory of the control CPU, the subsequent restoration of the hardware logic is not performed. If the hardware logic cannot be restored even though the module-specific backup parameter exists, a module-specific backup parameter restore error (error code: 17E0H) occurs.

Restoration of the hardware logic

When reading module-specific backup parameters is completed with no errors, the hardware logic is converted (restored) into the hardware logic for the new module and stored in the flash ROM.

Restrictions on the module-specific backup parameter

Hardware logic cannot be backed up or restored with a module-specific backup parameter in the following cases.

- When the control CPU is not the process CPU
- When the programmable controller is powered off and the flexible high-speed I/O control module is replaced with a new one
- When "Hardware logic control auto restoration executed/unexecuted" of the module parameter is set to "Disable"

In any of the cases above, back up or restore the hardware logic by the following method.

☞ Page 82 When the module-specific backup parameter is not used

When the module-specific backup parameter is not used

Back up or restore the hardware logic with the configuration tool.

- Save the hardware logic for a backup as a project file. (☞ Page 94 Project Management)
- Write the backed up hardware logic to the module for restoration. (☞ Page 115 Writing data to the module)

2 FUNCTIONS OF THE CONFIGURATION TOOL

This chapter describes the configuration tool for creating the hardware logic and writing it into the flexible high-speed I/O control module.

2.1 Starting and Exiting the Configuration Tool

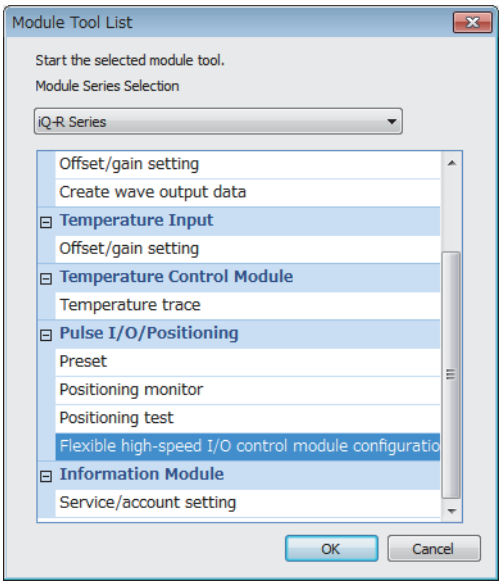
This section describes how to start or exit the configuration tool.

Start

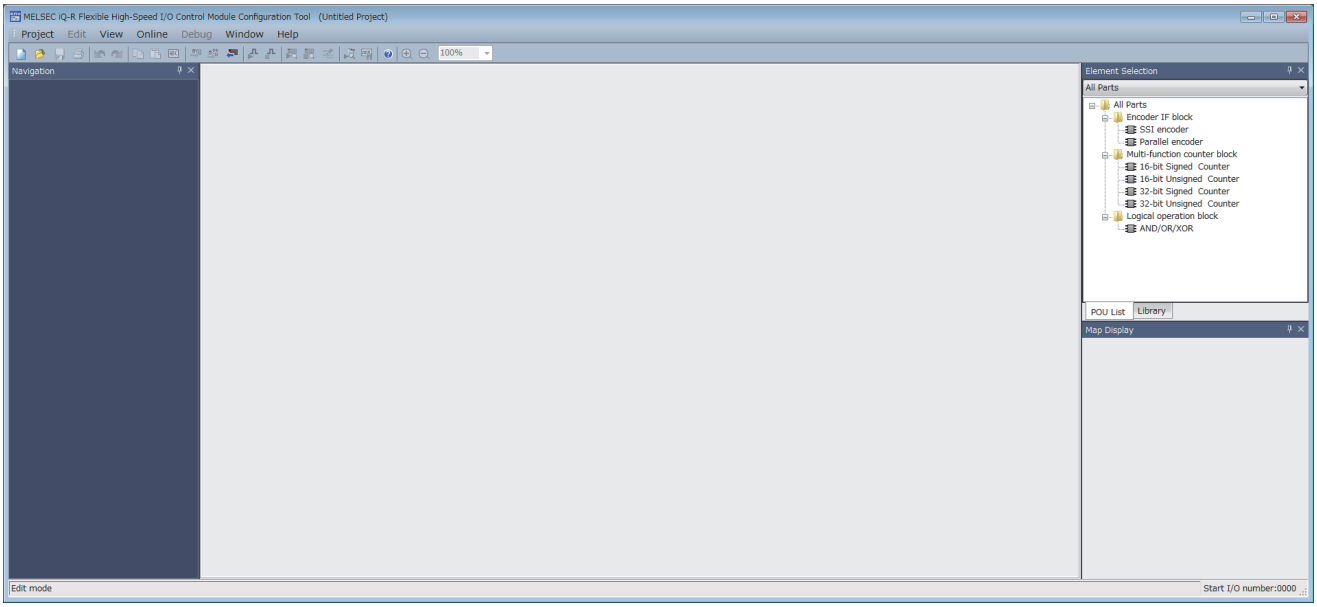
Follow the steps below.

[Tool] ⇨ [Module Tool List]

1. Select "iQ-R-Series" for "Module Series Selection".
2. Select [Pulse I/O/Positioning] ⇨ [Flexible high-speed I/O control module configuration tool], and click the [OK] button.



3. The configuration tool is started.



■Connection target

The configuration tool acquires information of the connection destination from GX Works3 at its start-up. Therefore, when the connection destination has been changed on GX Works3 after the start-up of the configuration tool, exit from the configuration tool once. And start it again.

Exit

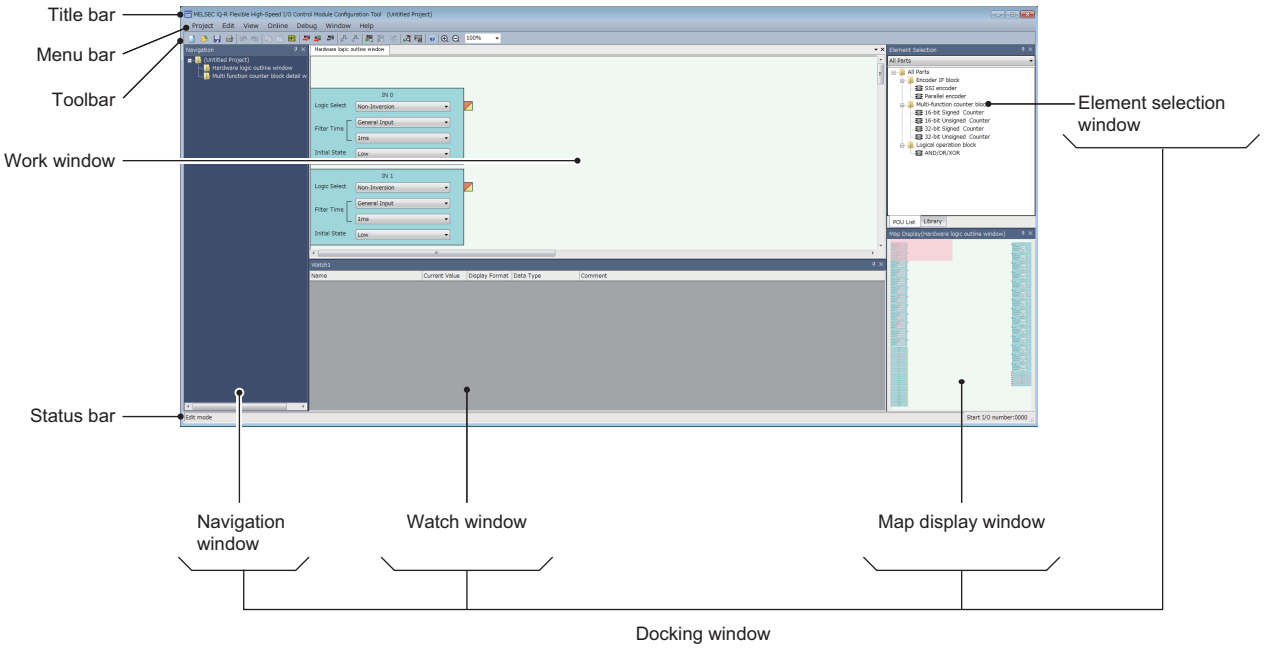
Select [Project] and [Exit] in the configuration tool.

Point

Even if GX Works3 is exited while the configuration tool is being used, the configuration tool can be operated independently. Settings and the monitor operation can be continued with the configuration tool.

2.2 Window Layout

The following figure shows the whole window layout.



For details on each item, refer to the following.

Window	Reference
Navigation window	☞ Page 87 Navigation window
Work window	☞ Page 89 Work window
Element Selection window	☞ Page 88 Element Selection window
Map display window	☞ Page 90 Map display window
Watch window	☞ Page 120 Watch Function

Window operation

■Displaying a docking window

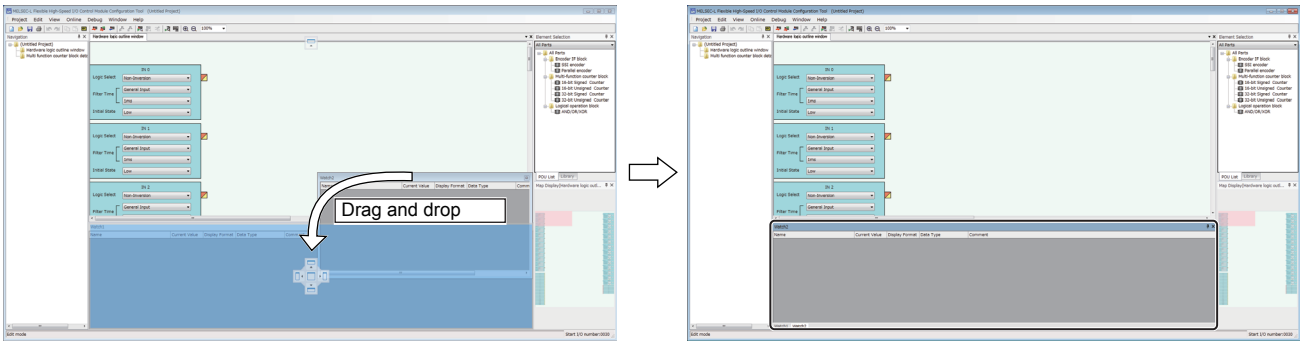
[View] ⇨ [Docking Window] ⇨ [(item to be displayed as a docking window)]

Point

If a window is not displayed by the above operation from the menu, select [Window] ⇨ [Return Window Layout to Initial Status].

■Switching the docking window displays between docking and floating

- Docking display: Drag and drop the title bar of a docking window with floating display into the guidance to dock the window. Or select a docking window with floating display and perform [Window] ⇨ [Docking].



Put the window onto the guidance.

A new tab appears after the window is docked.

- Floating display: Drag the title bar of a docking window that is docked with the main frame to a desired location to display the window independently. Or select a docking window that is docked and perform [Window] ⇨ [Floating].

■Switching work window displays between docking and floating

- Docking display: Select a work window with floating display and perform [Window] ⇨ [Docking].
- Floating display: Drag the title bar of a work window that is docked with the main frame to a desired location to display the window independently. Or select a work window that is docked and perform [Window] ⇨ [Floating].

Point

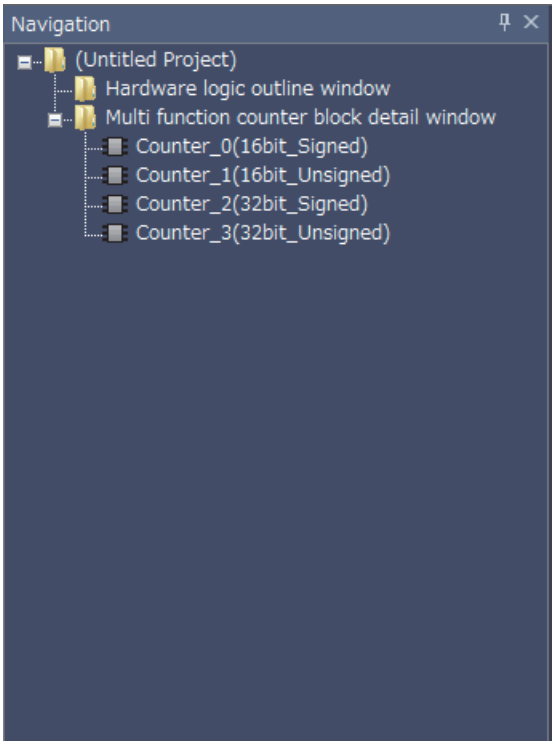
The docking window displays can be switched between docking display and floating display by double-clicking the title bar.

Navigation window

In the Navigation window, the Hardware logic outline window and names of multi function counter blocks arranged in the window (Counter_☐: A multi function counter block number comes in ☐.) are displayed in the tree format.

For details on the hardware logic outline window and multi function counter blocks, refer to the following.

☞ Page 144 CREATING A HARDWARE LOGIC

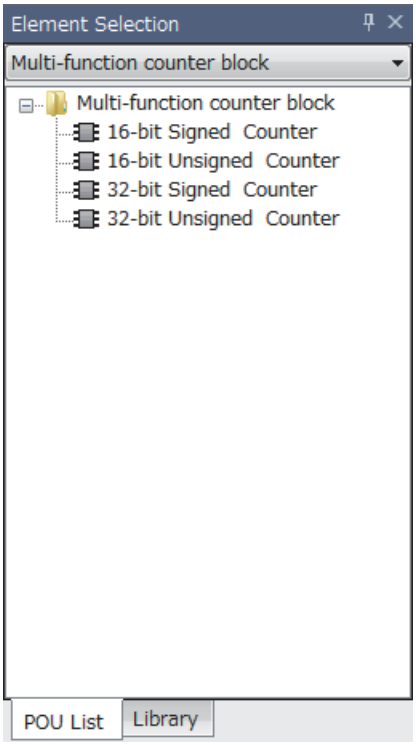


Element Selection window

In the Element Selection window, the main blocks that can be arranged when the hardware logic is created are displayed in the tree format.

This window displays only the blocks in a category selected from the drop-down menu at the upper section of the window.

When a multi function counter block is selected in the Element Selection window and arranged on the work window, the tab of the block is added in the work window.



"Library" tab

Select the "Library" tab to display the libraries provided by the manufacturer and the user libraries registered in the configuration tool.

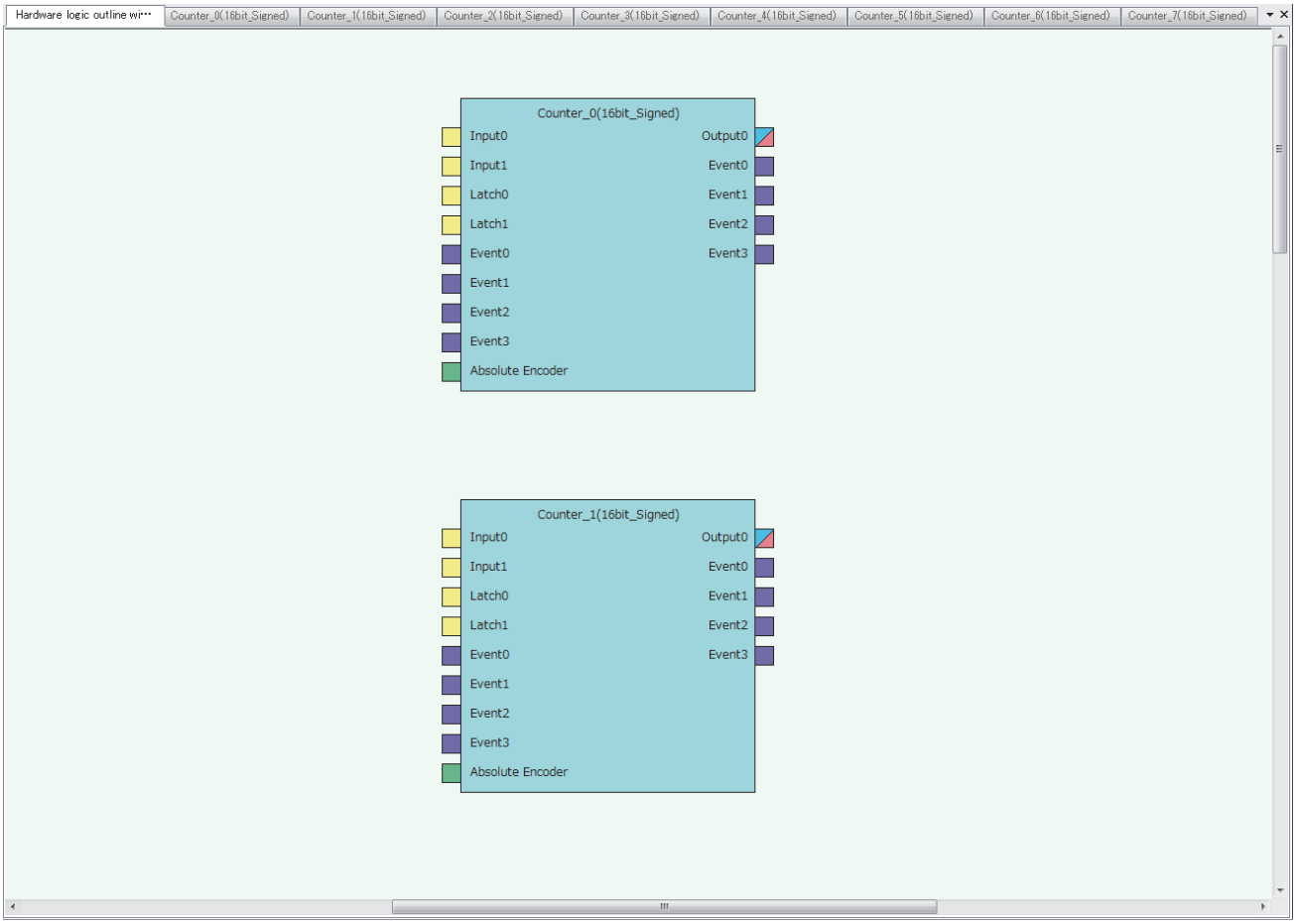
For the registration method, refer to the following.

☞ Page 111 Library Function

Work window

The hardware logic is created or the monitor display is executed in the work window.
One window out of the hardware logic outline window (one window) and multi function counter block detail windows (up to eight windows) is a target for the operation. The work window targeted for the operation is displayed in the map display window.
Switch the operation-target window with one of the following operations.

- Double-clicking the item to display in the Navigation window
- Double-clicking a multi function counter block in the hardware logic outline window
- Switching a window from [Window] menu
- Clicking a tab at the upper section of the work window (for a docked window) or clicking the title section (for a floating window)
- Clicking a point in the work window

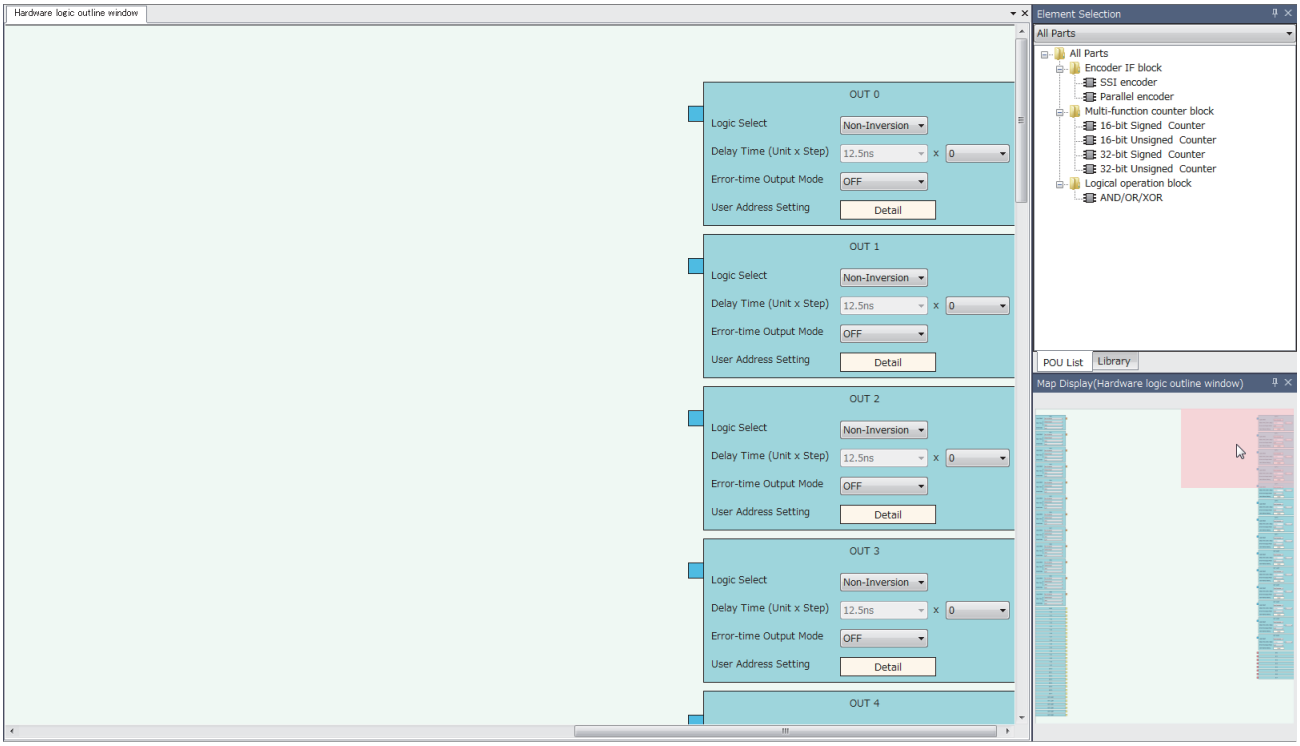


Map display window

The map display window displays a whole image of the hardware logic outline window or a multi function counter block detail window that is currently displayed in the work window.

Click the desired area to be displayed on the map display window, and the displayed portion in the work window will change.

The displayed area in the work window can also be changed by dragging the panel on the map display window.



2.3 List of Menus of the Configuration Tool

[Project] menu

Menu	Reference
[Project] ⇨ [New]	☞ Page 94 Creating a new project
[Project] ⇨ [Open]	☞ Page 94 Opening a project
[Project] ⇨ [Close]	—
[Project] ⇨ [Save]	☞ Page 95 Saving an existing project
[Project] ⇨ [Save As]	☞ Page 95 Saving a project with a new name
[Project] ⇨ [Project Verify]	☞ Page 96 Verifying a project file
[Project] ⇨ [Security]	☞ Page 97 Security
[Project] ⇨ [Print]	☞ Page 93 Printing
[Project] ⇨ [Exit]	☞ Page 84 Exit

[Edit] menu

Menu	Reference
[Edit] ⇨ [Undo]	—
[Edit] ⇨ [Redo]	
[Edit] ⇨ [Copy]	☞ Page 105 Copying a block
[Edit] ⇨ [Paste]	☞ Page 107 Pasting a block
[Edit] ⇨ [Insert and Paste]	
[Edit] ⇨ [Block Delete]	☞ Page 102 Deleting a block
[Edit] ⇨ [Export block]	☞ Page 111 Library Function
[Edit] ⇨ [Library Registration]	
[Edit] ⇨ [Library Delete]	
[Edit] ⇨ [Comment Insert]	☞ Page 108 Adding a comment

[View] menu

Menu	Reference
[View] ⇨ [Zoom] ⇨ [Set Zoom Factor]	—
[View] ⇨ [Zoom] ⇨ [Zoom In]	
[View] ⇨ [Zoom] ⇨ [Zoom Out]	
[View] ⇨ [Comment Display]	☞ Page 108 Adding a comment
[View] ⇨ [Docking Window] ⇨ [Navigation]	☞ Page 87 Navigation window
[View] ⇨ [Docking Window] ⇨ [Element Selection]	☞ Page 88 Element Selection window
[View] ⇨ [Docking Window] ⇨ [Map Display]	☞ Page 90 Map display window
[View] ⇨ [Docking Window] ⇨ [Watch 1] to [Watch 4]	☞ Page 120 Watch Function

[Online] menu

Menu	Reference
[Online] ⇨ [Start I/O number setting]	☞ Page 114 Start I/O number setting
[Online] ⇨ [Write to Module (execution memory)]	☞ Page 115 Writing data to the module
[Online] ⇨ [Write to Module (execution + Flash ROM)]	
[Online] ⇨ [Read from Module (Flash ROM)]	☞ Page 115 Reading data from the module
[Online] ⇨ [Verify with Module (Flash ROM)]	☞ Page 116 Verifying with the module
[Online] ⇨ [Module operation] ⇨ [Hardware logic control start]	☞ Page 117 Module operation
[Online] ⇨ [Module operation] ⇨ [Hardware logic control stop]	
[Online] ⇨ [Monitor] ⇨ [Start Monitoring (All Windows)]	☞ Page 118 Monitor
[Online] ⇨ [Monitor] ⇨ [End Monitoring (All Windows)]	
[Online] ⇨ [Monitor] ⇨ [Start Monitoring]	
[Online] ⇨ [Monitor] ⇨ [Stop Monitoring]	
[Online] ⇨ [Monitor] ⇨ [End Monitoring]	
[Online] ⇨ [Watch] ⇨ [Start Watching]	☞ Page 120 Watch Function
[Online] ⇨ [Watch] ⇨ [Stop Watching]	
[Online] ⇨ [Watch] ⇨ [Watch Items Batch Read] ⇨ [Watch 1] to [Watch 4]	

[Debug] menu

Menu	Reference
[Debug] ⇨ [Simulation]	☞ Page 123 Simulation function
[Debug] ⇨ [Logic Analyzer]	☞ Page 131 Logic analyzer function
[Debug] ⇨ [Continuous logging]	☞ Page 137 Continuous logging

[Window] menu

Menu	Reference
[Window] ⇨ [Close All Work Windows]	—
[Window] ⇨ [Return Window Layout to Initial Status]	☞ Page 85 Displaying a docking window
[Window] ⇨ [Floating]	☞ Page 86 Switching the docking window displays between docking and floating
[Window] ⇨ [Docking]	
[Window] ⇨ [(Information on the window that is being displayed)]	—

[Help] menu

Menu	Reference
[Help] ⇨ [Flexible High-Speed I/O Control Module Help]	☞ Page 143 Displaying the help
[Help] ⇨ [Version Information]	☞ Page 143 Checking the version of the configuration tool

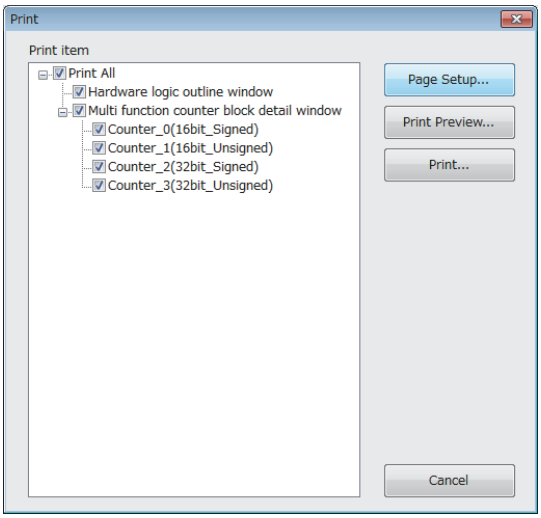
2.4 Printing

The contents in the Work window are printed.

Printing method

1. Open the "Print" window.

[Project] ⇨ [Print]



2. Select the items to be printed.

The following contents are printed depending on the items to be selected.

Item	Content
Hardware logic outline window	• Hardware logic outline window • Setting details of "Data Frame Setting" in an SSI encoder block (when the SSI encoder block is arranged in the Hardware logic outline window) • Setting details of "User Address Setting" of an external output block
Multi function counter block detail window	• Multi function counter block detail window • Setting details of "Detector Select" in an input signal event detection block • Setting details of "Detector Select" in a latch event detection block

3. Set the range of pages with the [Page Setup] button.

4. Check a print image with the [Print Preview] button.

5. Click the [Print] button.

2.5 Project Management

The configuration tool manages the hardware logic as a project.
This section describes the basic operations of the configuration tool for projects, such as creating, opening, and saving of a project.
Because a created project can be managed as a project file, changing of a project name, copying and pasting of a project, and other operations can be easily executed with Windows® Explorer.

Creating a new project

Create a new project.

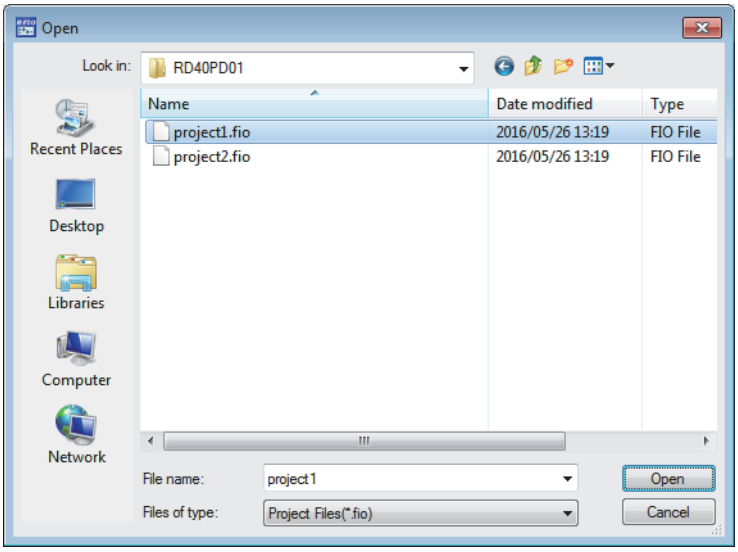
[Project] ⇨ [New]

Opening a project

Read a project saved in a hard disk or other areas in a personal computer.

1. Open the "Open" window.

[Project] ⇨ [Open]



2. Select a project to open and click the [Open] button.

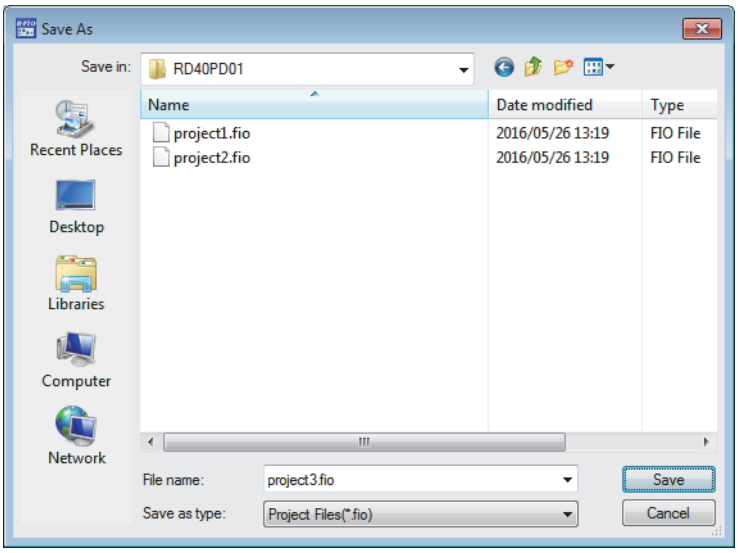
Saving a project file

Save a project file in a hard disk or other areas in a personal computer.

Saving a project with a new name

Name the project being edited and save the project file.

[Project] ⇨ [Save As]



Saving an existing project

Overwrite the hardware logic information being edited on an existing project file.

[Project] ⇨ [Save]

Verifying a project file

Verify the contents of a currently opened project and those of another project in a hard disk. Verification results are displayed in a list and mismatches can be checked.

Verifying the hardware logic

The following describes how to verify the hardware logic.

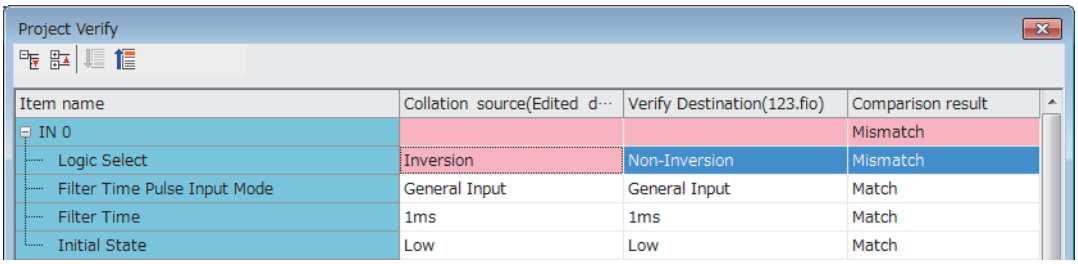
1. Select the following item.

[Project] ⇨ [Project Verify]

2. Select a project in a hard disk to be verified.

3. Verification results are displayed.

In the verification result window, the match/mismatch status (the links to the input terminals of each block, setting values of the block, and User Address) are displayed.



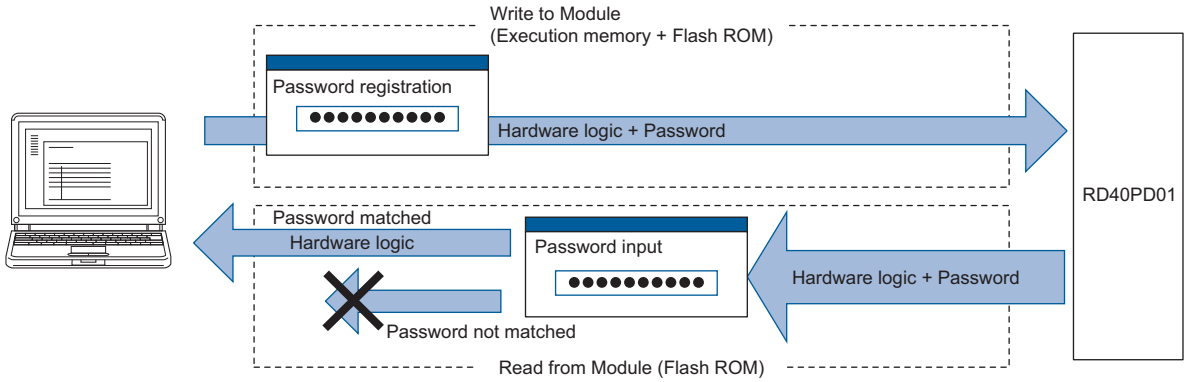
4. Double-click a mismatch to jump to the corresponding section in the work window. When the links are mismatched, the corresponding link in the work window is selected.

Security

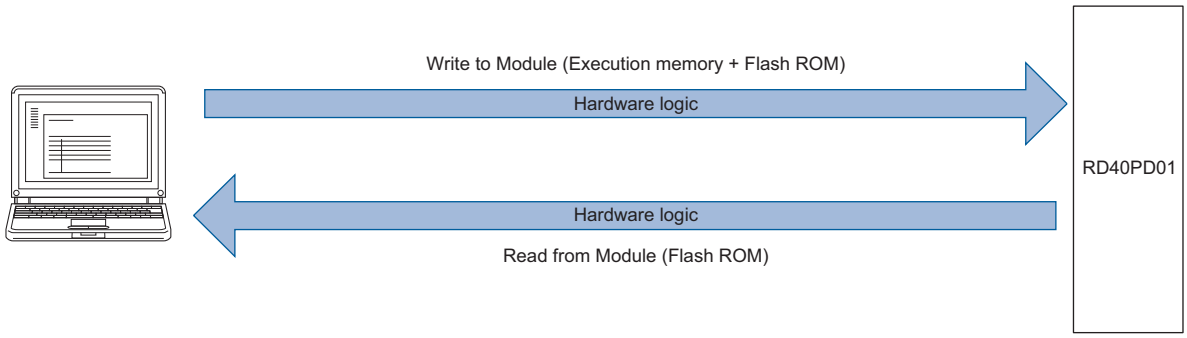
When the hardware logic is written into a flash ROM, add a password to prevent inappropriate access to read the data. After [Security] is set to "Enable" and [Write to Module (Execution memory + Flash ROM)] in [Online] is executed, the password input window is displayed. After a password is input and writing data into the module is completed, the security becomes valid.

To read the hardware logic to which the security has been enabled from a flash ROM to the module, input the set password.

- When the security is enabled



- When the security is disabled



Setting method

[Project] ⇨ [Security] ⇨ [Enable] or [Invalid]

When a new project is created, [Security] has been set to "Enable".

Operation details and restrictions

The security setting is saved in a project.

■Operations to be performed when "Write to Module (Execution memory + Flash ROM)" is executed

Security setting	Description
Enable	• The password registration window is displayed. • The hardware logic is written with a password.
Invalid*1	• The password registration window is not displayed. • The hardware logic is written. No password is added.

*1 When "Write to Module (Execution memory)" is executed, the same operations as the ones for when [Security] is "Invalid" are executed.

■Operations to be performed when "Read from Module (Flash ROM)" is executed

Hardware logic in the module	Description
With a password	• The password input window is displayed. • Only when the input password matches the password in the module, the hardware logic can be read.
Without a password	• The password input window is not displayed. • The hardware logic can be always read.

■Operations that change the security setting

When one of the following operations is performed, the security setting is changed.

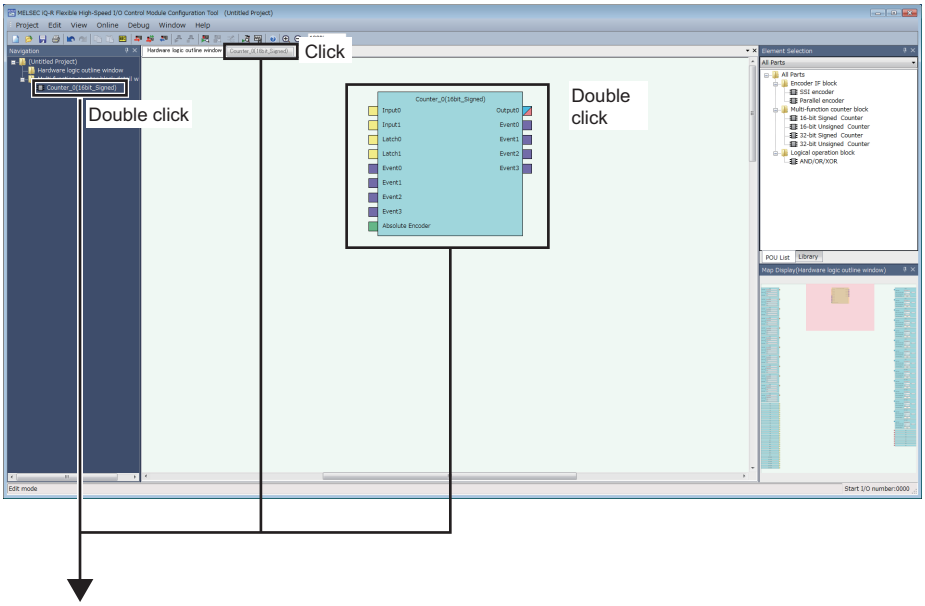
Operation	Security setting
Creating a new project	"Enable" is set.
Opening a project	The project setting is reflected.
Reading data from the module	The setting stored in the flash ROM of the flexible high-speed I/O control module is reflected.

2.6 Windows for Creating the Hardware Logic

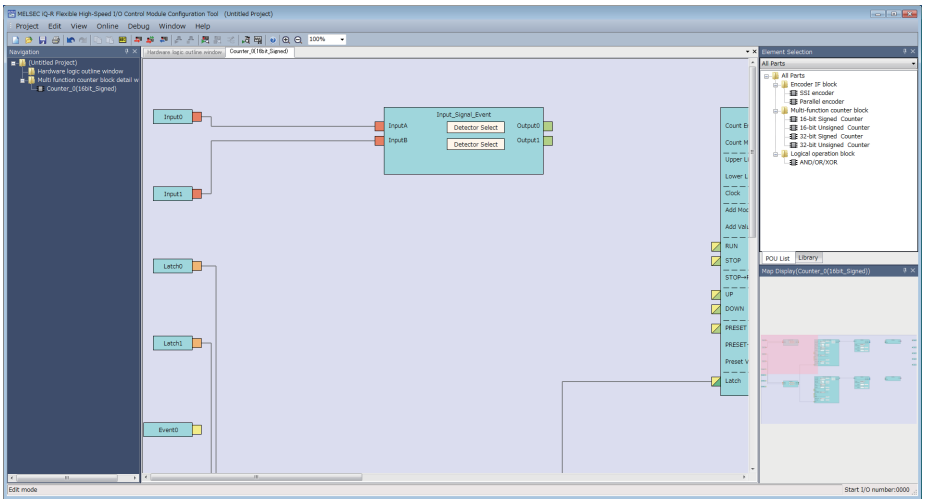
The following two types of window are prepared for creating the hardware logic. The window can be switched between the hardware logic outline window and the multi function counter block detail window.

Window	Description
Hardware logic outline window	This window is for creating the outline of the hardware logic in the flexible high-speed I/O control module. Multi function counter blocks are arranged, external I/O terminals are linked, and settings are configured in this window.
Multi function counter block detail window	This window is for configuring the detail settings of the multi function counter blocks arranged in the hardware logic outline window. Switch multi function counter block detail windows (up to eight windows) and edit each multi function counter block.

Hardware logic outline window



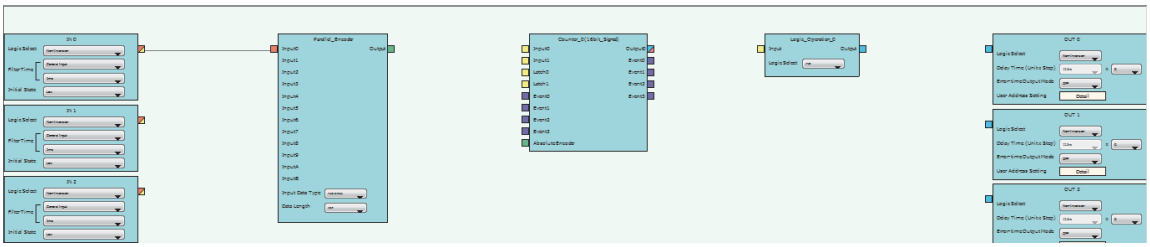
Multi function counter block detail window



Hardware logic outline window

The hardware logic outline window is composed of the following five sections. Depending on the section, the blocks that can be arranged differ.

Change the setting and linking of each block to create the hardware logic with various functions.



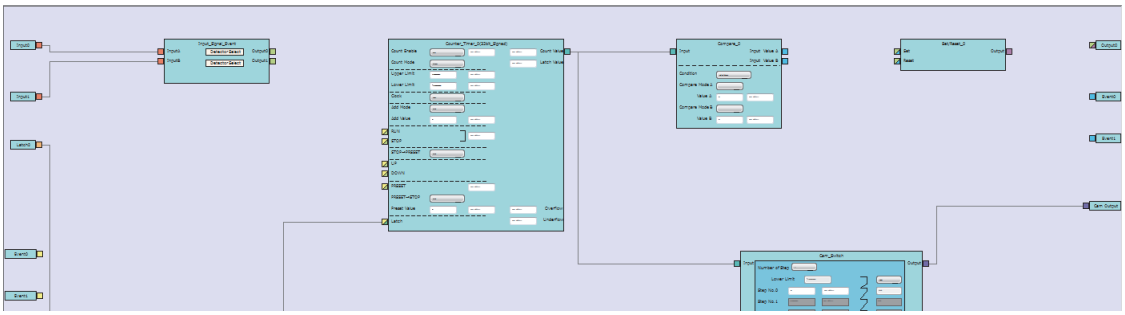
Section	Description	Blocks and terminals that can be arranged*1
First section	The blocks corresponding to inputs to the hardware logic have been arranged in this section. All the blocks in the first section are arranged when a project is started, and any blocks cannot be deleted or newly added from the Element Selection window.	• External input block • Inter-module synchronization signal input terminal • Y device terminal • OUT terminal
Second section	Blocks corresponding to encoder inputs are to be arranged in this section.	• Parallel encoder block • SSI encoder block
Third section	Blocks that execute count with external input signals and the internal clock are to be arranged in this section.	Multi function counter block
Fourth section	Blocks that perform logical operations to input signals and outputs of multi function counter blocks are to be arranged in this section.	Logical operation block
Fifth section	The blocks corresponding to outputs of the hardware logic have been arranged in this section. All the blocks in the fifth section are arranged when a project is started, and any blocks cannot be deleted or newly added from the Element Selection window.	• External output block • SI device terminal

*1 For details on each block and terminal, refer to the following.

☞ Page 144 CREATING A HARDWARE LOGIC

Multi function counter block detail window

A multi function counter block detail window is composed of the following six sections.
Change the setting and linking of each block to create various count operations.



Section	Description	Blocks and terminals that have been arranged*1
First section	The terminals corresponding to inputs to a multi function counter block have been arranged in this section. An input to a multi function counter block in the hardware logic outline window is handled as an input in the multi function counter block detail window.	• Input terminal • Latch input terminal • Event input terminal
Second section	Select a signal detection condition for each input signal. Desired conditions can be detected with the combinations of High/Low and rise/fall.	• Input signal event detection block • Latch event detection block
Third section	Counter timers that function depending on each event have been arranged. Switching of 16-bit signed counter/16-bit unsigned counter and 32-bit signed counter/32-bit unsigned counter is determined depending on the blocks arranged in the hardware logic outline window and cannot be performed in the multi function counter block detail window.	• Counter timer block (16bit_Unsigned) • Counter timer block (16bit_Signed) • Counter timer block (32bit_Unsigned) • Counter timer block (32bit_Signed)
Fourth section	An operation to compare a count value and a setting value of a counter timer is executed. Coincidence detections of count values can be executed with the comparison operation.	• Compare block • Pattern generator block • Cam switch block*2
Fifth section	Based on results of the comparison operation and event detections, the signals to be externally output are controlled.	• Set/reset block • Cam switch block*2
Sixth section	The terminals corresponding to outputs of a multi function counter block have been arranged in this section. Outputs from the multi function counter block detail window are handled as outputs from the multi function counter block in the hardware logic outline window.	• Output terminal • Event output terminal • Pattern generator output terminal • Cam switch output terminal

*1 For details on each block and terminal, refer to the following.
☞ Page 169 Multi Function Counter Block

*2 A cam switch block is arranged across the fourth and fifth sections.

How to use blocks

Arranging blocks

The following describes how to arrange a block.

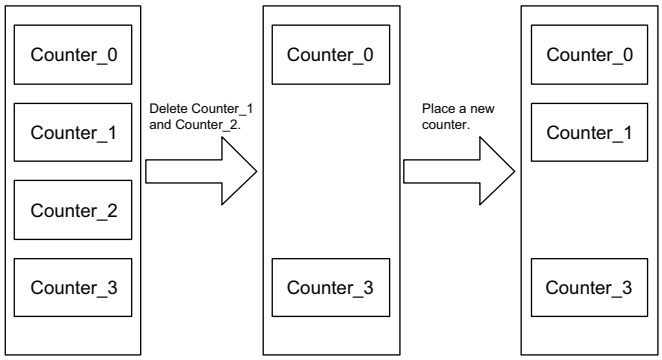
1. Select a block in the Element Selection window. Drag and drop the block into the hardware logic outline window.

When a block is dragged into the hardware logic outline window, the area to which the block can be dropped is highlighted.

2. When the selected block is dropped into the work window, the block is automatically arranged in the highlighted area.

Point

When three or more blocks have been arranged in the same section and blocks arranged between the top and bottom ones are deleted, that area becomes vacant. When a new block is arranged under this situation, the vacant area is highlighted and the block is arranged there.



Deleting a block

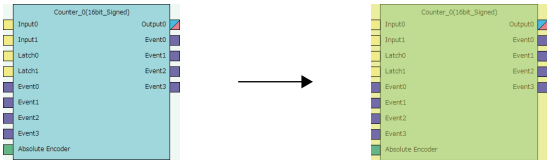
The following describes how to delete a block.

Only the blocks arranged from the Element Selection window can be deleted. The blocks that have been arranged by default cannot be deleted.

When a block is deleted, the link of the block is also deleted.

1. Click the block to be deleted.

The block is highlighted (displayed in yellow) and selected.



2. Right-click the block and select "Block Delete" from the context menu. Or press the Delete key.

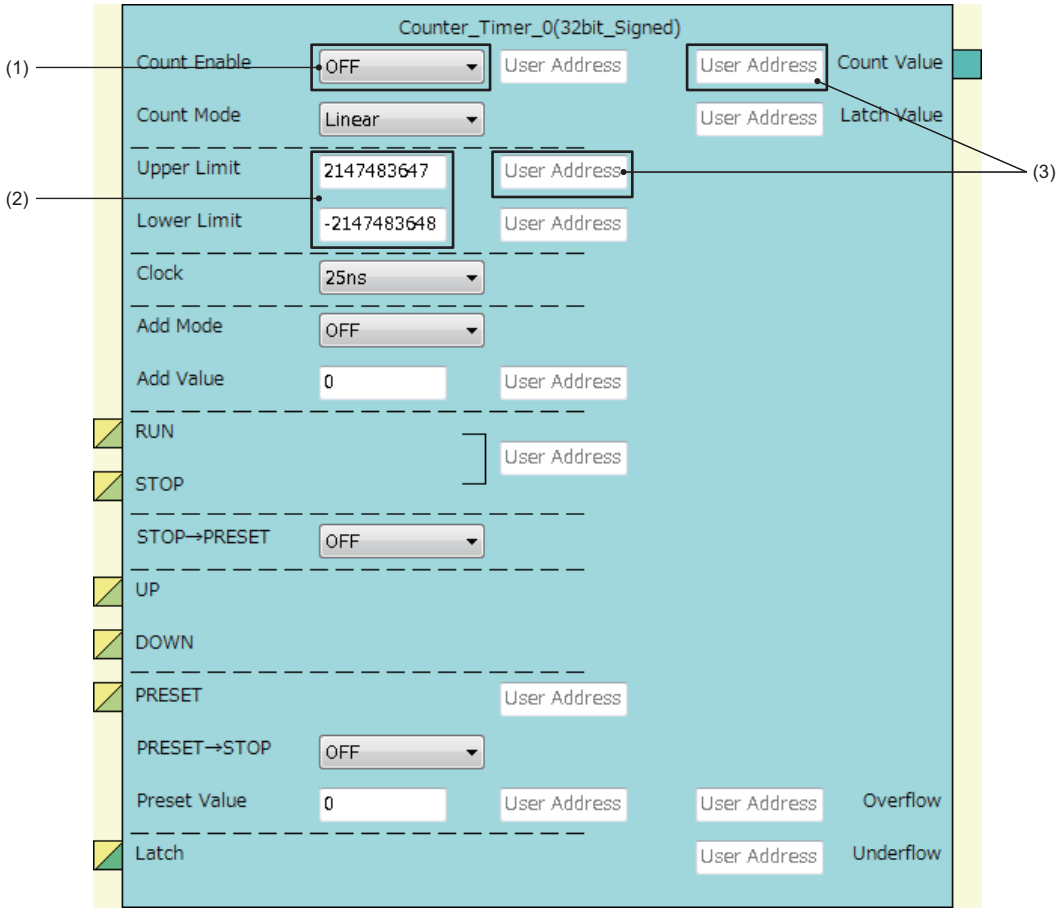
When a block is deleted, the link of the block is also deleted.

Point

- Users can also delete multi function counter blocks by right-clicking a multi function counter block name (Counter_☐) in the tree of the Navigation window and selecting "Block Delete" from the context menu.
- Select [Edit] ⇨ [Block Delete] to perform the same operation as selecting "Block Delete" from the context menu.

Block setting

Configure block settings by changing values in the drop-down lists and text boxes of each block.



No.	Item name	Description
(1)	Drop-down list	Select a setting value from the drop-down list.
(2)	Text box	Input a one-byte numerical value (decimal).
(3)	User Address	• By assigning buffer memory addresses to "User Address", input terminal status and parameter setting values can be changed with programs and values of a hardware logic can be monitored during the hardware logic control. For details, refer to the following (☞ Page 174 Assignment of "User Address"). • Input decimal values in the setting. • The range of settable buffer memory addresses is 1000 to 1099 (High speed area: 1000 to 1029, low speed area: 1030 to 1099). For details, refer to the following (☞ Page 103 Input range of "User Address").

■Input range of "User Address"

Input range	Description
1000 to 1029	• The monitor items and setting items assigned in this area are read or written at a high speed (100μs)*1. These items are also read or written when the flexible high-speed I/O control module sends an interrupt signal to the CPU module. • Assign even addresses to the parameters of two words (32 bits). Odd addresses cannot be assigned.
1030 to 1099	• The monitor items and setting items assigned in this area are read or written at a low speed (1ms)*1. • Assign even addresses to the parameters of two words (32 bits). Odd addresses cannot be assigned.

*1 This refresh timing is applied when the flexible high-speed I/O control module is in the normal operation mode. Refer to the following for the inter-module synchronous mode.
☞ Page 18 Update timings of the X/Y signals and Hardware logic area

Restriction

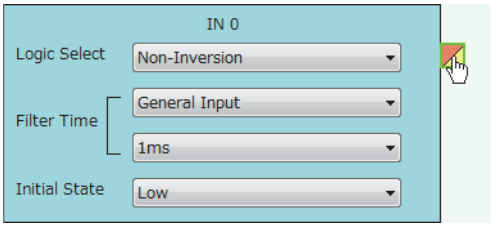
A single buffer memory address cannot be specified in several "User Address".
For example, when the buffer memory address 1000 has been specified in "Count Value" and 1000 is specified in "Latch Value", the value in User Address is returned to the initial value.

Linking blocks

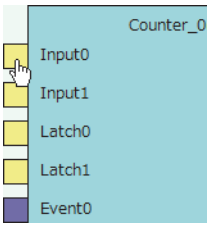
The following describes how to link blocks.

1. Click the terminal to be a start point.

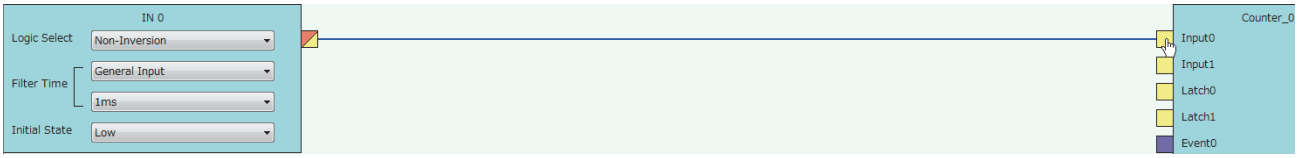
The outer frame of the selected terminal is highlighted.



2. Click the terminal to be an end point.

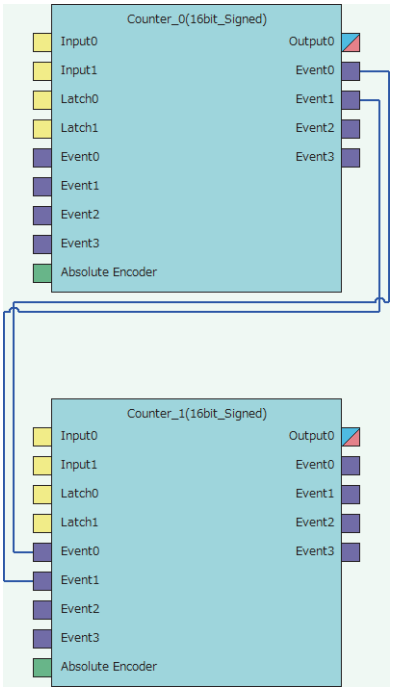


The start point and end point are automatically linked.



Point

Event terminals of a multi function counter block can be linked to Event terminals of another multi function counter block arranged in the same section.



■Linking conditions of terminals

Terminals in the same color can be linked. Terminals with two colors can be linked with the terminals with either of the two colors.

For details on the terminal colors, refer to the following.

☞ Page 144 Main Blocks in the Hardware Logic Outline Window

■Link type

The two link colors, blue and gray, are provided.

- A connection line between a linked output terminal and input terminal is blue.
- Gray connection lines indicate that the terminals have been automatically connected. Users cannot link the terminals.

■Deleting links

The following describes how to delete a link.

Only blue connection lines can be deleted. Gray connection lines cannot be deleted.

1. Right-click the link to be deleted. The link list is displayed in the context menu.
2. Select a link to be deleted and select "Wiring Delete".

Point

The link is highlighted (displayed in green) and selected when the link is clicked. The highlighted link can be deleted by pressing the Delete key.

Copying a block

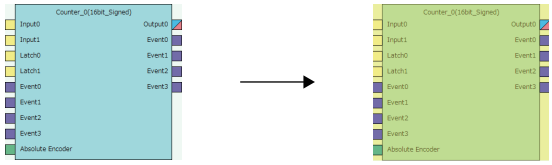
A block arranged in the hardware logic outline window can be copied by block unit. The setting values can be changed by block in a batch and an arranged block can be copied.

■Procedure of copying a block

The following describes the procedure of copying a block.

1. Click the block to be copied.

The block is highlighted (displayed in yellow) and selected.



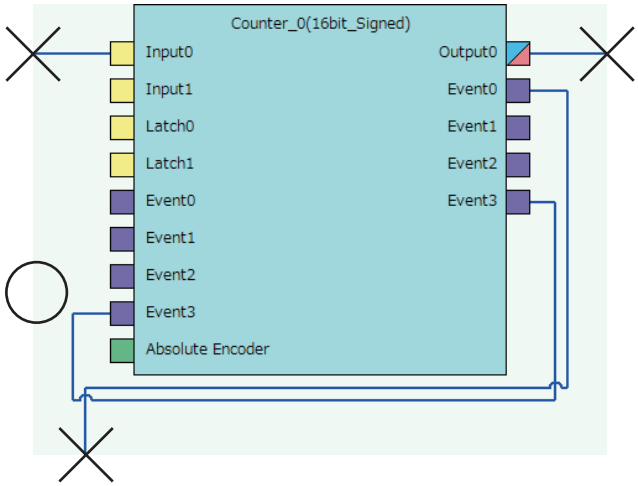
2. Right-click the block and select "Copy" from the context menu.

Point

- Users can also copy multi function counter blocks by right-clicking a multi function counter block name (Counter_☐) in the tree of the Navigation window and selecting "Copy" from the context menu.
- Select [Edit] ⇨ [Copy] to perform the same operation as selecting "Copy" from the context menu.

■Operation details and restrictions

- Multiple blocks cannot be copied at a time. Only a single block can be copied.
- The link between the block to be copied and other blocks is not copied. However, when the block to be copied is a multi function counter, link information in the multi function counter block detail window is also copied.
- The following table shows which main blocks can be copied.

Section*1	Target block	Copying	Item to be copied
1	External input block	Possible	Setting values of the block
	• Inter-module synchronization signal input terminal • Y device terminal • OUT terminal	Impossible	—
2	• Parallel encoder block • SSI encoder block	Possible	Setting values of the block
3	Multi function counter block	Possible	• Setting values (the setting value of User Address is not included), links, and comments in the multi function counter block detail window • Link between Event terminals of a multi function counter block in the hardware logic outline window (the links across blocks cannot be copied.) 
4	Logical operation block	Possible	The setting value of the block
5	External output block	Possible	Setting values of the block
	SI device terminal	Impossible	—

*1 This indicates a section in the hardware logic outline window.

Pasting a block

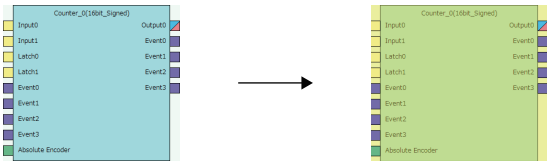
This function pastes the copy of a block. Select [Edit] ⇨ [Paste] for a block with the same type as the copy to overwrite the setting values with the ones of the copy at a time. Select [Edit] ⇨ [Insert and Paste] to add the copy as a new block in the hardware logic outline window.

■Procedure of [Paste]

The following describes the procedure of [Paste].

1. Click the block on which the copy is to be pasted.

The block is highlighted (displayed in yellow) and selected.



2. Right-click the block and select "Paste" from the context menu.

3. The setting values of the block are overwritten with the ones of the copy.

Point

- Users can also paste multi function counter blocks by right-clicking a multi function counter block name (Counter_☐) in the tree of the Navigation window and selecting "Paste" from the context menu.
- Select [Edit] ⇨ [Paste] to perform the same operation as selecting "Paste" from the context menu.

■Procedure of [Insert and Paste]

The following describes the procedure of [Insert and Paste].

1. Select the following item.

[Edit] ⇨ [Insert and Paste]

2. The copy of a block is added as a new block in the hardware logic outline window.

■Operation details and restrictions

- Selecting the block on which the copy of a block is to be pasted and selecting [Edit] ⇨ [Paste] overwrites the setting values of the target block with the ones of the copy at a time.
- Selecting [Edit] ⇨ [Insert and Paste] adds the copy of a block as a new block in the hardware logic outline window.
- Only a block with the same type as the copy of a block can be specified as a "Paste" target. If a block whose type is different from the one of the copy has been selected, the copy cannot be pasted. The setting values of a multi function counter block cannot be overwritten unless the data type of sign/unsigned and 16 bits/32 bits matches between the paste target and the copy.

Adding a comment

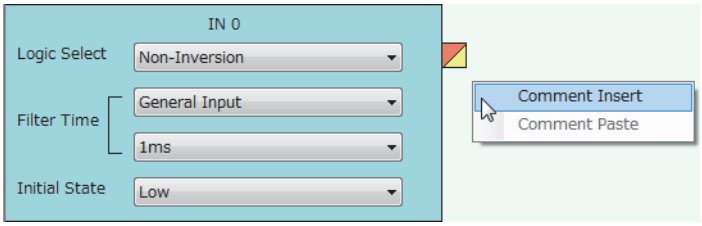
Comments can be arranged in the Work window.

Inserting a comment

The following describes the procedure of inserting a comment.

1. Right-click the area in which a comment is to be inserted in the Work window.

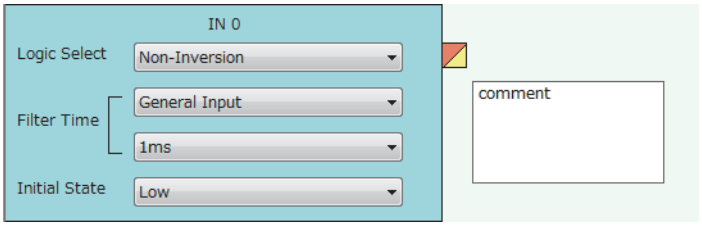
2. Select "Comment Insert" from the context menu.



Point

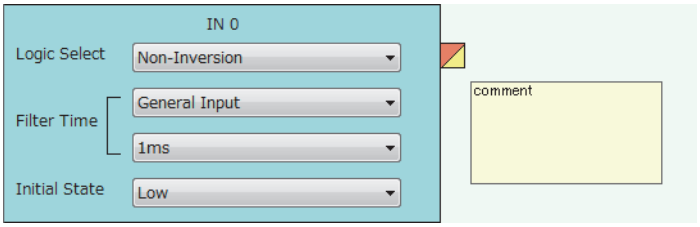
Comments can also be inserted by selecting [Edit] ⇨ [Comment Insert].

3. A new comment is inserted in the Work window. Enter a text in the comment.



4. To complete entering a text in the comment, click any area other than the comment in the Work window.

The comment is not selected (displayed in yellow).



5. To enter a text in the comment again, double-click the comment.

Deleting a comment

The following describes the procedure of deleting a comment.

1. Click the comment to be deleted.

The comment is selected (displayed in blue).

2. Right-click the comment and select "Comment Delete" from the context menu. Or press the Delete key.

Moving a comment

The following describes the procedure of moving a comment.

1. Click the comment to be moved.

The comment is selected (displayed in blue).

2. Drag and drop the comment into the area to which the comment is to be moved.

Cutting, copying, and pasting a comment

The following describes the procedure of cutting, copying, and pasting a comment.

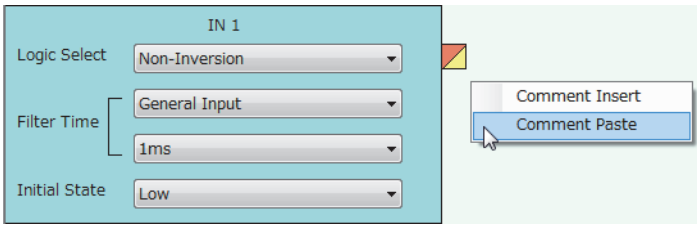
1. Click the comment to be cut or copied.

The comment is selected (displayed in blue).

2. Right-click the comment, and select "Comment Cut" or "Comment Copy" from the context menu.

3. Right-click the area on which the comment is to be pasted in the Work window.

4. Select "Comment Paste" from the context menu.



5. The cut or copied comment is pasted in the Work window.

Point

A comment can be overwritten with the cut or copied comment by selecting another comment and executing "Comment Paste".

Displaying or hiding a comment

Whether to display or hide a comment can be switched.

[View] ⇨ [Comment Display]

The setting of whether to display or hide a comment is also reflected to printed results.

2.7 Library Function

A library is a block in which the types and setting values of main blocks have been combined. Libraries can be shared among multiple projects by registering them in the configuration tool.
There are the following two types of libraries.

Library type	Description
Library provided by the manufacturer	The setting to enable specific functions has been applied to main blocks. Users can save time to create the hardware logic by using a library for a desired function.
User library	The blocks set by users can be exported as a user library. By exporting the blocks as a library, users can utilize the blocks that are frequently used for other projects and save time to create the hardware logic.

Export

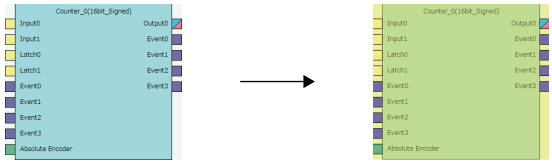
The blocks arranged in the hardware logic outline window can be exported as a library file (.fiolib).
By exporting the blocks whose operations have been checked as library files and registering them as libraries, users can utilize them and save time to create the hardware logic.
Users can name library files when exporting blocks as libraries.

Export procedure

The following describes the export procedure.

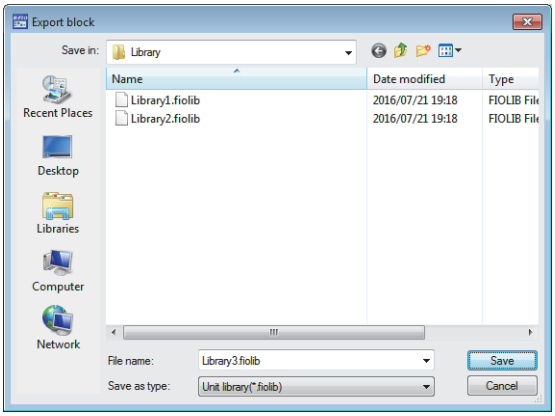
1. Click the block to be exported.

The block is highlighted (displayed in yellow) and selected.



2. Right-click the block and select "Export block" from the context menu.

3. Store the library file.

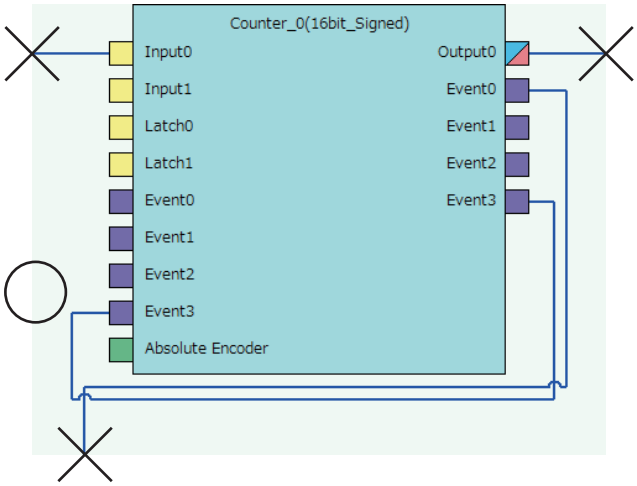


Point

Users can also export multi function counter blocks by right-clicking a multi function counter block name (Counter_☐) in the tree of the Navigation window and selecting "Export block" from the context menu.

Operation details and restrictions

- Multiple blocks cannot be exported at a time. Only a single block can be exported.
- The link between the block to be exported and other blocks is not exported. However, when the block to be exported is a multi function counter block, the link information in the multi function counter block detail window and the link between Event terminals of a multi function counter block in the hardware logic outline window are also exported.
- The following table shows which main block can be exported.

Section*1	Target block	Export	Item to be exported
1	• External input block • Inter-module synchronization signal input terminal • Y device terminal • OUT terminal	Impossible	—
2	• Parallel encoder block • SSI encoder block	Possible	Setting values of the block
3	Multi function counter block	Possible	• Setting values (the setting value of User Address is not included), links, and comments in the multi function counter block detail window • Link between Event terminals of a multi function counter block in the hardware logic outline window (the links across blocks cannot be exported.) 
4	Logical operation block	Impossible	—
5	• External output block • SI device terminal	Impossible	—

*1 This indicates a section in the hardware logic outline window.

Library operation

The following describes the library.

Restriction

No library exists immediately after the installation of GX Works3. Register libraries as necessary.

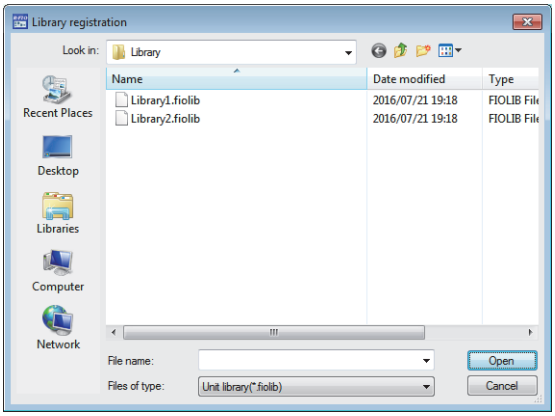
Registering a library

Register the libraries provided by the manufacturer and exported user libraries in the configuration tool. Registering libraries adds the blocks in library files in the "Library" tab in the Element Selection window.

1. Select the following item.

[Edit] ⇨ [Library Registration]

2. Open a library file to be registered in the configuration tool.



3. The registered library is added in the "Library" tab in the Element Selection window.

How to use a library

Registered libraries can be selected from the Element Selection window and dragged and dropped into the Work window to arrange them in the same way as main blocks.

1. Select a block in the Element Selection window. Drag and drop the block into the hardware logic outline window.
When a block is dragged into the hardware logic outline window, the area to which the block can be dropped is highlighted.

2. When the selected block is dropped into the Work window, the block is automatically arranged in the highlighted area.
The block name of the arranged library is the same as the main block. For example, when a registered block as a library is a 16-bit unsigned counter, the name of the block arranged in the work window is "Counter_☐(16bit_Unsigned)".

How to delete a library

Libraries that are no longer required can be deleted from the configuration tool.

1. Click the library to be deleted in the Element Selection window.

2. Right-click the library and select "Library Delete" from the context menu.

Even though the library is deleted, the library file is not deleted from the storage folder. To use the library again, register the library again.

Point

Select [Edit] ⇨ [Library Delete] to perform the same operation as selecting "Library Delete" from the context menu.

2.8 Online Functions

Connect the computer in which GX Works3 has been installed and the CPU module, and read or write data from/to the flexible high-speed I/O control module through the CPU module. The following table lists the online functions.

Function	Description	Reference
Start I/O number setting	Sets the start I/O number of the flexible high-speed I/O control module.	☞ Page 114 Start I/O number setting
Write to Module (Execution memory)	Writes the hardware logic to only the execution memory.	☞ Page 115 Writing data to the module
Write to Module (execution + flash ROM)	Writes the hardware logic to both the execution memory and the flash ROM.	
Read from Module (Flash ROM)	Reads the hardware logic saved in the flash ROM to the configuration tool.	☞ Page 115 Reading data from the module
Verify with Module (Flash ROM)	Verifies the hardware logic in the configuration tool and the setting data written in the flash ROM.	☞ Page 116 Verifying with the module
Module operation	Starts/stops the hardware logic control.	☞ Page 117 Module operation
Monitor	Monitors the ON/OFF states of I/O terminals and count values. The following lists the items that can be monitored.*1 • ON/OFF states of external input terminals corresponding to external input blocks (hardware logic outline window) • High/Low states of output terminals in an external input block (hardware logic outline window) • High/Low states of the Output 0 terminal in a multi function counter block (hardware logic outline window) • High/Low states of input terminals in an external output block (hardware logic outline window) • ON/OFF states of external output terminals corresponding to external output blocks (hardware logic outline window) • Internal action states of input terminals and count values in a counter timer block (multi function counter block detail window)	☞ Page 118 Monitor

*1 The states of I/O signals in the hardware logic are described as High and Low. For details, refer to the following.
☞ Page 145 Signal status name

Restriction

If a communication error occurs while an online function is being executed, the possible cause is one of the following causes. Check the target module and communication status.

- A module with the target start I/O number does not exist.
- A communication error has occurred during online access.
- Cable error

Start I/O number setting

Set the start I/O number of the flexible high-speed I/O control module.

[Online] ⇨ [Start I/O number setting]

Setting the I/O assignment to GX Works3 and the configuration tool enables writing of data into the flexible high-speed I/O control module and the monitor display through the CPU module.

However, match the I/O assignment of the flexible high-speed I/O control module in the configuration tool and that in GX Works3. If the I/O assignment is changed on GX Works3 and data writing to the module, the monitor display, or simulation is executed after the start I/O number has been set with the configuration tool, a communication error will occur.

Writing data to the module

Write the hardware logic to the execution memory of the flexible high-speed I/O control module.
The execution memory and the flash ROM are available as the write destination of the data. Select only the execution memory or both the execution memory and the flash ROM as the write destination.
Because the hardware logic written into the flash ROM is read to the execution memory at power-on, a control can be started without re-setting. However, the number of writable times to a flash ROM is 10000 times. Thus, using different write destinations as shown in the following examples is recommended.

- When the adjustment is repeated with changing the settings, select "Write to Module (execution memory)".
- After the adjustment is completed, select "Write to Module (execution + flash ROM)".

Users can set a password for the hardware logic by setting "Enable" in the [Security] menu of [Project]. For the hardware logic with a password, users are required to input the password for reading the hardware logic from the module.

Writing data

The following describes how to write data.

1. Select a writing method depending on a selected write destination.

[Online] ⇨ [Write to Module (execution memory)] or [Write to Module (execution + flash ROM)]

When "Write to Module (execution + flash ROM)" has been selected and the number of writes to flash ROM exceeds 10000 times, the error dialog box is displayed. Data can be written to the module even in such situation. However, the data written in the flash ROM is not guaranteed.

2. When the following conditions are satisfied, set a password.

- "Enable" has been set in the [Security] menu of [Project].
- [Write to Module (execution + flash ROM)] has been selected.

Point

- If data writing is executed while the hardware logic control is operating, the hardware logic control will stop and the data will be written to the module. After the data writing is completed, the hardware logic control restarts.
- If data writing is executed while the hardware logic control is stopped, the stop status of the hardware logic control will continue even after the completion of data writing.

Restriction

If the 'Hardware logic control start request' (Y4) in the program is turned on while writing to a module from the configuration tool, writing may fail.

Reading data from the module

Read the hardware logic saved in the flash ROM of the flexible high-speed I/O control module to the configuration tool.
The hardware logic being edited is overwritten with the read data. Save the hardware logic before the data reading as necessary.

Reading data

The following describes how to read data.

1. Select the following item.

[Online] ⇨ [Read from Module (flash ROM)]

2. To read the hardware logic with a password, input the password set when the data was written to the module.

Verifying with the module

Verify the hardware logic in the project file being edited and the hardware logic saved in the flash ROM of the flexible high-speed I/O control module. Verification results are displayed in a list and mismatches can be checked.

Verifying the hardware logic

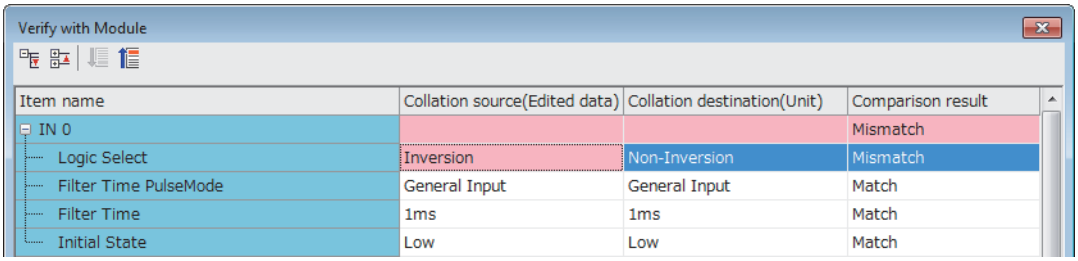
The following describes how to verify the hardware logic.

1. Select the following item.

[Online] ⇨ [Verify with Module (Flash ROM)]

2. Verification results are displayed.

In the verification result window, the match/mismatch status (the links to the input terminals of each block, setting values of the block, and User Address) are displayed.



Item name	Collation source(Edited data)	Collation destination(Unit)	Comparison result
IN 0			Mismatch
Logic Select	Inversion	Non-Inversion	Mismatch
Filter Time PulseMode	General Input	General Input	Match
Filter Time	1ms	1ms	Match
Initial State	Low	Low	Match

3. Double-click a mismatch to jump to the corresponding section in the Work window. When the links are mismatched, the corresponding link in the Work window is selected.

Module operation

The hardware logic control can be started or stopped with the configuration tool.
Start or stop the hardware logic control under the following situations.

Operation	Situation
Starting the hardware logic control	• Use this operation after the power is turned on. The hardware logic control is stopped after the power-on. Check the safety and select [Hardware logic control start] to start the control. • Use this operation after the completion of the simulation. The hardware logic control is stopped after the completion of the simulation. Select [Hardware logic control start] to start the control again.
Stopping the hardware logic control	Use this operation to check the module and system status. Select [Hardware logic control stop] to stop the control. When checking the module and system status, stop the hardware logic control for safety.

How to start or stop the hardware logic control

The following describes how to start or stop the hardware logic control.

[Online] ⇨ [Module operation] ⇨ [Hardware logic control start] or [Hardware logic control stop]

This operation can be performed during monitoring or watch operation. For details, refer to the following.

☞ Page 118 Monitor
☞ Page 120 Watch Function

Restriction

When the hardware logic control is stopped, the count value of the counter timer block is reset. To stop the count operation without resetting the count value, turn off the count enable of the counter timer block.

Checking the operating status of the hardware logic control

The operating status of the hardware logic control can be checked in the toolbar during monitor execution.

Hardware logic control status	Button status	Remarks
During hardware logic control		The "Hardware logic control stop" button can be pressed during the hardware logic control.
During hardware logic control stop		The "Hardware logic control start" button can be pressed during the hardware logic control stop.

When the hardware logic control is started or stopped from the configuration tool, 'Hardware logic control flag' (X4) turns on or off. Thus, users can check the operating status of the hardware logic control by monitoring the status of 'Hardware logic control flag' (X4) in GX Works3.

For details on 'Hardware logic control flag' (X4), refer to the following.

☞ Page 259 Hardware logic control flag

Relations of operations with output signals and window operations

The hardware logic control can be started or stopped with the configuration tool or output signals (Y4, Y5). The (start or stop) status of the hardware logic control changes depending on the last operation performed.

- When [Hardware logic control start] is selected with 'Hardware logic control stop request' (Y5) on, the hardware logic control is started.
- When [Hardware logic control stop] is selected with 'Hardware logic control start request' (Y4) on, the hardware logic control is stopped.

For details on 'Hardware logic control start request' (Y4) and 'Hardware logic control stop request' (Y5), refer to the following.

☞ Page 263 Hardware logic control start request
☞ Page 264 Hardware logic control stop request

Monitor

The High/Low states of I/O terminals and count values are displayed in the hardware logic outline window or a multi function counter block detail window.

The following lists the items that can be monitored.

Window	Block	Item
Hardware logic outline window	External input block	ON/OFF state of the corresponding external input terminal
		High/Low state of the output terminal
	Multi function counter block	High/Low state of the Output 0 terminal
	External output block	High/Low state of the input terminal
		ON/OFF state of the corresponding external output terminal
Multi function counter block detail window	Counter timer block	• Internal action state of the input terminal • Count value

How to use

The following shows how to start, stop, and end the monitoring.

[Online] ⇨ [Monitor] ⇨ [Start Monitoring] (or [Start Monitoring (All Windows)], [Stop Monitoring], or [End Monitoring] (or [End Monitoring (All Windows)])

The monitoring can be started, stopped, or ended under the following conditions.

Operation	Condition
[Start Monitoring]	Can be used when the window mode*1 is "Edit mode" or "Monitor mode".
[Start Monitoring (All Windows)]	Can be used other than in the situation where no project is opened.
[Stop Monitoring]	Can be used when the window mode*1 is "Monitor mode".
[End Monitoring]	Can be used when the window mode*1 is "Monitor mode".
[End Monitoring (All Windows)]	Can be used other than in the situation where no project is opened.

*1 The current window mode can be checked on the status bar.

Point

The monitoring and watch can be simultaneously started or stopped by performing [Online] ⇨ [Monitor] ⇨ [Start Monitoring (All Windows)] or [End Monitoring (All Windows)].

For details on the watch function, refer to the following.

☞ Page 120 Watch Function

Restriction

Although switching of the window between the hardware logic outline window and multi function counter block detail windows and changing of the display position by the map display window are allowed even during monitoring, operations other than them cannot be executed.

Monitor display

■Monitor display target

The following terminals can be monitored. Monitor values of the terminals that have not been linked are not displayed.

Window name	Block name		Terminal	Item to be monitored*2
Hardware logic outline window	External input block	IN 0 to IN B	—	ON/OFF state of the input terminal (external terminal)
			Output terminal	High/Low state of a terminal
	Multi function counter block	Counter_0 to Counter_7	Output 0 terminal	High/Low state of a terminal
	External output block	OUT 0 to OUT 7	Input terminal	High/Low state of a terminal
			—	ON/OFF state of the output terminal (external terminal)
		OUT 0_DIF to OUT 5_DIF	Input terminal	High/Low state of a terminal
			—	ON/OFF state of the output terminal (external terminal)
Multi function counter block detail window	Counter timer block	Counter_Timer_0, Counter_Timer_1*3	RUN terminal	Internal action state of the counter timer block to an input terminal*1
			STOP terminal	
			UP terminal	
			DOWN terminal	
			PRESET terminal	
			Count Value terminal	Count value

*1 When a buffer memory address is assigned to User Address of each terminal and a control is executed, the input state from the buffer memory cannot be monitored.
When checking the input state from the buffer memory, monitor the corresponding buffer memory area with "Device/Buffer Memory Batch Monitor" of GX Works3.

*2 For the input terminal (external terminal) of an external input block and the output terminal (external terminal) of an external output block, the latest ON/OFF state of each terminal is displayed even while the hardware logic is stopped.
For other items, monitor values are not updated while the hardware logic control is stopped.

*3 Counter_Timer_1 is not displayed when a 32-bit multi function counter block is used.

■Item to be monitored

Monitor values are displayed at the upper section of the terminal or block. The ON/OFF state or Low/High state is displayed as follows.

- OFF: OFF state or Low state
- ON: ON state or High state

When the monitoring has stopped, the values immediately before the monitoring stop are displayed.

2.9 Watch Function

Current values such as High/Low states of I/O terminals and count values can be displayed in a watch window.
The following table lists items that can be displayed.

Window	Block	Item
Hardware logic outline window	External input block	High/Low state of the output terminal
	Multi function counter block	High/Low state of the Output 0 terminal
	External output block	High/Low state of the input terminal
Multi function counter block detail window	Counter timer block	• Internal action state of the input terminal • Count value

An item to be watched must be registered with a watch window. Terminals and blocks that are arranged on the hardware logic in the module can be collectively registered with a watch window by reading out the hardware logic that was written to the module.

How to use

1. Register an item to be watched with a watch window. (☞ Page 121 Registration with a watch window)
The following are displayed when an item is registered on the watch window.

Watch1

Name	Current Value	Display Format	Data Type	Comment
External Input Block Output Terminal				External Input Block Output Terminal
IN 0	--	ON/OFF	Bit	External Input Block Output Terminal
IN 1	--	ON/OFF	Bit	External Input Block Output Terminal
IN 2	--	ON/OFF	Bit	External Input Block Output Terminal
IN 3	--	ON/OFF	Bit	External Input Block Output Terminal
IN 4	--	ON/OFF	Bit	External Input Block Output Terminal
IN 5	--	ON/OFF	Bit	External Input Block Output Terminal
IN 6	--	ON/OFF	Bit	External Input Block Output Terminal
IN 7	--	ON/OFF	Bit	External Input Block Output Terminal
IN 8	--	ON/OFF	Bit	External Input Block Output Terminal
IN 9	--	ON/OFF	Bit	External Input Block Output Terminal
IN A	--	ON/OFF	Bit	External Input Block Output Terminal
IN B	--	ON/OFF	Bit	External Input Block Output Terminal

Item	Description
Name	Indicates the terminal name or block name in the Work window.
Current Value	Indicates current values (of the item to be watched) obtained from the module when the watch starts.
Display Format	Indicates the display format of "Current Value".
Data Type	Indicates the data type of the item to be watched.
Comment	Indicates the detailed comment of the item to be watched.

- The watch can only be performed on the watch window. "Current Value" cannot be changed on the watch window.
- The description such as "Name" and "Display Format" cannot be changed.

2. Start or stop the watch.

[Online] ⇨ [Watch] ⇨ [Start Watching] or [Stop Watching]
During the watch operation, "(Watching)" is added to the title of watch window.

Point

The monitoring and watch can be simultaneously started or stopped by performing [Online] ⇨ [Monitor] ⇨ [Start Monitoring (All Windows)] or [End Monitoring (All Windows)].

Registration with a watch window

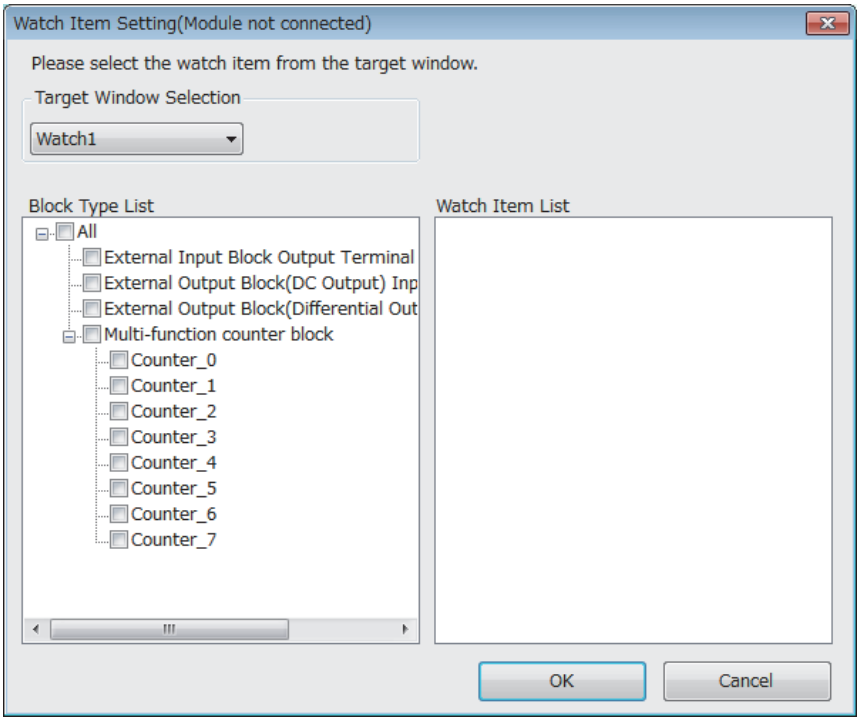
■Watch item selection

Select an item to be watched and register it with a watch window.

1. Display a watch window.

[View] ⇨ [Docking Window] ⇨ [Watch 1] to [Watch 4]

2. Right-click the watch window and select "Watch Item Setting" from the context menu.
3. Select a watch window to be used with "Target Window Selection".



4. Select an item to be registered with the watch window with "Block Type List" and "Watch Item List".
5. The item is registered with the watch window by clicking "OK".

■Registration from an external input block, a multi function counter block, or an external output block

Register an external input block, a multi function counter block, or an external output block with a watch window with mouse operation.

1. Click and select an external input block, a multi function counter block, or an external output block.
2. Right-click the block and perform the following operation from the context menu.

[Register to Watch Window] ⇨ [Watch 1] to [Watch 4]

Point

Drag-and-drop of an external input block, a multi function counter block, or an external output block into the watch window also registers the block with the watch window.

■Watch items batch read

Read out the hardware logic that was written to the module and collectively register terminals and blocks that are arranged on the hardware logic in the module with a watch window.

1. Select a watch window targeted for the watch items batch read.

[Online] ⇨ [Watch] ⇨ [Watch Items Batch Read] ⇨ [Watch 1] to [Watch 4]

2. The hardware logic written in the module is read out and items are registered with the watch window.

■Watch item deletion

An item registered with a watch window can be deleted.

1. Click and select an item to be deleted from a watch window.
2. Right-click the item and select "Delete" from the context menu. Or press the Delete key.

Watch display

■Watch target

The following table lists terminals targeted for the watch.

Window name	Block name		Terminal	"Current Value" display*2
Hardware logic outline window	External input block	IN 0 to IN B	Output terminal	High/Low state of a terminal
	Multi function counter block	Counter_0 to Counter_7	Output 0 terminal	High/Low state of a terminal
	External output block	OUT 0 to OUT 7	Input terminal	High/Low state of a terminal
		OUT 0_DIF to OUT 5_DIF	Input terminal	High/Low state of a terminal
Multi function counter block detail window	Counter timer block	Counter_Timer_0, Counter_Timer_1*3	RUN terminal	Internal action state of the counter timer block to an input terminal*1
			STOP terminal	
			UP terminal	
			DOWN terminal	
			PRESET terminal	
			Count Value terminal	Count value

*1 When a buffer memory address is assigned to User Address of each terminal and a control is executed, the input state from the buffer memory cannot be monitored.
When checking the input state from the buffer memory, monitor the corresponding buffer memory area with "Device/Buffer Memory Batch Monitor" of GX Works3.

*2 "Current Value" is not updated while the hardware logic control is stopped.

*3 Counter_Timer_1 is not displayed when a 32-bit multi function counter block is used.

■"Current Value" display

The ON/OFF state or Low/High state is displayed as follows.

- OFF: OFF state or Low state
- ON: ON state or High state

When the watch operation has stopped, the value immediately before the stop is still displayed.

2.10 Debug Function

This function performs simulation and logic analyzer, and starts/stops continuous logging.

Simulation function

The simulation function verifies the hardware logic written into the flexible high-speed I/O control module without wiring with external devices.

With the configuration tool, create "simulation input data", the substitute for external input signals, and write the data into the flexible high-speed I/O control module to operate the hardware logic.

Simulation results can be saved in CSV files. Saved results can be visually checked with GX LogViewer.

Restriction

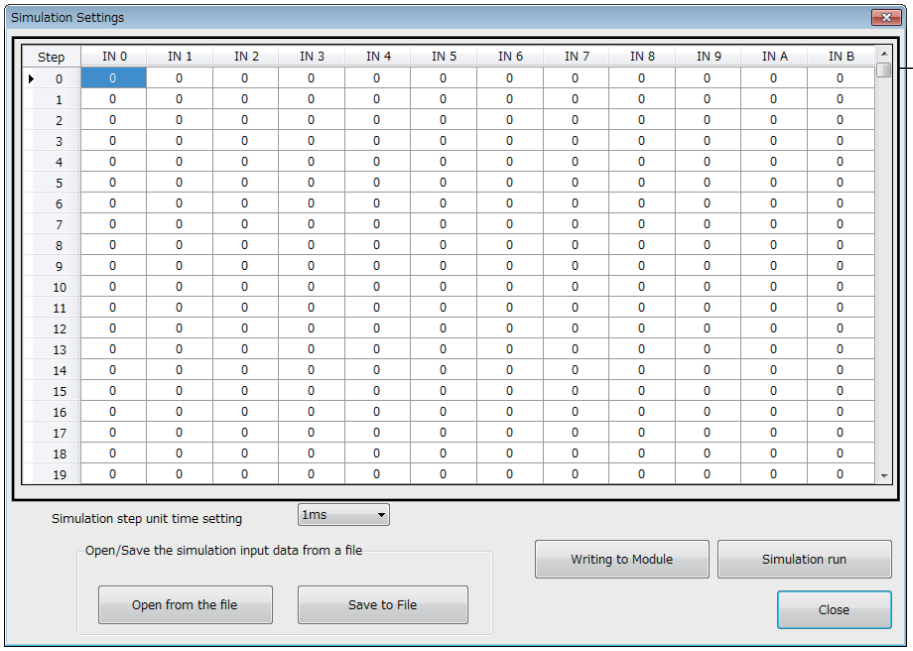
- During simulation, external outputs are actually turned on or off. Thus, execute the simulation under the situation in which the module is not connected with external devices or making sure that the ON/OFF of the output does not affect the system.
- Even though the simulation is executed with the hardware logic where an SI device terminal is linked, an interrupt request to the CPU module is not output.
- Even though the simulation is executed with the hardware logic where an SSI encoder block has been arranged, the communication with an SSI encoder will not be performed. Thus, position data of the SSI encoder is not reflected to simulation execution results.

Window layout

The following describes the configurations of the "Simulation Settings" window. In this window, "simulation input data" can be created and written into the flexible high-speed I/O control module or simulation is executed.

(Simulation input data is composed of external input signal data and "Simulation step unit time setting" in this window.)

[Debug] ⇨ [Simulation]



The following table lists the items to be displayed in the "Simulation Settings" window.

Item	Description
External input signal data setting area	To verify the hardware logic, set the status of data to be imported as external input signals (IN 0 to IN B). The setting value 0 indicates that an external input signal is off, and the setting value 1 indicates that an external input signal is on. Click a cell and switch the value between 0 (OFF) and 1 (ON). Columns in the area indicate the 12 points of the external input signals (IN 0 to IN B) and rows indicate 2048 steps (0 to 2047). In the simulation, the hardware logic is executed by one step at every cycle which is set in "Simulation step unit time setting".
"Simulation step unit time setting"	Set the cycle to switch external input signal data to the next step. Sampling of simulation results is also executed at this set cycle.
"Open from the file" button	Read the simulation input data saved in a CSV file to the "Simulation Settings" window.
"Save to File" button	Save the simulation input data in the "Simulation Settings" window into a CSV file.
"Writing to Module" button	Write the simulation input data set in the "Simulation Settings" window into the module. The written data is held until the flexible high-speed I/O control module is powered off. To execute the simulation, write simulation input data into the module in advance.
"Simulation run" button	Execute the simulation with the simulation input data and hardware logic written into the module. After the completion of the simulation, the "A simulation execution result is saved." dialog box is displayed. Save the simulation execution result.
"Close" button	Close the "Simulation Settings" window.

Data that can be acquired as simulation execution results

Users can acquire the following data by executing simulation. The execution result data acquired after simulation can be saved in a CSV format file. The file can be visually checked with GX LogViewer.

Window name	Block name		Terminal	Data to be acquired
Hardware logic outline window	External input block	IN 0 to IN B	Output terminal	High/Low state of a terminal
	External output block	OUT 0 to OUT 7	Input terminal	
		OUT 0_DIF to OUT 5_DIF	Input terminal	
Multi function counter block detail window*1	Counter timer block	Counter_Timer_0, Counter_Timer_1*2	RUN terminal	Internal action state of the counter timer block to an input terminal
			STOP terminal	
			UP terminal	
			DOWN terminal	
			PRESET terminal	
			Count Value terminal	Count value

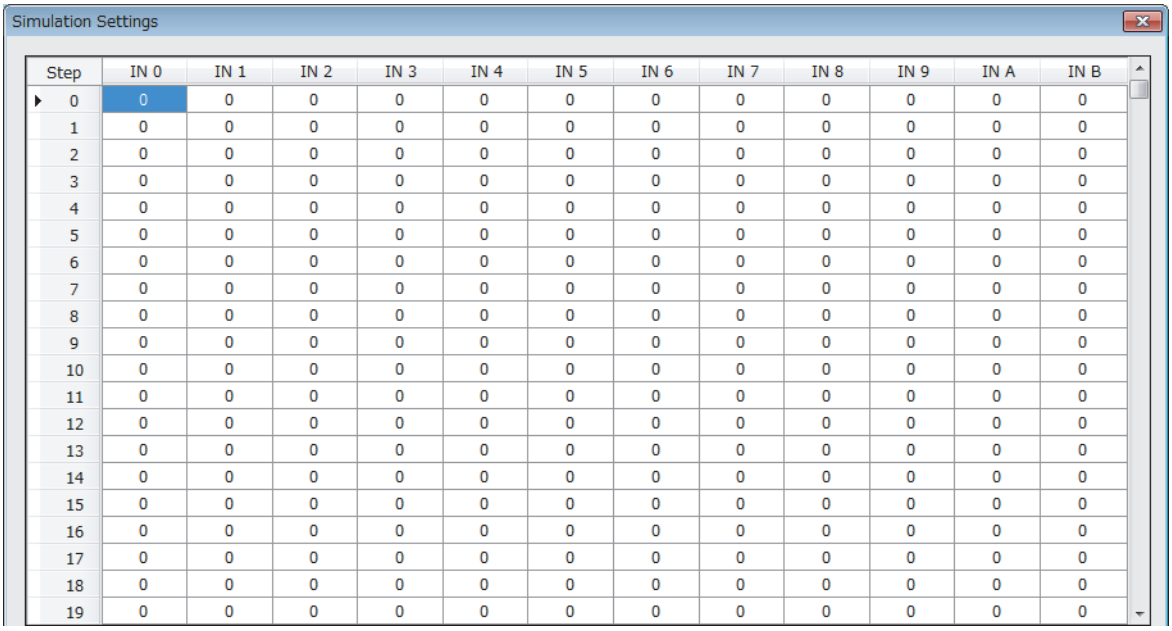
*1 Simulation results of the blocks arranged in the hardware logic outline window are acquired.
*2 Data of Counter_Timer_1 is not acquired when a 32-bit multi function counter block is used.

How to use

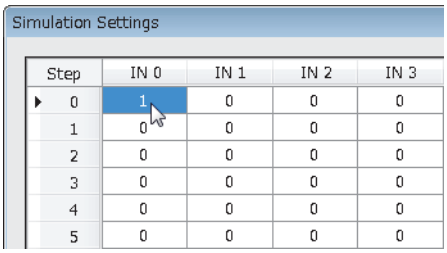
The following describes how to use the simulation function.

1. Open the "Simulation Settings" window.

[Debug] ⇨ [Simulation]



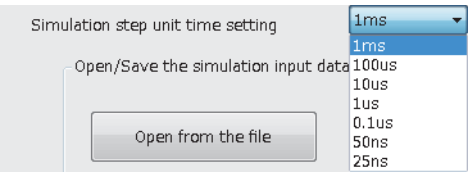
2. Click each cell in the external input signal data setting area and switch "0: External input signal OFF" and "1: External input signal ON". By default, "0: External input signal OFF" has been set in all cells.



3. Set "Simulation step unit time setting".

Changing "Simulation step unit time setting" changes the following timing.

- Timing to move external input data to the next step (Data is changed by one step at the set unit time.)
- Acquisition interval of execution result data (An execution result for one step is acquired at the set unit time.)



Users can save the set simulation input data with the "Save to File" button (Data is saved only in the CSV format).

The following shows the CSV file format specifications. When creating a CSV file with any method other than the one using the "Simulation Settings" window, create the file in the same format.

Item name	Character
Delimiter	Comma (,)
Return code	CRLF (0x0D, 0x0A)

	A	B	C	D	E	F
1	8					
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	0	0	0	0
10	0	0	0	0	0	0

Row	Description	
1st row	The simulation step unit time setting is output.	
	Simulation step unit time setting	Value output into a CSV file
	1ms	8
	100μs	7
	10μs	6
	1μs	5
	0.1μs	4
	50ns	3
	25ns	2
2nd to 2049th row	One row indicates one step, and external input signal data is output as "0: OFF" or "1: ON" in order of IN 0, IN 1 to IN B starting from the left column.	

Users can open a saved CSV file from "Open from the file" of "Open/Save the simulation input data from a file".

Restriction

- The "Writing to Module" button only writes simulation input data into the module. Write the hardware logic into the module in advance.
- The simulation input data written into the module is cleared when the module is powered off.
- When the external input signal data in the CSV file selected from "Open from the file" has less than 2048 steps or less than 12 rows, the data is not reflected in the "Simulation Settings" window. When the data has 2048 steps or more or 12 rows or more, the external input signal data of 2048 steps or 12 rows is read. The data that is not the read target is discarded.

4. Click "Writing to Module" to write the simulation input data into the flexible high-speed I/O control module.

Restriction

If a communication error has occurred during the communication with the module with the simulation function, the possible cause is one of the following causes. Check the target module and communication status.

- A module with the target start I/O number does not exist.
- A communication error has occurred during online access.
- Cable error

5. Click "Simulation run" to execute the simulation of the written simulation input data.

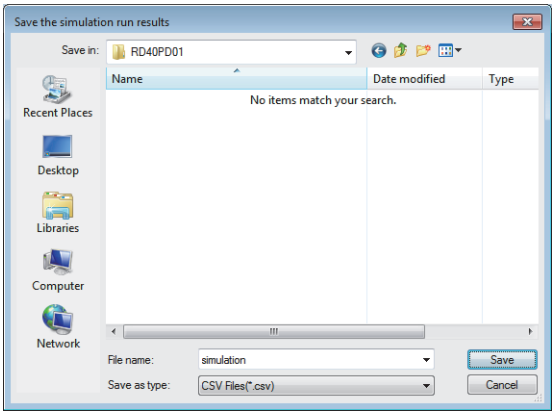
The simulation is executed by using the hardware logic and simulation input data written into the module. When the simulation input data has been edited, always press the "Writing to Module" button to write the data into the module.

Restriction

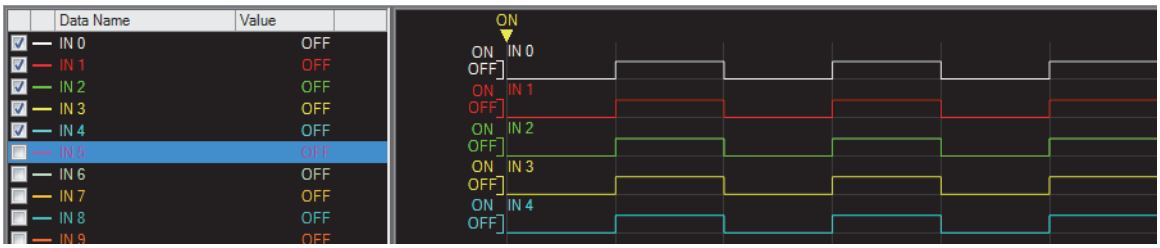
- If the simulation is started during a hardware logic control, the hardware logic control will stop. After the completion of the simulation, the stop status of the hardware logic control will continue. To start the hardware logic control after the simulation, turn on 'Hardware logic control start request' (Y4).
- If 'Hardware logic control start request' (Y4) or 'Hardware logic control stop request' (Y5) is turned on during the simulation, simulation execution results cannot be properly acquired.
- If 'Hardware logic control start request' (Y4) is turned on while simulation execution results are being acquired after the simulation, the acquisition of the simulation execution results will stop.
- When the simulation is executed during execution of the logic analyzer function, the logic analyzer function stops. The logic analyzer function will not restart after completion of the simulation execution. To restart it, issue a trigger start request.
- When the simulation is executed during continuous logging, the continuous logging stops. The continuous logging will not restart after completion of the simulation execution. To restart it, issue a continuous logging start request.
- When the simulation is executed during A/D conversion value logging, A/D conversion value logging stops. The A/D conversion value logging will not restart after completion of the simulation execution. To restart it, perform the A/D conversion value logging settings and turn on 'Hardware logic control start request' (Y4). For the A/D conversion value logging settings, refer to the following.

☞ Page 32 A/D Conversion Value Logging Function

6. Save the simulation execution results. (Data is saved only in the CSV format.)



7. The simulation execution results can be visually checked with GX LogViewer.



For details on GX LogViewer, refer to the following.

📖 GX LogViewer Version 1 Operating Manual

CSV file format of simulation execution results

The following shows the CSV file format specifications of simulation execution results.

Item name	Character
Delimiter	Comma (,)
Return code	CRLF (0x0D, 0x0A)

	A	B	C	D	E	F
1	[LOGGING]	RD40PD01_1	2	3	4	
2	BIT[1;0]	BIT[1;0]	BIT[1;0]	BIT[1;0]	BIT[1;0]	BIT[1;0]
3	IN 0	IN 1	IN 2	IN 3	IN 4	IN 5
4	1	1	1	1	1	1
5	1	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	0	0	0	0
10	0	0	0	0	0	0

■1st row

[LOGGING], RD40PD01_1, 2, 3, and 4 are output (fixed output).

■2nd row

The data type of the simulation execution result is output.

Data type	Bit	Word (unsigned) Double word (unsigned)	Word (signed) Double word (signed)
Value output into a CSV file	BIT[1;0]	ULONG[DEC.0]	LONG[DEC.0]

TRIGGER [*] is output in the last column of the 2nd row (fixed output).

■3rd row

The name of each input or output terminal is output.

I/O terminal	Name to be output
• Input terminal in the hardware logic outline window • Output terminal in the hardware logic outline window	Block name
• Input terminal in a multi function counter block detail window • Output terminal in a multi function counter block detail window	Multi function counter block name, block name, terminal name

Trigger is output in the last column of the 3rd row (fixed output).

■4th to 2051st row

The acquired data is chronologically output (at the set simulation step unit time).

Window name	Block name		Terminal	Data type	Data to be acquired
Hardware logic outline window	External input block	IN 0 to IN B	Output terminal	BIT[1;0]	Indicates the High/Low state of a terminal. 0: Low state 1: High state
	External output block	OUT 0 to OUT 7	Input terminal	BIT[1;0]	
		OUT 0_DIF to OUT 5_DIF	Input terminal	BIT[1;0]	
Multi function counter block detail window	Counter timer block	Counter_Timer_0, Counter_Timer_1*1	Count Value terminal	ULONG[DEC.0]/ LONG[DEC.0]	Count value
			UP terminal	BIT[1;0]	Indicates the internal action state of the counter timer block to an input terminal. 0: Internal action stop status 1: Internal action execution status
			DOWN terminal	BIT[1;0]	
			PRESET terminal	BIT[1;0]	
			RUN terminal	BIT[1;0]	
			STOP terminal	BIT[1;0]	

*1 Data of Counter_Timer_1 is not acquired when a 32-bit multi function counter block is used.

[*] is output in the last column of the 4th row (fixed output).

Point

The data in the 0th step of the simulation input data is reflected to the initial value of the Output terminal in the external input block.

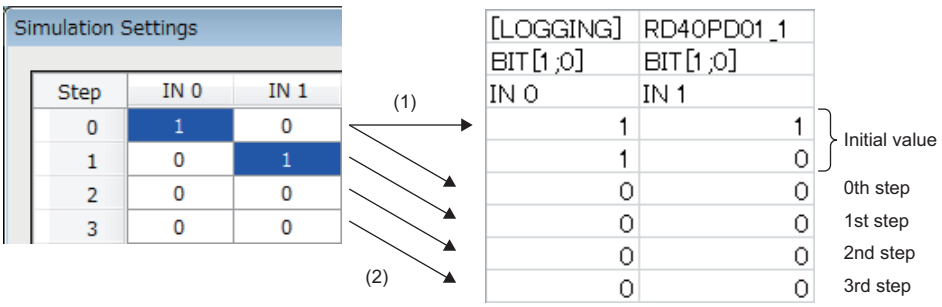
Because the processing time of the hardware logic affects the simulation step, the following time is required until the simulation input data changes the Output terminal in the external input block.

- When "Simulation step unit time setting" is "25ns", it takes time of 2 steps.
- When "Simulation step unit time setting" is other than "25ns", it takes time of 1 step.

Ex.

When simulation is executed with the simulation step unit time of 25ns and the external input block filter time of 0μs, the simulation result of an external input block is reflected as follows.

(When the filter time has been set, the simulation result is reflected after the time of the steps for the filter time passes.)



(1) Reflected to the initial values.

(2) Reflected to the processing of external input blocks after two steps.

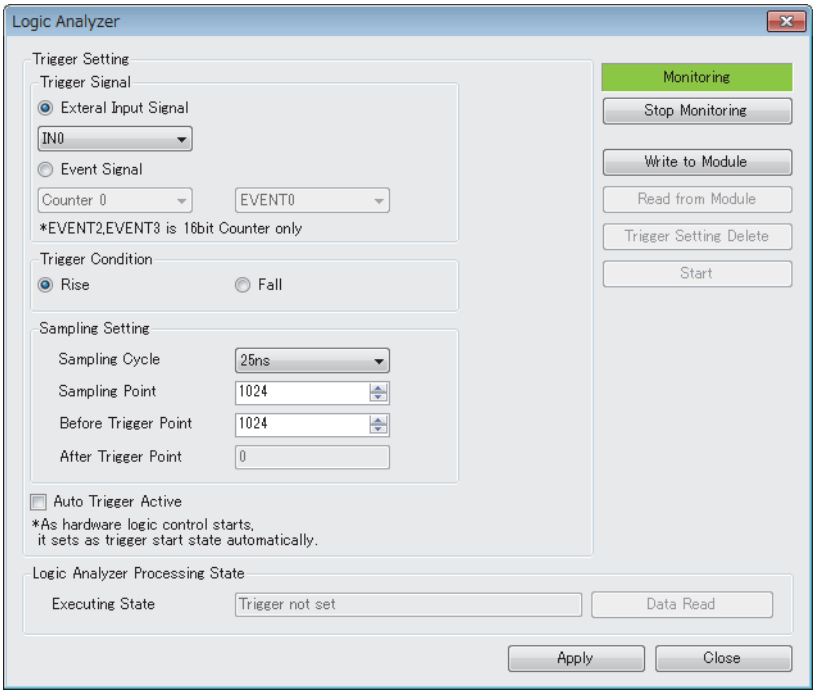
Logic analyzer function

When the trigger setting that has been set in advance and the states of the hardware logic match, the logic analyzer function acquires the states of the hardware logic before and after the match timing as sampling data.
Acquired sampling data can be saved in CSV files. Saved results can be visually checked with GX LogViewer.

Window layout

The following describes the configurations of the "Logic Analyzer" window. Setting a trigger, monitoring the trigger status, and reading sampling data can be performed in this window.
Open the "Logic Analyzer" window by performing any of the following operations.

Method	Operation
Opening the window from the menu bar	[Debug] ⇨ [Logic Analyzer]
Opening the window from an external input block or a multi function counter block	(1) Click and select an external input block or a multi function counter block. (2) Right-click the block and select "Logic Analyzer" from the context menu. (3) When a dialog box asking whether to set the selected block for a trigger is displayed, select [Yes].



The following table lists the items to be displayed in the "Logic Analyzer" window.

Item		Description
Trigger Signal		Select a signal to be a trigger. • When the "Logic Analyzer" window is opened from an external input block, the selected external input block is set for "Trigger Signal". • When the "Logic Analyzer" window is opened from a multi function counter block, the selected multi function counter block is set for "Trigger Signal".
Trigger Condition		Select a trigger condition from a rise or fall of the signal set for "Trigger Signal".
Sampling Setting	Sampling Cycle	Set a cycle to acquire sampling data.
	Sampling Point	Set the total number of points of sampling data to be acquired.
	Before Trigger Point	Set how many points of sampling data before a trigger occurs are to be acquired. A value equal to or lower than the setting value in "Sampling Point" can be set.
	After Trigger Point	A difference between "Sampling Point" and "Before Trigger Point" is automatically stored.
Auto Trigger Active		When this item is selected, the status switches to the trigger ready automatically when the hardware logic control is started.
Monitor status display label		The status of the monitor ("Monitoring" or "Stopping") is displayed on the monitor status display label. Switch the status of the monitor by using the "Start Monitoring" button or the "Stop Monitoring" button. • The "Stop Monitoring" button is displayed during monitoring. • The "Start Monitoring" button is displayed while the monitoring is stopped.
"Start Monitoring" button ("Stop Monitoring" button)		
"Write to Module" button		Write the trigger setting to the flexible high-speed I/O control module. Select the write destination from only the execution memory or both the execution memory and the flash ROM. For how to use different write destinations, refer to ☞ Page 132 Using different write destinations.
"Read from Module" button		Read data for the logic analyzer function from the flexible high-speed I/O control module and reflect the data to the "Logic Analyzer" window.
"Trigger Setting Delete" button		Delete the trigger setting written in the flexible high-speed I/O control module.
"Start" button ("Stop" button)		Switch the start state or stop state of the logic analyzer function. • When the logic analyzer function is in the start state, the "Stop" button is displayed. • When the logic analyzer function is in the stop state, the "Start" button is displayed. For details on the start state and stop state, refer to the following. ☞ Page 135 Switching the trigger status between the trigger ready and trigger stop
Logic Analyzer Processing State	Executing State	Display the execution status of the logic analyzer function of the flexible high-speed I/O control module.
	"Data Read" button	Read sampling data stored in the flexible high-speed I/O control module and save them in CSV files after the trigger condition is satisfied.
"Apply" button		Hold the contents of the trigger setting in the tool.
"Close" button		Close the "Logic Analyzer" window.

■Using different write destinations

The execution memory and the flash ROM are available as the write destination of the data. Select only the execution memory or both the execution memory and the flash ROM as the write destination.

Because the trigger setting written into the flash ROM is read to the execution memory at power-on, the logic analyzer function can be started without re-setting. However, the number of writable times to a flash ROM is 10000 times. Thus, using different write destinations as shown in the following examples is recommended.

- When the adjustment is repeated with changing the settings, write data only to the execution memory.
- After the adjustment is completed, write data to both the execution memory and flash ROM.

Data that can be acquired as sampling data

Window name	Block name		Terminal	Data to be acquired
Hardware logic outline window	External input block	IN 0 to IN B	Output terminal	High/Low state of a terminal
	External output block	OUT 0 to OUT 7	Input terminal	
		OUT 0_DIF to OUT 5_DIF	Input terminal	
Multi function counter block detail window*1	Counter timer block	Counter_Timer_0, Counter_Timer_1*2	RUN terminal	Internal action state of the counter timer block to an input terminal
			STOP terminal	
			UP terminal	
			DOWN terminal	
			PRESET terminal	
			Count Value terminal	Count value

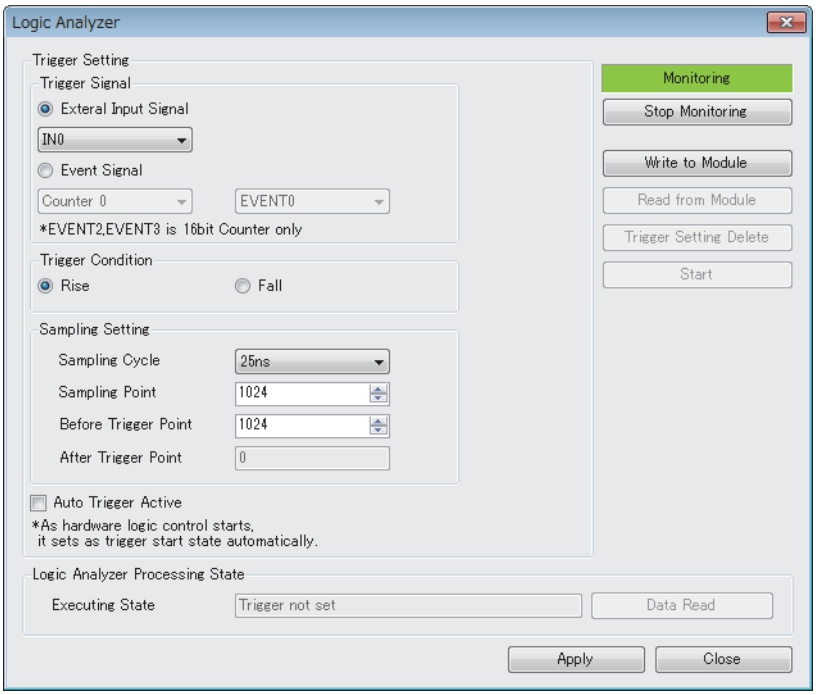
*1 Sampling data of the blocks arranged in the Hardware logic outline window are acquired.

*2 Data of Counter_Timer_1 is not acquired when a 32-bit multi function counter block is used.

How to use

The following describes how to use the logic analyzer function.

1. Write the hardware logic to the flexible high-speed I/O control module in advance.
2. Open the "Logic Analyzer" window.



3. Select a signal to be a trigger for "Trigger Signal".
4. In "Trigger Condition", select a trigger condition from a rise or fall of the signal set for "Trigger Signal".
5. In "Sampling Setting", set a cycle to acquire the sampling data.
6. To switch the trigger status to the trigger ready automatically when the hardware logic control is started, select "Auto Trigger Active".
7. Click "Write to Module" to write the trigger setting to the flexible high-speed I/O control module.

In addition, the trigger setting written in the flexible high-speed I/O control module can be operated with the following buttons.

- The trigger setting written in the flexible high-speed I/O control module can be read with the "Read from Module" button.
- The trigger setting written in the flexible high-speed I/O control module can be deleted with the "Trigger Setting Delete" button.

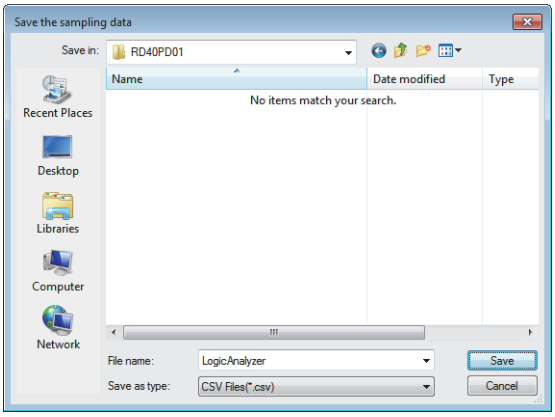
Restriction

If a communication error has occurred during the communication with the module with the logic analyzer function, the possible cause is one of the following causes. Check the target module and communication status.

- A module with the target start I/O number does not exist.
- A communication error has occurred during online access.
- Cable error

8. Click the "Start" button to switch the trigger status of the flexible high-speed I/O control module to the trigger ready.

9. When the trigger condition is satisfied and the sampling for the points specified in "After Trigger Point" is completed, "Executing State" becomes "Data Collection Complete". Click the "Data Read" button to save sampling data. (Data is saved only in the CSV format.)

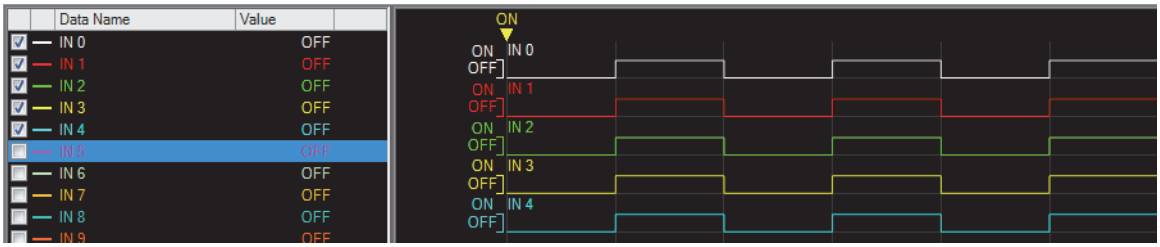


Point

If the trigger condition is satisfied again before the completion of sampling for the points specified in "After Trigger Point", next sampling starts at the timing when the second trigger condition is satisfied. For details, refer to the following.

☞ Page 296 When the trigger condition is satisfied twice or more times

10. The sampling data can be visually checked with GX LogViewer.



For details on GX LogViewer, refer to the following.

GX LogViewer Version 1 Operating Manual

Switching the trigger status between the trigger ready and trigger stop

The following describes how to switch the trigger status between the trigger ready and trigger stop.

■Switching the trigger stop to the trigger ready

Switch the trigger stop to the trigger ready with one of the following methods.

- Enabling the auto trigger and starting the hardware logic control
- Setting Start request (1) in 'Trigger start request' (Un\G120) in a program
- Clicking "Trigger Start" in the "Logic Analyzer" window

■Switching the trigger ready to the trigger stop

Switch the trigger ready to the trigger stop with one of the following methods.

- Stopping the hardware logic control
- Executing "Write to Module (execution memory)" or "Write to Module (execution + flash ROM)"
- Setting Stop request (1) to 'Trigger stop request' (Un\G121) in a program
- Clicking "Trigger Stop" in the "Logic Analyzer" window
- Detection of a trigger

Point

When the hardware logic control is stopped during hardware logic control and in the trigger ready, the trigger status switches to the trigger stop. At this time, the trigger stop is kept even if the hardware logic control is restarted. To switch the trigger status to the trigger ready, issue a trigger start request again.

CSV file format of sampling data

The following shows the CSV file format specifications of sampling data.

Item name	Character
Delimiter	Comma (,)
Return code	CRLF (0x0D, 0x0A)

	A	B	C	D	E	F	G	H
1	[LOGGING]	RD40PD01_1	2	3	4			
2	DATETIME[YYYY/MM/DD hh:mm:ss	us]	INDEX	BIT[1;0]	BIT[1;0]	BIT[1;0]	BIT[1;0]	BIT[1;0]
3	TIME	usec	INDEX	IN 0	IN 1	IN 2	IN 3	IN 4
4	2016/6/8 16:41	337000	1	1	1	1	1	1
5	2016/6/8 16:41	337000	2	1	1	1	1	1

■1st row

[LOGGING], RD40PD01_1, 2, 3, and 4 are output (fixed output).

■2nd row

- DATETIME[YYYY/MM/DD hh:mm:ss,us] and INDEX are output in the 1st to 3rd columns (fixed output).
- The data type of the sampling data is output in the 4th and later columns.

Data type	Bit	Word (unsigned) Double word (unsigned)	Word (signed) Double word (signed)
Value output into a CSV file	BIT[1;0]	ULONG[DEC.0]	LONG[DEC.0]

TRIGGER [*] is output in the last column of the 2nd row (fixed output).

■3rd row

- TIME, usec, and INDEX are output in the 1st to 3rd columns (fixed output).
- The name of each input terminal is output in the 4th and later columns.

I/O terminal	Name to be output
• Input terminal in the hardware logic outline window • Output terminal in the hardware logic outline window	Block name
• Input terminal in a multi function counter block detail window • Output terminal in a multi function counter block detail window	Multi function counter block name, block name, terminal name

Trigger is output in the last column of the 3rd row (fixed output).

■4th to 2051st row

- The time at which the sampling data is acquired is chronologically output (at the sampling period) in the first and second columns. The year/month/day hour:minute:second are output in the first column and the millisecond is output in increments of μ seconds in the second column.
- The acquired data is chronologically output (at the sampling period) in the third and later columns.

Window name	Block name		Terminal	Data type	Data to be acquired
Hardware logic outline window	External input block	IN 0 to IN B	Output terminal	BIT[1;0]	Indicates the High/Low state of a terminal. 0: Low state 1: High state
	External output block	OUT 0 to OUT 7	Input terminal	BIT[1;0]	
		OUT 0_DIF to OUT 5_DIF	Input terminal	BIT[1;0]	
Multi function counter block detail window	Counter timer block	Counter_Timer_0, Counter_Timer_1*1	Count Value terminal	ULONG[DEC.0]/ LONG[DEC.0]	Count value
			UP terminal	BIT[1;0]	Indicates the internal action state of the counter timer block to an input terminal. 0: Internal action stop status 1: Internal action execution status
			DOWN terminal	BIT[1;0]	
			PRESET terminal	BIT[1;0]	
			RUN terminal	BIT[1;0]	
			STOP terminal	BIT[1;0]	

*1 Data of Counter_Timer_1 is not acquired when a 32-bit multi function counter block is used.

[*] is output in the last column of the row where the trigger is detected (fixed output).

Continuous logging

A start request or stop request of continuous logging can be issued from the configuration tool. The continuous logging cycle can be set when issuing a start request.

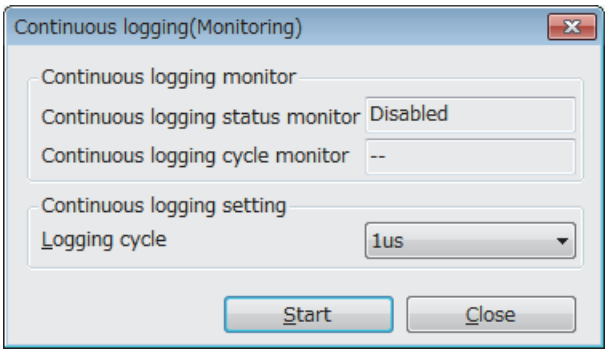
For details on the continuous logging function, refer to the following.

☞ Page 19 Continuous Logging Function

Window layout

The following describes the configurations of the "Continuous logging" window.

[Debug] ⇨ [Continuous logging]



The following table lists the items displayed in the "Continuous logging" window.

Item		Description
Continuous logging monitor	Continuous logging status monitor	Displays the data stored in Continuous logging status monitor (Un\G15010).
	Continuous logging cycle monitor	Displays the data stored in Continuous logging cycle monitor (μs) (Un\G15011). When continuous logging is not started, "-" is displayed.
Continuous logging setting	Logging cycle	Sets the cycle of continuous logging (any of a value between 1μs and 1000μs) at the issue of a start request of continuous logging.
"Start" button ("Stop" button)		Issues a start request or stop request of continuous logging. • While continuous logging is stopped, the "Start" button is displayed. • During execution of continuous logging, the "Stop" button is displayed.
"Close" button		Closes the "Continuous logging" window.

How to start or stop the continuous logging function

■Start request of continuous logging

1. Open the "Continuous logging" window.

[Debug] ⇨ [Continuous logging]

2. Set "Logging cycle".
3. Click the "Start" button.

■Stop request of continuous logging

1. Open the "Continuous logging" window.

[Debug] ⇨ [Continuous logging]

2. Click the "Stop" button.

2.11 Pattern Generator Function

The pattern generator function performs output from the external output terminals according to the output pattern data consisting of up to 8192 points that have been registered in the flexible high-speed I/O control module in advance.
If a counter timer block and pattern generator block are used together, periodic pattern output according to the clock in the flexible high-speed I/O control module and High/Low output corresponding to the input pulse from the external input terminals can be performed.
Users can easily create output pattern data, which eliminates the need for creating complicated hardware logic or programs and reduces the design man-hours.

Output pattern data

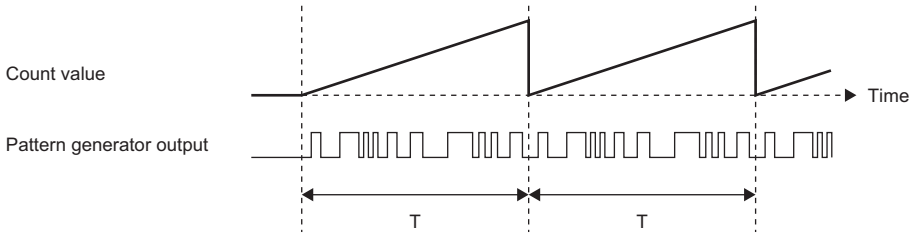
Output pattern data is a High/Low pattern table output from the external output terminals. The data consists of up to 8192 points and 10 output patterns can be created. When one of 10 output patterns is set for a pattern generator block, output from the external output terminals can be performed.
In the "Pattern generator output pattern setting" window, 10 output patterns can be created with simple mouse operation. Created output pattern data can be saved in CSV files. In addition, by reading saved CSV files to the "Pattern generator output pattern setting" window, users can utilize the output pattern data that is already created for other projects.

Operation of pattern output

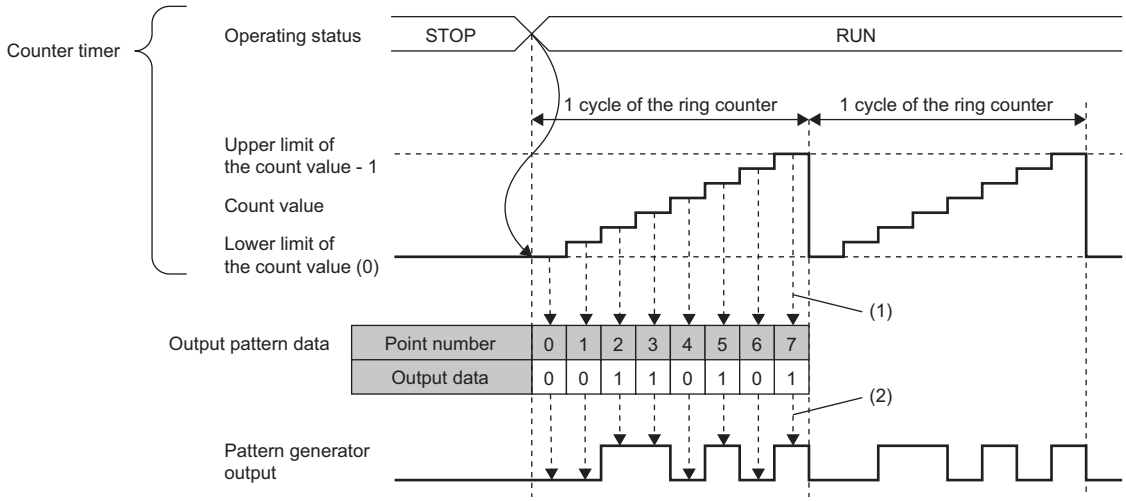
When a created output pattern is assigned to a pattern generator block, the pattern can be used. Outputs from the pattern generator block are determined depending on the count value of the counter timer block linked with the pattern generator block. The count value is a reference pointer of the output pattern data table. (When the count value is 100, the data of the point number 100 is output.)
Thus, the following outputs can be performed according to the counting method of the counter timer block.

■Counting at clock cycles set in a counter timer block

Counting every clock cycle set in a counter timer block outputs a registered output pattern at a fixed cycle (T) repeatedly.



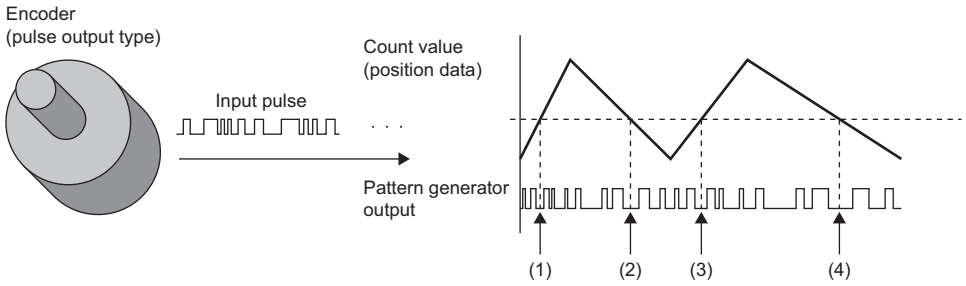
The point number of the registered output patterns changes at the fixed cycle. Thus, a same output pattern is output repeatedly at a cycle of "clock cycle × number of points".



(1) A point corresponding to the count value is referred to.
(2) Output is performed according to the setting of each point.

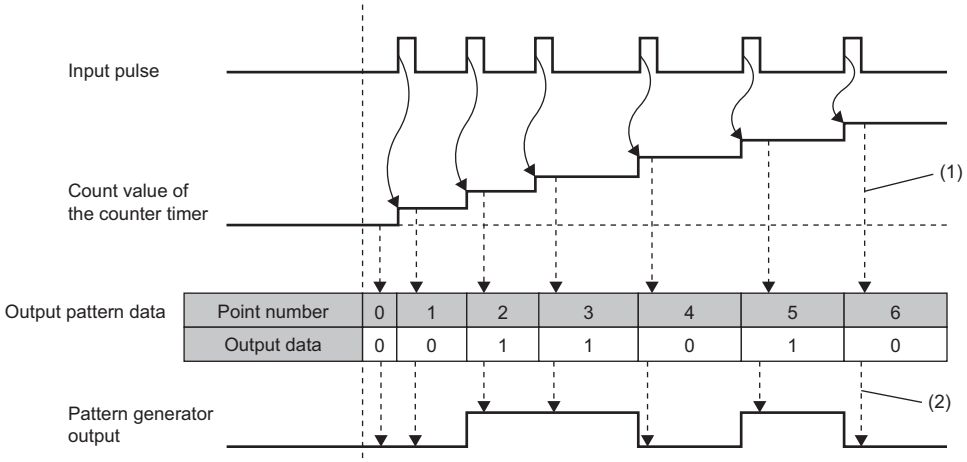
■Counting input pulses from an external input terminal

Output are performed according to the input pulses from an external input terminal. For example, output corresponding to position data can be performed.



(1) to (4) When the position data are the same, the output are also the same.

The point number of the registered output patterns changes by one point each time of a pulse input, and thus data corresponding to the number of input pulses is output.



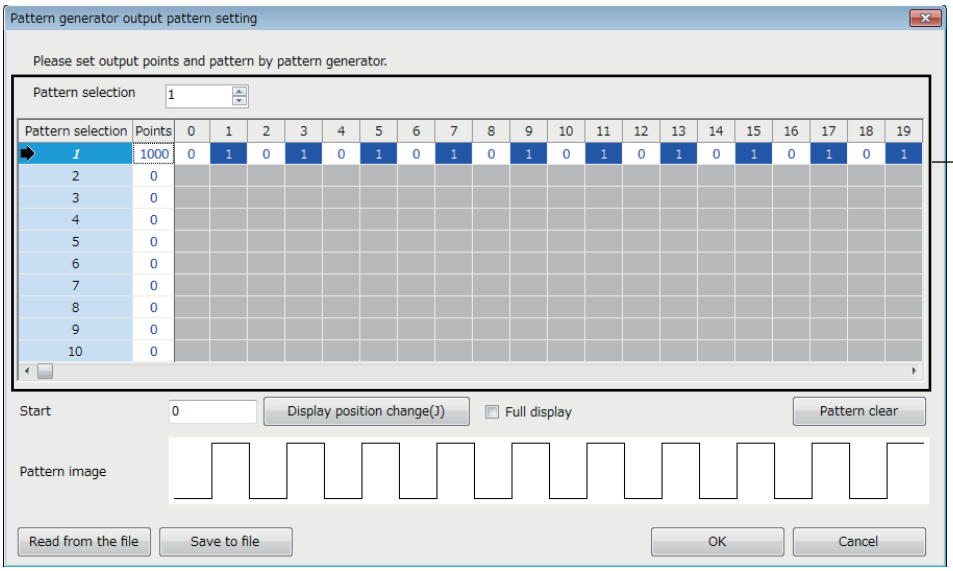
(1) A point corresponding to the count value is referred to.
(2) Output is performed according to the setting of each point.

Window layout

The following describes the configurations of the "Pattern generator output pattern setting" window. Set the output pattern data in this window.

To display the "Pattern generator output pattern setting" window, click "Detail" of the pattern generator block in the Multi function counter block detail window (Counter_☐(16bit_Unsigned)).

- Set output patterns at the upper section of the window.
- The images of set output patterns are displayed at the lower section of the window.



The following table lists the items to be displayed in the "Pattern generator output pattern setting" window.

Item		Description
Output pattern setting area	"Pattern selection"	Specify the pattern number to be assigned to a pattern generator block. Click the "OK" button to reflect the specified pattern number to "Pattern No." of the pattern generator block.
	"Points"	Set the number of points that composes an output pattern. The default value is 0, and 1 to 8192 can be set. • When 0 is set, the corresponding output pattern is disabled. • When 1 or larger number is set, the output patterns of the set number of points are enabled.
	"0" to "8191"	Set the output pattern of the pattern generator block from "0: Low output" or "1: High output" for each point number. The default value is "0: Low output". Every time the cell is clicked, the output pattern switches between "0: Low output" and "1: High output". Output patterns can be set for the number of output pattern points set in "Points".
"Start"		Set the start point to display the image of an output pattern.
"Display position" button		Change the start position of the image of an output pattern displayed in the window.
"Full display"		Switch the display range of the image of an output pattern. • Selected: Display the whole image with the range of the setting value in "Points". • Not selected: Display the image of the 20 points displayed at the upper section of the window.
"Pattern clear" button		Set "0: Low output" for all the selected output patterns.
"Pattern image"		Display the High/Low image of the selected output pattern from the start point.
"Read from the file" button		Read the output pattern data saved in a CSV file to the "Pattern generator output pattern setting" window.
"Save to File" button		Save the output pattern data in the "Pattern generator output pattern setting" window into a CSV file.
"OK" button		Set the output pattern set in the "Pattern generator output pattern setting" window for the hardware logic. The following items can be changed automatically according to the output pattern. • "Upper Limit" of "Counter_Timer_0" • "Lower Limit" of "Counter_Timer_0" To reflect the hardware logic where the output patterns have been set to the module, write the data into module.
"Cancel" button		Close the "Pattern generator output pattern setting" window.

Setting method

The following describes the setting method of the "Pattern generator output pattern setting" window.

1. Click "Detail" of a pattern generator block in the Multi function counter block detail window (Counter_☐(16bit_Unsigned)) to open the "Pattern generator output pattern setting" window.
2. Select a pattern to be output in "Pattern selection".
3. Set the number of output pattern points in "Points".
4. Click each cell of "0" to "8191" and switch the output pattern between "0: Low output" and "1: High output". By default, "0: Low output" has been set in all cells.

Point

Setting the output pattern "0: Low output" for the point number 0 is recommended. When the output pattern for the point number 0 is "1: High output", the output status is High because the count value when the hardware logic control is started is 0.

5. Save the set output pattern with the "Save to File" button (Data is saved only in the CSV format). Users can open a saved CSV file from the "Read from the file" button.
6. Click the "OK" button to set the output pattern set in the "Pattern generator output pattern setting" window for the hardware logic.

Point

- Created output pattern data can be saved in CSV files with the "Save to file" button. Thus, by opening the saved CSV files with the "Read from file" button, users can utilize the output pattern data for other projects.
- Output patterns of up to 8192 points cannot be displayed in the "Pattern generator output pattern setting" window at a time. In addition, clicking each point takes time. For convenience, edit output pattern data with a text editor and save it in the CSV format in advance. Then, read the CSV files to the "Pattern generator output pattern setting" window with the "Read from the file" button.

CSV file format of output pattern data

The following shows the CSV file format specifications of output pattern data.

Item name	Character
Delimiter	Comma (,)
Return code	CRLF (0x0D, 0x0A)

	A	B	C	D	E	F	G	H	I	J
1	100	0	1	0	1	0	1	0	1	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0

■Contents of rows

The rows correspond to the pattern numbers of output patterns. For example, the 1st row corresponds to the pattern number 1 of the output pattern.

■First column

The 1st column corresponds to the number of output pattern points. The following values can be set.

Value	Description
0	No output pattern is used. Output patterns in the 2nd column or later are disabled.
1 to 8191	The values are for setting values of the number of output pattern points. Set output patterns in the 2nd column or later. The output patterns of the points later than the setting value are disabled.

When a CSV file in which a value out of the above setting range has been set is read to the "Pattern generator output pattern setting" window, an error dialog box is displayed.

■2nd to 8193rd columns

- The columns correspond to output patterns. The point numbers 0 to 8191 are assigned to the 2nd to 8193rd columns in ascending order. "0: Low output" or "1: High output" can be set for each point number.
- The output pattern settings of the points later than the set number are disabled.
- When a CSV file in which a value other than "0: Low output" or "1: High output" has been set is read to the "Pattern generator output pattern setting" window, an error dialog box is displayed.

2.12 Help Function

As the help function, the software version of the configuration tool can be checked.

Checking the version of the configuration tool

Information including the software version of the configuration tool is displayed.

[Help] ⇨ [Version Information]

Displaying the help

Use the help to learn operation methods or understand the purposes of functions.

[Help] ⇨ [Flexible High-Speed I/O Control Module Help]
e-Manual Viewer is started and manuals are displayed.

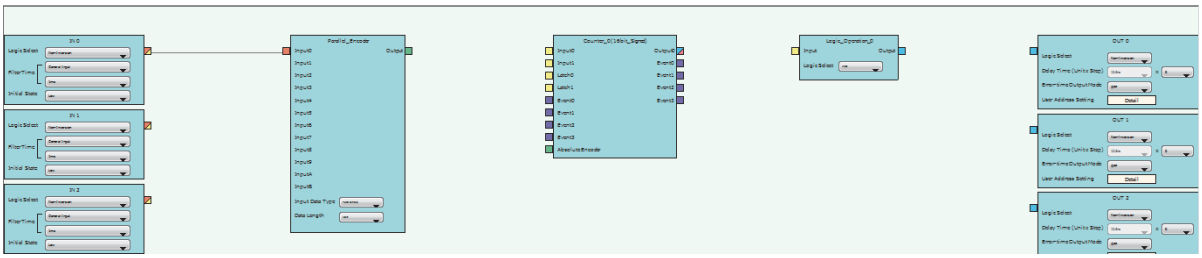
3 CREATING A HARDWARE LOGIC

This chapter describes how to create a hardware logic.

3.1 Main Blocks in the Hardware Logic Outline Window

With the flexible high-speed I/O control module, various controls are possible by arranging multi function counter blocks and various function blocks and linking these blocks.

In the Hardware logic outline window, all of external input blocks, inter-module synchronization signal input terminal, Y device terminals, and OUT terminals are arranged in the first section and all of external output blocks and SI device terminals are arranged in the fifth section when a project is started. No blocks are arranged in the second to fourth sections. Arrange blocks and link them according to the control to be executed with the flexible high-speed I/O control module.



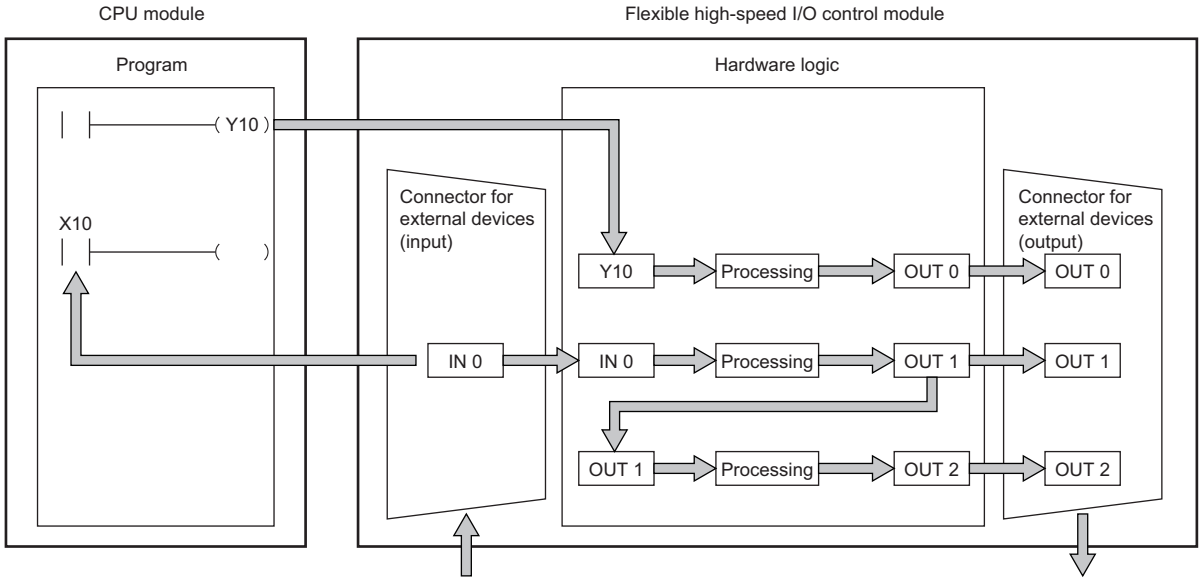
The following table shows sections where each main block can be arranged and the number of blocks that can be arranged in each section in the hardware logic outline window.

Section	Item name		Number of blocks that can be arranged*1	Remarks
First section	External input block		(12)	—
	Inter-module synchronization signal input terminal		(1)	A fixed scan clock synchronized with the inter-module synchronization cycle can be used as an input signal.
	Y device terminal		(16)	General command 0 to General command F (Y10 to Y1F) can be used as input signals.
	OUT terminal	DC output	(8)	Output signals from external output blocks in the fifth section can be used as input signals for the first section.
		Differential output	(6)	
Second section	Parallel encoder block		1	—
	SSI encoder block		2	—
Third section	Multi function counter block		8	—
Fourth section	Logical operation block		14	—
Fifth section	External output block	DC output	(8)	"OUT 0" to "OUT 7" are for DC output.
		Differential output	(6)	"OUT 0_DIF" to "OUT 5_DIF" are for differential output.
	SI device terminal		(8)	—

*1 The number in () indicates the number of blocks that are arranged by default and no additional blocks can be arranged.

Relationship of I/O in the hardware logic

This section describes the relationship between the I/O in the hardware logic and the I/O of X/Y devices and of the connectors for external devices.



Input terminal		Output terminal	
Y 10	Y device terminal	OUT 0	External output block
IN 0	External input block	OUT 1	External output block
OUT 1	OUT terminal	OUT 2	External output block

■Signal status name

In the flexible high-speed I/O control module, each signal status is called as shown below.

Signal status	Name	
Status of input signals from external devices	ON	OFF
Status of internal signals (terminals) of the hardware logic	High	Low
Status of DC output signals to external devices	ON	OFF
Status of differential output signals to external devices	High	Low

Link combination

The terminals to which the terminals of each main block can be linked are predetermined. An input terminal and an output terminal in the same color can be linked. Input terminals or output terminals cannot be linked each other.
The following table lists the combinations.

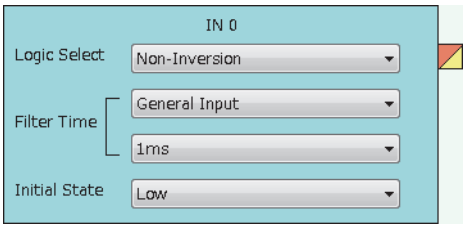
Output side		Input side		Color
Block name	**Terminal name**	**Block name**	**Terminal name**	
External input block	Output	Parallel encoder block	Input	Coral
		Multi function counter block	Input	Butterfly yellow
			Latch	
		Logical operation block	Input	
		SSI encoder block	Data	Coral
Inter-module synchronization signal input terminal	SYNC	Multi function counter block	Input	Butterfly yellow
			Latch	
		Logical operation block	Input	
Y device terminal	Y	Multi function counter block	Input	Butterfly yellow
			Latch	
		Logical operation block	Input	
OUT terminal	OUT OUT_DIF	Multi function counter block	Input	Butterfly yellow
			Latch	
		Logical operation block	Input	
Parallel encoder block	Output	Multi function counter block	Absolute Encoder	Celadon
SSI encoder block	Clock	External output block	OUT_DIF	Vermeer
	Output	Multi function counter block	Absolute Encoder	Celadon
Multi function counter block	Output	External output block	OUT	Vermeer
			OUT_DIF	
		SI device terminal	SI	Empire rose
	Event	Multi function counter block	Event	Columbine blue
	P.G. Output	External output block	OUT	Vermeer
			OUT_DIF	
	Cam Output	External output block	OUT	Vermeer
			OUT_DIF	
Logical operation block	Output	External output block	OUT	Vermeer
			OUT_DIF	

If multiple output terminals are linked to one input terminal, OR processing is executed on all input signals. If one output terminal is linked to multiple input terminals, the same signals are output for all the input terminals.

External input block

Outline

In the first section in the hardware logic outline window, 12 external input blocks ("IN 0" to "IN B") are arranged by default. Select which signal (among "IN 0" to "IN B") input from the connector for external devices is to be used as an input signal in the hardware logic. The selected input signal is output from the output terminal of the external input block through the filter. The output signal can be input to the input terminal of another block.



The external input block has the following functions.

- When "Logic Select" is set to "Inversion", inverted signals are output.
- The digital filter reduces the effect of external noise.
- The signal selected as an initial state is output when hardware logic control stops.

Parameter

The following table shows the parameters of the external input block.

Variable name	Data type	Valid range	Default value	Description
Logic Select	Bit	Non-Inversion Inversion	Non-Inversion	Set Non-Inversion or Inversion for input signals.
Filter Time	Word	☞ Page 148 "Filter Time"	• General Input (Pulse input mode) • 1ms (Input response time/ counting speed)	Select a type of the filter time. The filter time corresponding to the input response time of general-purpose input or the counting speed of each pulse input mode can be set.
Initial State	Bit	Low High	Low	Set the signal status during a hardware logic control stop to Low fixed or High fixed.

■"Filter Time"

General-purpose input or pulse input count can be selected for "Filter Time".

• General-purpose input (General Input 0μs to General Input 5ms)

Set this method to reduce noise. When a pulse width is less than the setting value of "Filter Time", the pulse is not detected as an input signal. If the setting value is too large, pulses other than noise cannot be detected. Thus, set the filter time suitable for the operating environment.

Input response time								
0μs	10μs	50μs	0.1ms	0.2ms	0.4ms	0.6ms	1ms (default value)	5ms

• Pulse input count (1-Phase Multiple of 1 (CW/CCW) 10kpps to 2-Phase Multiple of 4 8000kpps)

Set this method to count input pulses. A setting according to the pulse input mode and counting speed prevents incorrect count.

Pulse input mode	Counting speed							
1-Phase Multiple of 1 (CW/CCW)	10kpps (default value)	100kpps	200kpps	500kpps	1000kpps	2000kpps	—	—
1-Phase Multiple of 2	10kpps (default value)	100kpps	200kpps	500kpps	1000kpps	2000kpps	4000kpps	—
2-Phase Multiple of 1	10kpps (default value)	100kpps	200kpps	500kpps	1000kpps	2000kpps	—	—
2-Phase Multiple of 2	10kpps (default value)	100kpps	200kpps	500kpps	1000kpps	2000kpps	4000kpps	—
2-Phase Multiple of 4	10kpps (default value)	100kpps	200kpps	500kpps	1000kpps	2000kpps	4000kpps	8000kpps

■Setting method of "Filter Time"

1. Select a pulse input mode in the upper list box.

2. Select an item in the lower list box.

• When "General Input" has been selected in the upper list box, select an input response time.

• When an item other than "General Input" has been selected in the upper list box, select a counting speed.

If the item in the upper list box is changed to another one after an item is selected in the lower list box, the selection status of the lower list box returns to the default value.

■Minimum value of the pulse width that may be taken in as input

For general-purpose input, noise or others may be taken in as input depending on the filter time setting. The following table lists the minimum values of the pulse width that may be taken in as input for each filter time.

Filter time	Minimum value of the pulse width that may be taken in as input
10μs	6μs
50μs	32μs
0.1ms	0.06ms
0.2ms	0.12ms
0.4ms	0.24ms
0.6ms	0.36ms
1ms	0.6ms
5ms	3ms

■Link with SSI encoder blocks

When an external input block is linked with the "Data" terminal of an SSI encoder block, the "Filter Time" of the external input block is automatically changed according to the setting value of "Transmission Speed" of the SSI encoder block. Additionally, if the setting value of "Transmission Speed" of the SSI encoder block after linked is changed, the "Filter Time" of the external input block is automatically changed according to the new value.

When the link between the external input block and the "Data" terminal of the SSI encoder block is deleted, the "Filter Time" of the external input block returns to the default value.

■"Initial State"

When ON signals are constantly input from external devices during a hardware logic control stop, a rise is detected in an input signal event detection block at a hardware logic control start. To prevent rise detection at a hardware logic control start, set "Initial State" to "High".

Output

The following table shows the output of the external input block.

Variable name	Data type	Linkable block	Output value	Description
Output	Bit	• Parallel encoder • Multi function counter • Logical operation • SSI encoder	0, 1	Outputs the signals that have passed the filter. Selecting "Logic Select" outputs inverted signals.

Inter-module synchronization signal input terminal

Outline

In the first section in the hardware logic outline window, one inter-module synchronization signal input terminal ("SYNC") is arranged by default.

In the system using the inter-module synchronization function, a fixed scan clock synchronized with the inter-module synchronization cycle is input to the flexible high-speed I/O control module. This fixed scan clock can be used as an input signal of the hardware logic. When the inter-module synchronization function is not used, this fixed scan clock stops. Thus, input from the inter-module synchronization signal input terminal is always off.

Output

The following table shows the output of the inter-module synchronization signal input terminal.

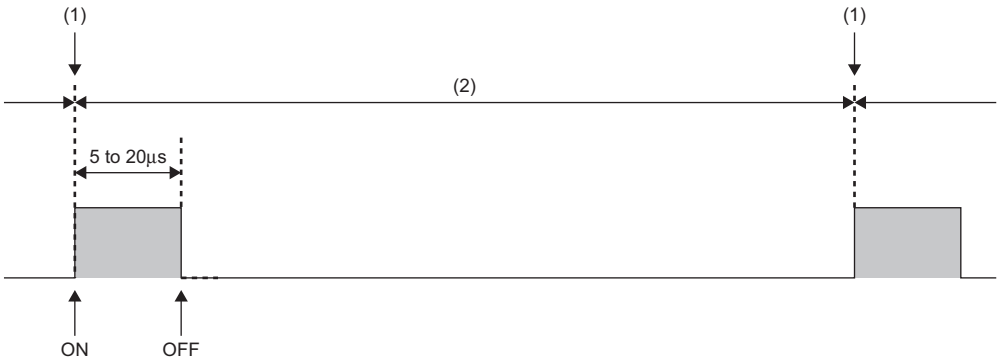
Variable name	Data type	Linkable block	Output value	Description
Input	Bit	• Multi function counter • Logical operation	0, 1	Outputs the inter-module synchronization signal as an input signal of the hardware logic.

■Input signal from the inter-module synchronization signal input terminal

The input signal from the inter-module synchronization signal input terminal turns on immediately after the flexible high-speed I/O control module receives an inter-module synchronization command from the inter-module synchronous master. Thus, the input signal is used as a trigger signal to start an inter-module synchronization cycle.

In addition, when the input signal is used together with the multi function counter block for which a one-shot timer (☞ Page 214 One-shot Timer) has been set, a clock with a specified duty ratio (example: 50%:50%) starting from the timing to start the inter-module synchronization cycle can be created.

The following shows the timing of the signals input from the inter-module synchronization signal input terminal to the hardware logic.



(1) Timing to start the inter-module synchronization cycle

(2) Inter-module synchronization cycle

ON: Indicates the ON timing. The input signal turns on immediately after the flexible high-speed I/O control module receives an inter-module synchronization command from the inter-module synchronous master.*1

OFF: Indicates the OFF timing. The input signal turns off 5 to 20µs after the flexible high-speed I/O control module receives an inter-module synchronization command.

*1 For the accuracy of the ON timing, refer to the section of the inter-module synchronization accuracy in the following manual.
MELSEC iQ-R Inter-Module Synchronization Function Reference Manual

Y device terminal

Outline

In the first section in the hardware logic outline window, 16 Y device terminals ("Y 10" to "Y 1F") are arranged by default. The ON/OFF states of 'General command 0 to General command F' (Y10 to Y1F) are output as signals and used as inputs in the hardware logic. Select ON or OFF of 'General command 0 to General command F' (Y10 to Y1F) with a program.

Output

The following table shows the output of the Y device terminal.

Variable name	Data type	Linkable block	Output value	Description
Input	Bit	• Multi function counter • Logical operation*1	0, 1	Outputs the ON/OFF states of 'General command 0 to General command F' (Y10 to Y1F) as signals.

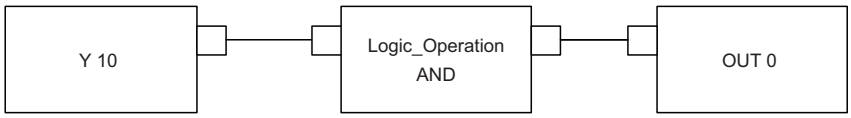
*1 A logical operation block can be linked only when an OUT terminal is not linked to the logical operation block.

Restriction

The timing of when signals output from Y device terminals are input to the link destination input terminal may delay due to the scan time or other causes. To input the signals to multiple connection destinations at the same time, using an external input block to which an external input signal is assigned is recommended.
For details on the external input block, refer to the following.
☞ Page 147 External input block

Point

To use the flexible high-speed I/O control module as an output module, link blocks in the Hardware logic outline window as shown below.



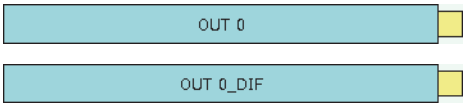
OUT terminal

Outline

In the first section in the hardware logic outline window, eight OUT terminals for DC output ("OUT 0" to "OUT 7") and six OUT terminals for differential output ("OUT 0_DIF" to "OUT 5_DIF") are arranged by default.

Signals output from external output blocks in the fifth section can be used as inputs in the hardware logic. For details on the external output block, refer to the following.

☞ Page 165 External output block



Output

The following table shows the output of the OUT terminal.

Variable name	Data type	Linkable block	Valid range	Description
Input	Bit	• Multi function counter • Logical operation*1	0, 1	When no external output blocks are linked, no signals are output from this terminal.

*1 Any of OUT 0 to OUT 7 or OUT 0_DIF to OUT 5_DIF can be linked to a logical operation block when a Y device terminal is not linked to the logical operation block.

Restriction

When "Logic Select" of an external output block is set to "Inversion", signals input to the external output block are inverted and output from the OUT terminal.

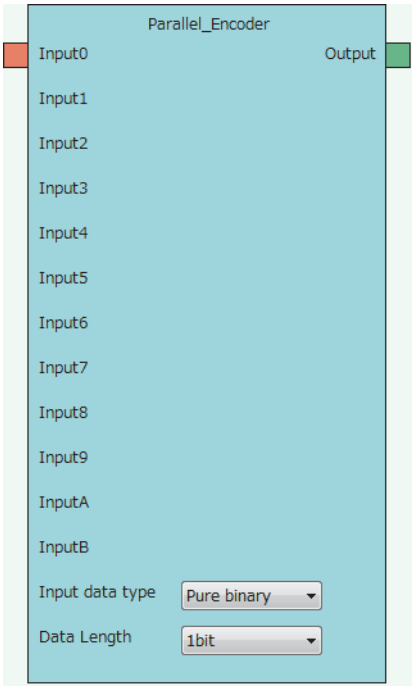
For details on the external output block, refer to the following.

☞ Page 165 External output block

Parallel encoder block

Outline

In the second section in the hardware logic outline window, one parallel encoder block ("Parallel_Encoder") can be arranged. The parallel encoder block transmits data with the encoder for parallel transmission and receives the data of the bit points set in the data length setting as the data of Input Data Type to convert this data into word data.



Input

The following table shows the inputs of the parallel encoder block.

Variable name	Data type	Linkable block	Valid range	Description
Input 0	Bit	External input	0, 1	These are input terminals of the parallel encoder. When the data length is changed, the number of displayed Input terminals is changed according to the data length. Then, the Input terminals are automatically linked to external input blocks in order from the first one ("IN 0"). The order of linking the Input terminals and the external input blocks cannot be changed.
Input 1				
Input 2				
Input 3				
Input 4				
Input 5				
Input 6				
Input 7				
Input 8				
Input 9				
Input A				
Input B				

Parameter

The following table shows the parameters of the parallel encoder block.

Variable name	Data type	Valid range	Default value	Description
Input Data Type	Word	Pure binary Gray code BCD	Pure binary	Set the input data type of the parallel interface. • Pure binary • Gray code • BCD
Data Length	Word	1 bit to 12 bit	1bit	Set the input data length for the parallel encoder. When the trigger for transfer timing is used for BCD or pure binary, only input data of up to 11 bits can be used.

■Link for using the parallel encoder block

The input from an encoder is stored as a preset value of the counter timer block. Because the preset value is not automatically applied to "Count Value", performing the preset function is required. Thus, when using the parallel encoder block, always link it to the "PRESET" terminal of the counter timer block.

For details on the counter timer block, refer to the following.

☞ Page 186 Counter timer block

Restriction

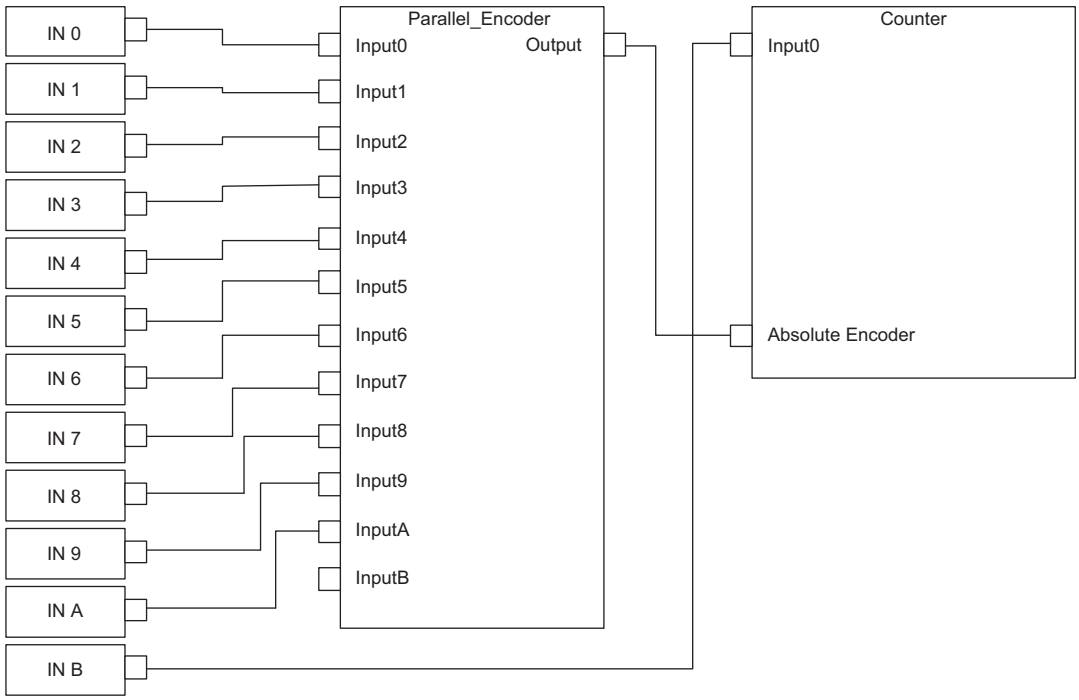
- There are the following types of multi function counter block which is linked from a parallel encoder block: 16 bits and 32 bits. When the parallel encoder block is linked to the "Absolute Encoder" terminal of a 16-bit multi function counter block, "Preset Value" of "Counter_Timer_0", out of the two counter timer blocks in the multi function counter block detail window, is updated. Make sure to link the parallel encoder block to the "PRESET" terminal of "Counter_Timer_0". (☞ Page 169 Multi Function Counter Block)
- When the parallel encoder block is linked to the "Absolute Encoder" terminal of a 32-bit multi function counter block, do not change the preset value of "Counter_Timer_0" in the multi function counter block detail window from 0 (default value). In addition, do not set "User Address" of the preset value of "Counter_Timer_0". When these settings are configured, the input from an encoder is not normally stored as a preset value.

■Setting and link of the encoder for which Input Data Type is set to Pure binary or BCD

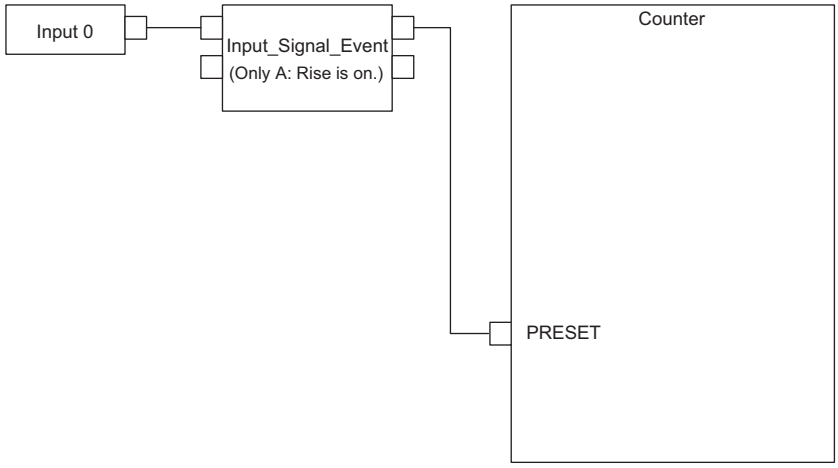
When data is input from the encoder for which Input Data Type is set to Pure binary or BCD, an error of 2 bits or more is generated depending on the timing of reading data. Because of this, when data is input from the encoder for which Input Data Type is set to Pure binary or BCD, specifying the read timing using a trigger signal for transfer timing is recommended.
To specify the read timing, follow the instructions below.

- Set "Data Length" to 11 bits or less.
- Link the terminal outputting the transfer timing signal to the "PRESET" terminal of the counter timer block.

■Link diagram of the hardware logic outline window



■Link diagram of the multi function counter block detail window



Output

The following table shows the output of the parallel encoder block.

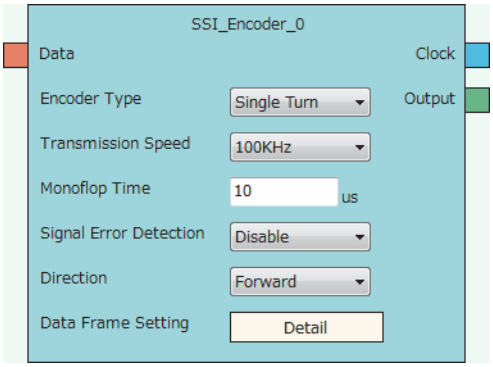
Variable name	Data type	Linkable block	Output value	Description
Output	Word	Multi function counter	0 to 4095	Inputs the value from the parallel encoder to the "Absolute Encoder" terminal of the multi function counter to use the value as a preset value of the multi function counter.

SSI encoder block

Outline

In the second section in the hardware logic outline window, two SSI encoder blocks ("SSI_Encoder_☐") can be arranged. The SSI encoder blocks perform serial communication with the absolute encoder having the SSI (Synchronous Serial Interface) output and acquire position data. The acquired position data is stored as a count value of the multi function counter block linked to this block.

- When "Signal Error Detection" is set to "Enable", a signal error detection processing is performed. If an error occurs on the external wiring to the SSI encoder, the SSI encoder block ☐ DATA signal wire reverse error (error code: 109☐H) or the SSI encoder block ☐ DATA signal error (error code: 10A☐H) is output.
- When "Parity" of "Data Frame Setting" is set to "Even" or "Odd", a parity check is performed. If an error is detected by the parity check, the SSI encoder block ☐ parity error (error code: 10B☐H) is output.
- If any of the above errors (error code: 109☐H to 10B☐H) is output during data receipt from the absolute encoder, the count value of the multi function counter block linked to this block does not change. The position data is reflected to the count value when data has been received properly at next transmission.



Input

The following table shows the input of the SSI encoder block.

Variable name	Data type	Linkable block	Valid range	Description
Data	Bit	External input	0, 1	Inputs data from the SSI encoder.

Parameter

The following table shows the parameters of the SSI encoder block.

Variable name	Data type	Valid range	Default value	Description
Encoder Type	Word	Single Turn Multi Turn	Single Turn	Set the type of the SSI encoder. • Single Turn • Multi Turn
Transmission Speed	Word	100kHz 200kHz 300kHz 400kHz 500kHz 1.0MHz 1.5MHz 2.0MHz	100kHz	Set the frequency of the clock signal to perform synchronous data communication with the SSI encoder.
Monoflop Time	Word	10 to 10000 (μs)	10 (μs)	Indicates the time until the next communication is ready to start after the output of the last clock signal. Set the monoflop time according to the specifications of the SSI encoder to be connected. For details on the monoflop time, refer to the following. ☞ Page 162 Monoflop time
Signal Error Detection	Word	Disable Enable	Disable	Set whether to detect an error (disconnection, short-circuit) of the data signal wire externally connected to the SSI encoder. • Error detection disabled: Disable • Error detection enabled: Enable For details on the signal error detection, refer to the following. ☞ Page 163 Signal error detection
Direction	Word	Forward Reverse	Forward	The counting direction of received data from the SSI encoder can be changed by setting this item. • Forward • Reverse For details on Direction, refer to the following. ☞ Page 162 Counting direction
Data Frame Setting	—	—	—	Click the [Detail] button to open the "Data Frame Setting" window in which the details of a data frame can be set.

• Parameters of "Data Frame Setting"

Variable name	Data type	Valid range	Default value	Description
Input Data Type	Word	Pure binary Gray code	Pure binary	Set the type of input data from the SSI encoder. • Pure binary • Gray code[*3]
Data Frame Length	Word	1 to 32 (bit)	1 (bit)	Specify the effective bit length of a received data frame from the SSI encoder.[*1]
Multi Turn Data Length	Word	0 to 32 (bit)	0 (bit)	Specify the effective bit length of multi turn data.[*1] When "Encoder Type" is set to "Multi Turn", this setting is enabled. When this setting is disabled, a default value is set.
Multi Turn Start Bit	Word	0 to 31 (bit)	0 (bit)	Specify the start bit position of multi turn data. When "Encoder Type" is set to "Multi Turn", this setting is enabled. When this setting is disabled, a default value is set.
Encoder Resolution	Word	0 to 4294967295	0	Set the encoder resolution to use the encoder with its resolution of single turn data not equal to 2^n (n: single turn data length).[*2] When the resolution is equal to 2^n, the setting change from the default value (0) is not required. (When 0 is set, the resolution is regarded as equal to 2^n.) When "Encoder Type" is set to "Multi Turn" and "Input Data Type" is set to "Gray code", this setting is disabled. When this setting is disabled, a default value is set.
Single Turn Data Length	Word	0 to 32 (bit)	0 (bit)	Specify the effective bit length of single turn data.[*1]
Single Turn Start Bit	Word	0 to 31 (bit)	0 (bit)	Specify the start bit position of single turn data. When "Encoder Type" is set to "Single Turn", this setting is enabled. When this setting is disabled, a default value is set.
Parity	Word	None Even Odd	None	Set whether to perform the parity check or the type of the parity check (even or odd) according to the SSI encoder to be connected. • No parity check: None • Even parity: Even • Odd parity: Odd

*1 Set values so that the total of set values for "Multi Turn Data Length" and "Single Turn Data Length" is equal to or smaller than the value set for "Data Frame Length". Note that the parity bit length is not included in these values.

*2 A value larger than the maximum value representable by "Single Turn Data Length" cannot be set for "Encoder Resolution".

*3 When using the encoder with the encoder type being multi turn and the input data type being gray code, make sure that its resolution of single turn data is equal to 2^n.

Output

The following table shows the outputs of the SSI encoder block.

Variable name	Data type	Linkable block	Output value	Description
Clock	Bit	External output	0, 1	Outputs the clock signal to perform synchronous data communication with the SSI encoder. When an SSI encoder block is arranged in the hardware logic outline window, the "Clock" terminal of the SSI encoder block and the "Input" terminal of an external output block are automatically linked. The link cannot be deleted or changed. The following are the link destinations. • SSI_Encoder_0: OUT 0_DIF • SSI_Encoder_1: OUT 1_DIF
Output	Word	Multi function counter*1	0 to 4294967295	Sets the position data acquired from the SSI encoder to a count value of the multi function counter block linked.

*1 The "Absolute Encoder" terminal of a 32-bit unsigned multi function counter block can be linked.

Restriction

The refreshing cycle of a count value is calculated by the following formula because the module processing time fluctuates within the range of 0 to 100μs.

Refreshing cycle = (Data frame length + P + 1) × Clock cycle + Monoflop time + Module processing time (0 to 100μs)

- P: 1 (for with parity) or 0 (for without parity)
- Clock cycle: Inverse of the transmission speed (for the transmission speed of 100kHz: 1/100000s = 10μs)

Note that if an SSI encoder block is used together with other functions, the following delay times are added to the refreshing cycle above.

- When an SI device terminal is used: Up to 200μs
- When the flexible high-speed I/O control module is in the inter-module synchronous mode and "User Address" of the input terminals or parameters is assigned to Hardware logic area (High speed area) (Un\G1000 to Un\G1029): Up to 100μs

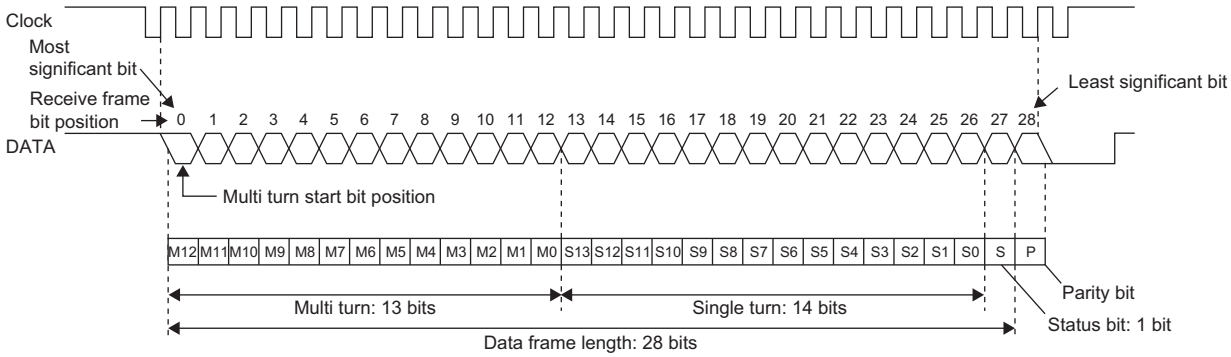
Setting examples of the SSI encoder block

This section shows setting examples of the SSI encoder block suitable for a receive frame from the SSI encoder and communication specifications.

■Multi turn

The following example is for the receive frame of a multi turn encoder. The receive frame consists of the elements below.

- Multi turn: 13 bits
- Single turn: 14 bits
- Status bit: 1 bit
- Parity bit: 1 bit



- Parameter

Encoder specifications[*1]		SSI encoder block setting		Remarks
		Item	Setting value	
Encoder type	Multi turn	Encoder Type	Multi Turn	—
Transmission speed	1MHz	Transmission Speed	1.0MHz	—
Monoflop time	16μs	Monoflop Time	16	—
—	—	Signal Error Detection	Enable	Set "Enable" to use the signal error detection. Set "Disable" not to use it.
—	—	Direction	Forward	Set "Forward" to count the position data from the SSI encoder in the forward direction. Set "Reverse" to reverse the counting direction.

*1 For details on the encoder specifications, refer to the manual for the encoder used.

- Parameters of "Data Frame Setting"

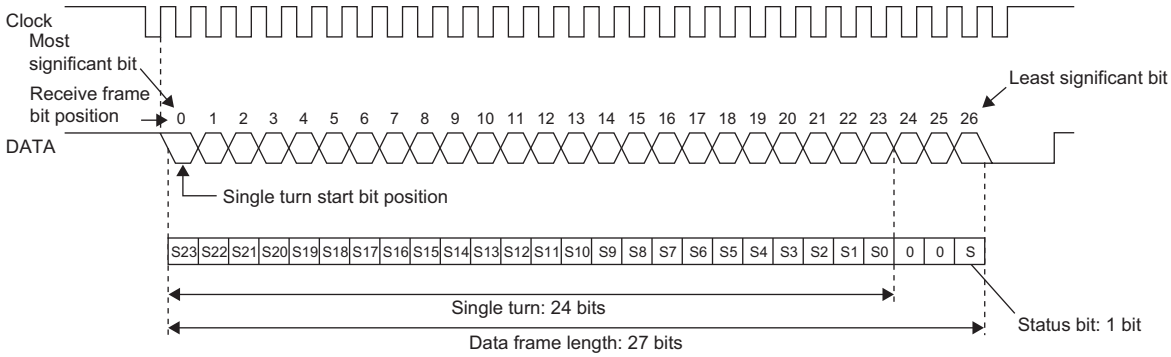
Encoder specifications[*2]		SSI encoder block setting		Remarks
		Item	Setting value	
Data type	Gray	Input Data Type	Gray code	—
Data frame length	28 bits	Data frame length	28	The parity bit is not included.
Multi turn data length	13 bits	Multi Turn Data Length	13	—
Multi turn data start bit position	0	Multi Turn Start Bit	0	Specify the receive frame bit position where multi turn data starts.
Encoder resolution	16384	Encoder Resolution	0	Changing the setting value from its default (0) is not required because the single turn data length is 14 bits and the encoder resolution is 16384 (= 2^{14}).
Single turn data length	14 bits	Single Turn Data Length	14	—
Single turn data start bit position	13	Single Turn Start Bit	0	Setting is not required.
Parity check	Odd parity	Parity	Odd	—

*2 For details on the encoder specifications, refer to the manual for the encoder used.

■Single turn

The following example is for the receive frame of a single turn encoder. The receive frame consists of the elements below.

- Single turn: 24 bits
- Status bit: 1 bit
- Parity bit: None



- Parameter

Encoder specifications*1		SSI encoder block setting		Remarks
		Item	Setting value	
Encoder type	Single turn	Encoder Type	Single turn	—
Transmission speed	2MHz	Transmission Speed	2.0MHz	—
Monoflop time	10μs	Monoflop Time	10	—
—	—	Signal Error Detection	Enable	Set "Enable" to use the signal error detection. Set "Disable" not to use it.
—	—	Direction	Reverse	Set "Forward" to count the position data from the SSI encoder in the forward direction. Set "Reverse" to reverse the counting direction.

*1 For details on the encoder specifications, refer to the manual for the encoder used.

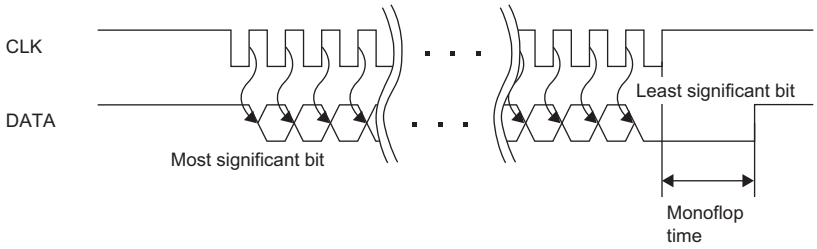
- Parameters of "Data Frame Setting"

Encoder specifications*2		SSI encoder block setting		Remarks
		Item	Setting value	
Data type	Pure binary	Input Data Type	Pure binary	—
Data frame length	27 bits	Data Frame Length	27	—
Multi turn data length	—	Multi Turn Data Length	0	Setting is not required.
Multi turn data start bit position	—	Multi Turn Start Bit	0	Setting is not required.
Encoder resolution	16777216	Encoder Resolution	0	Changing the setting value from its default value (0) is not required because the single turn data length is 24 bits and the encoder resolution is 16777216 (= 2^{24}).
Single turn data length	24 bits	Single Turn Data Length	24	—
Single turn data start bit position	0	Single Turn Start Bit	0	Specify the receive frame bit position where single turn data starts.
Parity check	—	Parity	None	—

*2 For details on the encoder specifications, refer to the manual for the encoder used.

Monoflop time

The monoflop time indicates the time set aside to refresh position data of the SSI encoder. The time is stipulated by an encoder type. If the monoflop time is set to be shorter than the stipulated time, correct position data cannot be received.
When CLK transmission from the connected device is stopped by the monoflop time, the DATA signal from the SSI encoder returns to High. This state indicates that the SSI encoder waits to start communication. At the timing the first fall of CLK from this state is detected, position data is refreshed to the latest value. And then, at a rise of CLK, data transmission resumes in order from the most significant bit.
With SSI communication specifications, after the receipt of the last bit of the receive frame, if CLK transmission resumes before the encoder-wait-to-start-communication state is established, the position data of the previous receive frame is sent from the encoder and received in the flexible high-speed I/O control module. In other words, if CLK transmission resumes before the monoflop time elapsed, the latest position data is not latched in the encoder, and thus the latest position data cannot be received in the flexible high-speed I/O control module.

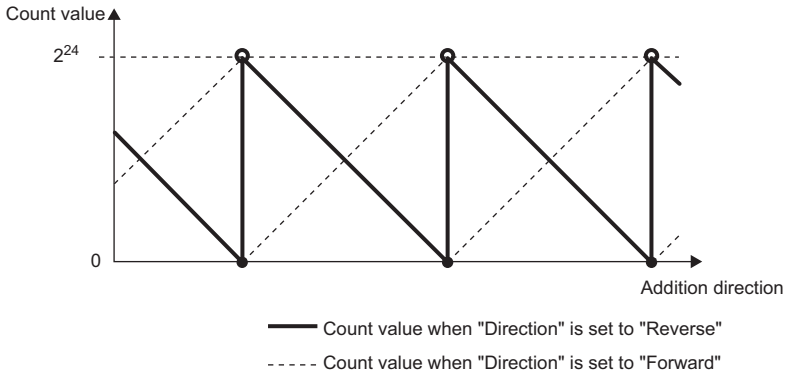


Counting direction

Set the counting direction of received position data from the SSI encoder.
When "Direction" is set to "Reverse", the position data from the encoder is processed using the calculation formulas in the table below.

Encoder type	Calculation formula
Multi turn	Count value = (2^m × Encoder resolution - 1) - Position data (m: Multi turn data length)
Single turn	Count value = (Encoder resolution - 1) - Position data

Ex.
The following is the example of using the encoder with multi turn data length of 12 bits and encoder resolution of 4096 (= 12 bits) and counting in the addition direction.
Count value = (2^{12} × 4096 - 1) - Position data = (2^{24} - 1) - Position data



Receive data monitor

Out of the data frame received from the SSI encoder, the information for the number of bits specified with "Data Frame Length" is stored in the following buffer memory areas. (The parity bit is not included.)

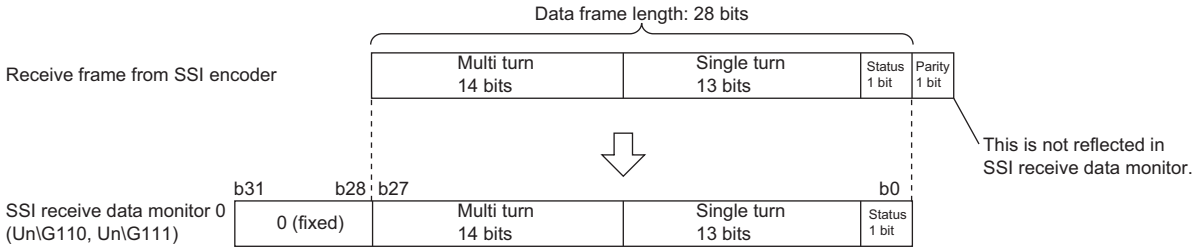
Buffer memory address	SSI encoder block
Un\G110, Un\G111	SSI_Encoder_0
Un\G114, Un\G115	SSI_Encoder_1

Some SSI encoders output not only position data (multi turn data, single turn data) but also status data in the data frame. The status of encoder can be checked by reading out values of above buffer memory areas with a program.
In addition, when a count value is not refreshed properly, its cause (whether it is a receive data error or a parameter setting error) can be identified by checking multi turn data and single turn data contained in the receive data monitor.

Ex.

The example shows the receive data monitor of when the SSI encoder with the data frame of the following configuration is used.

- Data frame length: 28 bits
- Multi turn data: 14 bits
- Single turn data: 13 bits
- Status: 1 bit
- Parity bit: 1 bit



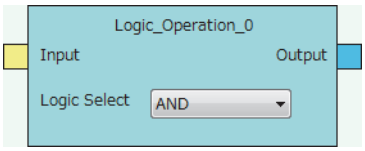
Signal error detection

An error of the external wiring to the SSI encoder can be detected.
When "Signal Error Detection" is set to "Enable", a signal error detection processing is performed. If an error occurs on the external wiring to the SSI encoder, the SSI encoder block ☐ DATA signal wire reverse error (error code: 109☐H) or the SSI encoder block ☐ DATA signal error (error code: 10A☐H) is output.
In addition, in the cycle where data is received from the SSI encoder with an error detected, the count value of the multi function counter block linked to the SSI encoder block is not refreshed.

Logical operation block

Outline

In the fourth section in the hardware logic outline window, 14 logical operation blocks ("Logic_Operation_0" to "Logic_Operation_D") can be arranged.
Any logical operation of AND operation, OR operation, or XOR operation is executed for all signals input to the "Input" terminal.



Restriction

Even though a project including an unlinked logical operation block for which Logic Select is set to AND (default) is saved, the arrangement status is not saved. When the saved project is opened, the logical operation block is not displayed. In addition, verifying with the module is executed on the assumption that the block is not arranged.

Input

The following table shows the input of the logical operation block.

Variable name	Data type	Linkable block	Valid range	Description
Input	Bit	• External input • Inter-module synchronization signal input • OUT*1*2 • Y device*2	0, 1	Multiple signals can be input to the "Input" terminal. Operations are executed for all input signals.

*1 When a logical operation block is linked with an OUT terminal, any of OUT 0 to OUT 7 or OUT 0_DIF to OUT 5_DIF can be used.
*2 Either of an OUT terminal or Y device terminal can be linked with a logical operation block.

Parameter

The following table shows the parameter of the logical operation block.

Variable name	Data type	Valid range	Default value	Description
Logic Select	Word	AND OR XOR	AND	Set the operation type for input signals.

Output

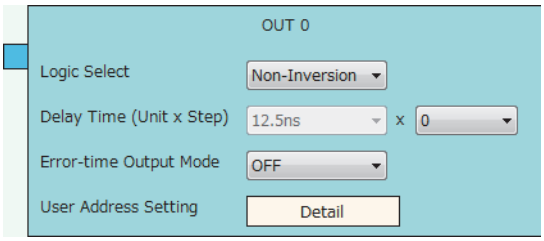
The following table shows the output of the logical operation block.

Variable name	Data type	Linkable block	Output value	Description
Output	Bit	External output	0, 1	Outputs the operation results of input signals.

External output block

Outline

In the fifth section in the hardware logic outline window, eight external output blocks for DC output ("OUT 0" to "OUT 7") and six external output blocks for differential output ("OUT 0_DIF" to "OUT 5_DIF") are arranged by default.
The result of the operation from the first to the fourth section in the hardware logic is output from the connector for external devices (OUT 0 to OUT 7, OUT 0_DIF to OUT 5_DIF).



The external output block has the following functions.

- When "Logic Select" is set to "Inversion", inverted signals are output.
- A delay between signals caused by different external interface circuits can be adjusted by setting a delay time.
- When an error occurs in the CPU module, signals are output according to the output setting specified with "Error-time Output Mode".

Input

The following shows the input of the external output block.

Variable name	Data type	Linkable block	Description
Input	Bit	• Multi function counter • Logical operation	Executes OR processing for all signals input to the "Input" terminal.
	Bit	SSI encoder	Only OUT 0_DIF and OUT 1_DIF are automatically linked with "Clock" of the SSI encoder block.

Parameter

The following table shows the parameters of the external output block.

Variable name	Data type	Valid range	Default value	Description
Logic Select	Bit	Non-Inversion Inversion	Non-Inversion	Set Non-Inversion or Inversion for input signals.
Delay Time (Step)	Word	0 to 64	0	Adjust the signal output timing. A delay time can be set with the combination of "Delay Time (Step)" and "Delay Time (Unit)". Setting the number of steps to 0 disables the delay. When "SYNC" is specified for "Delay Time (Unit)", the number of steps is 1.
Delay Time (Unit)	Word	12.5ns 25ns 50ns 0.1μs 1μs 10μs 100μs 1ms SYNC	12.5ns	Adjust the signal output timing. A delay time can be set with the combination of "Delay Time (Unit)" and "Delay Time (Step)". When "SYNC" is specified, the signal output timing is synchronized with the inter-module synchronization cycle and the signal output is held until the next inter-module synchronization cycle.*3
Error-time Output Mode	Word	OFF ON HOLD	OFF	Set output signals at occurrence of an error in the CPU module to OFF fixed, ON fixed, or holding output status. This setting is also applied to output signals at a hardware logic control stop.
User Address Setting	—	—	—	Set the buffer memory addresses used in the external output block. Click the [Detail] button to open the "User Address Setting" window.

• Parameters of "User Address Setting"

Variable name	Data type	Valid range	Description
Enable Forced Output	Word	1000 to 1099	Specify the buffer memory address to enable or disable the forced output.*1 Whether the forced output is enabled or disabled can be set by setting the following values for the specified buffer memory.*2 0: Forced output disabled 1: Forced output enabled
Forced Output	Word	1000 to 1099	Specify the buffer memory address to set the status of the forced output signal.*1 Whether the signal status is set to off or on can be set by setting the following values for the specified buffer memory. 0: OFF output 1: ON output
External terminal monitor	Word	1000 to 1099	Specify the buffer memory address to store the external terminal status.*1 0: OFF output 1: ON output

*1 The available buffer memory addresses are common in OUT 0 to OUT 7 and OUT 0_DIF to OUT 5_DIF. Each external output block is assigned to the bits of the specified buffer memory as follows.

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
00		OUT 5_DIF	OUT 4_DIF	OUT 3_DIF	OUT 2_DIF	OUT 1_DIF	OUT 0_DIF	OUT 7	OUT 6	OUT 5	OUT 4	OUT 3	OUT 2	OUT 1	OUT 0

*2 The external output block where Forced output enabled (1) is set outputs a signal according to the setting value (forced output value) of the buffer memory area specified with "Forced Output". The input from the "Input" terminal is not reflected to the external output.

*3 When "SYNC" is specified for "Delay Time (Step)" while the flexible high-speed I/O control module is in the normal operation mode, the inter-module synchronization signal is not input. Thus, the signal output from the connector for external devices does not change.

■Delay time

A delay time is calculated by multiplying "Delay Time (Unit)" by "Delay Time (Step)".

An error of one unit time may be generated in delay time. However, the error can be reduced by setting the delay time as shown below.

- Change the value of "Delay Time (Unit)" as small as possible.
- Set a large value for "Delay Time (Step)".

Ex.

The following table lists setting examples of the delay time of 20μs. Compared with example 2, an error is smaller in the setting of example 1.

Example	Delay Time(Step)	Delay Time(Unit)	Error
Example 1	20 steps	1μs	An error of maximum 1μs is generated in the output timing.
Example 2	2 steps	10μs	An error of maximum 10μs is generated in the output timing.

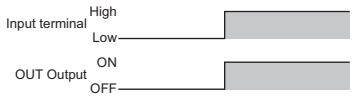
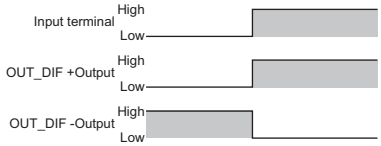
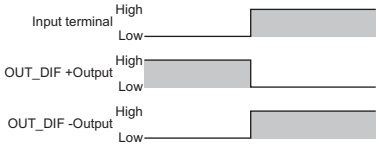
■Link with SSI encoder blocks

When an SSI encoder block is arranged in the hardware logic outline window, the "Clock" terminal of the SSI encoder block and the "Input" terminal of an external output block are automatically linked.

The default value is set for the setting value of the external output block automatically linked and the value cannot be changed.

High/Low states of external output signals

The following table lists the High/Low states of external output signals in each setting combination of input signals to external output blocks and "Logic Select".

Output type	"Logic Select"	
	"Non-Inversion"	"Inversion"
DC		
Differential		

When an error occurs in the CPU module, a signal is output according to the output setting of "Error-time Output Mode" independent of the setting of Inversion or Not-Inversion.

SI device terminal

Outline

In the fifth section in the hardware logic outline window, eight SI device terminals ("SI 0" to "SI 7") are arranged by default. When a signal is input to an SI device terminal, an interrupt request is output to the CPU module to start an interrupt program. The interrupt program corresponding to each SI device terminal needs to be set in GX Works3 in advance.

Output

The following table shows the output of the SI device terminal.

Variable name	Data type	Linkable block	Valid range	Description
Output	Bit	Multi function counter*1	0, 1	For details on interruption, refer to the following. ☞ Page 59 Interrupt Function

*1 An SI device terminal can be linked to any of the Output terminals of the multi function counter block Counter_0 to Counter_7.

3.2 Multi Function Counter Block

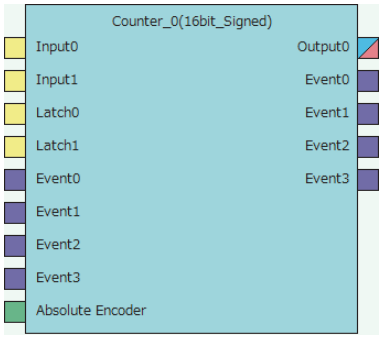
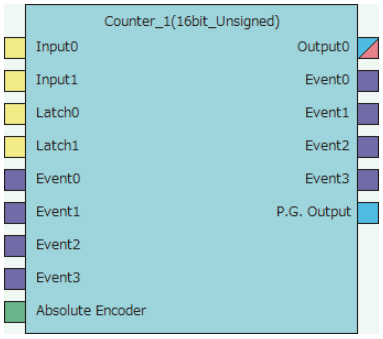
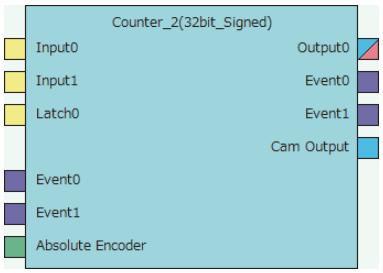
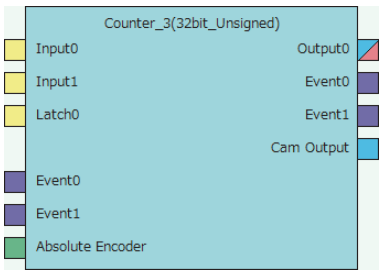
Outline

In the third section in the hardware logic outline window, up to eight multi function counter blocks ("Counter_0" to "Counter_7") can be arranged.

Set details on how to count pulses of input signals in the Multi function counter block detail window. To shift to the Multi function counter block detail window, double-click the corresponding block in the Hardware logic outline window.

Type

There are the following four types of multi function counter blocks.

Name	Block diagram	Specifications				
		Available count value	Number of Latch terminals	Number of Event terminals	Availability of "Cam Output" terminal	Availability of "P.G. Output" terminal
Counter_☐(16bit_Signed) (16-bit signed multi function counter block)		16-bit signed value	2	4	×	×
Counter_☐(16bit_Unsigned) (16-bit unsigned multi function counter block)		16-bit unsigned value	2	4	×	○
Counter_☐(32bit_Signed) (32-bit signed multi function counter block)		32-bit signed value	1	2	○	×
Counter_☐(32bit_Unsigned) (32-bit unsigned multi function counter block)		32-bit unsigned value	1	2	○	×

Input

The following table shows the inputs of the multi function counter block.

Variable name	Data type	Linkable block	Valid range	Description
Input 0 Input 1	Bit	• External input • Inter-module synchronization signal input • OUT • Y device	0, 1	The input terminals are for external input signals. Executes OR processing for all signals input to the "Input☐" terminal.
Latch 0 Latch 1	Bit	• External input • Inter-module synchronization signal input • OUT • Y device	0, 1	• The input terminals are for latch. Executes OR processing for all signals input to the "Latch☐" terminal. • The "Latch 1" terminal can be used in 16-bit multi function counter blocks only.
Event 0 Event 1 Event 2 Event 3	Bit	Multi function counter*1	0, 1	• The input terminals are for events. Executes OR processing for all signals input to the "Event☐" terminal. • The "Event 2" and "Event 3" terminals can be used in 16-bit multi function counter blocks only.
Absolute Encoder	Word	Parallel encoder	0 to 4095	Inputs the encoder value of the absolute encoder (parallel encoder block). The input encoder value is used as a preset value of the counter timer block in the multi function counter block detail window.
		SSI encoder	0 to 4294967295	Inputs the encoder value of the absolute encoder (SSI encoder block). The input encoder value is stored as a count value of the counter timer block in the multi function counter block detail window.

*1 Only the "Event☐" terminals of multi function counter blocks can be linked.

Output

The following table shows the outputs of the multi function counter block.

Variable name	Data type	Linkable block	Output value	Description
Output 0	Bit	• External output • SI device*2	0, 1	The output terminal is for external output signals and interruption.
Event 0 Event 1 Event 2 Event 3	Bit	Multi function counter*1	0, 1	The output terminals are for events. The "Event 2" and "Event 3" terminals can be used in 16-bit multi function counter blocks only. The linkable blocks differ for each Event terminal in the multi function counter block detail window. (☞ Page 207 Event output terminal)
P.G. Output	Bit	External output	0, 1	The output terminals are for external output signals. These terminals can be used in 16-bit unsigned multi function counter blocks only.
Cam Output	Bit	External output	0, 1	The output terminals are for external output signals. These terminals can be used in 32-bit multi function counter blocks only.

*1 Only the "Event☐" terminals of multi function counter blocks can be linked.
*2 Any of "SI ☐" terminals can be linked.

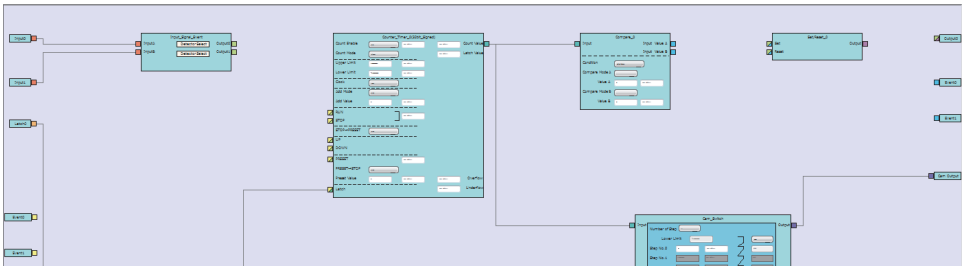
Point

When a 16-bit multi function counter block is arranged in the hardware logic outline window, two 16-bit counter timer blocks are arranged in the multi function counter block detail window. These two counter timer blocks can be simultaneously used. However, OR processing is executed on the outputs of these two counter timer blocks because a multi function counter block has only one output terminal.

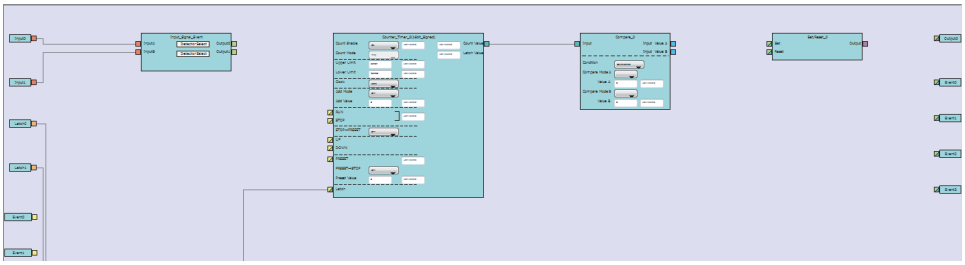
Internal blocks of multi function counter blocks

There are 14 types of internal blocks in a multi function counter block. Control operations can be freely customized by linking the internal blocks. When a multi function counter block is arranged in the hardware logic outline window, all internal blocks are arranged in the multi function counter block detail window.

- 32-bit multi function counter block detail window



- 16-bit multi function counter block detail window



Internal block list

The multi function counter internal blocks are as follows.

Section	Block name	Number of arranged items		Reference
		16-bit counter	32-bit counter	
First section	Input terminal	2	2	☞ Page 175 Input terminal
	Latch input terminal	2	1	☞ Page 175 Latch input terminal
	Event input terminal	4	2	☞ Page 176 Event input terminal
Second section	Input signal event detection block	1	1	☞ Page 177 Input signal event detection block
	Latch event detection block	1	1	☞ Page 184 Latch event detection block
Third section	Counter timer block	2	1	☞ Page 186 Counter timer block
Fourth section	Comparison block	2	1	☞ Page 196 Comparison block
	Pattern generator block	1*2	0	☞ Page 199 Pattern generator block
	Cam switch block*1	0	1	☞ Page 201 Cam switch block
Fifth section	Set/reset block	2	1	☞ Page 205 Set/reset block
Sixth section	Output terminal	1	1	☞ Page 206 Output terminal
	Event output terminal	4	2	☞ Page 207 Event output terminal
	Pattern generator output terminal	1*2	0	☞ Page 208 Pattern generator output terminal
	Cam switch output terminal	0	1	☞ Page 208 Cam switch output terminal

*1 This block is arranged across the fourth and fifth sections.
*2 This block is arranged only in 16-bit unsigned multi function counter blocks.

Link combination

An input terminal and output terminal in the same color can be linked among the internal blocks. Input terminals or output terminals cannot be linked each other.

The following table lists the combinations.

Output side		Input side		Color
Block name	**Terminal name**	**Block name**	**Terminal name**	
Input terminal	Input	Input signal event detection block	Input*1	Coral
Latch input terminal	Latch	Latch event detection block	Input*1	Pale orange
Event input terminal	Event	Counter timer block	RUN	Butterfly yellow
			STOP	
			UP	
			DOWN	
			PRESET	
			Latch	
Input signal event detection block	Output	Counter timer block	RUN	Hop green
			STOP	
			UP	
			DOWN	
			PRESET	
		Set/reset block	Set	
			Reset	
		Output terminal	Output	
		Event output terminal	Event*2	
Latch event detection block	Output	Counter timer block	Latch*1	Celadon
Counter timer block	Count Value	Comparison block	Input*1	Cyan blue
		Cam switch block	Input*1	
		Pattern generator block	Input*1	
Comparison block	Output	Set/reset block	Set	Vermeer
			Reset	
		Event output terminal	Event	
Cam switch block	Output	Cam switch output terminal	Cam Output*1	Columbine blue
Pattern generator block	Output	Pattern generator output terminal	P.G. Output*1	Columbine blue
Set/reset block	Output	Output terminal	Output	Lilac

*1 The terminals are linked by default. The links cannot be deleted.

*2 The terminals can be linked when a 16-bit multi function counter block is used. They cannot be linked when a 32-bit multi function counter block is used.

If multiple output terminals are linked to one input terminal, OR processing is executed on all input signals. If one output terminal is linked to multiple input terminals, the same signals are output for all the input terminals.

■Restrictions on linking the same-color terminals for 16-bit multi function counter blocks

For 16-bit multi function counter blocks, linkable combinations of blocks and terminals are restricted even if the terminals are in the same color.

• Input signal event detection block and event output terminal

Input signal event detection block (output side)		Event output terminal (input side)	
Block name	**Terminal name**	**Block name**	**Terminal name**
Input_Signal_Event	Output 0	Event output terminal	Event 0
			Event 1
	Output 1	Event output terminal	Event 2
			Event 3

• Comparison block and set/reset block

Comparison block (output side)		Set/reset block (input side)	
Block name	**Terminal**	**Block name**	**Terminal name**
Compare_0	Upper terminal	Set/Reset_0	Set
			Reset
	Lower terminal	Set/Reset_0	Set
			Reset
Compare_1	Upper terminal	Set/Reset_1	Set
			Reset
	Lower terminal	Set/Reset_1	Set
			Reset

• Comparison block and event output terminal

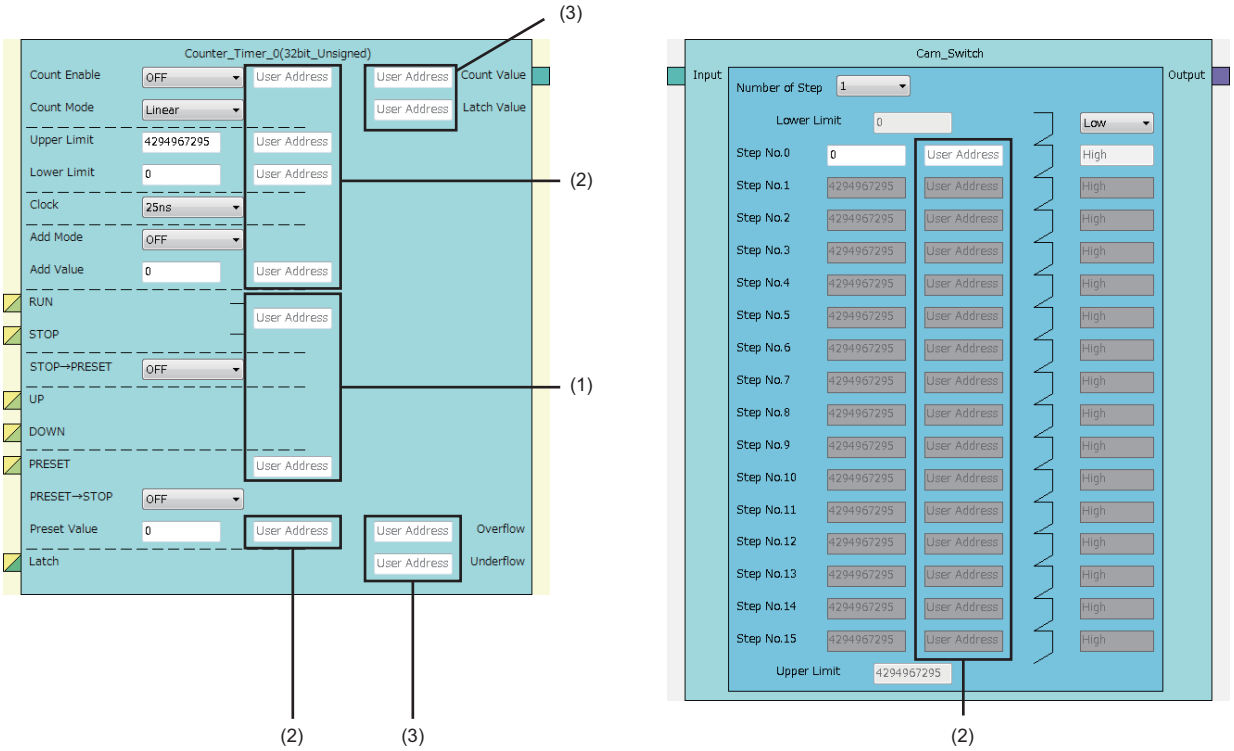
Comparison block (output side)		Event output terminal (input side)	
Block name	**Terminal**	**Block name**	**Terminal name**
Compare_0	Upper terminal	Event output terminal	Event 0
			Event 1
	Lower terminal	Event output terminal	Event 0
			Event 1
Compare_1	Upper terminal	Event output terminal	Event 2
			Event 3
	Lower terminal	Event output terminal	Event 2
			Event 3

• Counter timer block and comparison block (automatically linked in the following combinations by default)

Counter timer block (output side)		Comparison block (input side)	
Block name	**Terminal name**	**Block name**	**Terminal name**
Counter_Timer_0	Count Value	Compare_0	Input
Counter_Timer_1	Count Value	Compare_1	Input

Assignment of "User Address"

"User Address" can be set to particular internal blocks arranged in a multi function counter block detail window. By assigning buffer memory addresses to "User Address", the status of input terminals and parameter setting values can be changed with programs and values of the hardware logic can be monitored during the hardware logic control.



The application varies according to targets to which "User Address" is assigned.

No.	Assignment target	Application	Description
(1)	Input terminal	Write	The value of a buffer memory area specified with "User Address" is written to the hardware logic in a high-speed (100μs) or a low-speed (1ms) period*1. Even if the input terminal status changes due to signals in the hardware logic, the values in the buffer memory areas do not change and they are written in a high-speed (100μs) or a low-speed (1ms) period. Thus, do not assign "User Address" to the terminals linked in the hardware logic. Do not use buffer memory addresses as the monitors of input terminals because the input terminal status is not read.
(2)	Parameter	Write	The value of a buffer memory area specified with "User Address" is written to the hardware logic in a high-speed (100μs) or a low-speed (1ms) period*1.
(3)	Monitor	Read	A value of the hardware logic is read in a high-speed (100μs) or a low-speed (1ms) period*1. The value of a buffer memory area specified with "User Address" cannot be changed by users.

*1 This refresh timing is applied when the flexible high-speed I/O control module is in the normal operation mode. Refer to the following for the inter-module synchronous mode.

☞ Page 18 Update timings of the X/Y signals and Hardware logic area

Restriction

- A buffer memory address cannot be assigned to different user addresses.
- Only even addresses can be assigned to parameters and monitors of two words (32 bits).

Input terminal

Multi function

In the first section in a multi function counter block detail window, two input terminals ("Input 0", "Input 1") are arranged by default.

They are used to use signals input to a multi function counter block in the hardware logic outline window ("Input 0", "Input 1") as input signals in the multi function counter block detail window.

Output

The following table shows the output of the input terminal.

Variable name	Data type	Linkable block	Valid range	Description
Input	Bit	Input signal event detection*1	0, 1	Outputs the High/Low states of the Input terminal of a multi function counter block in the hardware logic outline window.

*1 The input terminals and the input signal event detection blocks are automatically linked in the following combinations, and the links cannot be deleted.
· "Input 0" terminal (input terminal) and "Input A" terminal (input signal event detection block)
· "Input 1" terminal (input terminal) and "Input B" terminal (input signal event detection block)

Latch input terminal

Multi function

In the first section in a multi function counter block detail window, latch input terminals are arranged by default. For a 16-bit counter, two latch input terminals ("Latch 0", "Latch 1") are arranged. For a 32-bit counter, one latch input terminal ("Latch 0") is arranged.

They are used to use latch signals input to a multi function counter block in the hardware logic outline window ("Latch 0", "Latch 1") as latch signals in the multi function counter block detail window.

Output

The following table shows the output of the latch input terminal.

Variable name	Data type	Linkable block	Valid range	Description
Latch	Bit	Latch event detection*1	0, 1	Outputs the High/Low states of the Latch terminal of a multi function counter block in the hardware logic outline window.

*1 The latch input terminals and the latch event detection blocks are automatically linked in the following combinations, and the links cannot be deleted.
· "Latch 0" terminal (latch input terminal) and "Input A" terminal (latch event detection block)
· "Latch 1" terminal (latch input terminal) and "Input B" terminal (latch event detection block)

Event input terminal

Multi function

In the first section in a multi function counter block detail window, event input terminals are arranged by default. For a 16-bit counter, four event input terminals ("Event 0" to "Event 3") are arranged. For a 32-bit counter, two event input terminals ("Event 0", "Event 1") are arranged.
They are used to use signals input to a multi function counter block in the hardware logic outline window ("Event 0" to "Event 3") as event signals in the multi function counter block detail window.

Output

The following table shows the output of the event input terminal.

Variable name	Data type	Linkable block	Valid range	Description
Event	Bit	Counter timer*1	0, 1	Outputs the High/Low states of Event terminals of a multi function counter block in the hardware logic outline window.

*1 The event input terminal can link with all the terminals of a counter timer block.

Input signal event detection block

Multi function

In the second section in a multi function counter block detail window, one input signal event detection block ("Input_Signal_Event") is arranged by default.

Set conditions to detect input signals of a multi function counter block. When the detection conditions are satisfied, the Output terminal turns to High. This operation can be utilized for the count-up or other functions of the counter timer block.

Input

The following table shows the inputs of the input signal event detection block.

Variable name	Data type	Linkable block	Valid range	Description
Input A	Bit	Input*1	0, 1	Signals from Input terminals are input. "Input A" is for a phase-A input, and "Input B" is for a phase-B input.
Input B				

*1 The input terminals and the input signal event detection blocks are automatically linked in the following combinations, and the links cannot be deleted.

· "Input 0" terminal (input terminal) and "Input A" terminal (input signal event detection block)

· "Input 1" terminal (input terminal) and "Input B" terminal (input signal event detection block)

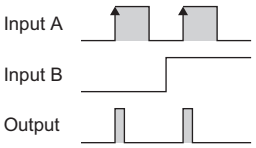
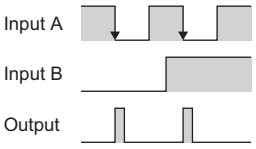
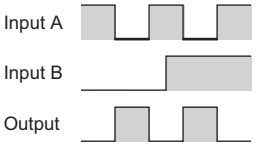
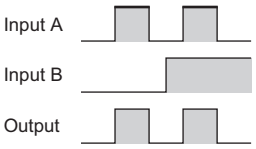
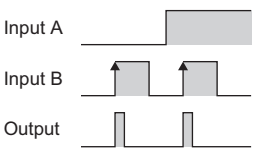
Parameter

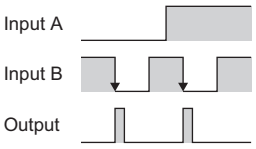
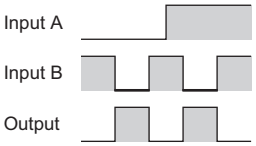
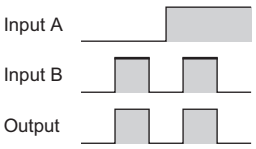
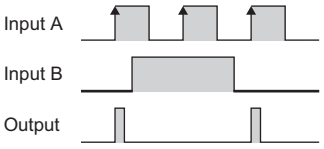
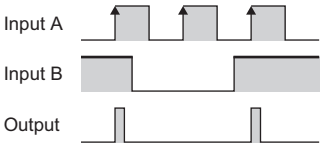
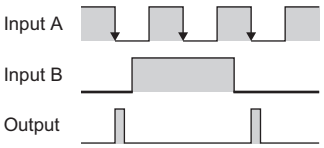
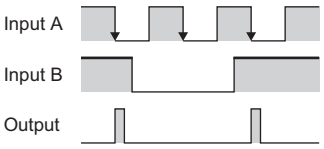
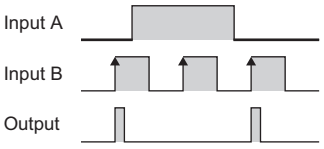
The following tables show the parameters of the input signal event detection block.

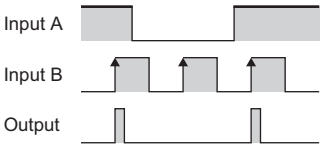
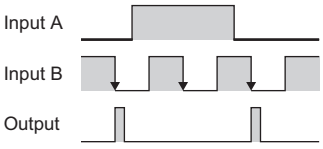
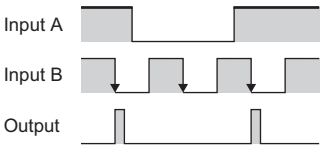
• Parameter of "Detector Select"

Variable name	Data type	Valid range	Default value	Description
Detector Select	—	—	—	Set the condition to detect signals. Click the [Detector Select] button to open the detection condition setting window ("Select Box").

• Parameters of "Select Box"

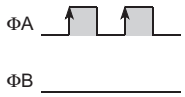
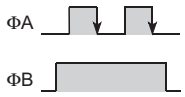
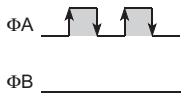
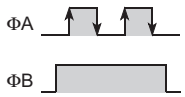
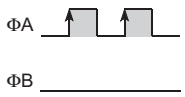
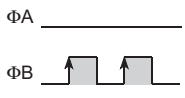
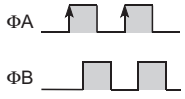
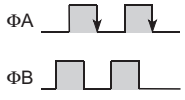
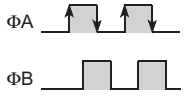
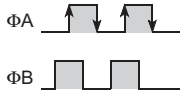
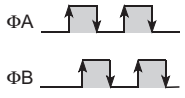
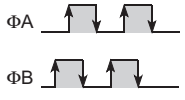
Variable name	Data type	Valid range	Default value	Description
Pulse Input Mode	Word	• User Setting • 1/2-Phase Multiple of 1(CW/CCW)+ • 1-Phase Multiple of 1- • 1-Phase Multiple of 2+ • 1-Phase Multiple of 2- • CW/CCW- • 2-Phase Multiple of 1- • 2-Phase Multiple of 2+ • 2-Phase Multiple of 2- • 2-Phase Multiple of 4+ • 2-Phase Multiple of 4-	User Setting	Set the method to detect input signals. According to the setting of this item, the following 16 settings are changed. If any of the following 16 settings is changed after this item has been set, "User Setting" is automatically set to this item.
A: Rise	Bit	OFF ON	OFF	When the rise of "Input A" has been detected, Output turns to High only for one clock cycle. The status of "Input B" does not affect the status of Output. 
A: Fall	Bit	OFF ON	OFF	When the fall of "Input A" has been detected, Output turns to High only for one clock cycle. The status of "Input B" does not affect the status of Output. 
A: Low	Bit	OFF ON	OFF	While "Input A" is Low, Output is High. The status of "Input B" does not affect the status of Output. 
A: High	Bit	OFF ON	OFF	While "Input A" is High, Output is High. The status of "Input B" does not affect the status of Output. 
B: Rise	Bit	OFF ON	OFF	When the rise of "Input B" has been detected, Output turns to High only for one clock cycle. The status of "Input A" does not affect the status of Output. 

Variable name	Data type	Valid range	Default value	Description
B: Fall	Bit	OFF ON	OFF	When the fall of "Input B" has been detected, Output turns to High only for one clock cycle. The status of "Input A" does not affect the status of Output. 
B: Low	Bit	OFF ON	OFF	While "Input B" is Low, Output is High. The status of "Input A" does not affect the status of Output. 
B: High	Bit	OFF ON	OFF	While "Input B" is High, Output is High. The status of "Input A" does not affect the status of Output. 
A: Rise+B: Low	Bit	OFF ON	OFF	When the rise of "Input A" and the Low state of "Input B" have been detected, Output turns to High only for one clock cycle. 
A: Rise+B: High	Bit	OFF ON	OFF	When the rise of "Input A" and the High state of "Input B" have been detected, Output turns to High only for one clock cycle. 
A: Fall+B: Low	Bit	OFF ON	OFF	When the fall of "Input A" and the Low state of "Input B" have been detected, Output turns to High only for one clock cycle. 
A: Fall+B: High	Bit	OFF ON	OFF	When the fall of "Input A" and the High state of "Input B" have been detected, Output turns to High only for one clock cycle. 
A: Low+B: Rise	Bit	OFF ON	OFF	When the Low state of "Input A" and the rise of "Input B" have been detected, Output turns to High only for one clock cycle. 

Variable name	Data type	Valid range	Default value	Description
A: High+B: Rise	Bit	OFF ON	OFF	When the High state of "Input A" and the rise of "Input B" have been detected, Output turns to High only for one clock cycle. 
A: Low+B: Fall	Bit	OFF ON	OFF	When the Low state of "Input A" and the fall of "Input B" have been detected, Output turns to High only for one clock cycle. 
A: High+B: Fall	Bit	OFF ON	OFF	When the High state of "Input A" and the fall of "Input B" have been detected, Output turns to High only for one clock cycle. 

■Pulse input modes and count timing

The following table shows the relationships between each pulse input mode and count timing.

"Pulse Input Mode"	Count timing		
1-phase multiple of 1 (1-Phase Multiple of 1+, 1-Phase Multiple of 1-)	At up count		The value is counted up at the rise (↑) of ΦA. ΦB is Low.
	At down count		The value is counted down at the fall (↓) of ΦA. ΦB is High.
1-phase multiple of 2 (1-Phase Multiple of 2+, 1-Phase Multiple of 2-)	At up count		The value is counted up at the rise (↑) and fall (↓) of ΦA. ΦB is Low.
	At down count		The value is counted down at the rise (↑) and fall (↓) of ΦA. ΦB is High.
CW/CCW (CW/CCW+, CW/CCW-)	At up count		The value is counted up at the rise (↑) of ΦA. ΦB is Low.
	At down count		ΦA is Low. The value is counted down at the rise (↑) of ΦB.
2-phase multiple of 1 (2-Phase Multiple of 1+, 2-Phase Multiple of 1-)	At up count		When ΦB is Low, the value is counted up at the rise (↑) of ΦA.
	At down count		When ΦB is Low, the value is counted down at the fall (↓) of ΦA.
2-phase multiple of 2 (2-Phase Multiple of 2+, 2-Phase Multiple of 2-)	At up count		When ΦB is Low, the value is counted up at the rise (↑) of ΦA. When ΦB is High, the value is counted up at the fall (↓) of ΦA.
	At down count		When ΦB is High, the value is counted down at the rise (↑) of ΦA. When ΦB is Low, the value is counted down at the fall (↓) of ΦA.
2-phase multiple of 4 (2-Phase Multiple of 4+, 2-Phase Multiple of 4-)	At up count		When ΦB is Low, the value is counted up at the rise (↑) of ΦA. When ΦA is High, the value is counted up at the rise (↑) of ΦB. When ΦB is High, the value is counted up at the fall (↓) of ΦA. When ΦA is Low, the value is counted up at the fall (↓) of ΦB.
	At down count		When ΦA is Low, the value is counted down at the rise (↑) of ΦB. When ΦB is High, the value is counted down at the rise (↑) of ΦA. When ΦA is High, the value is counted down at the fall (↓) of ΦB. When ΦB is Low, the value is counted down at the fall (↓) of ΦA.

■Detection conditions in pulse input modes

The following table shows the detection conditions in each pulse input mode.

Pulse input mode		"Pulse Input Mode"	Detection condition*1															
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1-phase multiple of 1	Up count	1-Phase Multiple of 1+	—	—	—	—	—	—	—	—	○	—	—	—	—	—	—	—
	Down count	1-Phase Multiple of 1-	—	—	—	—	—	—	—	—	—	—	—	○	—	—	—	—
1-phase multiple of 2	Up count	1-Phase Multiple of 2+	—	—	—	—	—	—	—	—	○	—	○	—	—	—	—	—
	Down count	1-Phase Multiple of 2-	—	—	—	—	—	—	—	—	—	○	—	○	—	—	—	—
CW/CCW	Up count	CW/CCW+	—	—	—	—	—	—	—	—	○	—	—	—	—	—	—	—
	Down count	CW/CCW-	—	—	—	—	—	—	—	—	—	—	—	—	○	—	—	—
2-phase multiple of 1	Up count	2-Phase Multiple of 1+	—	—	—	—	—	—	—	—	○	—	—	—	—	—	—	—
	Down count	2-Phase Multiple of 1-	—	—	—	—	—	—	—	—	—	—	○	—	—	—	—	—
2-phase multiple of 2	Up count	2-Phase Multiple of 2+	—	—	—	—	—	—	—	—	○	—	—	○	—	—	—	—
	Down count	2-Phase Multiple of 2-	—	—	—	—	—	—	—	—	—	○	○	—	—	—	—	—
2-phase multiple of 4	Up count	2-Phase Multiple of 4+	—	—	—	—	—	—	—	—	○	—	—	○	—	○	○	—
	Down count	2-Phase Multiple of 4-	—	—	—	—	—	—	—	—	—	○	○	—	○	—	—	○

*1 The numbers of Detection condition indicate the following parameters.
1: "A: Rise"
2: "A: Fall"
3: "A: Low"
4: "A: High"
5: "B: Rise"
6: "B: Fall"
7: "B: Low"
8: "B: High"
9: "A: Rise+B: Low"
10: "A: Rise+B: High"
11: "A: Fall+B: Low"
12: "A: Fall+B: High"
13: "A: Low+B: Rise"
14: "A: High+B: Rise"
15: "A: Low+B: Fall"
16: "A: High+B: Fall"

■Links of "Pulse Input Mode"

Link the Output terminal where "Pulse Input Mode" is set to "UP count" to the "UP" terminal of a counter timer block. Link the Output□ terminal where "Pulse Input Mode" is set to "Down count" to the "DOWN" terminal of a counter timer block. Otherwise, values are not correctly counted.

Output

The following table shows the outputs of the input signal event detection block.

Variable name	Data type	Linkable block	Output value	Description
Output 0 Output 1	Bit	• Counter timer*1 • Set/reset • Output • Event output*2	0, 1	Outputs signals detected with the detector.

*1 The "RUN", "STOP", "PRESET", "UP", or "DOWN" terminal of counter timer blocks can be linked.

*2 The terminals can be linked when a 16-bit multi function counter block is used, as shown below. They cannot be linked when a 32-bit multi function counter block is used.

· Event output terminals that can be linked with the "Output 0" terminal: "Event 0" terminal, "Event 1" terminal

· Event output terminals that can be linked with the "Output 1" terminal: "Event 2" terminal, "Event 3" terminal

Setting method

Set the input signal event detection block with the following procedure.

1. Click the [Detector Select] button to open "Select Box".
2. Set "Pulse Input Mode" according to the pulse input mode of external devices.
3. Change the settings in "A: Rise" to "A: High+B: Fall" as necessary.

- If any of the settings has been changed, "Pulse Input Mode" is set to "User Setting".
- When the items of "A: Rise" to "A: High+B: Fall" have been set with "Pulse Input Mode" not set and any of the detection conditions of the pulse input mode is satisfied, the corresponding pulse input mode is applied to "Pulse Input Mode" when "Select Box" is opened again after closing of it.

Restriction

When "User Setting" is set in "Pulse Input Mode", all the items in "A: Rise" to "A: High+B: Fall" are set to "OFF".

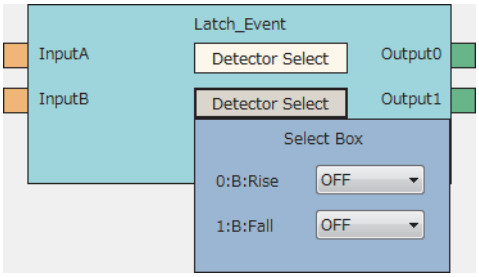
Latch event detection block

Multi function

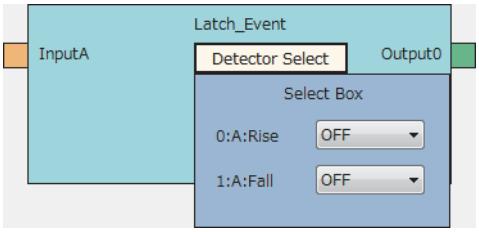
In the second section in a multi function counter block detail window, one latch event detection block ("Latch_Event") is arranged by default.

Set conditions to detect latch input signals of the multi function counter block. When the detection conditions are satisfied, the Output terminal turns to High.

- 16-bit multi function counter block



- 32-bit multi function counter block



Input

The following table shows the inputs of the latch event detection block.

Variable name	Data type	Linkable block	Valid range	Description
Input A	Bit	Latch input*1	0, 1	Signals from Latch terminals are input.
Input B*2				

*1 The latch input terminals and the latch event detection blocks are automatically linked in the following combinations, and the links cannot be deleted.
· "Latch 0" terminal (latch input terminal) and "Input A" terminal (latch event detection block)
· "Latch 1" terminal (latch input terminal) and "Input B" terminal (latch event detection block)

*2 The terminal is not displayed when a 32-bit multi function counter block is used.

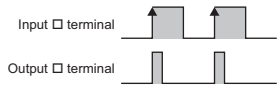
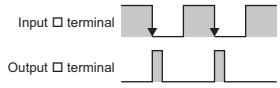
Parameter

The following tables show the parameters of the latch event detection block.

• Parameter of "Detector Select"

Variable name	Data type	Valid range	Default value	Description
Detector Select	Word	—	—	Set the condition to detect signals. Click the [Detector Select] button to open the "Select Box" window. The Detector Select on the upper side is for detecting signals of "Input A". The Detector Select on the lower side is for detecting signals of "Input B".

• Parameters of "Select Box"

Variable name	Data type	Valid range	Default value	Description
A: Rise ("B: Rise" for lower-side Detector Select for a 16-bit multi function counter block)	Bit	OFF ON	OFF	When the rise of Input has been detected, Output turns to High only for one clock cycle. 
A: Fall ("B: Fall" for lower-side Detector Select for a 16-bit multi function counter block)	Bit	OFF ON	OFF	When the fall of Input has been detected, Output turns to High only for one clock cycle. 

Output

The following table shows the outputs of the latch event detection block.

Variable name	Data type	Linkable block	Valid range	Description
Output 0	Bit	Counter timer*1	0, 1	Outputs signals detected with the detector.
Output 1*2				

*1 The latch event detection block and the Latch terminals of the counter timer blocks are automatically linked in the following combinations, and the links cannot be deleted.
· "Output 0" terminal (latch event detection block) and "Latch" terminal ("Counter_Timer_0")
· "Output 1" terminal (latch event detection block) and "Latch" terminal ("Counter_Timer_1") (This combination cannot be used for 32-bit multi function counter blocks.)

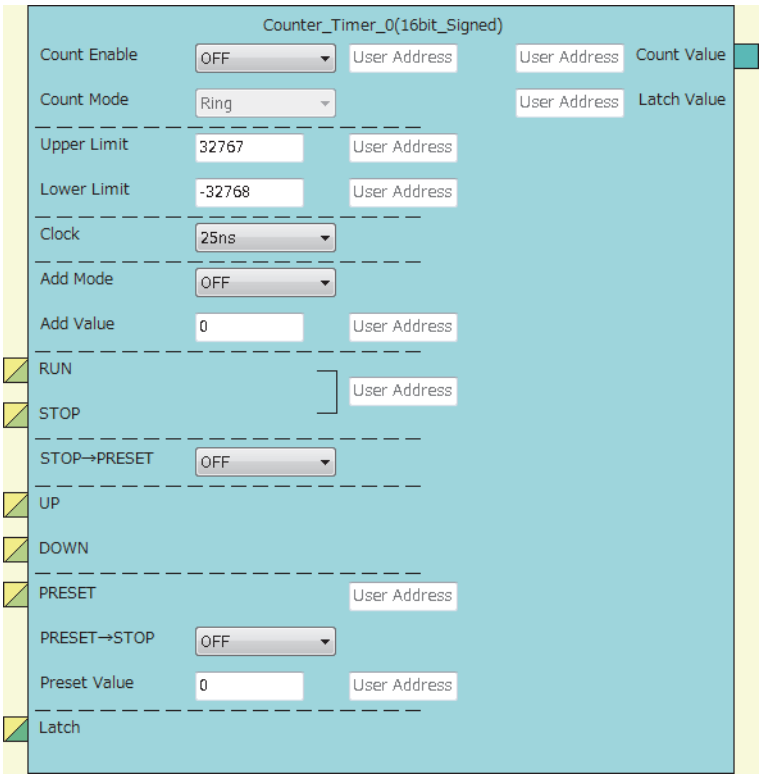
*2 The terminal is not displayed when a 32-bit multi function counter block is used.

Counter timer block

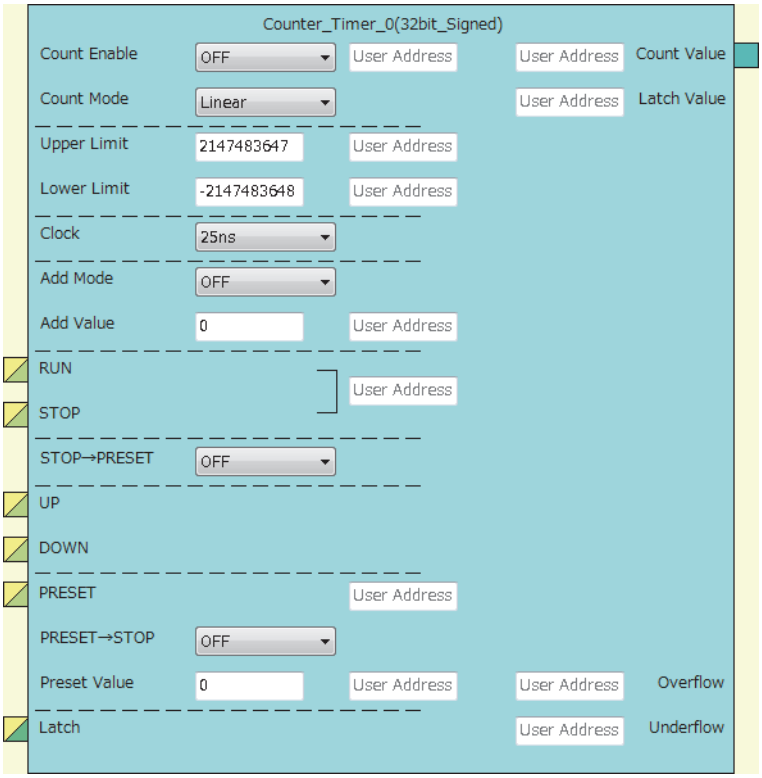
Multi function

In the third section in a multi function counter block detail window, counter timer blocks are arranged by default. For a 16-bit counter timer block, two counter timer blocks ("Counter_Timer_0"(16bit_Unsigned/Signed), "Counter_Timer_1"(16bit_Unsigned/Signed)) are arranged. For a 32-bit counter timer block, one counter timer block ("Counter_Timer_0"(32bit_Unsigned/Signed)) is arranged.
The counter operates according to the signals of the event input terminals, the input signal event detection block, and the latch event detection block and the count values are output.

- 16-bit multi function counter block



- 32-bit multi function counter block



The counter timer block has the following functions.

- Input pulses are counted.
- Counting is performed every clock cycle.
- Either of the ring counter mode or the linear counter mode can be selected (For 16-bit counter timer blocks, only the ring counter mode can be selected).
- Count values can be preset and latched.
- Addition values can be set (addition mode).
- For 32-bit counter timer blocks, an overflow and an underflow can be detected.

Input

The following table shows the inputs of the counter timer block.

Variable name	Data type	Linkable block	Valid range	Description
RUN[*1*6]	Bit	• Event input • Input signal event detection	0, 1	Adds 1 to the count value every preset clock cycle.[*2] Multiple signals can be input to the terminal. OR processing is executed for all input signals.
STOP[*1*6]	Bit	• Event input • Input signal event detection	0, 1	Stops the counter. Multiple signals can be input to the terminal. OR processing is executed for all input signals.
UP[*3*6]	Bit	• Event input • Input signal event detection	0, 1	Adds an addition value to the count value when a signal generated in the previous block is input to the terminal. When the addition mode is off, 1 is added to the count value.[*2] Multiple signals can be input to the terminal. OR processing is executed for all input signals.
DOWN[*3*6]	Bit	• Event input • Input signal event detection	0, 1	Subtracts an addition value from the count value when a signal generated in the previous block is input to the terminal. When the addition mode is off, 1 is subtracted from the count value.[*2] Multiple signals can be input to the terminal. OR processing is executed for all input signals.
PRESET[*1*3]	Bit	• Event input • Input signal event detection	0, 1	Presets the count value. Multiple signals can be input to the terminal. OR processing is executed for all input signals.
Latch[*4]	Bit	• Latch event detection • Event input[*5]	0, 1	Latches the count value to a register.
User Address	Word	—	1000 to 1099	Assign a buffer memory address to change the terminal status with a program during operation. Changing the value of the buffer memory area whose address is assigned changes the set value in the hardware logic.

*1 When a buffer memory address is assigned to "User Address", control can be performed with a program without link. Assign the same buffer memory address to the "RUN" terminal and the "STOP" terminal. (bit 0: RUN, bit 1: STOP)

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	STOP	RUN

*2 When the addition mode is on, the addition value is added or subtracted.

*3 Events are detected every clock cycle of the counter timer block. Thus, while a High-state signal is input to an Event terminal, an event occurs every clock cycle.

*4 The latch event detection block and the "Latch" terminals of the counter timer blocks are automatically linked in the following combinations, and the links cannot be deleted.
· "Output 0" terminal (latch event detection block) and "Latch" terminal ("Counter_Timer_0")
· "Output 1" terminal (latch event detection block) and "Latch" terminal ("Counter_Timer_1")

*5 On the rising edge (Low state → High state) of the event input terminal, the count value is latched.

*6 Do not link these terminals when an SSI encoder block is linked with a multi function counter block in the hardware logic outline window. Otherwise, correct encoder values may not be acquired.

Point

The High/Low states of the input terminals are detected per clock cycle. When the High states of multiple terminals are detected, only the input of the terminal with the highest priority becomes valid. The inputs of terminals with lower priority become invalid. Link the terminals so that multiple signals are not input at the same time. (Creating a hardware logic where the input to the STOP terminal is detected during the High state is recommended.)

The following shows the priority of the terminals.

1. "PRESET" terminal
2. "STOP" terminal
3. "RUN" terminal
4. "UP" terminal
5. "DOWN" terminal

The RUN terminal holds an event. Thus, when the High states of the PRESET terminal and the RUN terminal for one clock cycle are detected, the PRESET terminal becomes valid. The RUN terminal becomes valid in the next clock cycle.

Parameter

The following table shows the parameters of the counter timer block.

Variable name	Data type	Valid range		Default value	Description
Count Enable	Bit	OFF ON		ON	Set the count enable to valid or invalid. OFF: Invalid ON: Valid
Count Mode	Bit	Linear Ring		Linear	Set the counter mode to the ring counter or the linear counter. The counter mode is fixed to "Ring" for 16-bit counter timer blocks. Linear: Linear counter Ring: Ring counter
Upper Limit	Word	16-bit signed counter*1	-32768 to 32767	32767	Set the upper limit value of the counter timer.
		16-bit unsigned counter*1	0 to 65535	65535	
		32-bit signed counter*1	-2147483648 to 2147483647	2147483647	
		32-bit unsigned counter*1	0 to 4294967295	4294967295	
Lower Limit	Word	16-bit signed counter*1	-32768 to 32767	-32768	Set the lower limit value of the counter timer.
		16-bit unsigned counter*1	0 to 65535	0	
		32-bit signed counter*1	-2147483648 to 2147483647	-2147483648	
		32-bit unsigned counter*1	0 to 4294967295	0	
Clock	Word	25ns 50ns 0.1μs 1μs 10μs 100μs 1ms		25ns	Set the clock cycle. To count input pulses, set "25ns" for "Clock". Setting a value other than "25ns" may not count the input pulses correctly.
Add Mode	Bit	OFF ON		OFF	Set the addition mode to valid or invalid. OFF: Invalid ON: Valid
Add Value	Word	16-bit signed counter*1	0 to 65535	0	Set the value of addition/subtraction for count.
		16-bit unsigned counter*1			
		32-bit signed counter*1	0 to 4294967295		
		32-bit unsigned counter*1			
STOP → PRESET	Bit	OFF ON		OFF	Set whether to perform the preset function or not at an occurrence of a STOP event. OFF: The preset function is not performed. ON: The preset function is performed.
PRESET → STOP	Bit	OFF ON		OFF	Set whether to stop counting or not at an occurrence of a PRESET event. OFF: Counting is continued. ON: Counting is stopped.
Preset Value	Word	16-bit signed counter*1	-32768 to 32767	0	Set the preset value.
		16-bit unsigned counter*1	0 to 65535		
		32-bit signed counter*1	-2147483648 to 2147483647		
		32-bit unsigned counter*1	0 to 4294967295		

*1 The valid range is determined by the type of the multi function counter block arranged in the hardware logic outline window.

Restriction

The settings of "Clock" of two 16-bit counter timer blocks must be the same. When one "Clock" setting is changed, the other "Clock" setting is changed automatically.

The following table shows "User Address" used for the parameters.

Variable name	Data type	Valid range	Setting range	Description
User Address	Word	—	1000 to 1099	Only even addresses can be assigned to the parameters of two words (32 bits).

Output

The following table shows the output of the counter timer block.

Variable name	Data type	Linkable block	Output value	Description
Count Value	Word	• Comparison*3 • Cam switch*2 • Pattern generator*4	• -32768 to 32767 (16-bit signed counter)*1 • 0 to 65535 (16-bit unsigned counter)*1 • -2147483648 to 2147483647 (32-bit signed counter)*1 • 0 to 4294967295 (32-bit unsigned counter)*1	Outputs the value counted by the counter timer.

*1 The valid range is determined by the type of the multi function counter block arranged in the hardware logic outline window.

*2 The cam switch block can be linked only with a 32-bit counter timer block. The 32-bit counter timer block and the cam switch block are linked automatically and the link cannot be deleted.

*3 Counter timer blocks and Comparison blocks are linked automatically in the following combinations, and the links cannot be deleted.
· "Count Value" terminal of "Counter_Timer_0" and "Input" terminal of "Compare_0"
· "Count Value" terminal of "Counter_Timer_1" and "Input" terminal of "Compare_1" (This combination cannot be used for 32-bit multi function counter blocks.)

*4 The pattern generator block can be linked only with a 16-bit unsigned counter timer block. A 16-bit unsigned counter timer block and the pattern generator block is linked automatically and the link cannot be deleted.

Restriction

The count value is cleared to 0 when the hardware logic control is stopped.
To stop the count operation without clearing the count value, set "Count Enable" to "OFF". To switch the on/off of "Count Enable" during the hardware logic control, assign a buffer memory address to "User Address" of "Count Enable".

Monitor

The following table shows the monitors of the counter timer block.

Variable name	Data type	Description
Count Value	Word	A count value is stored in the buffer memory area specified with "User Address".
Latch Value	Word	A latched count value is stored in the buffer memory area specified with "User Address".
Overflow	Word	When an overflow is detected, 1 is stored in the buffer memory area specified with "User Address".*1
Underflow	Word	When an underflow is detected, 1 is stored in the buffer memory area specified with "User Address".*1

*1 Even when no "User Address" is assigned, an overflow and an underflow are detected and the multi function counter block ☐ overflow error (error code: 100☐H) and the multi function counter block ☐ underflow error (error code: 101☐H) occur. These errors are detected only when the linear counter mode is set for 32-bit counter timer blocks.

The following table shows "User Address" used for the monitors.

Variable name	Data type	Linkable block	Valid range	Description
User Address	Word	—	1000 to 1099	Only even addresses can be assigned to monitors of two words (32 bits).

Counting method

In the flexible high-speed I/O control module, the following counting methods are provided: counting addition pulses and subtraction pulses from external devices and performing up count per clock cycle.

The counting method can be set by linking blocks in the hardware logic.

Counting method	Link in hardware logic
Counting the input pulses from external devices	Link the "Output" terminals of the input signal event detection block where the pulse input mode is set according to the external input pulse to the "UP" terminal and "DOWN" terminal of the counter timer block.
Counting based on the clock cycle inside the module	Link terminals to the "RUN" terminal and the "STOP" terminal.

In a counter timer block where the "UP" terminal, "DOWN" terminal, "RUN" terminal, and "STOP" terminal are linked, either of the above counting methods can be used.

■Operation of when input pulses from external devices are counted

When the High state of the "UP" terminal is detected, the addition value is added to the count value. When the High state of the "DOWN" terminal is detected, the addition value is subtracted from the count value. When the High states of the "UP" terminal and the "DOWN" terminal are detected at the same time, the addition value is added to the count value and the detection of the High state of the "DOWN" terminal is ignored.

Point

While input pulses are counted, the count operation can be stopped externally because the "STOP" terminal has higher priority than the "UP" terminal and the "DOWN" terminal.

■Operation of when counting is performed based on the clock cycle inside the module

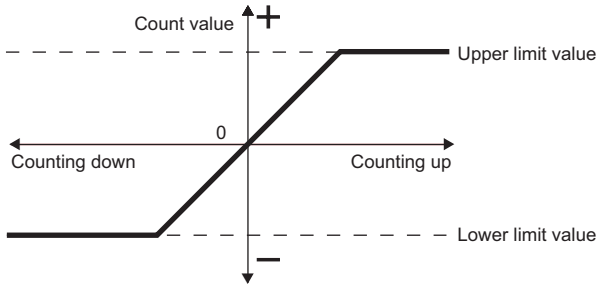
The count operation is performed as a timer. When the High state of the "RUN" terminal is detected, counting up the count value is started. Set the up count cycle with "Clock". When the High state of the "STOP" terminal is detected, counting up the count value is stopped. When the High states of the "RUN" terminal and the "STOP" terminal are detected at the same time, only the detection of the High state of the "STOP" terminal becomes valid. The detection of the High state of the "RUN" terminal is ignored.

Linear counter mode

When the linear counter mode is selected, the count operation is performed between the lower limit value and the upper limit value. The following table shows the setting ranges (lower limit value and upper limit value).

Multi function counter block	Setting range
32-bit unsigned	0 to 4294967295
32-bit signed	-2147483648 to 2147483647

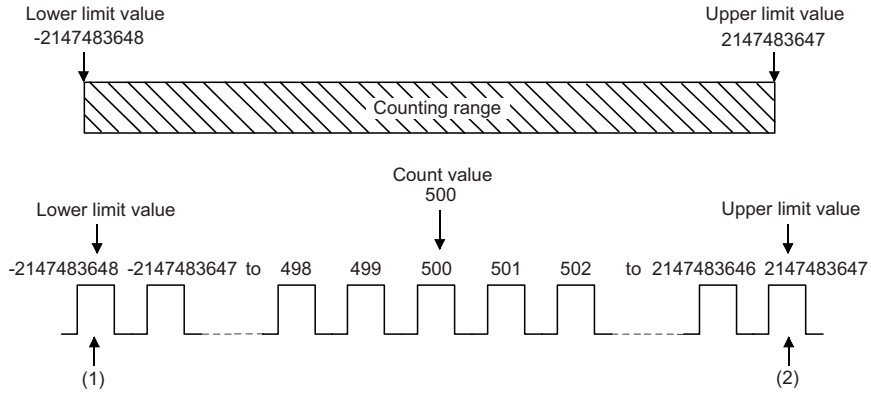
The linear counter mode can be set only for a 32-bit counter timer block.



If "Count Value" has reached the upper limit value in up count, "Count Value" does not change in the next up count and the overflow error (error code: 100☐H) occurs. "Count Value" does not change until the preset function is performed even if down count is performed.
If "Count Value" has reached the lower limit value in down count, "Count Value" does not change in the next down count and the underflow error (error code: 101☐H) occurs. "Count Value" does not change until the preset function is performed even if up count is performed.

Ex.

The following figure shows how the counting range and the count value change in a 32-bit signed counter timer block when the preset function is performed with the following settings: the upper limit value is 2147483647, the lower limit value is -2147483648, and "Preset Value" is 500.



(1) If "Count Value" has reached the lower limit value, an underflow occurs when a subtraction pulse is counted.
(2) If "Count Value" has reached the upper limit value, an overflow occurs when an addition pulse is counted.

■Overflow/underflow error

- In a 32-bit counter timer block with the linear counter mode setting, when "Count Value" exceeds the upper limit value in up count, the overflow error (error code: 100□H) is stored in Latest error code (Un\G100). When "Count Value" falls below the lower limit value in down count, the underflow error (error code: 101□H) is stored in Latest error code (Un\G100).
- When the overflow error or underflow error has occurred, the count stops and "Count Value" does not change even if up count or down count is performed.
- These errors can be cleared by performing the preset function. Performing the preset function stores a preset value in "Count Value" and restarts the count. The value stored in Latest error code (Un\G100) is held until the error is reset. Clear the error using Error clear request (YF).

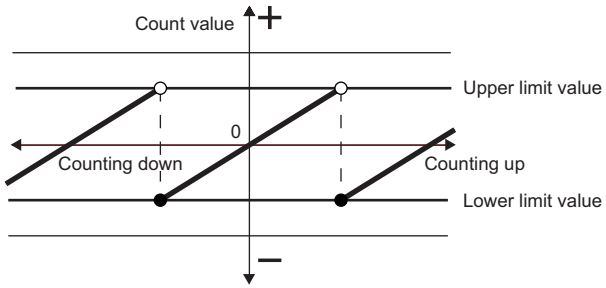
Restriction

Set values as follows in linear counter mode.

- Set the upper limit value and the lower limit value so that the lower limit value is smaller than the upper limit value. When the upper limit value is equal to or smaller than the lower limit value, the overflow error or underflow error occurs and the count operation is not performed.
- Set the preset value so that it is equal to or larger than the lower limit value and is equal to or smaller than the upper limit value.

Ring counter mode

When the ring counter mode is selected, the count operation is performed between "Lower Limit" and "Upper Limit" repeatedly. The overflow error and underflow error do not occur.

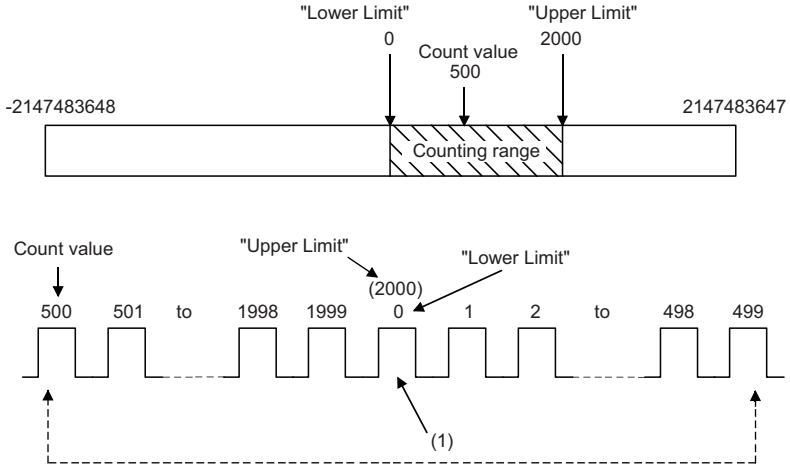


■Lower limit value ≤ Count value < Upper limit value

In up count, when "Count Value" has reached "Upper Limit", the value of "Lower Limit" is stored in "Count Value" automatically. In down count, the count value is held at the value of "Lower Limit" even if "Count Value" has reached "Lower Limit", and the value obtained by subtracting 1 from "Upper Limit" is stored in the count value in the next down count.
In both up count and down count, the value of "Upper Limit" is not stored in "Count Value". However, these specifications exclude the case where "Count Enable" is turned on from off or the values of "Count Value" and "Upper Limit" are the same when the preset function is performed.

Ex.

The following figure shows how the counting range and the count value change when the preset function is performed with the following settings: "Lower Limit" is 0, "Upper Limit" is 2000, and "Preset Value" is 500.



(1) The value 2000 of "Upper Limit" is not stored in "Count Value".

Point

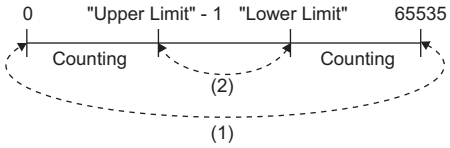
When the same values are set for "Upper Limit" and "Lower Limit", the ring counter operation can be performed with the maximum valid range.
The following show examples for a 32-bit signed counter timer block.

- When up count is performed with the count value 2147483647, the count value becomes -2147483648.
- When down count is performed with the count value -2147483648, the count value becomes 2147483647.

When up count is performed from 2147483646 with "Upper Limit" = 2147483647 and "Lower Limit" = -2147483648, the count value becomes -2147483648. Thus, the count range is one less than that of when the same value is set to the "Upper Limit" and the "Lower Limit".

■Upper limit value ≤ Lower limit value

In up count, when "Count Value" has reached the value of "Upper Limit" -1, "Lower Limit" is stored in "Count Value" in the next up count. In down count, when "Count Value" has reached the "Lower Limit", the value of "Upper Limit" - 1 is stored in "Count Value" in the next down count. When the same values are set to "Upper Limit" and "Lower Limit", counting becomes possible within the entire range for 32-bit signed/unsigned and 16-bit signed/unsigned count timer blocks.
The following figure shows a count operation example for a 16-bit unsigned counter timer block.



(1) When the count value is counted down from 0, the value becomes 65535. When the count value is counted up from 65535, the value becomes 0.
(2) When the count value is counted down from "Lower Limit", the value becomes "Upper Limit" - 1. When the count value is counted up from "Upper Limit" - 1, the value becomes "Lower Limit".

Restriction

Set values as follows in ring counter mode.

- Make sure that the count value is within the counting range set with an upper limit value and lower limit value. When the count value is out of counting range, set the preset value within the counting range and perform the preset function. If the count operation is continued with the count value out of counting range, the operation cannot be guaranteed.
- Set the preset value within the counting range set with an upper limit value and lower limit value.

Addition mode

When "Add Mode" is set to "ON", the following operations are performed.

- For the counting based on the clock cycle, a value set as the addition value is added every clock cycle.
- For counting input pulses, a value set with "Add Value" is added when an addition pulse is input and a value set with "Add Value" is subtracted when a subtraction pulse is input.

■When the addition mode is on in the linear counter mode

When the addition mode is on in the linear counter mode, the count value does not change even if the count operation is performed with the following status.
"Count Value" + "Add Value" > "Upper Limit"

Ex.

When "Add Value" is 5, "Upper Limit" is 1000, and "Count Value" is 998 in a 32-bit signed counter timer block, "Count Value" remains 998 even if up count is performed, and an overflow occurs.

■When the addition mode is on in the ring counter mode

When the addition mode is on in the ring counter mode and up count is performed in the state in which the total of "Count Value" and "Add Value" exceeds "Upper Limit", the count value is as follows.
Count value after addition = "Lower Limit" + ("Add Value" - ("Upper Limit" - Current count value))

Ex.

When "Add Value" is 5, "Upper Limit" is 1000, "Lower Limit" is 0, "Count Value" is 998, and up count is performed, the current count value becomes 3.

Restriction

To turn on the addition mode in the ring counter mode, set a value which satisfies the following condition for "Add Value". When a value which does not satisfy the following condition is set, "Count Value" may exceed "Upper Limit" or fall below "Lower Limit".

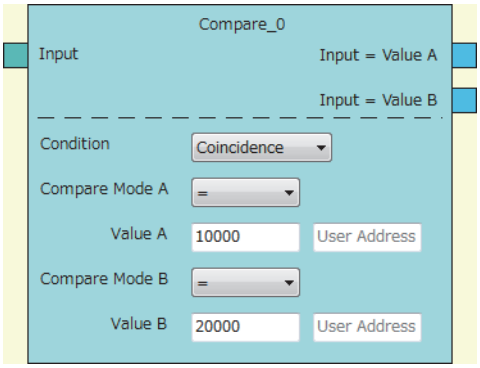
- "Add Value" < ("Upper Limit" - "Lower Limit")

Comparison block

Multi function

In the fourth section in a multi function counter block detail window, comparison blocks are arranged by default. For a 16-bit counter, two comparison blocks ("Compare_0", "Compare_1") are arranged. For a 32-bit counter, one comparison block ("Compare_0") is arranged.

"Count Value" in a counter timer block is compared with the value set with the parameter. If a condition is satisfied, the state of the "Output" terminal turns to High. Two comparison conditions can be set for one counter timer block.



Input

The following table shows the input of the comparison block.

Variable name	Data type	Linkable block	Valid range	Description
Input	Word	Counter timer	• -32768 to 32767 (16-bit signed counter)*1 • 0 to 65535 (16-bit unsigned counter)*1 • -2147483648 to 2147483647 (32-bit signed counter)*1 • 0 to 4294967295 (32-bit unsigned counter)*1	Inputs the count value of the counter timer block.

*1 The valid range is determined by the type of the multi function counter block arranged in the hardware logic outline window.

■Automatic link

Counter timer blocks and comparison blocks are linked automatically in the following combinations, and the links cannot be deleted.

Multi function counter block	Counter timer block	Comparison block
16-bit multi function counter block	"Count Value" terminal of "Counter_Timer_0"	"Input" terminal of "Compare_0"
	"Count Value" terminal of "Counter_Timer_1"	"Input" terminal of "Compare_1"
32-bit multi function counter block	"Count Value" terminal of "Counter_Timer_0"	"Input" terminal of "Compare_0"

Parameter

The following table shows the parameters of the comparison block.

Variable name	Data type	Valid range	Default value	Description
Condition	Word	Coincidence Range	Coincidence	Specify a comparison method. • Coincidence • Range
Compare Mode A Compare Mode B	Word	(Blank) = > < >= <= <>	(Blank)	Select a mode for comparing the count value and the compare value. When "Condition" is set to "Range", this setting is disabled.
Compare Value A Compare Value B	Word	• -32768 to 32767 (16-bit signed counter)*1 • 0 to 65535 (16-bit unsigned counter)*1 • -2147483648 to 2147483647 (32-bit signed counter)*1 • 0 to 4294967295 (32-bit unsigned counter)*1	0	Set the value to be compared with the count value.

*1 The valid range is determined by the type of the multi function counter block arranged in the hardware logic outline window.

The following table shows "User Address" used for the parameters.

Variable name	Data type	Valid range	Setting range	Description
User Address	Word	—	1000 to 1099	Only even addresses can be assigned to the parameters of two words (32 bits).

Ex.

The following table shows examples of comparison with "Count Value" of the counter timer block when "Compare Value" is set to 1000 and "Compare Mode" is set to >.

"Count Value"	Comparison result
900	The comparison condition is not met because "Count Value" is equal to or smaller than "Compare Value". Thus, the "Output" terminal status is turned to Low.
1100	The comparison condition is met because "Count Value" is larger than "Compare Value". Thus, the "Output" terminal status is turned to High.

Output

The following table shows the output of the comparison block.

■When "Condition" is set to "Coincidence"

Variable name	Data type	Linkable block	Output value	Description
Input ☐ Value A*1	Bit	• Set/reset • Event output	0, 1	Outputs the comparison result between "Count Value" and "Compare Value A".
Input ☐ Value B*1	Bit	• Set/reset • Event output	0, 1	Outputs the comparison result between "Count Value" and "Compare Value B".

*1 The setting of "Compare Mode" is reflected to ☐.

Point

A value other than 0 is recommended for the compare value. When the compare value is set to 0, the status of the Output terminal turns to High soon after the hardware logic control starts because the count value soon after the hardware logic control starts is 0.

■When "Condition" is set to "Range"

Variable name	Data type	Linkable block	Output value	Description
Value A <= Input <= Value B	Bit	• Set/reset • Event output	0, 1	Compares "Count Value" with the range using "Compare Value A" and "Compare Value B" and outputs the result. • Within range: High • Out of range: Low
Input < Value A, Value B < Input	Bit	• Set/reset • Event output	0, 1	Compares "Count Value" with the range using "Compare Value A" and "Compare Value B" and outputs the result. • Within range: Low • Out of range: High

Restriction

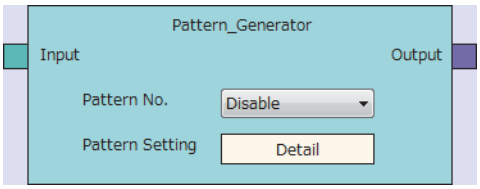
Set the compare value so that "Compare Value A" is equal to or smaller than "Compare Value B". When "Compare Value A > Compare Value B" is set, values are output as follows.

- Output of "Value A <= Input <= Value B" is always Low.
- Output of "Input < Value A, Value B < Input" is always High.

Pattern generator block

Multi function

In the fourth section in the multi function counter block detail window (Counter_☐(16bit_Unsigned)), one pattern generator block ("Pattern_Generator") is arranged by default. No pattern generator block is arranged in the multi function counter block detail window (Counter_☐(16bit_Signed)), (Counter_☐(32bit_Signed)), or (Counter_☐(32bit_Unsigned)).
A pattern generator block performs pattern output according to the output pattern data consisting of up to 8192 points that have been set for the block. Set Low output (0) or High output (1) for the output pattern of each point number. The count value of a counter timer block linked with the Input terminal is a point number. Thus, changing the count value continuously allows performing pattern output.



Input

The following table shows the input of the pattern generator block.

Variable name	Data type	Linkable block	Valid range	Description
Input	Word	Counter timer*2	0 to 8191*1	Inputs the count value of the counter timer block.

*1 When the value exceeds the range of 0 to 8191, it is regarded as a lower 13-bit value. For example, when 8192 is input, it is regarded as 0 and when 10000 is input, it is regarded as 1808.
*2 The "Count Value" terminal of "Counter_Timer_0" and the pattern generator block is linked automatically and the link cannot be deleted.

Parameter

The following table shows the parameters of the pattern generator block.

Variable name	Data type	Valid range	Default value	Description
Pattern No.	Word	Disable 1 to 10	Disable	Specify the output pattern number of an output pattern output from the pattern generator block. An output pattern number indicates the number of the output pattern data created in the "Pattern generator output pattern setting" window.
Pattern Setting	—	—	—	Click the [Detail] button to open the "Pattern generator output pattern setting" window.

For details on the "Pattern generator output pattern setting" window, refer to the following.
☞ Page 138 Pattern Generator Function

Output

The following table shows the output of the pattern generator block.

Variable name	Data type	Linkable block	Output value	Description
Output	Bit	Pattern generator output	0, 1	Outputs the output pattern according to the output pattern set for "Pattern No.". When "Disable" is set for "Pattern No.", the output value is 0.

Operation details

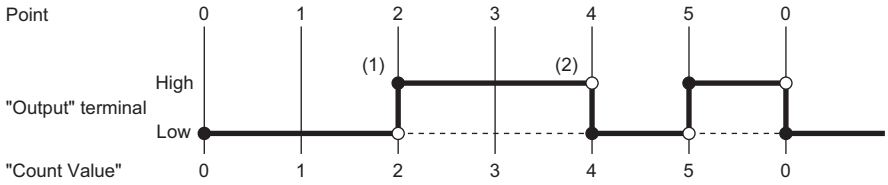
■Change timing of the pattern generator output

The pattern generator output switches when the count value of the counter timer block is changed.

Ex.

The following shows an operation example when output patterns of 6 points are set.

Item		Setting value
Counter timer block	"Count Mode"	"Ring"
	"Upper Limit"	6
	"Lower Limit"	0
Output pattern	Number of points	6
	Output value of the point 0	0
	Output value of the point 1	0
	Output value of the point 2	1
	Output value of the point 3	1
	Output value of the point 4	0
	Output value of the point 5	1



No.	Description
(1)	When the count value is changed from 1 to 2, the output turns to High, which is the output value of the point 2.
(2)	When the count value is changed from 3 to 4, the output turns to Low, which is the output value of the point 4. After that, the output switches between Low and High depending on the output value of the point same as the count value.

■When a count value exceeding the number of output patterns is input

When a count value exceeding the number of output patterns is input, the pattern generator output is Low.

Ex.

When the number of output patterns is 100 and the count value is within the range of 100 to 8191, the pattern generator output is Low.

■When a count value exceeding 8191 (8192 to 65535) is input

When a count value exceeding 8191 (8192 to 65535) is input, the count value input to the pattern generator block is converted into a value within the range of 0 to 8191 according to the following formula.

- Remainder of "Count value divided by 8192"

Ex.

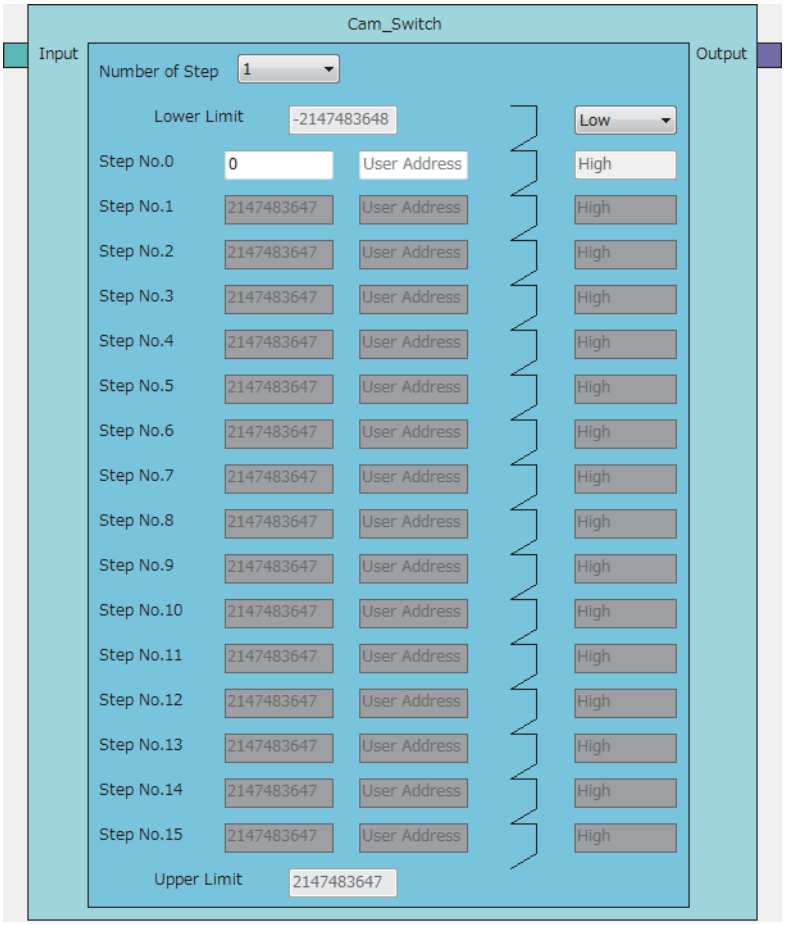
When the count value is 20000, a value obtained by dividing 20000 by 8192 is 2 with a remainder of 3616 according to the above formula. Thus, the point number is 3616.

Cam switch block

Multi function

In a multi function counter block detail window, one cam switch block ("Cam_Switch") linked to a counter timer block is arranged across the fourth and fifth sections. However, no cam switch block is arranged in a 16-bit multi function counter block.

The values in the cam switch block are compared with "Count Value" in the counter timer block and the High/Low signals are output.



Input

The following table shows the input of the cam switch block.

Variable name	Data type	Linkable block	Valid range	Description
Input	Word	Counter timer*1	• -2147483648 to 2147483647 (32-bit signed counter)*2 • 0 to 4294967295 (32-bit unsigned counter)*2	Inputs the compare value of the counter timer block. When "User Address" of the step No. is not used, set the refreshing cycle of the count value to 0.1μs or more. When "User Address" is used, set it to 0.2μs or more.*3

*1 "Count Value" in the counter timer block and the cam switch block is linked automatically and the link cannot be deleted.

*2 The valid range is determined by the type of the multi function counter block arranged in the hardware logic outline window.

*3 When "User Address" is used, variability of 100ns or less occurs when the output turns to High or Low.

Parameter

The following table shows the parameters of the cam switch block.

Variable name	Data type	Valid range	Default value	Description
Number of Step	Word	1 to 16	1	Set the number of steps of the cam switch. A step indicates the point in which the Low state is switched to the High state or the High state is switched to the Low state. The valid Step No.☐ is determined according to the set value of "Number of Step".
Lower Limit	Word	—	—	The value set with "Lower Limit" in the counter timer block is applied. Change the setting value in the counter timer block.
Lower Limit Output State	Word	Low High	Low	Set the initial state of output. When "Count Value" of the counter timer block is less than the input value of "Step No.0", the output status turns to the initial state. The output status of the step No. later than "Step No.0" inverts to the High state or the Low state for every step No.
Step No.0	Word	• -2147483648 to 2147483647 (32-bit signed counter)*1 • 0 to 4294967295 (32-bit unsigned counter)*1	0	Compares the values of step numbers with "Count Value" in the counter timer block. When "Count Value" is equal to or larger than the input value, the previous status is inverted.
Step No.1			Maximum value in the valid range	
Step No.2				
Step No.3				
Step No.4				
Step No.5				
Step No.6				
Step No.7				
Step No.8				
Step No.9				
Step No.10				
Step No.11				
Step No.12				
Step No.13				
Step No.14				
Step No.15				
Upper Limit	Word	—	—	The value set with "Upper Limit" in the counter timer block is applied. Change the setting value in the counter timer block.

*1 The valid range is determined by the type of the multi function counter block arranged in the hardware logic outline window.

The following table shows "User Address" used for the parameters.

Variable name	Data type	Valid range	Description
User Address	Word	1000 to 1098	Only even addresses can be assigned to the parameters of two words (32 bits).

Output

The following table shows the output of the cam switch block.

Variable name	Data type	Linkable block	Output value	Description
Output	Bit	Cam switch output*1	0, 1	Outputs signals created in the cam switch block. Only the "Cam Output" terminal can be linked to the "Output" terminal.

*1 The cam switch block and "Cam Output" terminal are linked automatically and the link cannot be deleted.

Operation examples

The following describes operation examples of the cam switch block.

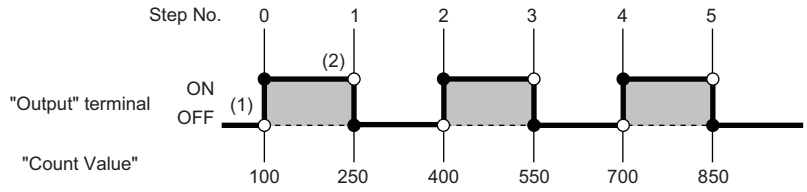
■When the number of steps is set to 6

• Setting details

Item	Setting value
Number of Step	6
Lower Limit*1	-2147483648
Lower Limit Output State	Low
Step No.0	100
Step No.1	250
Step No.2	400
Step No.3	550
Step No.4	700
Step No.5	850
Upper Limit*1	2147483647

*1 Set it in the counter timer block.

• Operation



No.	Description
(1)	For "Lower Limit" ≤ Count value < "Step No.0", the output status set with "Lower Limit Output State" is applied. When the count value is equal to or larger than the set value of "Step No.0", the output status turns to High.
(2)	When the count value is equal to or larger than the set value of "Step No.1", the output status turns to Low. After that, the output status turns to High or Low depending on the magnitude relationship between the count value and "Step No.□".

■When the number of steps is set to 2

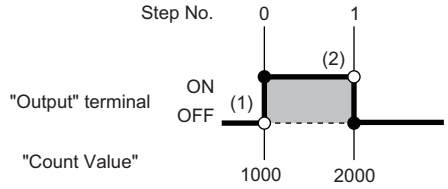
• Setting details

Item	Setting value
Number of Step	2
Lower Limit*1	-2147483648
Lower Limit Output State	Low*2
Step No.0	1000
Step No.1	2000
Upper Limit*1	2147483647

*1 Set it in the counter timer block.

*2 When "Low" is set, operations of output within the range are performed. When "High" is set, operations of output out of the range are performed.

• Operation



No.	Description
(1)	For "Lower Limit" ≤ Count value < "Step No.0", the output status set with "Lower Limit Output State" is applied. When the count value is equal to or larger than the set value of "Step No.0", the output status turns to High.
(2)	When the count value is equal to or larger than the set value of "Step No.1", the output status turns to Low.

Setting method

Set the cam switch block with the following procedure.

1. Set the number of steps of the cam switch for "Number of Step".
2. Set "Step No.0" to "Step No.15" so that the setting values in "Lower Limit", "Step No.0" to "Step No.15", and "Upper Limit" are arranged in ascending order.

Restriction

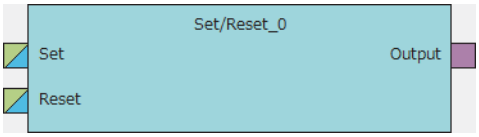
When the setting values in "Lower Limit", "Step No.0" to "Step No.15", and "Upper Limit" are not arranged in ascending order, the control for the cam switch block is not normally performed.

Set/reset block

Multi function

In the fifth section in a multi function counter block detail window, set/reset blocks are arranged by default. For a 16-bit counter, two set/reset blocks ("Set/Reset_0" and "Set/Reset_1") are arranged. For a 32-bit counter, one set/reset block ("Set/Reset_0") is arranged.

The signal input to the "Set" terminal is used as a trigger to output High fixed signals or the signal input to the "Reset" terminal is used as a trigger to output Low fixed signals.



Input

The following table shows the inputs of the set/reset block.

Variable name	Data type	Linkable block	Valid range	Description
Set	Bit	• Input signal event detection • Comparison	0, 1	Outputs the High fixed signal when an input signal is detected. Even if the input signal turns to Low, output continues until a signal is input to "Reset".
Reset	Bit	• Input signal event detection • Comparison	0, 1	Outputs the Low fixed signal when an input signal is detected.

Restriction

- When both the "Set" terminal status and "Reset" terminal status turn to High at the same time, the "Reset" terminal becomes valid.

■Link of the set/reset block

- The set/reset block in a 16-bit counter can be linked to the following comparison blocks.

Set/reset block	Comparison block
"Set/Reset_0"	"Compare_0"
"Set/Reset_1"	"Compare_1"

Output

The following table shows the output of the set/reset block.

Variable name	Data type	Linkable block	Output value	Description
Output	Bit	Output	0, 1	Outputs High fixed or Low fixed signals.

Output terminal

Multi function

In the sixth section in a multi function counter block detail window, one output terminal ("Output 0") is arranged by default. When terminals have been linked to the output terminal, the signals input to the output terminal can be output from the "Output 0" terminal in the multi function counter block arranged in the hardware logic outline window.

Output

The following table shows the output of the output terminal.

Variable name	Data type	Linkable block	Valid range	Description
Output	Bit	• Input signal event detection • Set/reset	0, 1	Outputs the High/Low states of the "Output 0" terminal of a multi function counter block in the hardware logic outline window. This terminal is used for link in the multi function counter block detail window. Multiple signals can be input to the terminal. OR processing is executed for all input signals.

Event output terminal

Multi function

In the sixth section in a multi function counter block detail window, event output terminals are arranged by default. For a 16-bit counter, four event output terminals ("Event 0" to "Event 3") are arranged. For a 32-bit counter, two event output terminals ("Event 0", "Event 1") are arranged.

Event output terminals output the input signals from "Event☐" terminals in the multi function counter block.

- 16-bit multi function counter block

- 32-bit multi function counter block

Output

The following table shows the output of the event output terminal.

Variable name	Data type	Linkable block	Valid range	Description
Event	Bit	• Input signal event detection • Comparison	0, 1	Outputs the High/Low states of "Event☐" terminals of a multi function counter block in the hardware logic outline window. This terminal is used for link in the multi function counter block detail window. Multiple signals can be input to the terminal. OR processing is executed for all input signals.

■Link of event output terminals

- The event output terminals in a 16-bit counter can be linked to the following comparison blocks.

Event output terminal	Comparison block
• "Event 0" • "Event 1"	"Compare_0"
• "Event 2" • "Event 3"	"Compare_1"

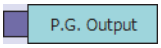
- For event output terminals in a 16-bit counter, linkable "Output" terminals of the input signal event detection block are different as shown below. Event output terminals are not linkable in a 32-bit counter.

Event output terminal	Input signal event detection block
• "Event 0" • "Event 1"	"Output 0"
• "Event 2" • "Event 3"	"Output 1"

Pattern generator output terminal

Multi function

In the sixth section in the multi function counter block detail window (Counter_☐(16bit_Unsigned)), one pattern generator output terminal ("P.G. Output") is arranged by default. No pattern generator output terminal is arranged in the multi function counter block detail window (Counter_☐(16bit_Signed)), (Counter_☐(32bit_Signed)), or (Counter_☐(32bit_Unsigned)).
The signals input to the pattern generator output terminal can be output from the "P.G. Output" terminal in the multi function counter block arranged in the hardware logic outline window.



Output

The following table shows the output of the pattern generator output terminal.

Variable name	Data type	Linkable block	Valid range	Description
P.G. Output	Bit	Pattern generator*1	0, 1	Outputs the High/Low state of the "P.G. Output" terminal of a multi function counter block in the hardware logic outline window.

*1 The pattern generator block and the pattern generator output terminal are linked automatically and the link cannot be deleted.

Cam switch output terminal

Multi function

In the sixth section in a 32-bit multi function counter block detail window, one cam switch output terminal ("Cam Output") is arranged by default. No cam switch output terminal is arranged in a 16-bit multi function counter block.
The arranged cam switch output terminal is linked with the cam switch block.
The signals input to the cam switch output terminal can be output from the "Cam Output" terminal in the multi function counter block arranged in the hardware logic outline window.

Output

The following table shows the output of the cam switch output terminal.

Variable name	Data type	Linkable block	Valid range	Description
Cam Output	Bit	Cam switch*1	0, 1	Outputs the High/Low states of the "Cam Output" terminal of a multi function counter block in the hardware logic outline window.

*1 The cam switch block and the cam switch output terminal are linked automatically and the link cannot be deleted.

4 BLOCK LINK EXAMPLES

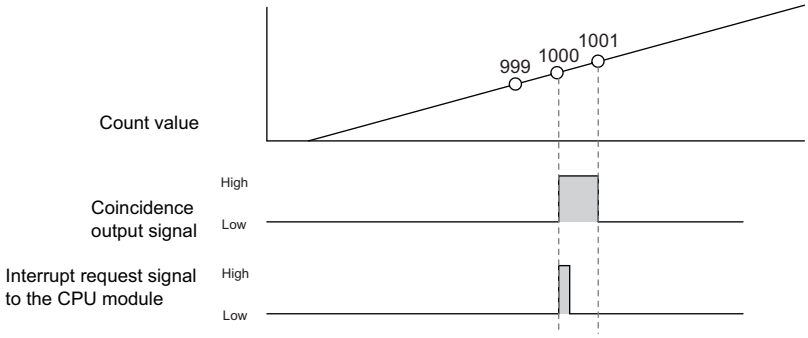
Various controls can be performed with the combinations of various blocks. This chapter shows link examples to perform the following controls.

Control name	Description	Reference
Coincidence detection	Turns on the output of the multi function counter block at coincidence detection.	☞ Page 209 Coincidence Detection
One-shot timer	Turns on the output of the multi function counter block for the specified time period. After the specified time period, the output is turned off until the next event detection.	☞ Page 214 One-shot Timer
Event generation	Outputs an external input signal as an event signal. The output signal can be linked as an event signal of other multi function counter blocks.	☞ Page 218 Event Generation
Cam switch	Turns on or off the output when a count value within the upper and lower limit value range becomes equal to or larger than the setting value of the corresponding steps.	☞ Page 220 Cam Switch
PWM output	Outputs a PWM wave with the specified cycle and duty ratio.	☞ Page 222 PWM Output
Fixed cycle output	Outputs one pulse at every specified cycle.	☞ Page 224 Fixed Cycle Output
Latch counter	Latches (holds) a count value.	☞ Page 226 Latch Counter
Ratio conversion	Outputs an input signal multiplied with the set ratio (x/y).	☞ Page 229 Ratio Conversion
Pulse measurement	Measures the ON width and OFF width of an input pulse.	☞ Page 232 Pulse Measurement
A/D conversion value logging	Outputs 1000 fixed cycle pulses of 100μs width from one input pulse.	☞ Page 235 A/D conversion value logging

4.1 Coincidence Detection

The output of the multi function counter block is turned on when a count value matches with a compare value. The compare value is required to be set in advance as a coincidence detection value.

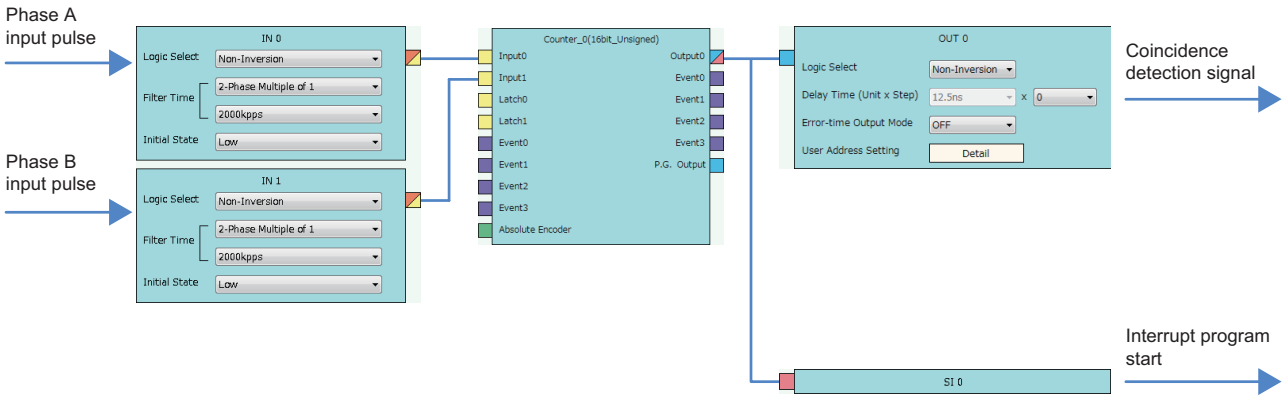
The following describes a link and parameter example of coincidence detection. With the example setting, an external output signal turns on when the count value becomes 1000 and an interrupt request is sent to the CPU module. Note that this link example is for when a 16-bit unsigned multi function counter block is used.



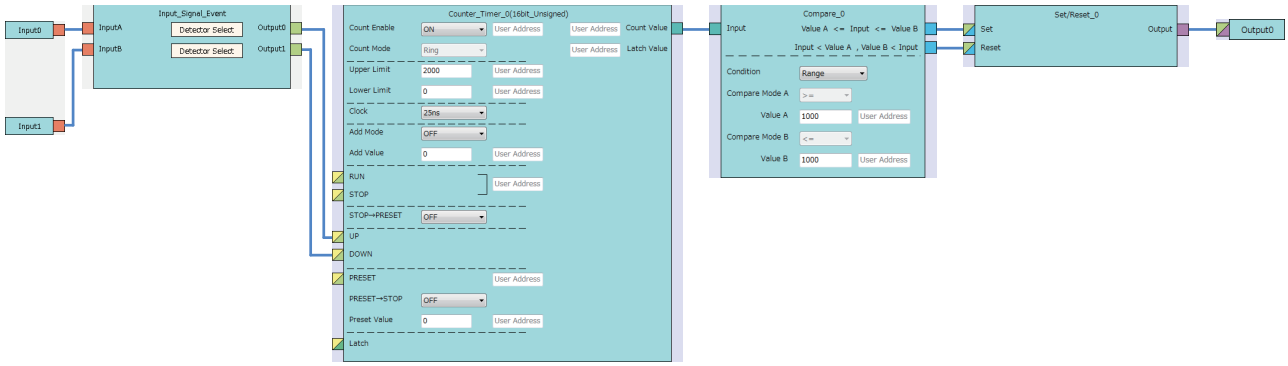
Link and parameter

The following shows a link example of the hardware logic outline window and a link example and parameter setting example of the multi function counter block detail window.

■Hardware logic outline window



■Multi function counter block detail window

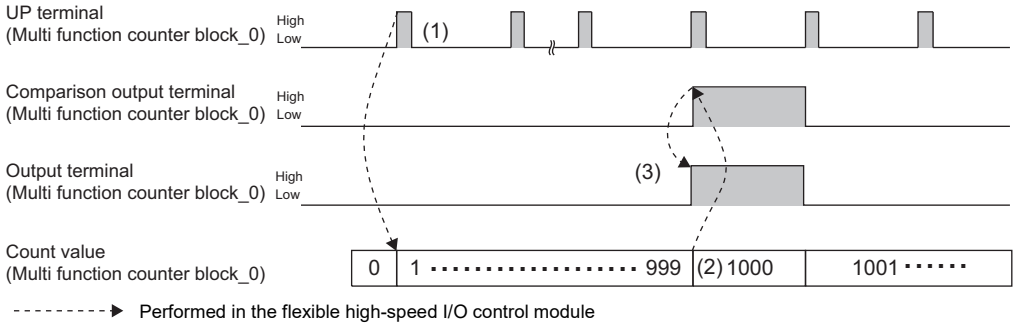


Block	Variable name	Setting value	Description
Input 0	—	—	—
Input 1	—	—	—
Input_Signal_Event	Detector Select	1/2-Phase Multiple of 1(CW/CCW)+	Set this parameter to detect an addition pulse.
	Detector Select	2-Phase Multiple of 1-	Set this parameter to detect a subtraction pulse.
Counter_Timer_0	Count Enable	ON	Set this parameter to "ON".
	Count Mode	Ring	Set this parameter according to the control.
	Upper Limit	2000	Set this parameter according to the control.
	Lower Limit	0	Set this parameter according to the control.
	Clock	25ns	Set this parameter to "25ns". (To prevent input pulses from missing)
	Add Mode	OFF	Set this parameter to "OFF", because the addition mode is not used in this link example.
	Add Value	0	
Compare_0	Condition	Range	Set this parameter to "Range".
	Compare Mode A	>=	Set this parameter to ">=".
	Compare Mode B	<=	Set this parameter to "<=".
	Compare Value A	1000	Set a value used as a "coincidence detection value".
	Compare Value B	1000	Set a value used as a "coincidence detection value".
Set/Reset_0	—	—	—
Output 0	—	—	—

The items other than the above need not to be set.

Operation

The following shows operation of coincidence detection when the count value becomes 1000.

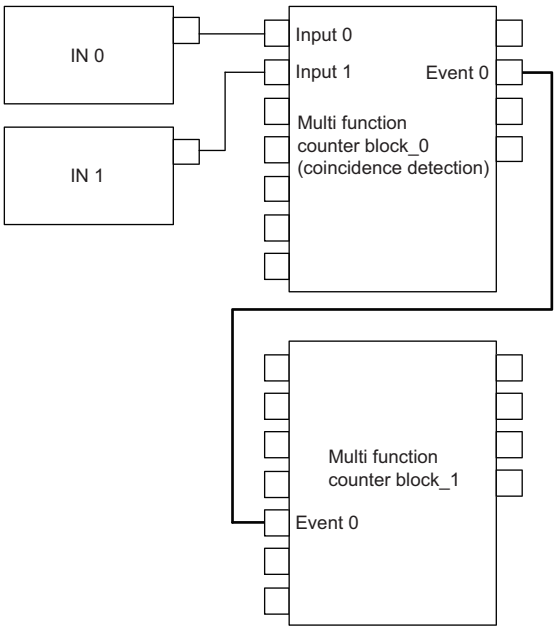


No.	Description
(1)	The count value is increased by one every time the UP terminal of the Counter_0 block turns to High.
(2)	When the count value becomes 1000 (Coincidence detection value), the compare output terminal turns to High.
(3)	Output 0 turns to High and to Low, and an interrupt request is sent to the CPU module.

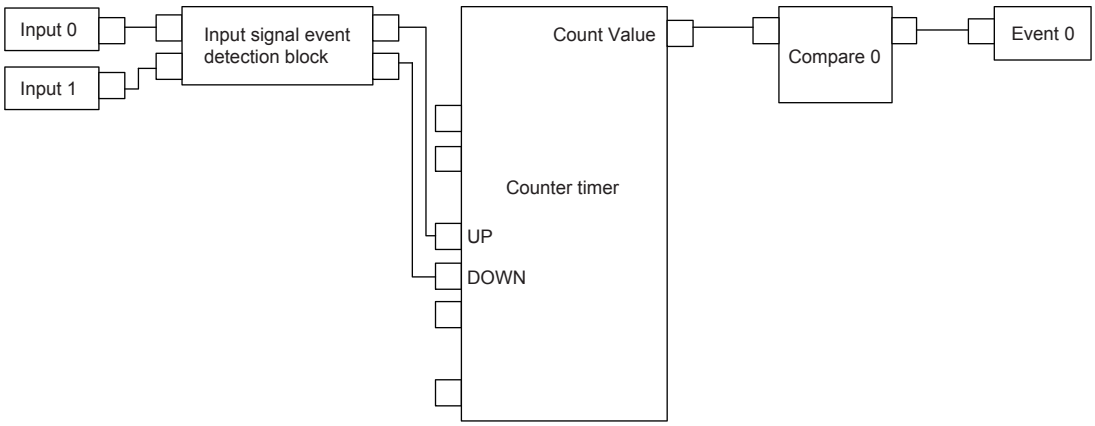
Point

The multi function counter block with a link for coincidence detection can be used as a trigger of other multi function counter blocks. The following shows link examples of the hardware logic outline window and multi function counter block _0 detail window.

• Hardware logic outline window



• Multi function counter block _0 detail window (coincidence detection)

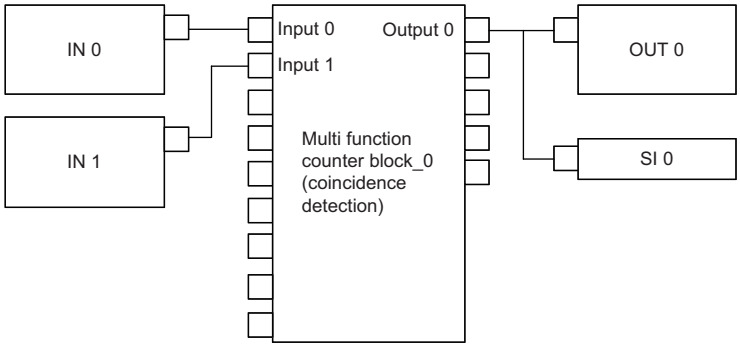


The multi function counter block_0 counts external input signals. When the count value becomes equal to the compare value, an event signal is output to the multi function counter block_1.

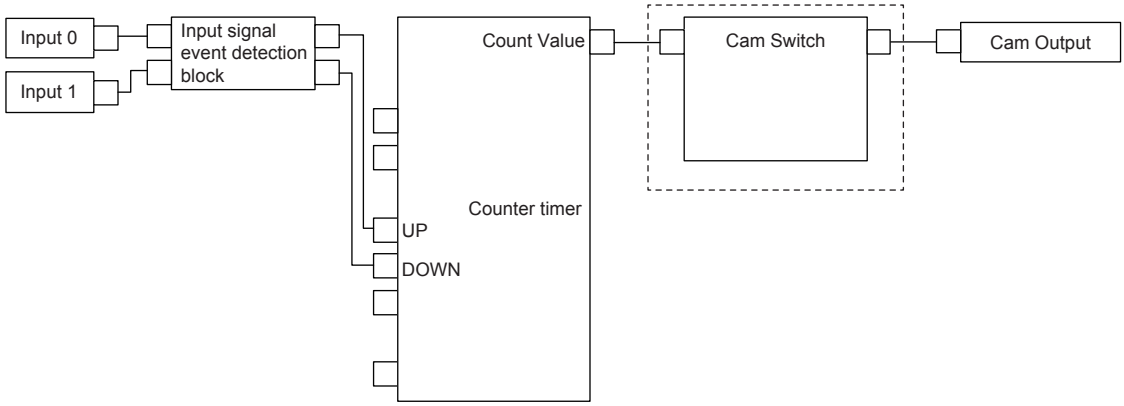
Precautions for using 32-bit multi function counter blocks

Coincidence detection control can be set even with 32-bit multi function counter blocks. However, usable blocks are different.
The following shows link examples of the hardware logic outline window and multi function counter block detail window.

- Hardware logic outline window



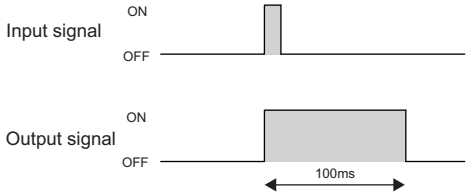
- Multi function counter block _0 detail window (coincidence detection)



Create the hardware logic by using a cam switch block, as shown in the dotted line frame.

4.2 One-shot Timer

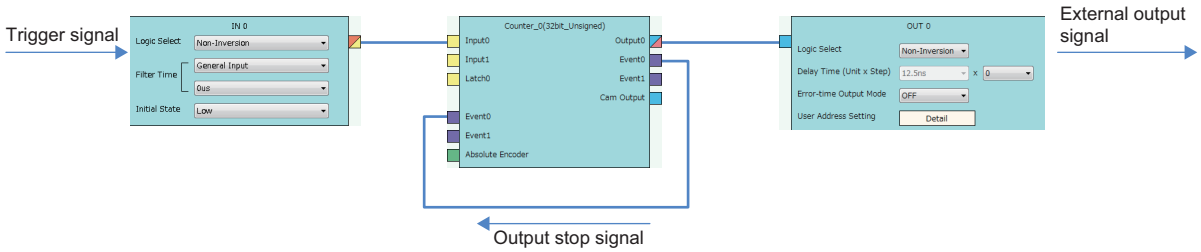
The output signal is turned on for the specified time period using an input signal as a trigger.
The following describes a link and parameter example of a one-shot timer. With the example setting, an output is turned on for 100ms using an external input signal as a trigger. Note that this link example is for when a 32-bit unsigned multi function counter block is used.



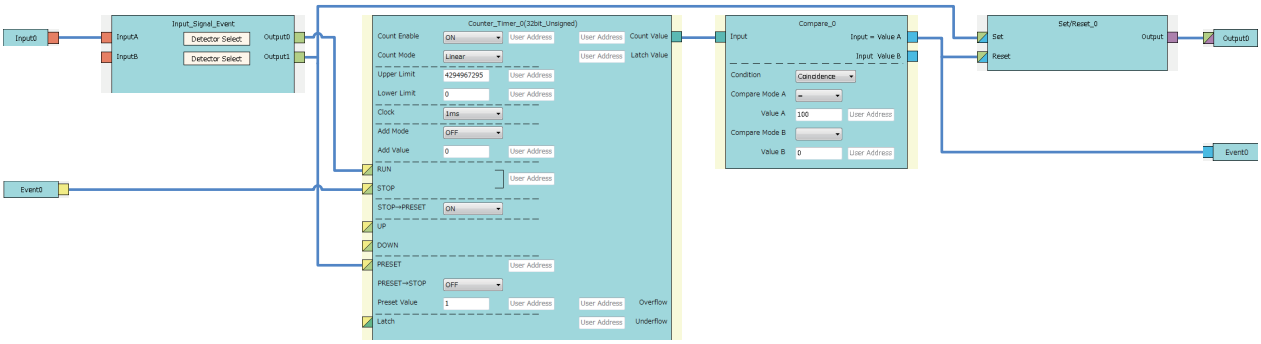
Link and parameter

The following shows a link example of the hardware logic outline window and a link example and parameter setting example of the multi function counter block detail window.

■Hardware logic outline window



■Multi function counter block detail window



Block	Variable name	Setting value	Description
Input 0	—	—	—
Event 0	—	—	—
Input_Signal_Event	Detector Select	User Setting (ON only in A: High)	Set this parameter to detect the phase A input signal.
	Detector Select	User Setting (ON only in A: Rise)	
Counter_Timer_0	Count Enable	ON	Set this parameter to "ON".
	Count Mode	Linear	Set this parameter according to the control.
	Upper Limit	4294967295	Set this parameter according to the control.
	Lower Limit	0	Set this parameter according to the control.
	Clock	1ms	Set this parameter according to the control.*1
	Add Mode	OFF	Set this parameter according to the control.*2
	Add Value	0	
	STOP → PRESET	ON	Set this parameter to "ON", to perform a preset when a stop event occurs.
	PRESET → STOP	OFF	Set this parameter to "OFF".
	Preset Value	1	Set this parameter according to the control.*1
Compare_0	Condition	Coincidence	Set this parameter to "Coincidence".
	Compare Mode A	=	Set this parameter to "=".
	Compare Value A	100	Set this parameter according to the control.*1
Set/Reset_0	—	—	—
Output 0	—	—	—

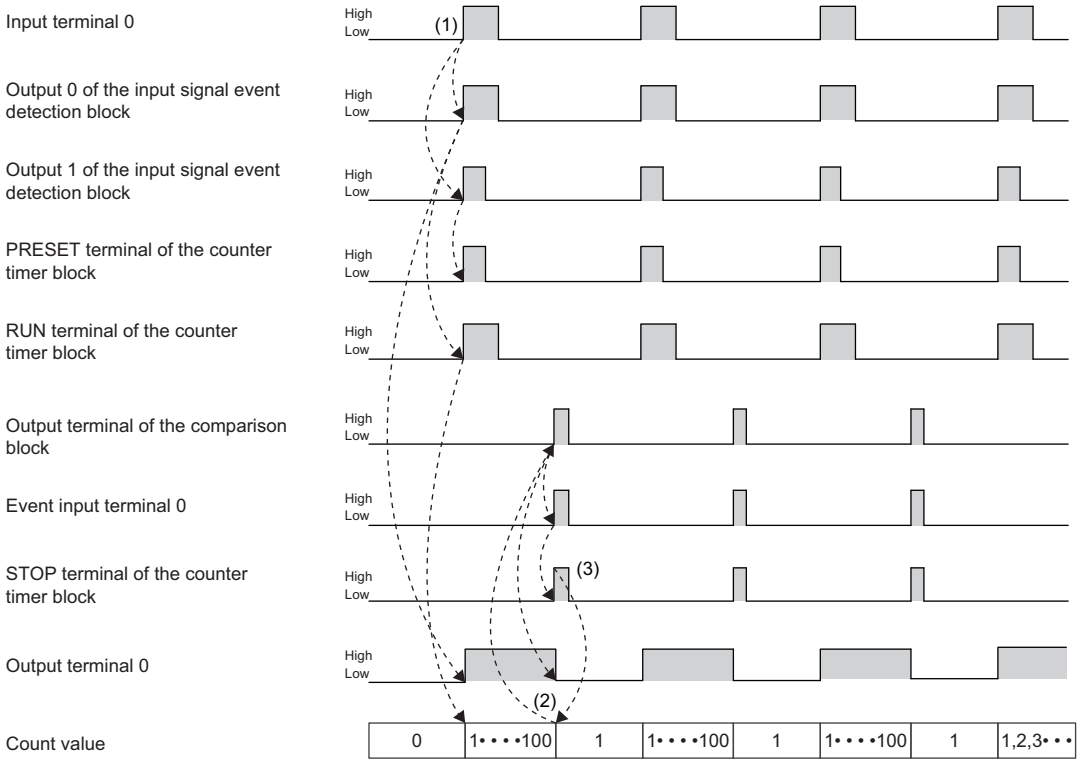
*1 Output ON time = Clock cycle × (Compare value - Preset value + 1)

*2 In the addition mode, the output ON time calculated from the following is applied.
Output ON time = (Clock cycle × (Compare value - Preset value + 1)) ÷ Addition value

The items other than the above need not to be set.

Operation

The following shows operation of the one-shot timer when a trigger signal is input.

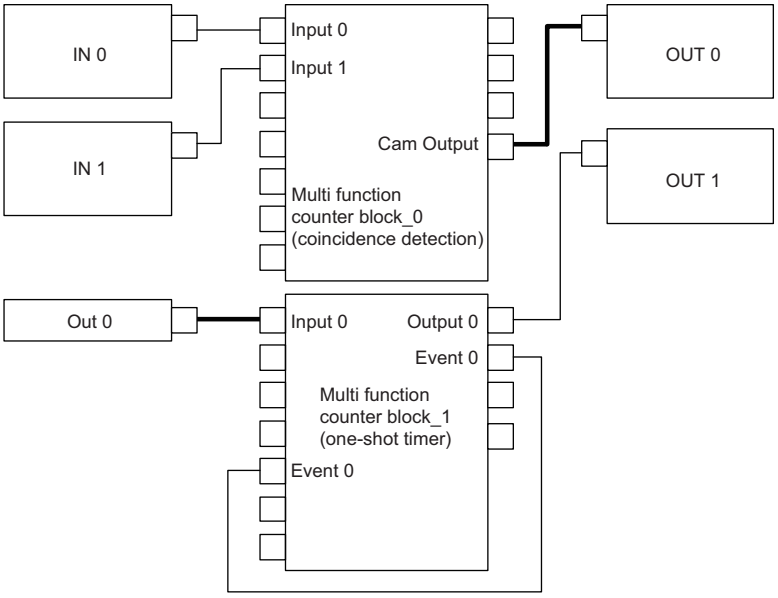


No.	Description
(1)	When a signal is input to the input terminal 0, the signal is output from the output terminal 1 of the input signal event detection block. The output signal turns the PRESET terminal of the counter timer block to High and to Low, performing the preset. (Count value: 0) The signal output from the output terminal 0 of the input signal event detection block turns the RUN terminal of the counter timer block to High, starting counting per clock cycle. In addition, outputs from the output terminal 0 start. (The RUN terminal has a lower priority than the PRESET terminal. Thus, counting per clock cycle starts after the preset.)
(2)	When the count value becomes 100, the Output terminal of the comparison block turns to High. The event output terminal 0, event input terminal 0, and the STOP terminal of the counter timer block turn to High in the described order. In addition, outputs from the output terminal 0 stop.
(3)	When the STOP terminal of the counter timer block turns to High, counting per clock cycle stops.

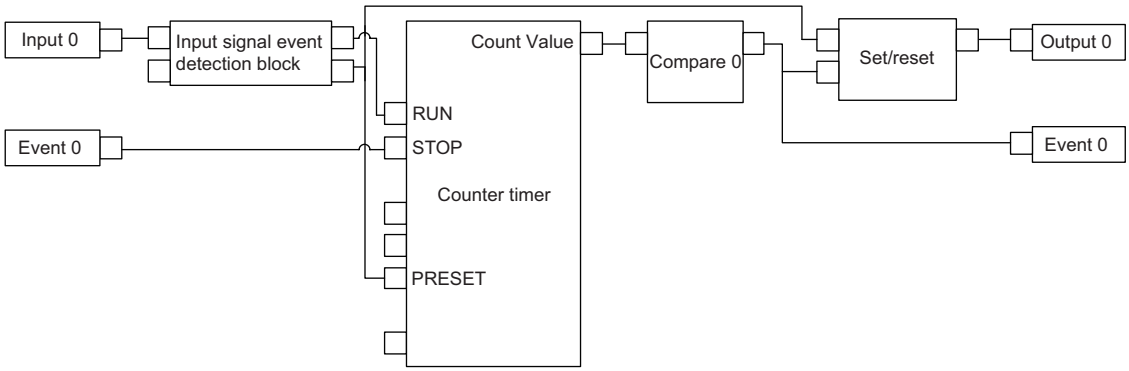
Point

If the multi function counter block with a link for one-shot timer is used together with other multi function counter blocks with links for coincidence detection or other functions, any output ON time for external devices can be set. The following shows link examples of the hardware logic outline window and multi function counter block_1 detail window.

• Hardware logic outline window



• Multi function counter block_1 detail window (one-shot timer)



4.3 Event Generation

An external input signal is output as an event signal.
Only two event signals can be generated with one multi function counter block. If several multi function counter blocks are used together, three or more event signals can be used with one multi function counter block.
The following describes a link and parameter example of event generation. With the example setting, an event is generated under the following condition and input to the Event terminal of another multi function counter block as an event signal.
The following describes a link and parameter example of event generation. With the example setting, an event is generated under the following condition and input to the Event terminal of another multi function counter block as an event signal.

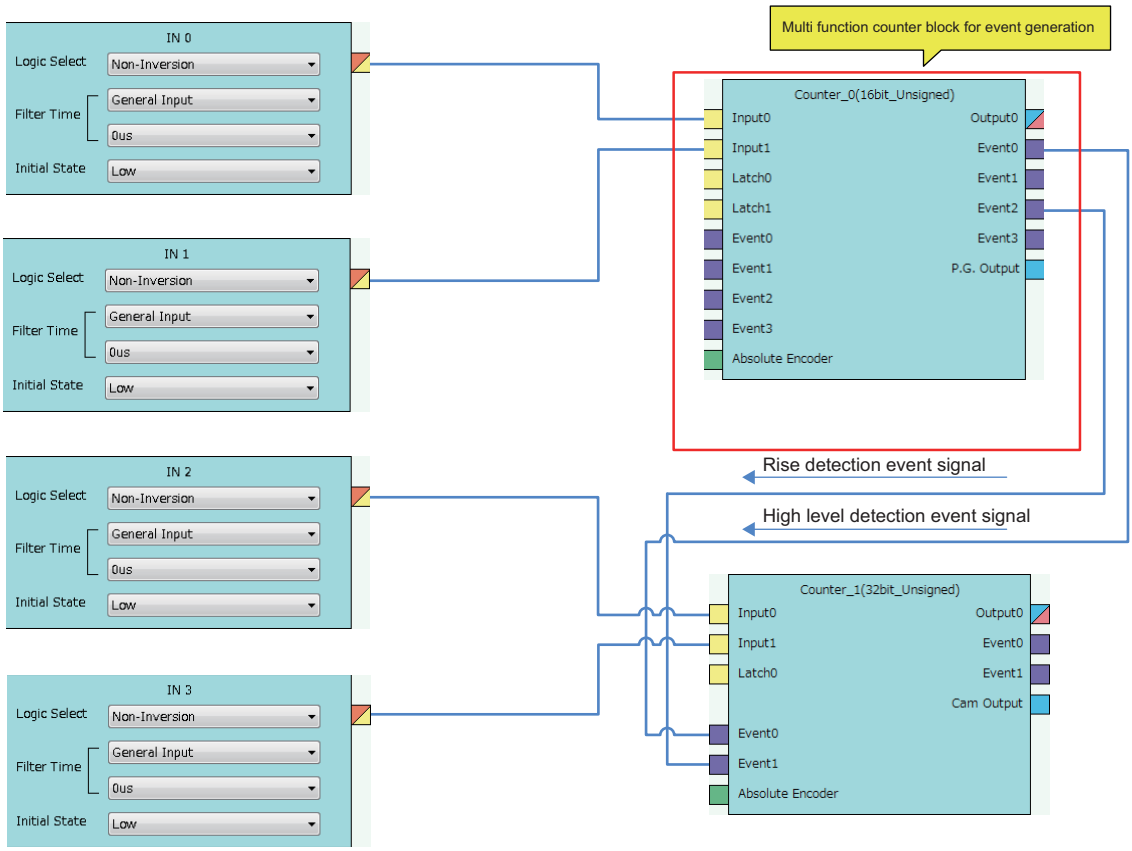
- A signal output from the Event 0 terminal of the multi function counter block _0: Rise of the phase A input
- A signal output from the Event 2 terminal of the multi function counter block _0: High of the phase B input

Note that this link example is for when a 16-bit unsigned multi function counter block is used. Event generation is supported only by 16-bit multi function counter blocks.

Link and parameter

The following shows a link example of the hardware logic outline window and a link example and parameter setting example of the multi function counter block detail window.

■Hardware logic outline window

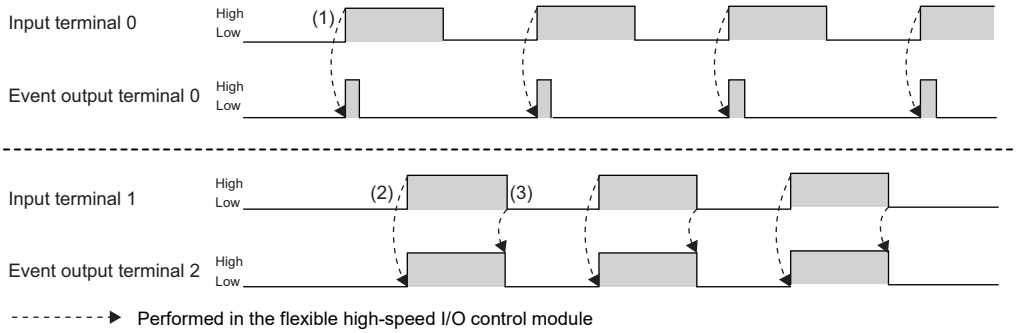


■Multi function counter block detail window

Block	Variable name	Setting value	Description
Input 0	—	—	—
Input 1	—	—	—
Input_Signal_Event	Detector Select	User Setting (ON only in A: Rise)	Set conditions to generate an event signal.
	Detector Select	User Setting (ON only in B: High)	
Event 0	—	—	—
Event 2	—	—	—

Operation

The following shows operation where the input signal 1 turns to High when a rise of the input signal 0 is detected.



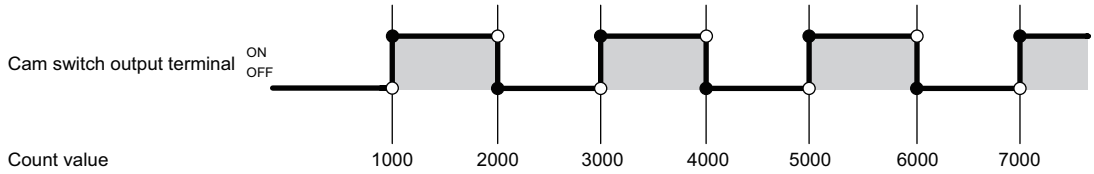
No.	Description
(1)	When a rise of the input terminal 0 is detected, the event output terminal 0 turns to High and to Low.
(2)	When a rise of the input terminal 1 is detected, the event output terminal 2 turns to High.
(3)	When a fall of the input terminal 1 is detected, the event output terminal 2 turns to Low.

4.4 Cam Switch

The preset output status of the coincidence output range is compared with a count value to switch on or off the output of the multi function counter block at each step.

The following is an example assuming that an output is switched by seven steps as follows. Note that this link example is for when a 32-bit unsigned multi function counter block is used. (The cam switch is not supported by 16-bit multi function counter blocks.)

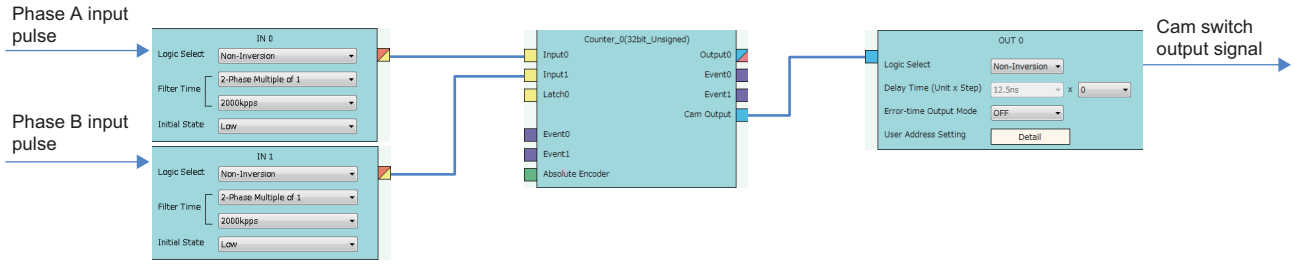
Step No.	Count value	Output
—	0 (lower limit value) to 999	Low
0	1000 to 1999	High
1	2000 to 2999	Low
2	3000 to 3999	High
3	4000 to 4999	Low
4	5000 to 5999	High
5	6000 to 6999	Low
6	7000 to 10000 (upper limit value)	High



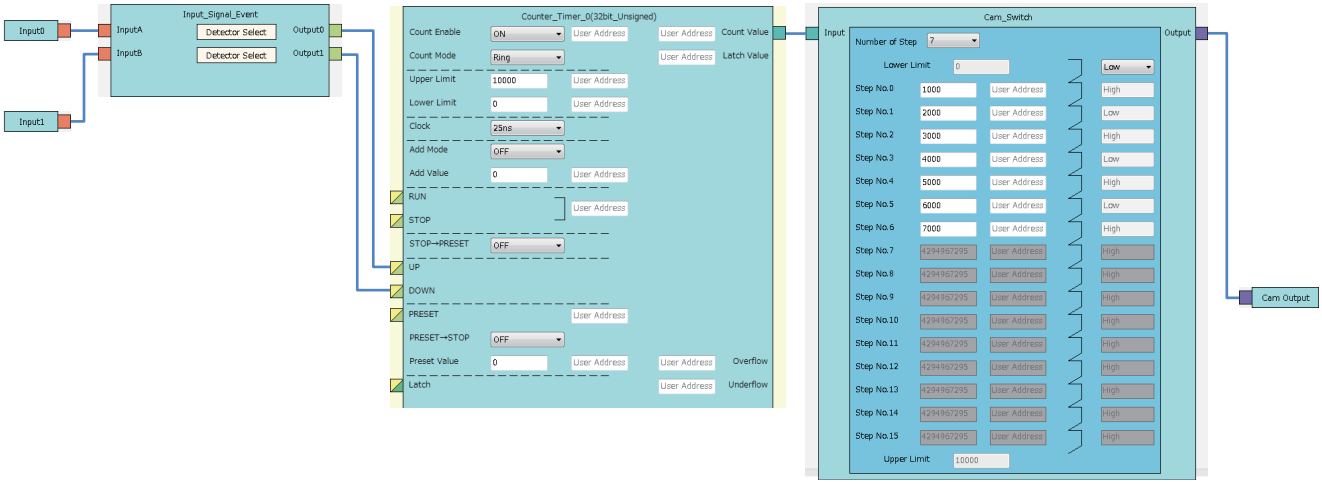
Link and parameter

The following shows a link example of the hardware logic outline window and a link example and parameter setting example of the multi function counter block detail window.

■Hardware logic outline window



■Multi function counter block detail window

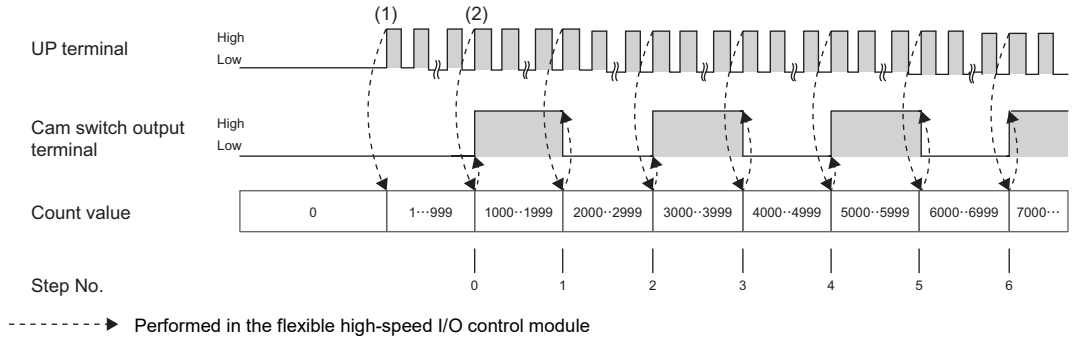


Block	Variable name	Setting value	Description
Input 0	—	—	—
Input 1	—	—	—
Input_Signal_Event	Detector Select	1/2-Phase Multiple of 1(CW/CCW)+	Set this parameter to detect an addition pulse.
	Detector Select	2-Phase Multiple of 1-	Set this parameter to detect a subtraction pulse.
Counter_Timer_0	Count Enable	ON	Set this parameter to "ON".
	Count Mode	Ring	Set this parameter according to the control.
	Upper Limit	10000	Set this parameter according to the control.
	Lower Limit	0	Set this parameter according to the control.
	Clock	25ns	Set this parameter to "25ns". (To prevent input pulses from missing)
	Add Mode	OFF	Set this parameter according to the control.
	Add Value	0	
Cam_Switch	Number of Step	7	Set "7", which is the number of steps of the cam switch, to enable the step No.0 to 6.
	Lower Limit Output State	Low	Set this parameter according to the control.
	Step No.0	1000	Set a count value at which the output status of the subsequent step is enabled.
	Step No.1	2000	
	Step No.2	3000	
	Step No.3	4000	
	Step No.4	5000	
	Step No.5	6000	
	Step No.6	7000	
Cam Output	—	—	—

The items other than the above need not to be set.

Operation

The following shows operation where a count value is compared with the setting value of each step.

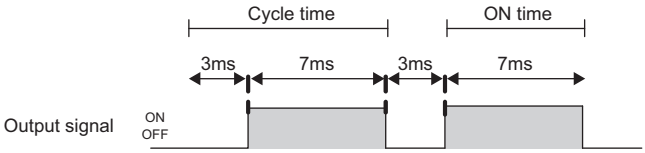


No.	Description
(1)	The count value is increased by one every time the UP terminal turns to High.
(2)	When the count value reaches the setting value of the step No.0, an external output is inverted. Also, every time the count value reaches each setting value of the step No.1 or later, an external output is inverted.

4.5 PWM Output

The multi function counter block outputs a PWM wave with a specified duty ratio.

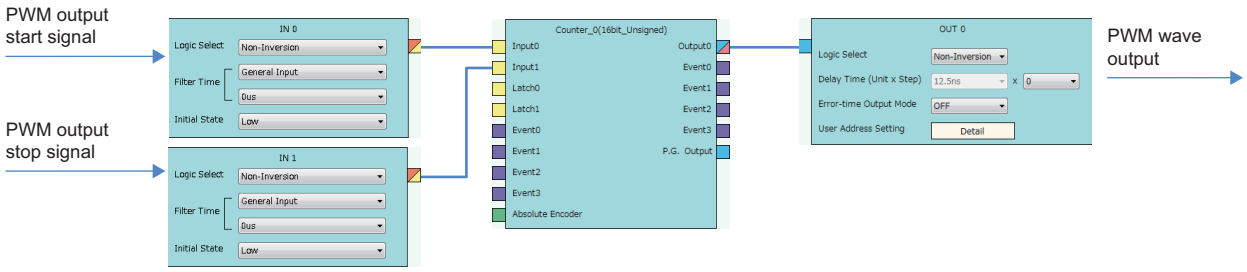
The following describes a link and parameter example of a PWM output. With the example setting, when a rise of the input signal 0 is detected, output of the following PWM wave starts, and when a rise of the input signal 1 is detected, the output stops. Note that this link example is for when a 16-bit unsigned multi function counter block is used.



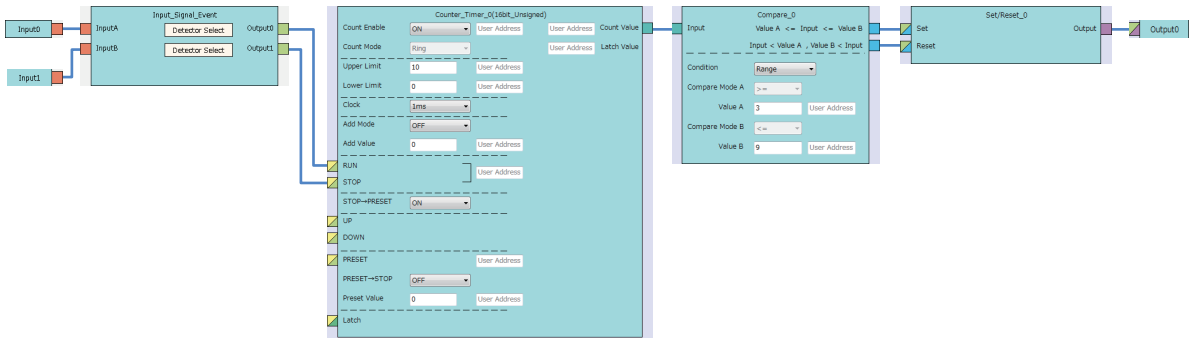
Link and parameter

The following shows a link example of the hardware logic outline window and a link example and parameter setting example of the multi function counter block detail window.

■Hardware logic outline window



■Multi function counter block detail window



Block	Variable name	Setting value	Description
Input 0	—	—	—
Input 1	—	—	—
Input_Signal_Event	Detector Select	User Setting (ON only in A: Rise)	Set conditions to detect a PWM output start signal.
	Detector Select	User Setting (ON only in B: Rise)	Set conditions to detect a PWM output stop signal.
Counter_Timer_0	Count Enable	ON	Set this parameter to "ON".
	Count Mode	Ring	Set this parameter to "Ring".
	Upper Limit	10	Set this parameter according to the control.*1
	Lower Limit	0	Set this parameter according to the control.*1
	Clock	1ms	Set this parameter according to the control.*1*2
	Add Mode	OFF	Set this parameter according to the control.*3
	Add Value	0	
	STOP → PRESET	ON	Set this parameter to "ON", to perform a preset when a stop event occurs.
	Preset Value	0	Set the same value as that in "Lower Limit".
Compare_0	Condition	Range	Set this parameter to "Range".
	Compare Mode A	>=	Set this parameter to ">=".
	Compare Mode B	<=	Set this parameter to "<=".
	Compare Value A	3	Set this parameter according to the control.*2
	Compare Value B	9	Set this parameter according to the control.*2
Output 0	—	—	—

*1 Cycle time = (Upper limit value - Lower limit value) × Clock cycle

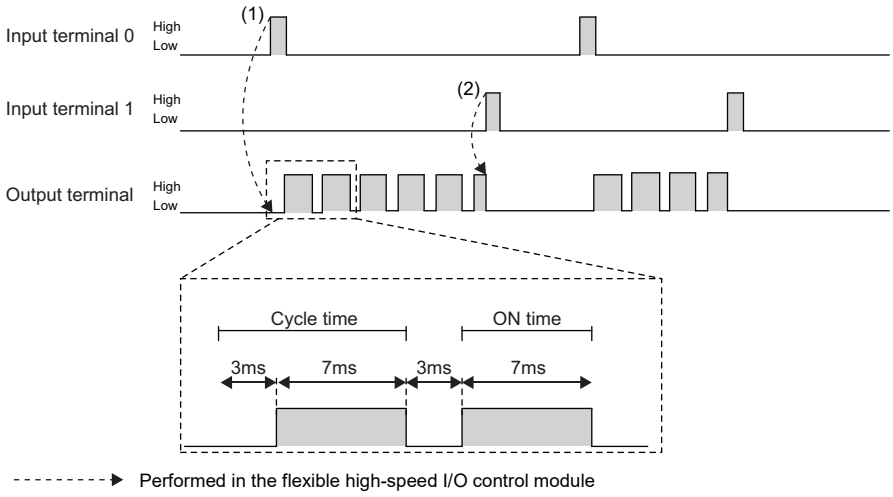
*2 ON time = Cycle time - (Compare value A × Clock cycle) Set the value of compare value A + ON time for the compare value B.

*3 In the addition mode, the cycle time calculated from the following is applied.
Cycle time = (Upper limit value - Lower limit value) ÷ Addition value × Clock cycle

The items other than the above need not to be set.

Operation

The following shows operation of when the PWM output start signal (input terminal 0) is input and when the PWM output stop signal (input terminal 1) is input.

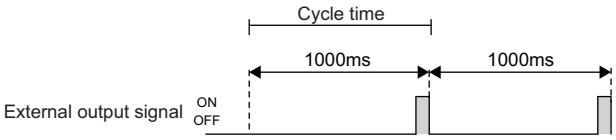


No.	Description
(1)	When a signal is input to the input terminal 0, output of the PWM wave starts.
(2)	When a signal is input to the input terminal 1, the output of the PWM wave stops.

4.6 Fixed Cycle Output

One pulse is output from the multi function counter block at specified cycle time.

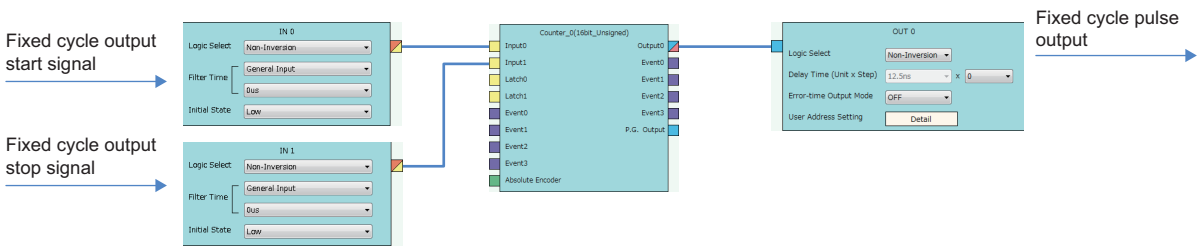
The following describes a link and parameter example of a fixed cycle output. With the example setting, when a rise of the input signal 0 is detected, output of the following fixed cycle pulse starts, and when a rise of the input signal 1 is detected, the output stops. Note that this link example is for when a 16-bit unsigned multi function counter block is used.



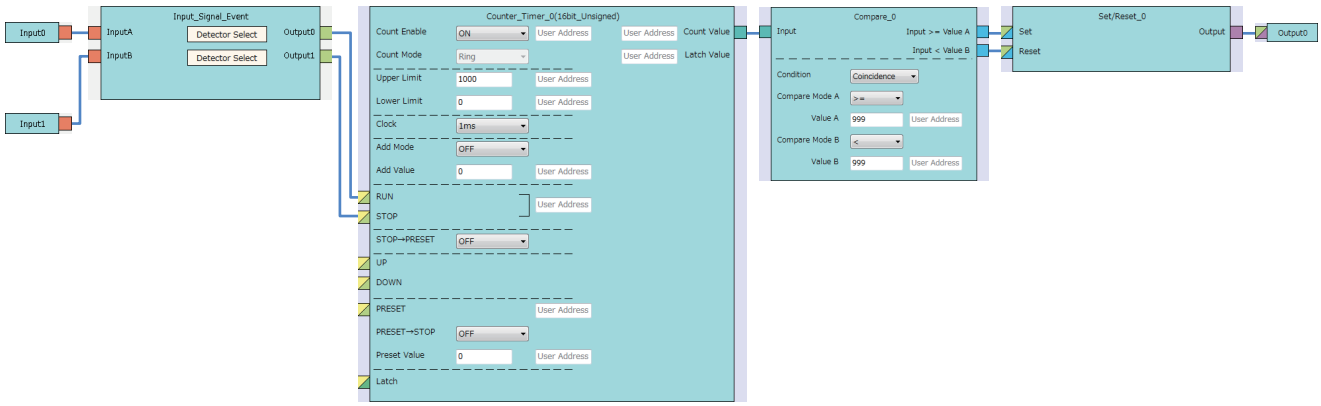
Link and parameter

The following shows a link example of the hardware logic outline window and a link example and parameter setting example of the multi function counter block detail window.

■Hardware logic outline window



■Multi function counter block detail window



Block	Variable name	Setting value	Description
Input 0	—	—	—
Input 1	—	—	—
Input_Signal_Event	Detector Select	User Setting (ON only in A: Rise)	Set conditions to detect a fixed cycle output start signal.
	Detector Select	User Setting (ON only in B: Rise)	Set conditions to detect a fixed cycle output stop signal.
Counter_Timer_0	Count Enable	ON	Set this parameter to "ON".
	Count Mode	Ring	Set this parameter to "Ring".
	Upper Limit	1000	Set this parameter according to the control.*1
	Lower Limit	0	Set this parameter according to the control.*1
	Clock	1ms	Set this parameter according to the control.*1
	Add Mode	OFF	Set this parameter according to the control.*2
	Add Value	0	
	STOP → PRESET	OFF	Set this parameter to "OFF", not to perform a preset when a stop event occurs.
	Preset Value	0	Set the same value as that in "Lower Limit".
Compare_0	Condition	Coincidence	Set this parameter to "Coincidence".
	Compare Mode A	>=	Set this parameter to ">=".
	Compare Mode B	<	Set this parameter to "<".
	Compare Value A	999	Set a value of "Upper limit value of the counter timer block - 1".
	Compare Value B	999	Set a value of "Upper limit value of the counter timer block - 1".
Output 0	—	—	—

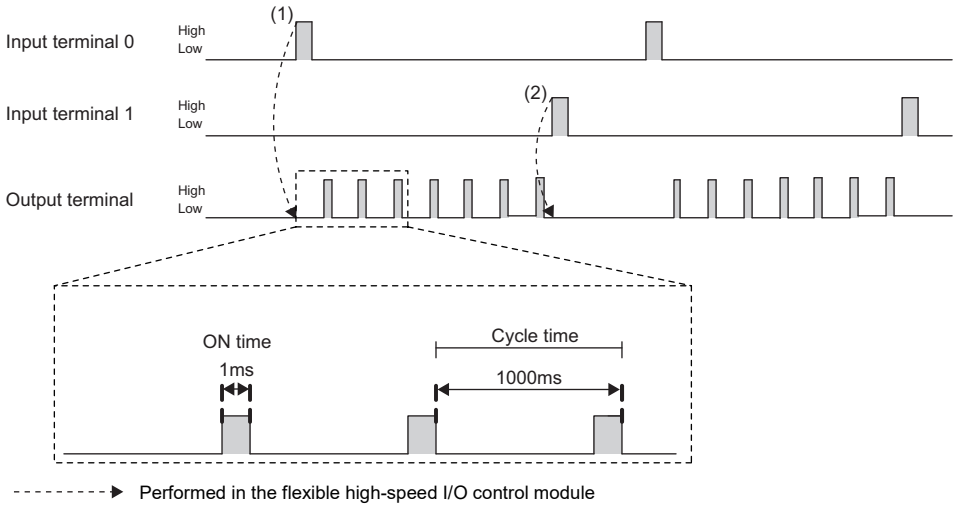
*1 Cycle time = (Upper limit value - Lower limit value) × Clock cycle

*2 In the addition mode, the cycle time calculated from the following is applied.
Cycle time = (Upper limit value - Lower limit value) ÷ Addition value × Clock cycle
However, when the count value and the compare value do not match each other with the set addition value, fixed cycle pulses are not output.

The items other than the above need not to be set.

Operation

The following shows operation of when the fixed cycle output start signal (input terminal 0) is input and when the fixed cycle output stop signal (input terminal 1) is input.



No.	Description
(1)	When a signal is input to the input terminal 0, pulse output starts. One pulse is output per cycle.
(2)	When a signal is input to the input terminal 1, the pulse output stops.

4.7 Latch Counter

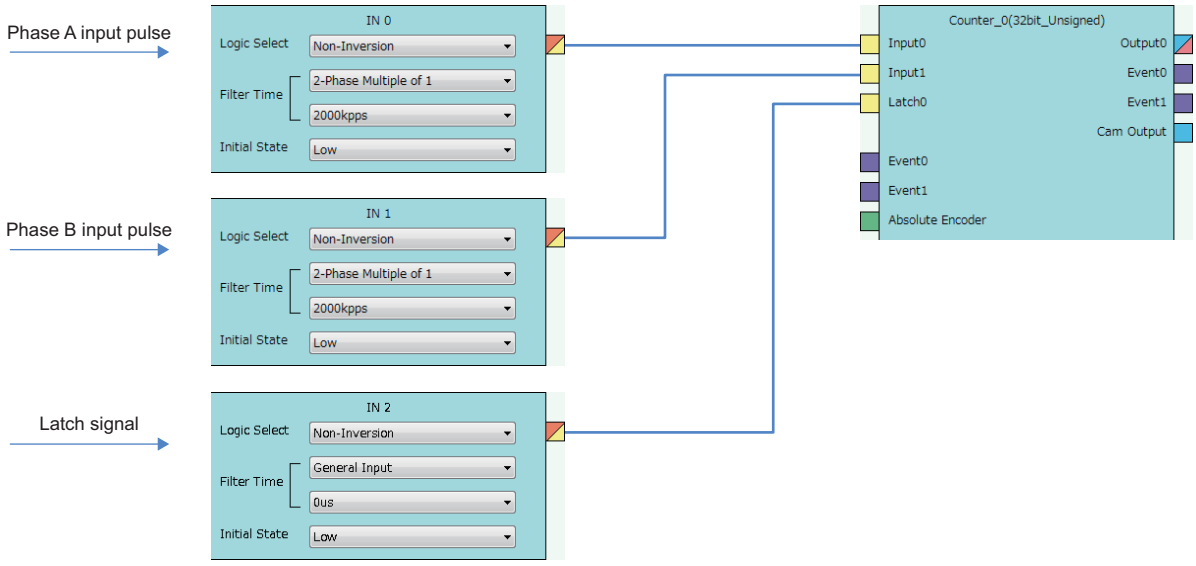
The current value is latched (held) as a latch count value.

The following describes a link and parameter example of a latch counter. With the example setting, when a rise of the latch input signal is detected, the current value is latched. Note that this link example is for when a 32-bit unsigned multi function counter block is used.

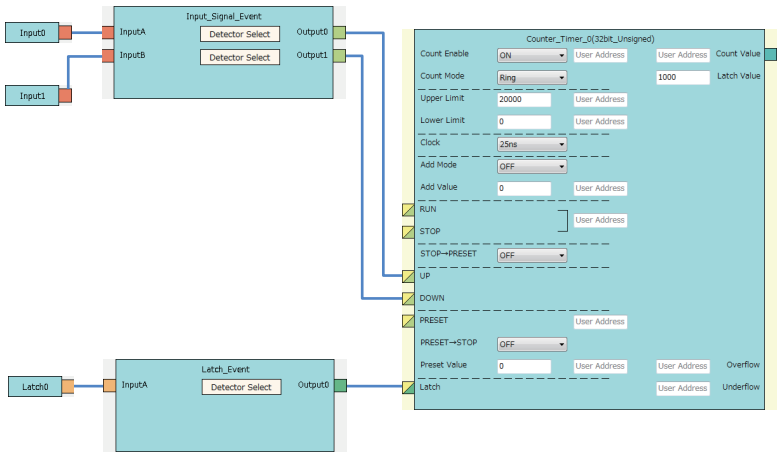
Link and parameter

The following shows a link example of the hardware logic outline window and a link example and parameter setting example of the multi function counter block detail window.

■Hardware logic outline window



■Multi function counter block detail window

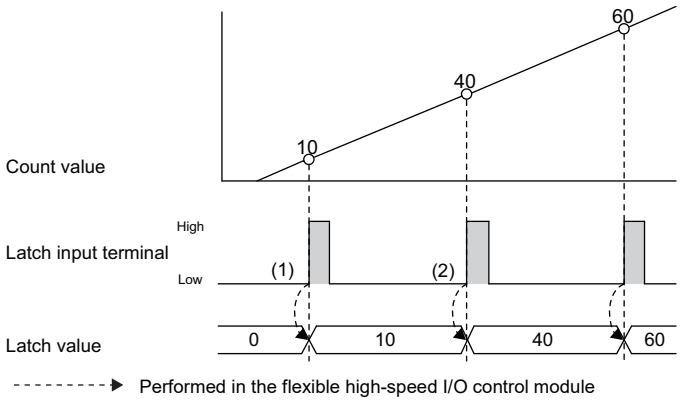


Block	Variable name	Setting value	Description
Input 0	—	—	—
Input 1	—	—	—
Latch 0	—	—	—
Input_Signal_Event	Detector Select	2-Phase Multiple of 1+	Set this parameter to detect an addition pulse.
	Detector Select	2-Phase Multiple of 1-	Set this parameter to detect a subtraction pulse.
Latch_Event	Detector Select	A: Rise	Set conditions to detect a latch signal.
Counter_Timer_0	Count Enable	ON	Set this parameter to "ON".
	Count Mode	Ring	Set this parameter according to the control.
	Upper Limit	20000	Set this parameter according to the control.
	Lower Limit	0	Set this parameter according to the control.
	Clock	25ns	Set this parameter to "25ns". (To prevent input pulses from missing)
	Add Mode	OFF	Set this parameter according to the control.
	Add Value	0	
	Preset Value	0	Set the same value as that in "Lower Limit".
	Latch Value (User Address)	1000	Assign a buffer memory address.

The items other than the above need not to be set.

Operation

The following shows operation of when a latch signal is input.

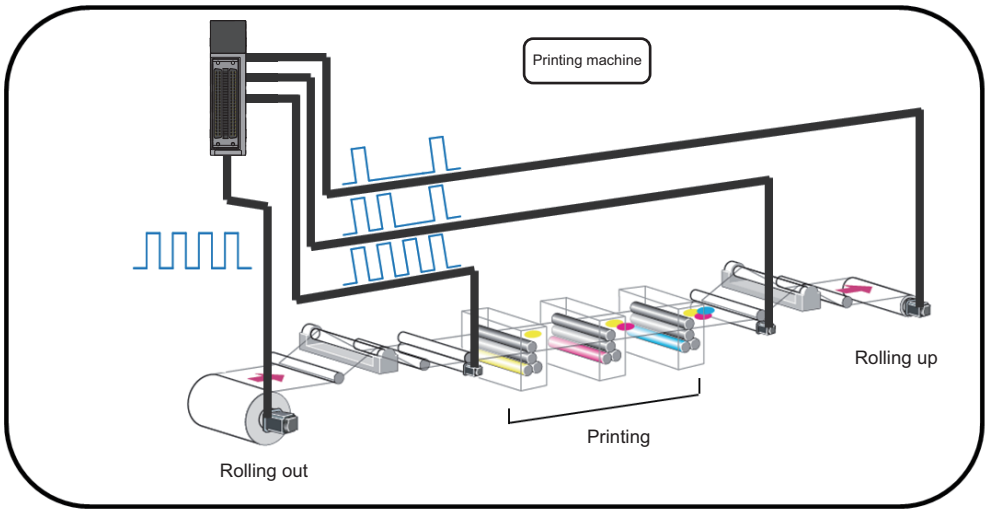


No.	Description
(1)	When a signal is input to the latch input terminal, the current count value is stored in the latch value.
(2)	When a signal is input to the latch input terminal again, the latch value is updated.

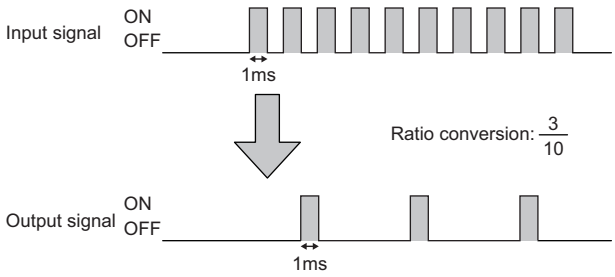
4.8 Ratio Conversion

An input signal multiplied with the set ratio (x/y) is output from the multi function counter block.
Ratio conversion enables the following controls.

- An encoder output of the motor at the unwinding part is input to the programmable controller to control the motor speed of lines with distribution output.
- The motor speed of multiple lines, for example, a printer, can be controlled.



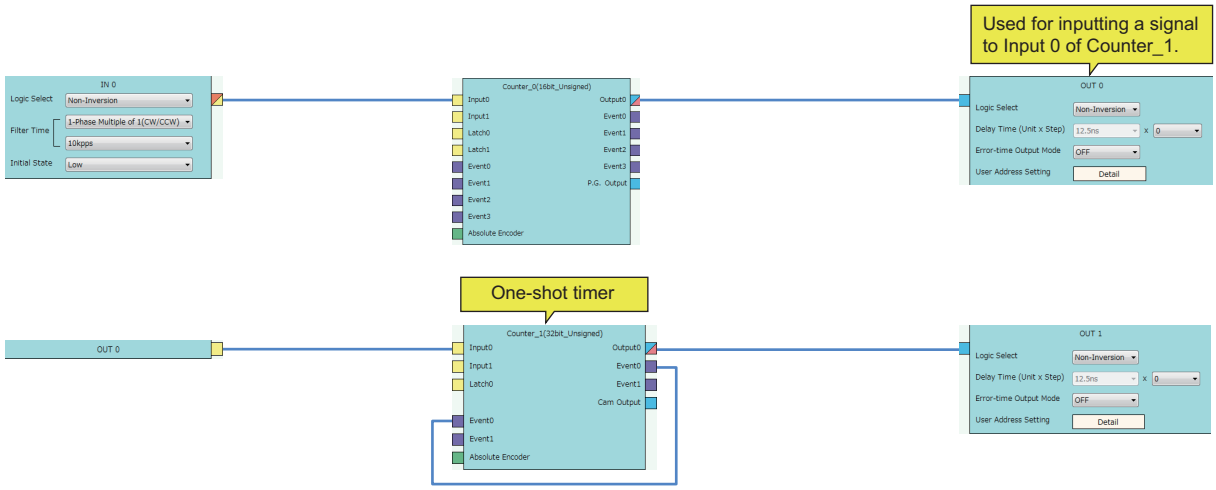
The ratio (x/y) used in ratio conversion must be "y ≥ x" when an input signal multiplied by x/y is output. Conversions with the ratio exceeding one time cannot be performed.
The following describes a link and parameter example of a ratio conversion. With the example setting, when a 1ms input signal is output for 1ms after multiplied with 3/10. Note that this link example is for when a 16-bit unsigned multi function counter block is used.



Link and parameter

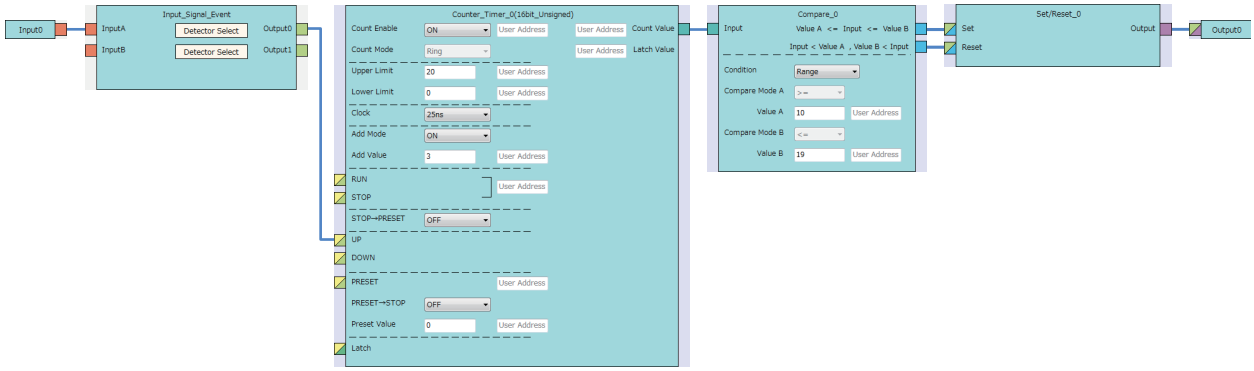
The following shows a link example of the hardware logic outline window and a link example and parameter setting example of the multi function counter block detail window.

■Hardware logic outline window



■Multi function counter block detail window

• Counter_0 (16bit_Unsigned)

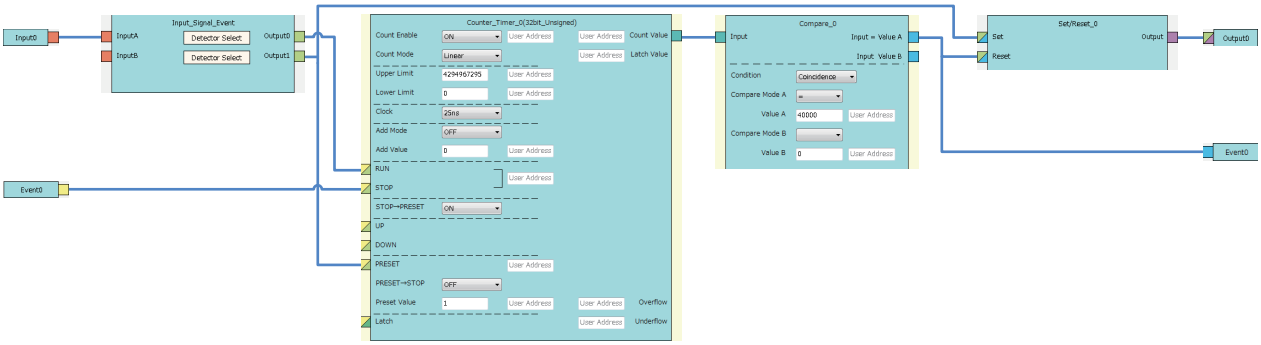


Block	Variable name	Setting value	Description
Input 0	—	—	—
Input_Signal_Event	Detector Select	User Setting (A: Rise ON, A: Fall ON)	Set "A: Rise" and "A: Fall" to "ON" and the others to "OFF".
	Detector Select	—	This parameter is not used in this link example.
Counter_Timer_0	Count Enable	ON	Set this parameter to "ON".
	Count Mode	Ring	Set this parameter to "Ring".
	Upper Limit	20	Set this parameter according to the control.*1
	Lower Limit	0	Set "0".
	Clock	25ns	Set this parameter to "25ns". (To prevent input pulses from missing)
	Add Mode	ON	Set this parameter to "ON".
	Add Value	3	Set this parameter according to the control.*1
Compare_0	Condition	Range	Set this parameter to "Range".
	Compare Mode A	>=	Set this parameter to ">=".
	Compare Mode B	<=	Set this parameter to "<=".
	Compare Value A	10	Set the value of "Upper limit value of the counter timer block ÷ 2".
	Compare Value B	19	Set the value of "Upper limit value of the counter timer block - 1".
Set/Reset_0	—	—	—
Output 0	—	—	—

*1 Conversion ratio = Addition value ÷ (Upper limit value ÷ 2)

The items other than the above need not to be set.

• Counter_1 (32bit_Unsigned)



Block	Variable name	Setting value	Description
Input 0	—	—	—
Input_Signal_Event	Detector Select	User Setting (ON only in A: High)	Set this parameter to detect the phase A input signal.
	Detector Select	User Setting (ON only in A: Rise)	
Counter_Timer_0	Count Enable	ON	Set this parameter to "ON".
	Count Mode	Linear	Set this parameter according to the control.
	Upper Limit	4294967295	Set this parameter according to the control.
	Lower Limit	0	Set this parameter according to the control.
	Clock	25ns	Set this parameter according to the control.*2
	Add Mode	OFF	Set this parameter according to the control.*3
	Add Value	0	
	STOP → PRESET	ON	Set this parameter to "ON", to perform a preset when a stop event occurs.
	PRESET → STOP	OFF	Set this parameter to "OFF".
	Preset Value	1	Set this parameter according to the control.*2
Compare_0	Condition	Coincidence	Set this parameter to "Coincidence".
	Compare Mode A	=	Set this parameter to "=".
	Compare Value A	40000	Set this parameter according to the control.*2
Set/Reset_0	—	—	—
Output 0	—	—	—
Event 0	—	—	—

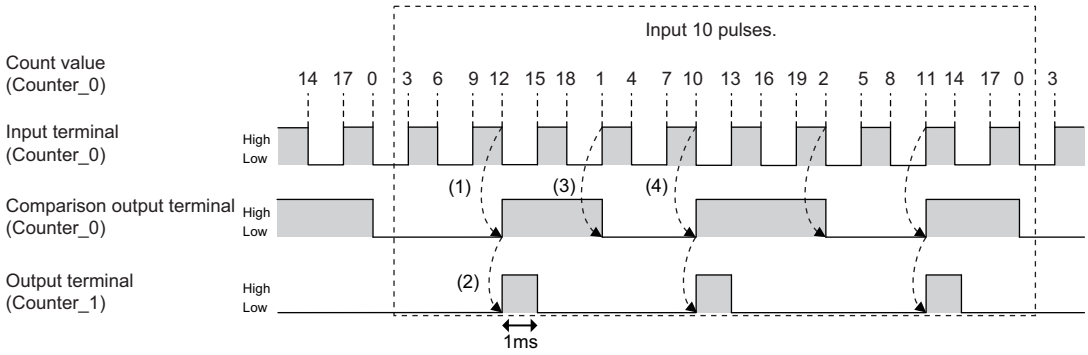
*2 Output ON time = Clock cycle × (Compare value - Preset value + 1)

*3 In the addition mode, the output ON time calculated from the following is applied.
Output ON time = (Clock cycle × (Compare value - Preset value + 1)) ÷ Addition value

The items other than the above need not to be set.

Operation

The following shows operation of ratio conversion.

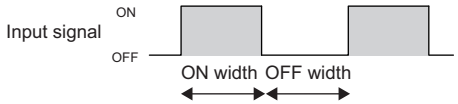


Performed in the flexible high-speed I/O control module

No.	Description
(1)	When the count value (Counter_0) becomes equal to or larger than the compare value, the compare output terminal (Counter_0) turns to High.
(2)	When the rise (Low to High) of the comparison output terminal (Counter_0) is detected, the output terminal (Counter_1) turns to High, and the output terminal (Counter_1) turns to Low in 1ms.
(3)	When the count value (Counter_0) falls below the compare value, the comparison output terminal (Counter_0) turns to Low.
(4)	When the count value (Counter_0) becomes equal to or larger than the compare value again, the comparison output terminal (Counter_0) turns to High. The operations (1) to (3) above are repeated, and signals of 3 pulses are output for inputs of 10 pulses.

4.9 Pulse Measurement

The ON width or OFF width of an input signal is measured.

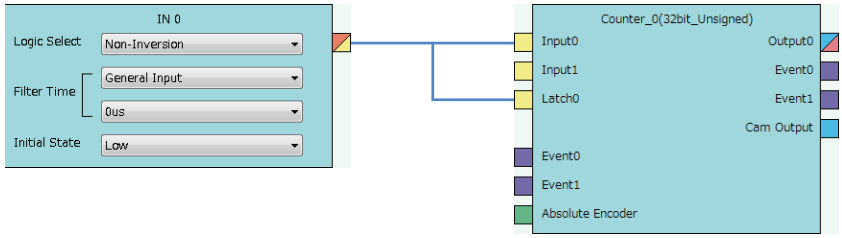


The following describes a link and parameter example of a pulse measurement. With the example setting, the ON width (latch value × 25ns) of the input signal is measured. Note that this link example is for when a 32-bit unsigned multi function counter block is used.

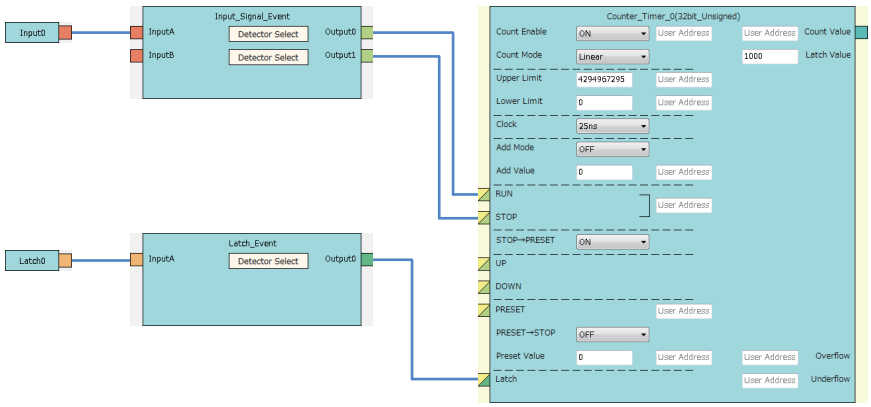
Link and parameter

The following shows a link example of the hardware logic outline window and a link example and parameter setting example of the multi function counter block detail window.

■Hardware logic outline window



■Multi function counter block detail window



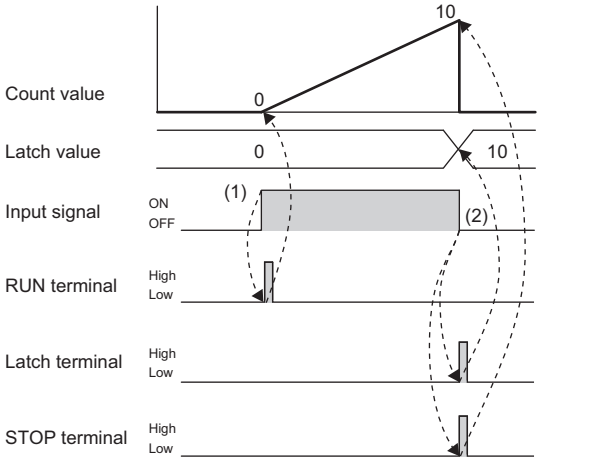
Block	Variable name	Setting value	Description
Input 0	—	—	—
Latch 0	—	—	—
Input_Signal_Event	Detector Select	User Setting (ON only in A: Rise)*1	Set ON only in "A: Rise".
	Detector Select	User Setting (ON only in A: Fall)*1	Set ON only in "A: Fall".
Latch_Event	Detector Select	User Setting (ON only in A: Fall)*1	Set ON only in "A: Fall".
Counter_Timer_0	Count Enable	ON	Set this parameter to "ON".
	Count Mode	Linear	Set this parameter to "Linear".
	Upper Limit	4294967295	Set "4294967295", to widen the pulse measurement width.
	Lower Limit	0	Set "0".
	Clock	25ns	Set this parameter to "25ns". (To prevent input pulses from missing)
	Add Mode	OFF	Set the addition mode to "OFF". (No addition in this example)
	Add Value	0	
	STOP → PRESET	ON	Set this parameter to "ON", to perform a preset when a stop event occurs.
	Preset Value	0	Set this parameter to 0.
	Latch Value (User Address)	1000	Assign a buffer memory address.

*1 Set the following to measure the OFF width.
"Detector Select" of "Input_Signal_Event" linked with the RUN terminal: ON only in A: Fall
"Detector Select" of "Input_Signal_Event" linked with the STOP terminal: ON only in A: Rise
"Detector Select" of "Latch_Event" linked with the Latch terminal: ON only in A: Rise

The items other than the above need not to be set.

Operation

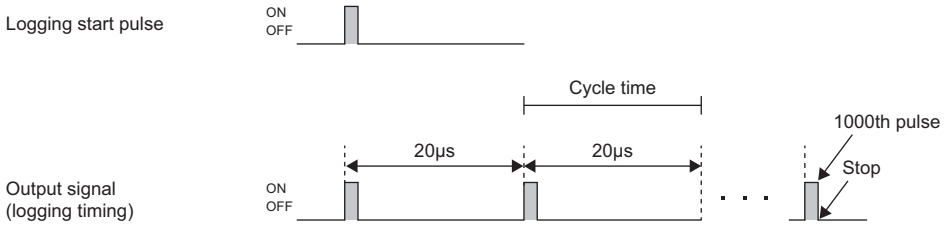
The following shows operation of pulse measurement.



No.	Description
(1)	When the input signal is turned on, the RUN terminal turns to High and counting per clock cycle starts.
(2)	When the input signal is turned off, the following two operations are simultaneously performed. • The Latch terminal turns to High and the current count value is latched. • The STOP terminal turns to High, counting per clock cycle stops and preset is performed.

4.10 A/D conversion value logging

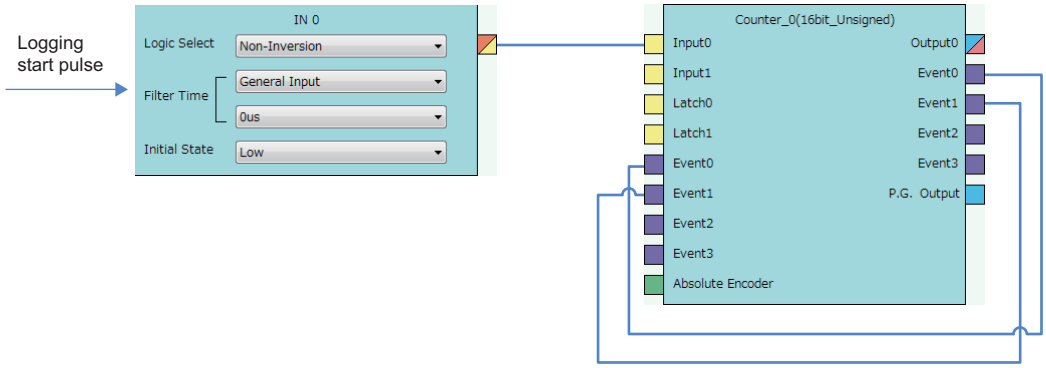
The following shows a wiring example applied to the A/D conversion value logging function.
When the logging start pulse is input, A/D conversion values are logged every 20μs, and logging stops when 1000 points have been logged.



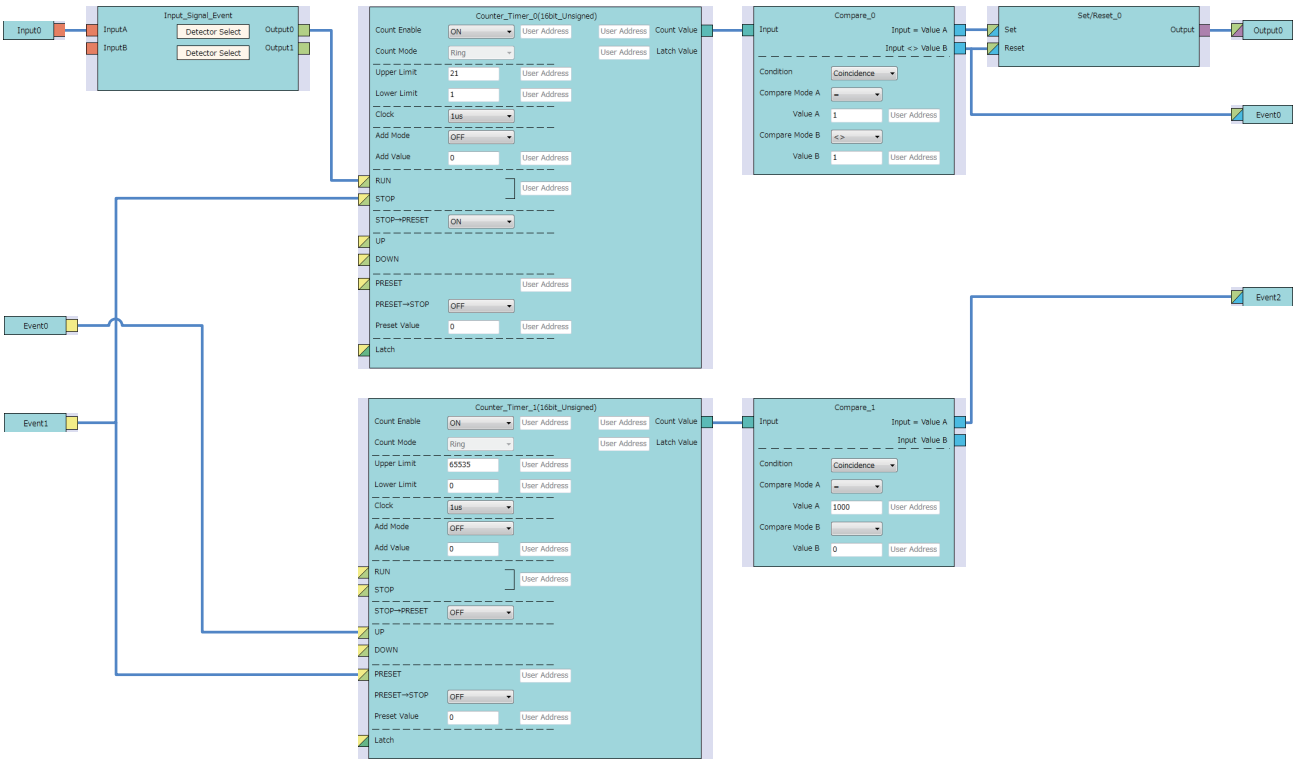
Link and parameter

The following shows a link example of the hardware logic outline window and a link example and parameter setting example of the multi function counter block detail window.

■Hardware logic outline window



■Multi function counter block detail window



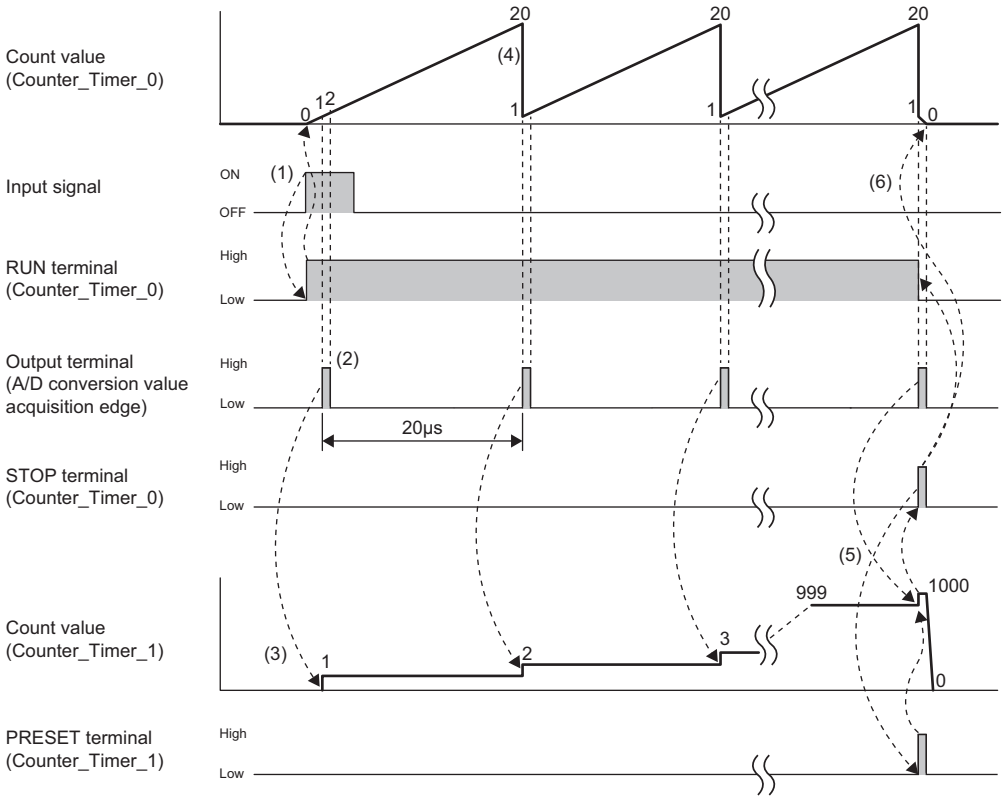
Block	Variable name	Setting value	Description
Input 0	—	—	—
Input_Signal_Event	Detector Select (upper side)	User Setting (ON only in A: Rise)	Set ON only in "A: Rise".
	Detector Select (lower side)	—	—
Counter_Timer_0	Count Enable	ON	Set this parameter to "ON".
	Upper Limit	21	Set the upper limit for a ring count with a 20μs cycle. Set "21" for the upper limit value and "1" for the lower limit value.
	Lower Limit	1	
	Clock	1μs	Set this parameter to "1μs".
	STOP→PRESET	ON	Set this parameter to "ON", to perform a preset when a stop event occurs.
	Preset Value	0	Set this parameter to 0.
Compare_0	Condition	Coincidence	Set this parameter to "Coincidence".
	Compare Mode A	=	Set this parameter to "=".
	Compare Mode B	<>	Set this parameter to "<>".
	Compare Value A	1	Set this parameter to "1".
	Compare Value B	1	Set this parameter to "1".
Set/Reset_0	—	—	—
Output 0	—	—	—
Event 0	—	—	—
Counter_Timer_1	Count Enable	ON	Set this parameter to "ON".
	Upper Limit	65535	Default value
	Lower Limit	0	Default value
	Clock	1μs	Set this parameter to "1μs".
	Preset Value	0	Set this parameter to 0.
Compare_1	Condition	Coincidence	Set this parameter to "Coincidence".
	Compare Mode A	=	Set this parameter to "=".
	Compare Value A	1000	Set 1000 as comparison value A for the count value.
Event 2	—	—	—

The items other than the above need not to be set.

Operation

The following shows the operation at the time of A/D conversion value logging.

The A/D conversion value acquisition timing is when the output terminal turns to High.



No.	Description
(1)	When the input signal is turned on, the RUN terminal of counter timer 0 turns to High and counting per clock cycle starts.
(2)	The count value is 1 and the output terminal turns to High for one clock. (A/D conversion value acquisition edge)
(3)	The count value of counter timer 1 becomes 1 through the Event terminal.
(4)	The value returns to the lower limit value (1) through the ring counter, and the output terminal turns to High again.
(5)	The count value of counter timer 1 becomes 1000, and the STOP terminal of counter timer 0 is set to High through an Event.
(6)	The count value of counter timer 0 is preset to 0 through STOP → PRESET, and the initial status is restored.

5 PARAMETER SETTINGS

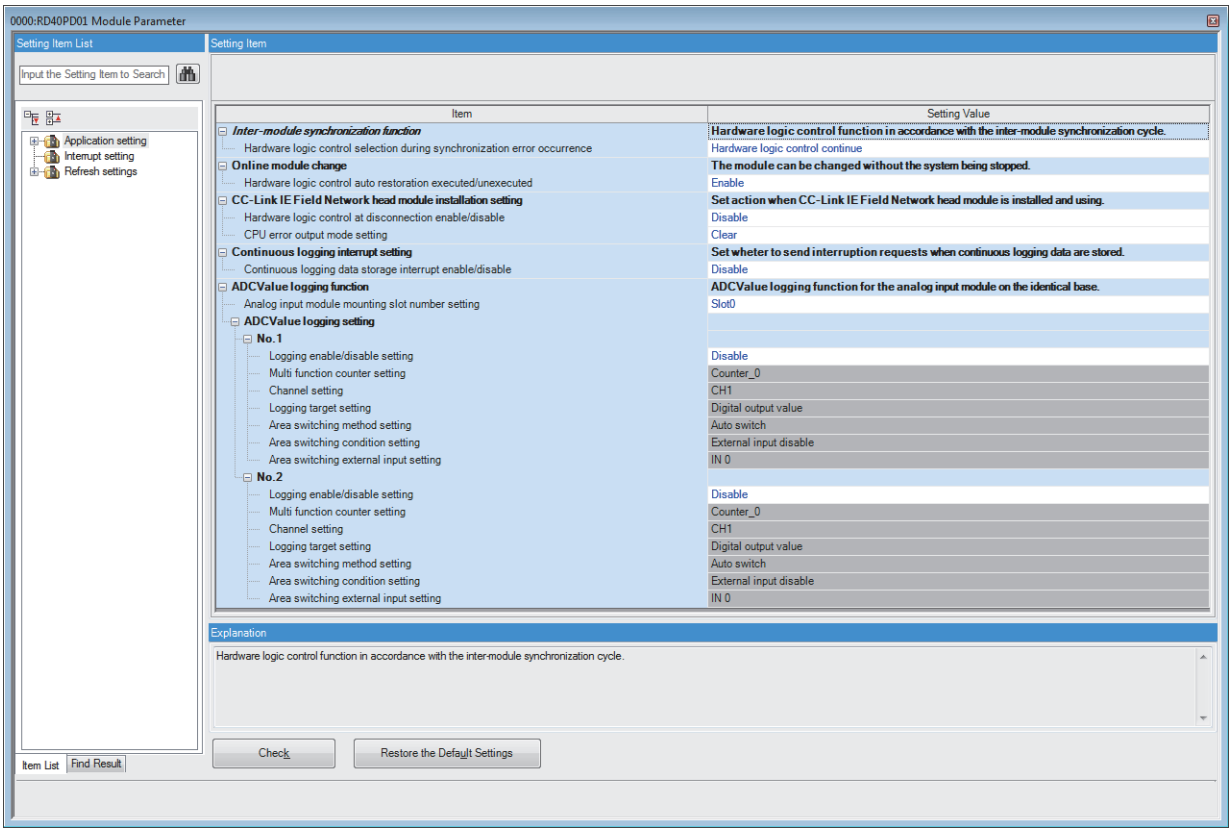
5.1 Application Setting

Setting method

Open "Application setting" of the engineering tool.

1. Start "Module Parameter".

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ Module model name ⇨ [Module Parameter] ⇨ [Application setting]



2. Click the item to be changed to enter the setting value.

• Item where a value is selected from the drop-down list

Click the [▼] button of the item to be set, and from the drop-down list that appears, select the value.

• Item where a value is entered into the text box

Double-click the item to be set to enter the numeric value.

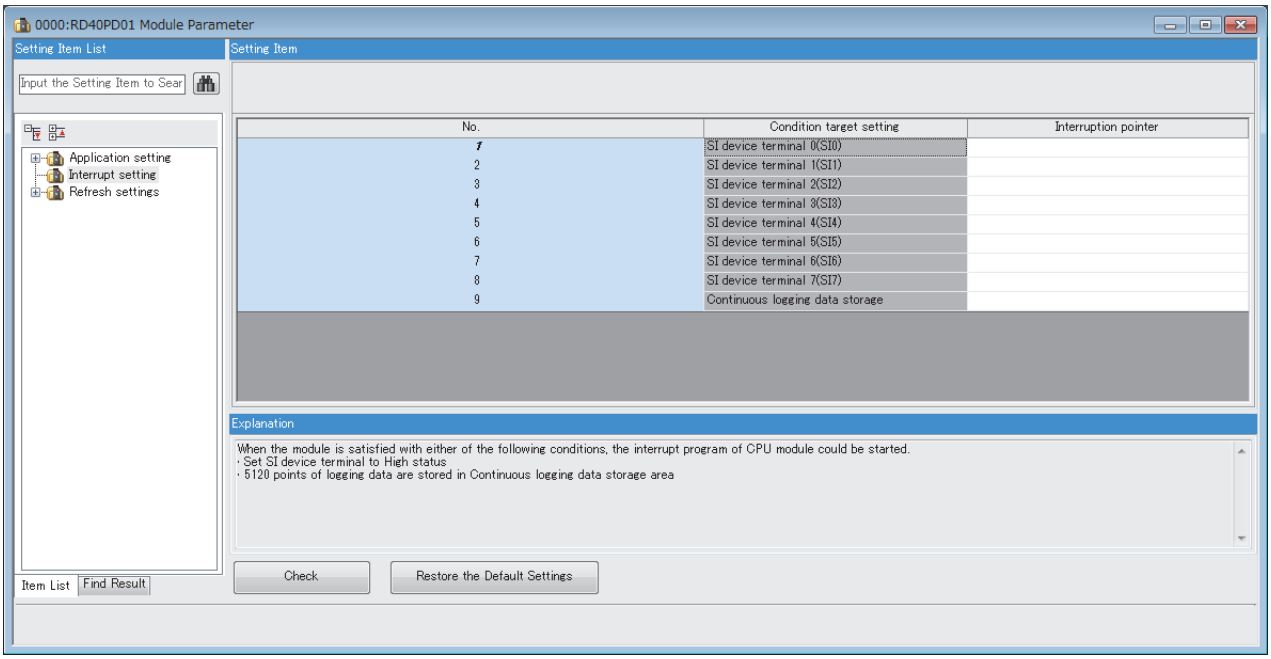
5.2 Interrupt Settings

Setting method

Open "Interrupt setting" of the engineering tool.

1. Start "Module Parameter".

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ Module model name ⇨ [Module Parameter] ⇨ [Interrupt setting]



2. Click the item of interrupt setting number (No.1 to 9) to be changed to enter the setting value.

- Item where a value is entered into the text box

Double-click the item to be set to enter the numeric value.

5.3 Refresh Settings

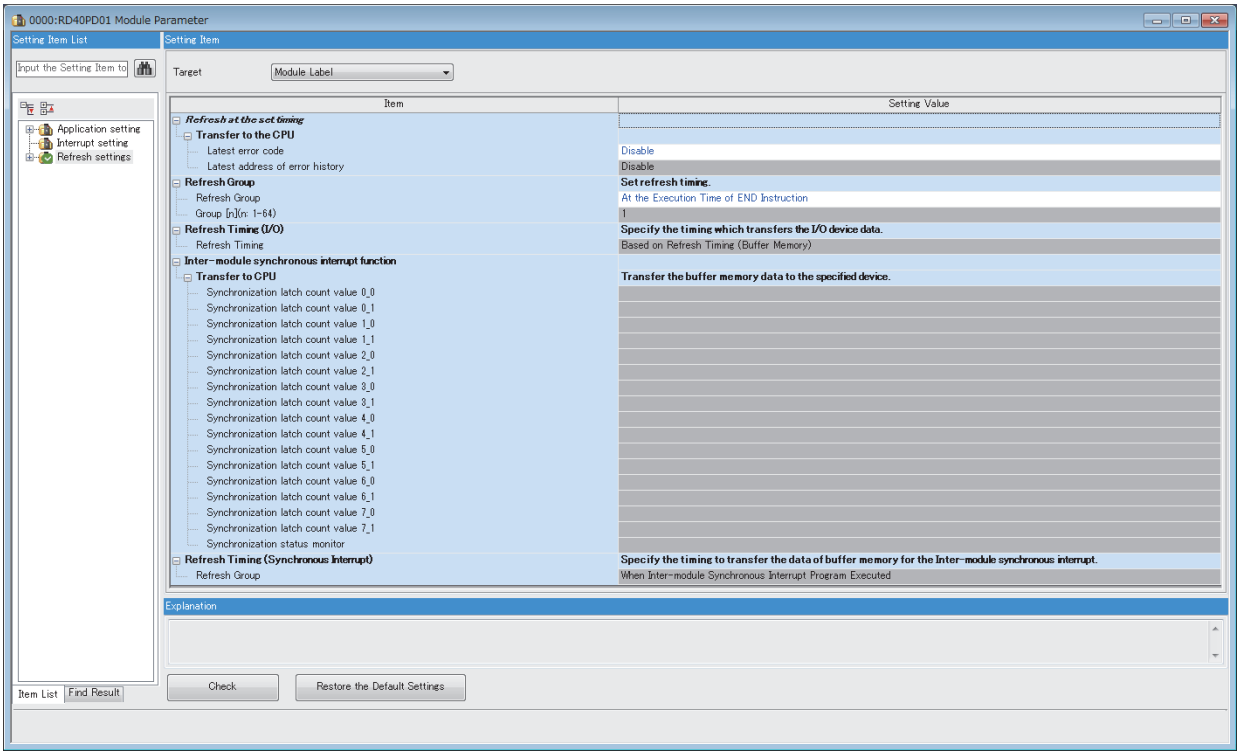
Setting method

Set the buffer memory area of the flexible high-speed I/O control module to be refreshed.

The refresh settings eliminate the need for reading data by programming.

1. Start "Module Parameter".

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ Module model name ⇨ [Module Parameter] ⇨ [Refresh setting]



2. Click "Target" and set the refresh target.

• When "Target" is "Refresh Data Register (RD)"

Setting the start device for "Start Device Name" automatically sets the transfer destinations for all the items.

• When "Target" is "Device"

Double-click the item to be set to enter the refresh target device.

3. Click "Refresh Timing" to set the refresh timing.

Set "Refresh Timing" to "At the Execution Time of END Instruction" or "At the Execution Time of Specified Program".

When "At the Execution Time of Specified Program" is set, double-click "Refresh Group [n](n: 1-64)" and set a value of 1 to 64.

Point

The module label cannot be set for "Target" for the flexible high-speed I/O control module.

Refresh processing time

The refresh processing time [μs] is a constituent of the scan time of the CPU module. For details on the scan time, refer to the following.

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

The refresh processing time [μs], which is taken for refresh, is given by:

• Refresh processing time [μs] = Refresh read time (time for transferring refresh data to the CPU module)

According to the setting of "Target", the refresh read time varies.

When "Target" is "Refresh Data Register (RD)"

The following table lists the refresh read time when the R□CPU is used.

Model	Classification	When the refresh settings are used	When the inter-module synchronization function is used
RD40PD01	Refresh read time	13.57μs	14.32μs

When "Target" is "Device"

Calculate the refresh read 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.

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

6 TROUBLESHOOTING

This chapter describes errors that may occur when the flexible high-speed I/O control module is used, and those troubleshooting.

6.1 Troubleshooting with the LEDs

Checking the display status of LEDs enables the primary diagnosis without an engineering tool and can narrow the range of trouble causes.

The status of the flexible high-speed I/O control module can be checked using the RUN LED and ERR LED. The following table lists the correspondence relation between each LED and the status of the flexible high-speed I/O control module.

Name	Description
RUN LED	Indicates the operating status of the module. On: Normal operation Flashing (1s cycles): During simulation Flashing (400ms cycles): When a module is selected for an online module change Off: When 5V power off, a watchdog timer error has occurred, or exchanging the module is allowed in the process of the online module change
ERR LED	Indicates the error status of the module.*1 On: An error has occurred. Off: Normal operation

*1 For details, refer to the following.
☞ Page 250 List of Error Codes

6.2 Checking the State of the Module

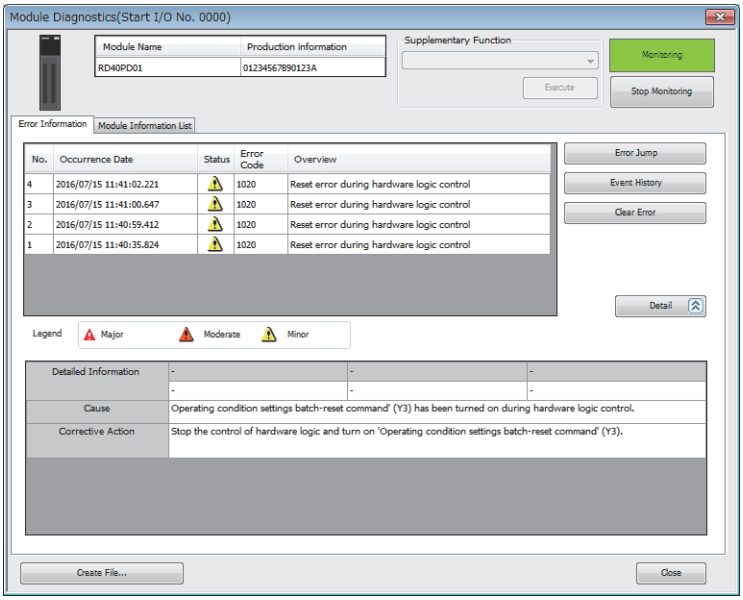
The following functions are available in the "Module Diagnostics" window of the flexible high-speed I/O control module.

Function	Application
Error Information	Displays the descriptions of errors that have occurred. Clicking the [Event History] button displays the errors that have occurred on the flexible high-speed I/O control module and the history of the errors detected and the operations executed on each module.
Module Information List	Displays each status information of the flexible high-speed I/O control module.

Error information

Check the descriptions and the actions of the errors that have occurred.

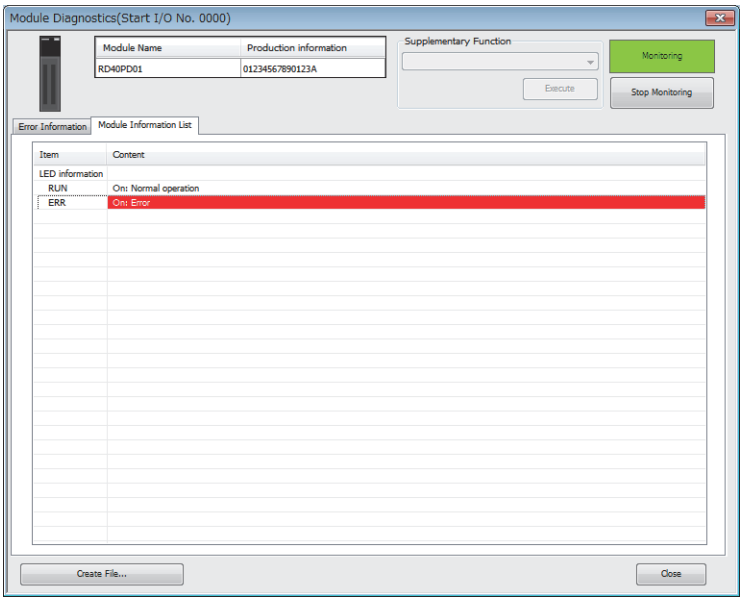
[Diagnostics] ⇨ [System Monitor] ⇨ Right-click the module to be checked. ⇨ "Module Diagnostics"



Item	Description
Cause	Displays the detailed error causes.
Corrective Action	Displays the actions to eliminate the error causes.

Module information list

Switch to the "Module Information List" tab to display status information of the flexible high-speed I/O control module.



Item	Description
LED information	Displays the LED status of the flexible high-speed I/O control module.

6.3 Troubleshooting by Symptom

RUN LED is flashing or turns off

RUN LED is flashing

Check item	Action
Is the simulation being executed?	Check that the RUN LED turns on after the completion of the simulation.
Is the module selected as an online module change target?	Turn on the module selection cancellation request flag (SM1615).

RUN LED turns off

Check item	Action
Has the power been supplied?	Check that the voltage supplied to the power supply module is within the rated range.
Is the capacity of the power supply module sufficient?	Calculate the total current consumption of the connected modules including the CPU module, input/output modules, and intelligent function module to check that the power capacity is sufficient.
Has the module been properly connected?	Check that the module has been properly connected.
Is exchanging the module allowed in the process of the online module change?	Execute the online module change. For details, refer to the following. 📖 MELSEC iQ-R Online Module Change Manual
Other than the above	Reset the CPU module and check that the RUN LED turns on. If the RUN LED still does not turn on, the possible cause is a failure of the module. Please consult your local Mitsubishi representative.

ERR LED turns on

Check item	Action
Is there an error?	Check 'Latest error code' (Un\G100) and take actions described in the following section. ☞ Page 250 List of Error Codes

Inputs from external devices are not performed

Check item	Action
Is the upper section LED of the indicator LED on?	Check the external wiring and make necessary corrections.
Is the control of the hardware logic working?	When 'Hardware logic control flag' (X4) is off, turn on 'Hardware logic control start request' (Y4) to start the hardware logic control.
Are the block layout and links between blocks in the hardware logic correct?	Check the block layout and links between blocks in the hardware logic and make necessary corrections.

Incorrect inputs from external devices are performed

Check item		Action
Measures to reduce noise	Is the filter time setting of external input blocks correct?	Set the filter time of the external input blocks longer.
	Are twisted pair shielded cables used for the pulse input cables?	Use twisted pair shielded cables for the pulse input cables.
	Have measures to reduce noise been taken in the control panel or for adjacent devices?	Take measures to reduce noise such as attaching a CR surge suppressor to the magnet switch or other device.
	Is there a sufficient distance between the high voltage device and the pulse input cables?	Wire the pulse input cables alone when placing them in a duct, and keep a distance of 150mm or more from the power cables in the control panel.
	Is the module affected by noise through the grounding area?	Separate the grounding cable of the flexible high-speed I/O control module from the grounding area.
	Are the power cables and I/O cables bundled together?	Do not bundle the power cables and I/O cables together.
	Is the external wiring connected to unused terminals?	Do not connect the external wiring to unused terminals.

Outputs to external devices are not performed

Check item	Action
Is the lower section LED of the indicator LED on?	Check the block layout and links between blocks in the hardware logic and make necessary corrections.
Is the external wiring correct?	Check the external wiring and make necessary corrections.
Is the control of the hardware logic working?	When 'Hardware logic control flag' (X4) is off, turn on 'Hardware logic control start request' (Y4) to start the hardware logic control.
Are the block layout and links between blocks in the hardware logic correct?	Check the block layout and links between blocks in the hardware logic and make necessary corrections.

The hardware logic does not function

Check item	Action
Is the control of the hardware logic working?	When 'Hardware logic control flag' (X4) is off, turn on 'Hardware logic control start request' (Y4) to start the hardware logic control.
Are the block layout and links between blocks in the hardware logic correct?	Check the block layout and links between blocks in the hardware logic and make necessary corrections.
Is the CPU module indicating an error?	If the CPU module is in an error status, refer to the troubleshooting in the user's manual for the CPU module used and take corrective actions to restore normal operation of the CPU module.

A multi function counter does not start counting or does not count properly

The module does not start counting

Check item	Action
Is Count Enable set to ON?	Set Count Enable to ON using a program or the configuration tool.
Is the CPU module indicating an error?	If the CPU module is in an error status, refer to the troubleshooting in the user's manual for the CPU module used and take corrective actions to restore normal operation of the CPU module.
Is the external wiring for ΦA and ΦB correct?	Check the external wiring and make necessary corrections.
Is the control of the hardware logic working?	When 'Hardware logic control flag' (X4) is off, turn on 'Hardware logic control start request' (Y4) to start the hardware logic control.
Are the block layout and links between blocks in the hardware logic correct?	Check the block layout and links between blocks in the hardware logic and make necessary corrections.

The module does not count properly

Check item	Action
Are the current value or other values read in increments of one word (16 bits) using a program when 32-bit counter timer blocks are used?	Read the values of two words (32 bits) in a batch.
Was the preset function performed out of the count range of a counter?	Correct the preset value within the count range and perform the preset function again.
Is the filter time setting of external input blocks correct?	Check the filter time setting and make necessary corrections.
When the same pulse is input to the multi function counter blocks that have the same links, is the same value counted?	If the count values are different, the possible cause is a failure of the module. Please consult your local Mitsubishi representative.
Does the input pulse wave satisfy the performance specifications?	Observe and check the pulse wave with a synchroscope. If the input pulse does not satisfy the performance specifications, input pulses that satisfy the performance specifications.
Is the external wiring with the SSI encoder correct?	Check the external wiring. (📖 MELSEC iQ-R Flexible High-Speed I/O Control Module User's Manual (Startup))
Is the setting value of the SSI encoder block correct?	Check and correct the value according to the SSI encoder to be connected.
Is the cable length within the range of the maximum cable length?	Check the cable length and size. (📖 MELSEC iQ-R Flexible High-Speed I/O Control Module User's Manual (Startup)) Or slow the SSI transmission speed.
Are shielded twisted pair cables used?	Use shielded twisted pair cables.
Does noise affect anything?	Take measures to reduce noise such as attaching a surge suppressor to the magnet switch.
Is there a sufficient distance between the high voltage device and the signal wires?	Wire the signal wires alone and keep them 150mm or more away from power cables.

■Pulse shaping method

As one of measures against external noise or waveform distortion, the following describes the shaping method of a pulse waveform with dummy resistors.

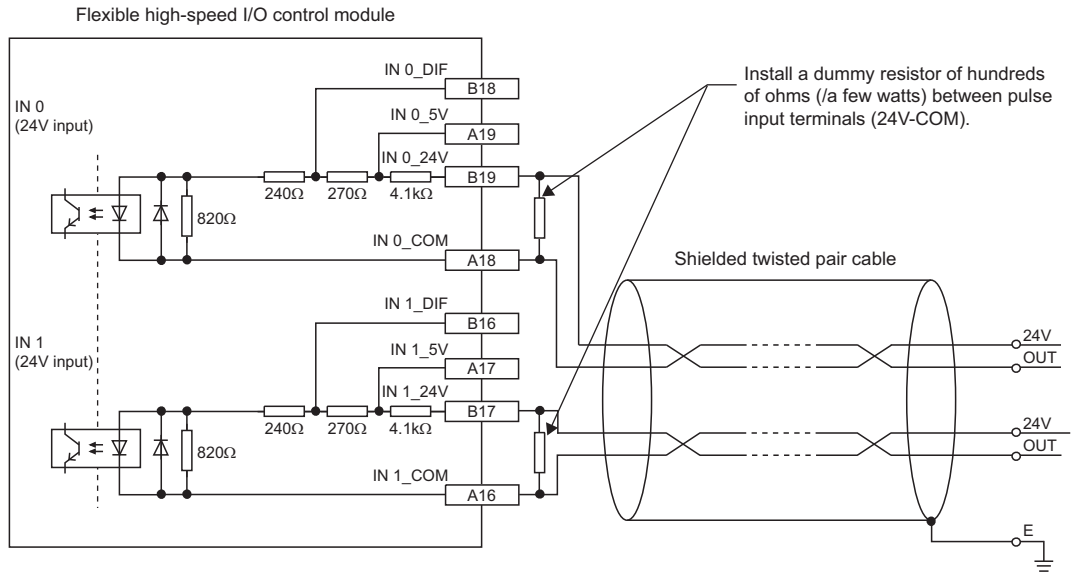
An effective method for pulse shaping is to apply dummy resistors of several hundreds ohms (/several watts) across pulse input terminals connected to an encoder to increase a load current through the cables. This method becomes more effective as the load current value increases.

With this shaping method, the following effects can be obtained.

- When the wiring distance between an encoder and the flexible high-speed I/O control module is long, this shaping method improves the waveform distortion and the pulse waveform becomes stable.
- When the pulse waveform is unstable due to noise such as external noise, this shaping method stabilizes the pulse waveform and the noise effects can be reduced.

Ex.

The following figure shows a connection example of dummy resistors at 24VDC.



The following example describes how to select a dummy resistor (how to calculate a resistance constant and rated power of a dummy resistor).

Target	Calculation method
Resistance constant of a dummy resistor (at 24VDC input)	Calculate a resistance constant (R) as follows. • R = V ÷ I = 24V ÷ 35mA = 680Ω
Rated power of a dummy resistor (at 24VDC input)	Calculate the power (P1) as follows. • P1 = V × I = 24V × 35mA = 0.84W (Approx. 1W) Calculate the power (P2) including margins from the power (P1). • P2 = P1 × 2 = 0.84 × 2 = 1.68W (Approx. 2W)

As a result of the above calculations, apply dummy resistors of 680Ω(/2W) across pulse input terminals in this case.

Interrupt requests are not properly sent to the CPU module

Check item	Action
Is a link to the SI device terminal in the hardware logic correct?	Check the block layout and links between blocks in the hardware logic and make necessary corrections.
Is the coincidence detection disabled because the addition mode has been applied?	To use the coincidence detection interrupt and addition mode together, preset the value used for the coincidence detection.
Is [Interrupt setting] of [Module Parameter] correct?	Review [Interrupt setting].

The synchronization latch count value does not change

Check item		Action
Checking the system parameter settings of GX Works3	Is the flexible high-speed I/O control module selected for a synchronization target module?	Check 'Synchronization status monitor' (Un\G8173). When 'Synchronization status monitor' (Un\G8173) is Not the inter-module synchronization target (0), the module is not selected for a synchronization target. Set the module for a synchronization target in "System Parameter" of GX Works3.
Checking the operating status of the CPU module	Is the RUN/STOP/RESET switch set to STOP or is the CPU module in the stop error status?	Set the RUN/STOP/RESET switch to RUN. When a stop error has occurred, take actions described in the list of error codes. ☞ Page 250 List of Error Codes
Checking the program	Is the control of the hardware logic working?	When 'Hardware logic control flag' (X4) is off, turn on 'Hardware logic control start request' (Y4) to start the hardware logic control.
	Is the inter-module synchronous interrupt program (I44) prepared?	To acquire the synchronization latch count value, refresh the value by executing the inter-module synchronous interrupt program.
	Has the EI instruction been executed?	To execute the inter-module synchronous interrupt program, execute the EI instruction in the main program.
Checking the hardware logic	Are the block layout and links between blocks in the hardware logic correct?	Check the block layout and links between blocks in the hardware logic and make necessary corrections.

A/D conversion values are not logged

Check item		Action
Monitor check	Did A/D conversion value logging of the flexible high-speed I/O control module start?	Check 'No.1 ADCValue logging enable/disable monitor' (Un\G170) and 'No.2 ADCValue logging enable/disable monitor' (Un\G180). If the monitor status is Disabled (0), logging has not started. Set the A/D conversion value logging settings in either the module parameter settings in GX Works3 or the buffer memory settings, and start hardware logic control.
Analog input module check	Are the A/D conversion values of the analog input module updated?	Check whether the digital output values and digital operation values of the target analog input module are updated. If they are not updated, check the troubleshooting items of the analog input module.
Consistency check with A/D conversion value logging settings	In the flexible high-speed I/O control module, is a disabled A/D conversion channel of the analog input module being targeted?	If a disabled A/D conversion channel is targeted, only 0's will be logged. Either correct the parameter of the flexible high-speed I/O control module to specify an enabled conversion channel, or modify the target channel on the analog input module side to enable conversion.
Hardware logic check	Are the block layout and links between blocks in the hardware logic correct?	Use the debug function or other means to check that the hardware logic specified by the parameter satisfies the output conditions of the multi function counter block. If the output conditions are not satisfied, correct the hardware logic.

6.4 List of Error Codes

When an error occurs during operation, the flexible high-speed I/O control module stores an error code in 'Latest error code' (Un\G100) of the buffer memory. In addition, 'Error flag' (XF) turns on. Turning on 'Error clear request' (YF) clears the error code in 'Latest error code' (Un\G100), and 'Error flag' (XF) turns off.

Error codes of the flexible high-speed I/O control module are classified in minor errors or moderate errors.

- Minor error: An error caused by an incorrect setting in the program such as incorrect execution timing. After the error cause is eliminated by reviewing the program, each function can be executed normally. (1000H to 1FFFH)
- Moderate error: An error such as hardware failure and an execution error of the inter-module synchronization function. The hardware logic control does not continue. (2000H to 2FFFH, 3000H to 3FFFH)

The following table shows the error codes stored.

☐ in the error code: Indicates the number (0 to 7) of the multi function counter block where an error has occurred.

◇ in the error code: Indicates the number (0 to 1) of the SSI encoder block where an error has occurred.

△ in the error code: Indicates the number (No.1 or No.2) of the ADCValue logging where an error has occurred.

Error code (hexadecimal)	Error name	Description and cause	Operation at the error	Action
0000H	—	No error has occurred.	—	—
100☐H	Multi function counter block ☐ overflow error	A count value has exceeded the upper limit value when 32-bit multi function counter block is set to linear counter mode.	The count operation stops for the corresponding multi function counter block.	Perform the preset function to the corresponding multi function counter block.
101☐H	Multi function counter block ☐ underflow error	A count value has fallen below the lower limit value when 32-bit multi function counter block is set to linear counter mode.	The count operation stops for the corresponding multi function counter block.	Perform the preset function to the corresponding multi function counter block.
1020H	Reset error during hardware logic control	'Operating condition settings batch-reset command' (Y3) has been turned on during hardware logic control.	The control of hardware logic continues.	Stop the control of hardware logic and turn on 'Operating condition settings batch-reset command' (Y3).
1040H	Reset error without any data writes to a flash ROM	'Operating condition settings batch-reset command' (Y3) has been turned on while any setting data has not been written to a flash ROM.	The operating condition settings batch-reset for hardware logic is not performed.	Set operating conditions in a flash ROM and turn on 'Operating condition settings batch-reset command' (Y3).
1060H	Flash ROM data error	The setting data in the flash ROM are faulty.	The control can be started with non-wired hardware logic.	Write the setting data again with a configuration tool. Take measures to reduce noise with a shielded cable for connection. If the error occurs again, the possible cause is a failure of the module. Please consult your local Mitsubishi representative.
1061H	Trigger setting data error	The trigger setting data in the flash ROM are faulty.	The trigger setting becomes disabled, and thus the logic analyzer function cannot be started.	Write the trigger setting again with a configuration tool. Take measures to reduce noise with a shielded wire for connection. If the error occurs again, the possible cause is a failure of the module. Please consult your local Mitsubishi representative.
1080H	Number of writes to a flash ROM over error	The number of writes to a flash ROM has exceeded 10000 times.	The control of hardware logic can be started.	Any further writes to a flash ROM may not be reflected correctly.
1081H	Number of trigger setting writes over error	The number of trigger setting writes to a flash ROM has exceeded 10000 times.	The logic analyzer function can be started.	Any further trigger setting writes to a flash ROM may not be reflected correctly.

Error code (hexadecimal)	Error name	Description and cause	Operation at the error	Action
109◇H	SSI encoder block ◇ DATA signal wire reverse error	The positive and negative of the DATA signal wire are reversely connected.	Position data from the encoder cannot be received properly.	Check the external wiring for the DATA signal.
10A◇H	SSI encoder block ◇ DATA signal error	The DATA signal input state is Low at the start of data transfer, or is High just after the transfer is completed.		Check the SSI transmission speed and SSI code length. In addition, consider the possibility of noise effects. Check the cable laying, shielding, and cable length.
10B◇H	SSI encoder block ◇ parity error	A parity error has occurred.		Check the external wiring to the SSI encoder and settings of the SSI encoder block.
10C0H	Trigger start error at trigger setting disabled	The trigger start request was issued while the trigger setting for the logic analyzer function was disabled.	'Trigger status monitor' (Un\G123) remains set to Trigger stop (0).	Write the trigger setting with a configuration tool and issue the trigger start request.
10C1H	Trigger start error during simulation execution	The trigger start request was issued while the simulation was being executed.	'Trigger status monitor' (Un\G123) remains set to Trigger stop (0).	Issue the trigger start request after the simulation execution is completed.
10D0H	Hardware logic setting error at the start of continuous logging	The continuous logging was started with the following settings configured in the hardware logic in operation. • "User Address" is assigned to Hardware logic area (High speed area) (Un\G1000 to Un\G1029) • SSI encoder block	The continuous logging is not started.	Start the continuous logging with the following settings not configured in the hardware logic. • "User Address" is assigned to Hardware logic area (High speed area) (Un\G1000 to Un\G1029) • SSI encoder block (When using "User Address", assign it to Hardware logic area (Low speed area) (Un\G1030 to Un\G1099).)
10D1H	Continuous logging start error during simulation	The continuous logging was started during simulation.	The continuous logging is not started.	Start the continuous logging after the completion of the simulation.
10D2H	Continuous logging start error with the logic analyzer function operating	The continuous logging was started while the logic analyzer function was operating.	The continuous logging is not started.	Start the continuous logging after the logic analyzer function is stopped. (Check that Trigger stop (0) is stored in 'Trigger status monitor' (Un\G123) to confirm that the logic analyzer function is stopped.)
10D3H	Continuous logging start error during inter-module synchronization	The continuous logging was started when this module was set as the inter-module synchronization target.	The continuous logging is not started.	The continuous logging function cannot be used when this module is set as the inter-module synchronization target. To use the continuous logging function, set this module as a non-target of inter-module synchronization in "Inter-module Synchronization Setting" of "System Parameter".
10D4H	Continuous logging start error with hardware logic control stopped	The continuous logging was started while the hardware logic control was stopped.	The continuous logging is not started.	Start the continuous logging after the start of hardware logic control.
10D5H	Trigger start error during continuous logging	The trigger start request was issued from a program during continuous logging.	The logic analyzer function is not started.	Issue the trigger start request after the continuous logging is stopped.
10D6H	Continuous logging cycle setting range error	A value other than 1μs (0) to 1000μs (3) is set in Continuous logging cycle setting.	The continuous logging is not started.	Set Continuous logging cycle setting to a value within 1μs (0) to 1000μs (3).
10D7H	Continuous logging start error during ADCValue logging	The continuous logging was started while the ADCValue logging was being executed.	The continuous logging is not started.	Start the continuous logging after the ADCValue logging is stopped. (Check that Disabled (0) is stored in 'No.△ ADCValue logging enable/disable monitor' (Un\G170, Un\G180) to confirm that the A/D conversion value logging function is stopped.)

Error code (hexadecimal)	Error name	Description and cause	Operation at the error	Action
10E0H	A/D conversion value logging function start error during inter-module synchronization	The A/D conversion value logging function was set to be enabled when this module was set as the inter-module synchronization target.	The ADCValue logging is not started.	The A/D conversion value logging function cannot be used when this module is set as the inter-module synchronization target. To use the A/D conversion value logging function, set this module as a non-target of inter-module synchronization in "Inter-module Synchronization Setting" of "System Parameter".
10E1H	Analog input module mounting slot number setting error	The slot set in 'Analog input module mounting slot number setting' (Un\G140) is an empty slot or a slot where a module other than the target analog input module is mounted.	The ADCValue logging is not started.	Set 'Analog input module mounting slot number setting' (Un\G140) to the number of the slot where the target analog input module is mounted.
10E2H	Analog input module communication error	An error has occurred on communications between modules. Or the ADCValue logging was executed while the target analog input module was in the process of the online module change.	The ADCValue logging is stopped.	Do not replace the target analog input module using the online module change. The module may be affected by noise if the error has occurred with a cause other than the above. Check and adjust the cable wiring and the installation environment of the programmable controllers, and restart the system. If the error occurs again even after the adjustment, the possible cause is a failure of the module. Please consult your local Mitsubishi representative.
110△H	No.△ ADCValue logging counter setting error	A value other than 0 to 7 is set in 'No.△ ADCValue logging counter setting' (Un\G151, Un\G161).	No.△ ADCValue logging is not started.	Set 'No.△ ADCValue logging counter setting' (Un\G151, Un\G161) to a value within 0 to 7.
111△H	No.△ ADCValue logging channel setting error	A value out of the range is set in 'No.△ ADCValue logging channel setting' (Un\G152, Un\G162).	No.△ ADCValue logging is not started.	Set 'No.△ ADCValue logging channel setting' (Un\G152, Un\G162) to a value within the following ranges. • Analog input module with four channels: CH1 (0) to CH4 (3) • Analog input module with eight channels: CH1 (0) to CH8 (7)
112△H	No.△ ADCValue logging area switching external input setting error	A value other than 0H to BH is set in 'No.△ ADCValue logging area switching external input setting' (Un\G156, Un\G166).	'No.△ ADCValue logging area switching condition setting' becomes Disable external input (0).	Set 'No.△ ADCValue logging area switching external input setting' (Un\G156, Un\G166) to a value within 0H to BH.
17E0H	Module-specific backup parameter restore error	The hardware logic cannot be restored with the module-specific backup parameter.	The hardware logic stored in the flash ROM in the new module after online module change becomes valid.	• The module-specific backup parameter file may be damaged. Execute "Write to Module (execution + flash ROM)". • The module-specific backup parameter cannot be used when the flexible high-speed I/O control module is mounted on the redundant extension base unit. Set "Hardware logic control auto restoration executed/unexecuted" to "Disable".
17E1H	Module-specific backup parameter creation error	The module-specific backup parameter was not created.	The hardware logic stored in the flash ROM in the new module after online module change becomes valid.	Check the free space on the data memory of the control CPU and the SD memory card, and execute "Write to Module (execution + flash ROM)" again.

Error code (hexadecimal)	Error name	Description and cause	Operation at the error	Action
2600H	Inter-module synchronization cycle crossing over error	It was detected that the processing was performed across the cycles with the inter-module synchronization function operating.	The operations differ depending on the setting of "Hardware logic control selection during synchronization error occurrence". For details, refer to the following. ☞ Page 63 Inter-Module Synchronization Function	Correct settings to extend the inter-module synchronization cycle or reduce the number of steps of the inter-module synchronous interrupt program.
2601H	Inter-module synchronization cycle skip error	It was detected that the processing of a cycle was skipped with the inter-module synchronization function operating.	The control of hardware logic stops regardless of the setting of "Hardware logic control selection during synchronization error occurrence".	Execute the EI instruction in the main program and enable an interrupt. Correct settings to extend the inter-module synchronization cycle or reduce the number of steps of the inter-module synchronous interrupt program. If the error occurs again even after the above actions are taken, the possible cause is a failure of the module. Please consult your local Mitsubishi representative.
2610H	Inter-module synchronization signal error	A synchronization loss was detected with the inter-module synchronization function operating.	The operations differ depending on the setting of "Hardware logic control selection during synchronization error occurrence". For details, refer to the following. ☞ Page 63 Inter-Module Synchronization Function	The module may be affected by noise. Check and adjust the cable wiring and the installation environment of the programmable controllers, and restart the system. If the error occurs again even after the adjustment, the possible cause is a failure of the module. Please consult your local Mitsubishi representative.
3001H	Hardware error	A hardware error has occurred.	The control of hardware logic stops.	Turn off and on the power, or reset the CPU module. If the error occurs again, the possible cause is a failure of the module. Please consult your local Mitsubishi representative.

APPENDICES

Appendix 1 Module Label

The functions of the flexible high-speed I/O control module can be set with module labels.

Module labels of I/O signals

The module label names of I/O signals are defined as follows.
"Module name"_"Module number".b"Label name" or "Module name"_"Module number".b"Label name"_D

Ex.
RD40PD01_1.bModuleREADY_D

■Module name
The character string of a module model name is given.

■Module number
A number starting from 1 is added to identify modules that have the same module name.

■Label name
The label identifier unique to a module is given.

■_D
This string indicates that the module label is for the direct access input (DX) or direct access output (DY). A module label without the string is for the input (X) or output (Y) of the refresh processing.

Module labels of buffer memory areas

The module label name of a buffer memory area is defined with the following structure:
"Module name"_"Module number"."Data type"_D["Logging condition number"]."Data type""Label name"_D

Ex.
RD40PD01_1.uLatestErrorCode_D
RD40PD01_1.stnADCValueLogging_Parameter_D[1].uLoggingEnableDisableSetting_D

■Module name
The character string of a module model name is given.

■Module number
A number starting from 1 is added to identify modules that have the same module name.

■Data type
The data type to sort a buffer memory area is given. Each data type is as follows:

Data type	Description
stnADCValueLogging_Monitor	A/D conversion value logging function monitor
stnADCValueLogging_Control	A/D conversion value logging function control
stnADCValueLogging_Parameter	A/D conversion value logging function parameter

■Logging condition number
The number of the logging condition of the A/D conversion value logging function is given.

■Data type

The string that represents the data size of a buffer memory area is given. Each data type is as follows:

Data type	Description
u	Word [Unsigned]/Bit string [16-bit]
w	Word [Signed]
d	Double word [Signed]

■Label name

The label identifier unique to a module is given.

■_D

This string indicates that the module label is for the direct access. Values that are read from or written to the module label is reflected in the module instantly.

Appendix 2 I/O Signals

List of I/O signals

The following tables list the I/O signals of the flexible high-speed I/O control module.
For details on the I/O signals, refer to the following.
☞ Page 258 Details of input signals
☞ Page 263 Details of output signals

Point

- The I/O numbers (X/Y) listed below are shown on the assumption that the start I/O number of the flexible high-speed I/O control module is set to 0.
- The use prohibited signals listed below are used by the system and are not available for users. If a user uses these signals (turning on), the performance of the flexible high-speed I/O control module is not guaranteed.

Input signal

Device number	Signal name
X0	Module READY
X1, X2	Use prohibited
X3	Operating condition settings batch-reset complete flag
X4	Hardware logic control flag
X5, X6	Use prohibited
X7	Hardware logic control stop flag at disconnection
X8 to XC	Use prohibited
XD	No.1 ADCValue logging area switching request complete flag
XE	No.2 ADCValue logging area switching request complete flag
XF	Error flag
X10	IN 0
X11	IN 1
X12	IN 2
X13	IN 3
X14	IN 4
X15	IN 5
X16	IN 6
X17	IN 7
X18	IN 8
X19	IN 9
X1A	IN A
X1B	IN B
X1C to X1F	Use prohibited

Output signal

Device number	Signal name
Y0 to Y2	Use prohibited
Y3	Operating condition settings batch-reset command
Y4	Hardware logic control start request
Y5	Hardware logic control stop request
Y6	Hardware logic control stop signal at disconnection
Y7	Hardware logic control stop flag clear request at disconnection
Y8 to YC	Use prohibited
YD	No.1 ADCValue logging area switching request
YE	No.2 ADCValue logging area switching request
YF	Error clear request
Y10	General command 0
Y11	General command 1
Y12	General command 2
Y13	General command 3
Y14	General command 4
Y15	General command 5
Y16	General command 6
Y17	General command 7
Y18	General command 8
Y19	General command 9
Y1A	General command A
Y1B	General command B
Y1C	General command C
Y1D	General command D
Y1E	General command E
Y1F	General command F

Details of input signals

The following describes the details of the input signals for the flexible high-speed I/O control module which are assigned to the CPU module.

The I/O numbers (X/Y) in this section apply when the start I/O number of the flexible high-speed I/O control module is set to "0".

Module READY

This signal turns on when the control is ready after the CPU module is powered on or is reset.

In the following case, 'Module READY' (X0) turns off.

- When a watchdog timer error has occurred in the flexible high-speed I/O control module (The control is not performed.)

■Device number

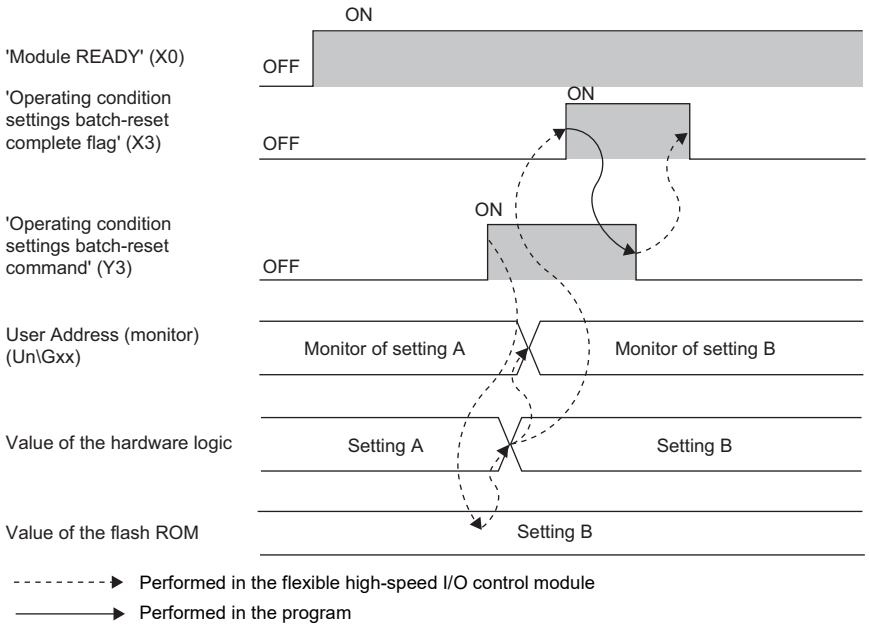
The following shows the device number of this input signal.

Signal name	Device number
Module READY	X0

Operating condition settings batch-reset complete flag

When the setting of the execution memory is reset with the value stored in the flash ROM in the flexible high-speed I/O control module, this signal is used as an interlock to turn on or off 'Operating condition settings batch-reset command' (Y3).

- When 'Operating condition settings batch-reset command' (Y3) is turned on, this signal turns on. When the reset of the hardware logic setting is completed with an error, this signal also turns on.
- When 'Operating condition settings batch-reset command' (Y3) is turned off, this signal turns off.



■Device number

The following shows the device number of this input signal.

Signal name	Device number
Operating condition settings batch-reset complete flag	X3

Hardware logic control flag

While the hardware logic is operating, 'Hardware logic control flag' (X4) is on.

■ON of 'Hardware logic control flag' (X4)

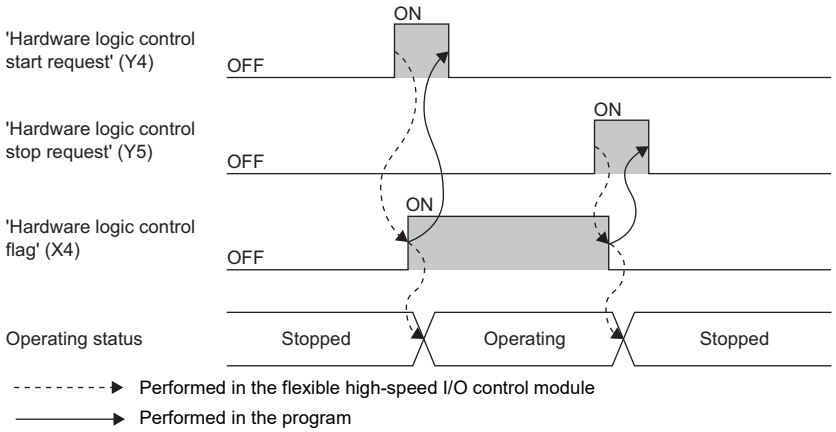
In the following states, this signal turns on.

- When 'Hardware logic control start request' (Y4) is turned on
- When the hardware logic control start request is issued with the configuration tool

■OFF of 'Hardware logic control flag' (X4)

In the following states, this signal turns off.

- When the power supply is turned on
- When the CPU module is reset
- When 'Hardware logic control stop request' (Y5) is turned on (If 'Hardware logic control start request' (Y4) and 'Hardware logic control stop request' (Y5) are turned on at the same time, 'Hardware logic control stop request' (Y5) takes priority and 'Hardware logic control flag' (X4) is not turned on.)
- When the hardware logic control stop request is issued with the configuration tool
- From when the "Write to Module (execution memory)" or "Write to Module (execution + flash ROM)" is executed with the configuration tool until the hardware logic control restarts
- When the simulation is executed with the configuration tool
- When 'Hardware logic control stop flag at disconnection' (X7) turns on
- When "Hardware logic control stop" is set for "Hardware logic control selection during synchronization error occurrence" and the inter-module synchronization cycle crossing over error (error code: 2600H) or inter-module synchronization signal error (error code: 2610H) occurs
- When an inter-module synchronization cycle skip error (error code: 2601H) occurs
- When a stop error occurs in the control CPU



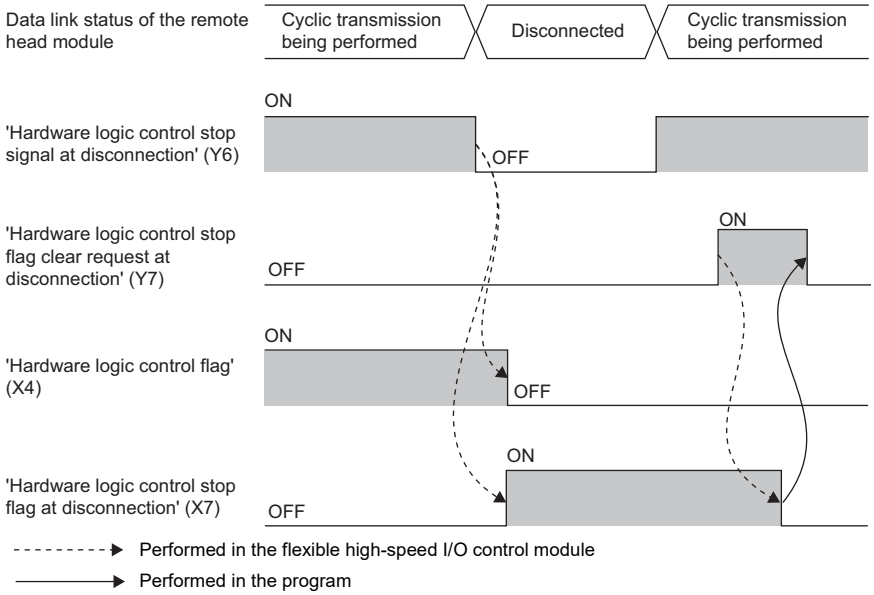
■Device number

The following shows the device number of this input signal.

Signal name	Device number
Hardware logic control flag	X4

Hardware logic control stop flag at disconnection

- This signal is enabled when "Enable" is set for "Hardware logic control at disconnection enable/disable" in "Application setting".
- When 'Hardware logic control stop signal at disconnection' (Y6) is turned off during hardware logic control, the hardware logic control is stopped and this signal turns on. When 'Hardware logic control stop signal at disconnection' (Y6) is turned off while the hardware logic control is stopped, this signal does not turn on.
- When 'Hardware logic control stop flag clear request at disconnection' (Y7) is turned on, this signal turns off.
- When 'Hardware logic control start request' (Y4) is turned on while this signal is on, the request is ignored. Turn off this signal by turning on 'Hardware logic control stop flag clear request at disconnection' (Y7). Then, turn on 'Hardware logic control start request' (Y4).



■Device number

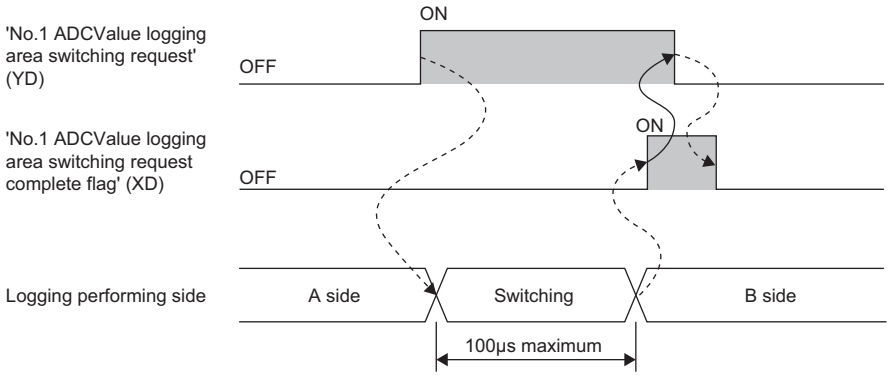
The following shows the device number of this input signal.

Signal name	Device number
Hardware logic control stop flag at disconnection	X7

No.☐ ADCValue logging area switching request complete flag

If trigger switching is set as the switching method for the ADCValue logging data storage area, when switching of the logging side is completed, this flag turns on. No.☐ is the number of the logging condition of the A/D conversion value logging function. The device used depends on whether this input signal is No.1 or No.2.

The maximum time taken to switch the logging side is 100μs.



■Device number

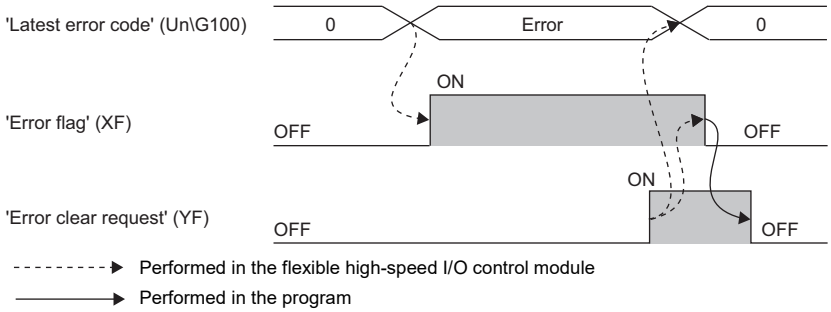
The following shows the device number of this input signal.

Signal name	Device number
No.1 ADCValue logging area switching request complete flag	XD
No.2 ADCValue logging area switching request complete flag	XE

A

Error flag

This signal turns on when an error occurs.



■OFF of 'Error flag' (XF)

When the error cause has been eliminated and 'Error clear request' (YF) is turned on and off, the following flag and error code are cleared.

- 'Error flag' (XF)
- 'Latest error code' (Un\G100)

When 'Clear setting of error history' (Un\G8002) is set to Clear the history. (1), Error history No. ☐ (Un\G8010 to Un\G8169) is also cleared.

■Device number

The following shows the device number of this input signal.

Signal name	Device number
Error flag	XF

IN 0 to IN B

Set input values to external input blocks ("IN 0" to "IN B").

■Device number

The following shows the device number of this input signal.

Signal name	Device number
IN 0 to IN B	X10 to X1B

Details of output signals

The following describes the details of the output signals for the flexible high-speed I/O control module which are assigned to the CPU module.
The I/O numbers (X/Y) in this section apply when the start I/O number of the flexible high-speed I/O control module is set to "0".

Operating condition settings batch-reset command

This signal is used to reset the setting of the execution memory with the value stored in the flash ROM in the flexible high-speed I/O control module. When this signal is turned on, the values in the execution memory and Hardware logic area (Un\G1000 to Un\G1099) are reset to the values stored in the flash ROM.
For the timing of turning on and off this signal, refer to the following.
☞ Page 258 Operating condition settings batch-reset complete flag

■Precautions

- Turn on this signal while the hardware logic is stopped. When this signal is turned on in the operation of the hardware logic, the reset error during hardware logic control (error code: 1020H) occurs. The setting of the hardware logic is not reset and the operation of the hardware logic continues.
- When no data has been written to the flash ROM and this signal is turned on, the reset error without any data writes to a flash ROM (error code: 1040H) occurs.

■Device number

The following shows the device number of this output signal.

Signal name	Device number
Operating condition settings batch-reset command	Y3

Hardware logic control start request

This signal is used to start the operation of the hardware logic. Because the operation of the hardware logic is stopped when the power supply is turned on, this signal needs to be turned on to start the operation.
When the auto trigger of the logic analyzer function is enabled, a trigger start request is issued at the same time as when the hardware logic control is started.
For the timing of turning on and off this signal, refer to the following.
☞ Page 259 Hardware logic control flag

■Device number

The following shows the device number of this output signal.

Signal name	Device number
Hardware logic control start request	Y4

Hardware logic control stop request

This signal is used to stop the operation of the hardware logic. When the operation of the hardware logic is stopped by turning on this signal, the count value is reset. To stop only the count operation without resetting the count value, set "OFF" for "Count Enable" of the counter timer block.

When the hardware logic control is stopped while Trigger ready (1) is stored in 'Trigger status monitor' (Un\G123), the logic analyzer function is stopped. In addition, Trigger stop (0) is stored in 'Trigger status monitor' (Un\G123).

For the timing of turning on and off this signal, refer to the following.

☞ Page 259 Hardware logic control flag

■Device number

The following shows the device number of this output signal.

Signal name	Device number
Hardware logic control stop request	Y5

Hardware logic control stop signal at disconnection

This signal is enabled when "Enable" is set for "Hardware logic control at disconnection enable/disable" in "Application setting".

Under the condition where the remote head module is not mounted, do not set "Enable" for "Hardware logic control at disconnection enable/disable" in "Application setting". If "Enable" is set, this signal needs to be on at the start of the hardware logic control.

■Starting the hardware logic control

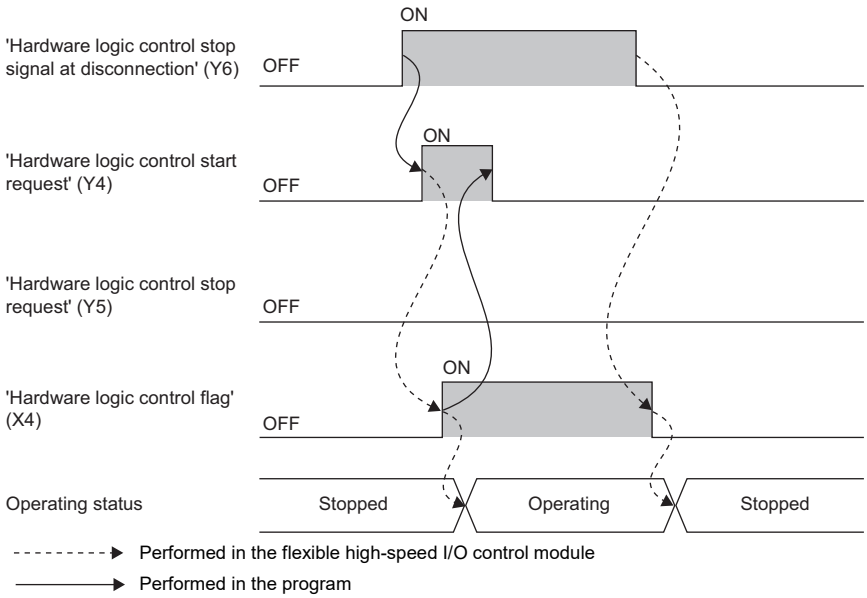
When "Hardware logic control at disconnection enable/disable" is set to "Enable" and both of the following conditions are satisfied, the hardware logic control is started.

- When this signal is on
- When 'Hardware logic control start request' (Y4) is turned on

■Stopping the hardware logic control

When any of the following conditions is satisfied, the hardware logic control is stopped.

- When this signal is turned off
- When 'Hardware logic control stop request' (Y5) is turned off while this signal is on



■Device number

The following shows the device number of this output signal.

Signal name	Device number
Hardware logic control stop signal at disconnection	Y6

Hardware logic control stop flag clear request at disconnection

This signal is enabled when "Enable" is set for "Hardware logic control at disconnection enable/disable" in "Application setting".

Turn on and off this signal to turn off 'Hardware logic control stop flag at disconnection' (X7).

For the timing of turning on and off this signal, refer to the following.

☞ Page 260 Hardware logic control stop flag at disconnection

■Device number

The following shows the device number of this output signal.

Signal name	Device number
Hardware logic control stop flag clear request at disconnection	Y7

No.☐ ADCValue logging area switching request

If trigger switching is set as the switching method for the ADCValue logging data storage area, when the logging side switches, this flag turns on and off again. No.☐ is the number of the logging condition of the A/D conversion logging function.

The device used depends on whether this output signal is No.1 or No.2.

For the timing of turning on and off this signal, refer to the following.

☞ Page 261 No.☐ ADCValue logging area switching request complete flag

■Device number

The following shows the device number of this output signal.

Signal name	Device number
No.1 ADCValue logging area switching request	YD
No.2 ADCValue logging area switching request	YE

Error clear request

Turn on and off this signal to clear 'Error flag' (XF), 'Latest error code' (Un\G100), and Error history No. ☐ (Un\G8010 to Un\G8169).

For the timing of turning on and off this signal, refer to the following.

☞ Page 262 Error flag

■Device number

The following shows the device number of this output signal.

Signal name	Device number
Error clear request	YF

General command 0 to General command F

Signals in the hardware logic can be controlled from a program.

Turning on or off 'General command 0 to General command F' (Y10 to Y1F) outputs the internal signals of the hardware logic from Y device terminals.

For details on the Y device terminals, refer to the following.

☞ Page 151 Y device terminal

■Device number

The following shows the device number of this output signal.

Signal name	Device number
General command 0 to General command F	Y10 to Y1F

Appendix 3 Buffer Memory Areas

List of buffer memory addresses

The following table lists the buffer memory addresses of the flexible high-speed I/O control module. For details on the buffer memory areas, refer to the following.

☞ Page 271 Details of buffer memory areas

The buffer memory areas of the flexible high-speed I/O control module are classified by the following data types.

Data type	Description	
Control data	Description	Use this data to control the flexible high-speed I/O control module.
	Read/write attribute	Data can be read and written from/to this area.
	Setting method	Set this data using an engineering tool or in a program.
	Setting timing	As soon as values are changed, the set values are enabled.
Monitor data	Description	Use this data to monitor the status of the flexible high-speed I/O control module.
	Read/write attribute	Reading data is only allowed. Writing data is not allowed.
	Setting method	—
	Setting timing	—
Setting data	Description	Set this data according to the connected device and the application of the system.
	Read/write attribute	Data can be read and written from/to this area.
	Setting method	Set this data using the engineering tool or the configuration tool.
	Setting timing	After changing values, turn on and off 'Hardware logic control start request' (Y4) to enable the set values.

Point

Do not write data to the system areas and areas whose data types are monitor in the buffer memory. Writing data into these areas can cause the malfunction of the module.

Address (decimal)	Address (hexadecimal)	Name	Default value*1	Data type	Auto refresh
0 to 99	0H to 63H	System area	—	—	—
100	64H	Latest error code	0	Monitor	○
101	65H	System area	—	—	—
102	66H	Cumulative number of write accesses to a flash ROM	0	Monitor	—
103	67H				
104 to 109	68H to 6DH	System area	—	—	—
110, 111	6EH, 6FH	SSI receive data monitor 0	0	Monitor	—
112, 113	70H, 71H	System area	—	—	—
114, 115	72H, 73H	SSI receive data monitor 1	0	Monitor	—
116 to 119	74H to 77H	System area	—	—	—
120	78H	Trigger start request	0	Control	—
121	79H	Trigger stop request	0	Control	—
122	7AH	Trigger setting status monitor	0	Monitor	—
123	7BH	Trigger status monitor	0	Monitor	—
124	7CH	Sampling data acquired flag	0	Monitor	—
125 to 127	7DH to 7FH	System area	—	—	—
128, 129	80H, 81H	Cumulative number of trigger setting writes	0	Monitor	—
130 to 139	82H to 8BH	System area	—	—	—
140	8CH	Analog input module mounting slot number setting	0	Setting	—
141 to 144	8DH to 90H	System area	—	—	—
145	91H	Analog input module mounting slot number setting monitor	0	Monitor	—
146 to 149	92H to 95H	System area	—	—	—
150	96H	No.1 ADCValue logging enable/disable	0	Setting	—
151	97H	No.1 ADCValue logging counter setting	0	Setting	—
152	98H	No.1 ADCValue logging channel setting	0	Setting	—

Address (decimal)	Address (hexadecimal)	Name	Default value*1	Data type	Auto refresh
153	99H	No.1 ADCValue logging target setting	0	Setting	—
154	9AH	No.1 ADCValue logging area switching method setting	0	Setting	—
155	9BH	No.1 ADCValue logging area switching condition setting	0	Setting	—
156	9CH	No.1 ADCValue logging area switching external input setting	0H	Setting	—
157 to 159	9DH to 9FH	System area	—	—	—
160	A0H	No.2 ADCValue logging enable/disable	0	Setting	—
161	A1H	No.2 ADCValue logging counter setting	0	Setting	—
162	A2H	No.2 ADCValue logging channel setting	0	Setting	—
163	A3H	No.2 ADCValue logging target setting	0	Setting	—
164	A4H	No.2 ADCValue logging area switching method setting	0	Setting	—
165	A5H	No.2 ADCValue logging area switching condition setting	0	Setting	—
166	A6H	No.2 ADCValue logging area switching external input setting	0H	Setting	—
167 to 169	A7H to A9H	System area	—	—	—
170	AAH	No.1 ADCValue logging enable/disable monitor	0	Monitor	—
171	ABH	No.1 ADCValue logging counter setting monitor	0	Monitor	—
172	ACH	No.1 ADCValue logging channel setting monitor	0	Monitor	—
173	ADH	No.1 ADCValue logging target setting monitor	0	Monitor	—
174	AEH	No.1 ADCValue logging area switching method setting monitor	0	Monitor	—
175	AFH	No.1 ADCValue logging area switching condition setting monitor	0	Monitor	—
176	B0H	No.1 ADCValue logging area switching external input setting monitor	0H	Monitor	—
177 to 179	B1H to B3H	System area	—	—	—
180	B4H	No.2 ADCValue logging enable/disable monitor	0	Monitor	—
181	B5H	No.2 ADCValue logging counter setting monitor	0	Monitor	—
182	B6H	No.2 ADCValue logging channel setting monitor	0	Monitor	—
183	B7H	No.2 ADCValue logging target setting monitor	0	Monitor	—
184	B8H	No.2 ADCValue logging area switching method setting monitor	0	Monitor	—
185	B9H	No.2 ADCValue logging area switching condition setting monitor	0	Monitor	—
186	BAH	No.2 ADCValue logging area switching external input setting monitor	0H	Monitor	—
187 to 189	BBH to BDH	System area	—	—	—
190	BEH	No.1 ADCValue logging data storage monitor	0	Monitor	—
191	BFH	No.1 ADCValue logging data A side storage flag	0	Control	—
192	C0H	No.1 ADCValue logging data B side storage flag	0	Control	—
193	C1H	No.1 ADCValue logging data A side storage count value	0	Monitor	—
194	C2H	No.1 ADCValue logging data B side storage count value	0	Monitor	—
195	C3H	No.1 ADCValue logging reset request	0	Control	—
196 to 199	C4H to C7H	System area	—	—	—
200	C8H	No.2 ADCValue logging data storage monitor	0	Monitor	—
201	C9H	No.2 ADCValue logging data A side storage flag	0	Control	—
202	CAH	No.2 ADCValue logging data B side storage flag	0	Control	—
203	CBH	No.2 ADCValue logging data A side storage count value	0	Monitor	—
204	CCH	No.2 ADCValue logging data B side storage count value	0	Monitor	—
205	CDH	No.2 ADCValue logging reset request	0	Control	—
206 to 699	CEH to 2BBH	System area	—	—	—
700 to 715	2BCH to 2CBH	Synchronization latch count value	0	Monitor	○
716 to 999	2CCH to 3E7H	System area	—	—	—
1000 to 1029	3E8H to 405H	Hardware logic area (High speed area)	0	Monitor/ control	—
1030 to 1099	406H to 44BH	Hardware logic area (Low speed area)	0	Monitor/ control	—
1100 to 7999	44CH to 1F3FH	System area	—	—	—
8000	1F40H	Latest address of error history	0	Monitor	○
8001	1F41H	System area	—	—	—
8002	1F42H	Clear setting of error history	0	Control	—

Address (decimal)	Address (hexadecimal)	Name				Default value*1	Data type	Auto refresh
8003 to 8009	1F43H to 1F49H	System area				—	—	—
8010	1F4AH	Error history No.1	Error code			0	Monitor	—
8011	1F4BH		Error time	First two digits of the year	Last two digits of the year			
8012	1F4CH			Month	Day			
8013	1F4DH			Hour	Minute			
8014	1F4EH			Second	Day of the week			
8015	1F4FH			Millisecond				
8016 to 8019	1F50H to 1F53H	System area				—	—	—
8020 to 8025	1F54H to 1F59H	Error history No.2	Same as Error history No.1			0	Monitor	—
8026 to 8029	1F5AH to 1F5DH	System area				—	—	—
8030 to 8035	1F5EH to 1F63H	Error history No.3	Same as Error history No.1			0	Monitor	—
8036 to 8039	1F64H to 1F67H	System area				—	—	—
8040 to 8045	1F68H to 1F6DH	Error history No.4	Same as Error history No.1			0	Monitor	—
8046 to 8049	1F6EH to 1F71H	System area				—	—	—
8050 to 8055	1F72H to 1F77H	Error history No.5	Same as Error history No.1			0	Monitor	—
8056 to 8059	1F78H to 1F7BH	System area				—	—	—
8060 to 8065	1F7CH to 1F81H	Error history No.6	Same as Error history No.1			0	Monitor	—
8066 to 8069	1F82H to 1F85H	System area				—	—	—
8070 to 8075	1F86H to 1F8BH	Error history No.7	Same as Error history No.1			0	Monitor	—
8076 to 8079	1F8CH to 1F8FH	System area				—	—	—
8080 to 8085	1F90H to 1F95H	Error history No.8	Same as Error history No.1			0	Monitor	—
8086 to 8089	1F96H to 1F99H	System area				—	—	—
8090 to 8095	1F9AH to 1F9FH	Error history No.9	Same as Error history No.1			0	Monitor	—
8096 to 8099	1FA0H to 1FA3H	System area				—	—	—
8100 to 8105	1FA4H to 1FA9H	Error history No.10	Same as Error history No.1			0	Monitor	—
8106 to 8109	1FAAH to 1FADH	System area				—	—	—
8110 to 8115	1FAEH to 1FB3H	Error history No.11	Same as Error history No.1			0	Monitor	—
8116 to 8119	1FB4H to 1FB7H	System area				—	—	—
8120 to 8125	1FB8H to 1FBDH	Error history No.12	Same as Error history No.1			0	Monitor	—
8126 to 8129	1FBEH to 1FC1H	System area				—	—	—
8130 to 8135	1FC2H to 1FC7H	Error history No.13	Same as Error history No.1			0	Monitor	—
8136 to 8139	1FC8H to 1FCBH	System area				—	—	—
8140 to 8145	1FCCH to 1FD1H	Error history No.14	Same as Error history No.1			0	Monitor	—
8146 to 8149	1FD2H to 1FD5H	System area				—	—	—
8150 to 8155	1FD6H to 1FDBH	Error history No.15	Same as Error history No.1			0	Monitor	—
8156 to 8159	1FDCH to 1FDFH	System area				—	—	—
8160 to 8165	1FE0H to 1FE5H	Error history No.16	Same as Error history No.1			0	Monitor	—
8166 to 8169	1FE6H to 1FE9H	System area				—	—	—
8170	1FEAH	RUN LED status monitor				0	Monitor	—
8171	1FEBH	ERR LED status monitor				0	Monitor	—
8172	1FECH	System area				—	—	—
8173	1FEDH	Synchronization status monitor				0	Monitor	○
8174 to 15007	1FEEH to 3A9FH	System area				—	—	—
15008	3AA0H	Continuous logging cycle setting				0	Control	—
15009	3AA1H	Continuous logging start/stop request				0	Control	—
15010	3AA2H	Continuous logging status monitor				0	Monitor	—
15011	3AA3H	Continuous logging cycle monitor (μs)				0	Monitor	—
15012	3AA4H	Continuous logging data A side storage flag				0	Control	—
15013	3AA5H	Continuous logging data B side storage flag				0	Control	—
15014, 15015	3AA6H, 3AA7H	Continuous logging data points				0	Monitor	—
15016 to 15019	3AA8H to 3AABH	System area				—	—	—

Address (decimal)	Address (hexadecimal)	Name	Default value*1	Data type	Auto refresh
15020 to 20139	3AACH to 4EABH	Continuous logging data storage area (A side)	0	Monitor	—
20140 to 25259	4EACH to 62ABH	Continuous logging data storage area (B side)	0	Monitor	—
25260 to 29999	62ACH to 752FH	System area	—	—	—
30000 to 34095	7530H to 852FH	No.1 ADCValue logging data storage area (A side)	0	Monitor	—
34096 to 38191	8530H to 952FH	No.1 ADCValue logging data storage area (B side)	0	Monitor	—
38192 to 42287	9530H to A52FH	No.2 ADCValue logging data storage area (A side)	0	Monitor	—
42288 to 46383	A530H to B52FH	No.2 ADCValue logging data storage area (B side)	0	Monitor	—
46384 to 65535	B530H to FFFFH	System area	—	—	—

*1 The default value to be set after the power is turned on or the CPU module is reset.

Details of buffer memory areas

The following section describes the details on the buffer memory areas of the flexible high-speed I/O control module.

Latest error code

This area stores the latest error code detected in the flexible high-speed I/O control module.

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

☞ Page 250 List of Error Codes

■How to clear an error

Turn on and off 'Error clear request' (YF).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Latest error code	100

Cumulative number of write accesses to a flash ROM

This area stores the cumulative number of writes to a flash ROM. When a hardware logic is written to the flash ROM of the flexible high-speed I/O control module with the configuration tool, the stored value is increased by one.

When the number of writes exceeds the allowable number of writes to the flash ROM (10000 times), the written hardware logic data cannot be assured. To decrease the number of writes to the flash ROM, write hardware logics to the execution memory for adjustment. After the adjustment is completed, write the hardware logic to the flash ROM.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Cumulative number of write accesses to a flash ROM	102, 103

SSI receive data monitor 0

Out of the data frames received from SSI encoder 0, the information for the number of bits specified with "Data Frame Length" is stored in this area at the communication cycle of the SSI encoder.

The parity bit is not reflected to this area.

When the data frame length is smaller than 32 bits, the least significant bit of the data frame is stored in the bit 0 of Un\G110.

For details, refer to the following.

☞ Page 156 SSI encoder block

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
SSI receive data monitor 0	110, 111

SSI receive data monitor 1

Out of the data frames received from SSI encoder 1, the information for the number of bits specified with "Data Frame Length" is stored in this area at the communication cycle of the SSI encoder.

The data to be stored in this area is the same as the one of 'SSI receive data monitor 0' (Un\G110, Un\G111).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
SSI receive data monitor 1	114, 115

Trigger start request

This area issues a trigger start request of the logic analyzer function.

When Start request (1) is set in 'Trigger start request' (Un\G120) during the hardware logic control after the trigger setting has been written to the flexible high-speed I/O control module, the trigger status of the flexible high-speed I/O control module is switched to the trigger ready. After the trigger status has been switched to the trigger ready, No request (0) is automatically stored in 'Trigger start request' (Un\G120).

Setting value	Setting details
0	No request
1	Start request

- When a value other than the above is set, the value is handled as No request (0).
- When Start request (1) is set in 'Trigger start request' (Un\G120) before the trigger setting is written to the flexible high-speed I/O control module, a trigger start error at trigger setting disabled (error code: 10C0H) occurs and the trigger status is not switched to the trigger ready.
- When Start request (1) is set in 'Trigger start request' (Un\G120) during simulation, a trigger start error during simulation execution (error code: 10C1H) occurs and the trigger status is not switched to the trigger ready.
- When Start request (1) is set in 'Trigger start request' (Un\G120) while Trigger ready (1) is stored in 'Trigger status monitor' (Un\G123), the trigger ready is continued.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Trigger start request	120

Trigger stop request

This area clears the trigger ready of the logic analyzer function.

When Stop request (1) is set in 'Trigger stop request' (Un\G121) while Trigger ready (1) is stored in 'Trigger status monitor' (Un\G123), the flexible high-speed I/O control module clears the trigger ready and the trigger status is switched to the trigger stop. After the status has been switched to the trigger stop, No request (0) is automatically stored in 'Trigger stop request' (Un\G121).

Setting value	Setting details
0	No request
1	Stop request

- When a value other than the above is set, the value is processed as No request (0).
- When Stop request (1) is set in 'Trigger stop request' (Un\G121) while Trigger stop (0) is stored in 'Trigger setting status monitor' (Un\G122), the trigger stop is continued.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Trigger stop request	121

Trigger setting status monitor

This area stores the trigger setting status of the logic analyzer function.

Stored value	Description
0	Trigger setting disable
1	Trigger setting enable (Auto trigger disable)
2	Trigger setting enable (Auto trigger enable)

- Write the trigger setting to the flexible high-speed I/O control module with the configuration tool. (☞ Page 131 Logic analyzer function)
- If the trigger setting is written to the flash ROM, the trigger setting is read from the flash ROM at power-on of the module. As a result, Trigger setting enable (Auto trigger disable) (1) or Trigger setting enable (Auto trigger enable) (2) is stored in this area at power-on of the module.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Trigger setting status monitor	122

Trigger status monitor

This area stores the trigger status of the logic analyzer function.

Stored value	Description
0	Trigger stop
1	Trigger ready

■Storing Trigger stop (0)

This value indicates a status in which generating sampling data by the logic analyzer function is stopped because of any of the following conditions.

- After the trigger condition is satisfied, acquisition of sampling data is completed.
- Stop request (1) is stored in 'Trigger stop request' (Un\G121).
- A trigger stop request from the configuration tool is received.
- The trigger is stopped due to simulation execution.
- The hardware logic control is stopped by a hardware logic control stop request from the configuration tool or 'Hardware logic control stop request' (Y5).
- "Write to Module (execution memory)" or "Write to Module (execution + flash ROM)" is executed with the configuration tool.

■Storing Trigger ready (1)

This value indicates a status in which sampling data is being generated until the trigger condition is satisfied after a trigger start request is received because of any of the following conditions.

- Start request (1) is stored in 'Trigger start request' (Un\G120) during the hardware logic control.
- A trigger start request from the configuration tool is received during the hardware logic control.
- When the auto trigger is enabled, the hardware logic control is started by a hardware logic control start request from the configuration tool or 'Hardware logic control start request' (Y4).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Trigger status monitor	123

Sampling data acquired flag

This area stores the sampling data acquisition status of the logic analyzer function.

Stored value	Description
0	Sampling data is not acquired
1	Sampling data is acquired

- When the trigger is detected while Trigger ready (1) is stored in 'Trigger status monitor' (Un\G123), the flexible high-speed I/O control module generates sampling data. After the generation is completed, Sampling data is acquired (1) is stored in 'Sampling data acquired flag' (Un\G124).
- While Sampling data is acquired (1) is stored in 'Sampling data acquired flag' (Un\G124), sampling data can be read.
- When the value in 'Trigger start request' (Un\G120) is changed from No request (0) to Start request (1), Sampling data is not acquired (0) is stored in 'Sampling data acquired flag' (Un\G124).
- When the simulation is executed, Sampling data is not acquired (0) is stored in 'Sampling data acquired flag' (Un\G124).
- When "Write to Module (execution memory)" or "Write to Module (execution + flash ROM)" is executed with the configuration tool, Sampling data is not acquired (0) is stored in 'Sampling data acquired flag' (Un\G124).
- When 'Operating condition settings batch-reset command' (Y3) is turned on, Sampling data is not acquired (0) is stored in 'Sampling data acquired flag' (Un\G124).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Sampling data acquired flag	124

Cumulative number of trigger setting writes

This area stores the cumulative number of writes of the trigger setting to a flash ROM. When the trigger setting is written to the flash ROM of the flexible high-speed I/O control module with the setting tool, the stored value is increased by one. In addition, when "Trigger Setting Delete" is clicked in the "Logic Analyzer" window of the configuration tool, the stored value is increased by one.

When the number of writes exceeds the allowable number of writes to the flash ROM (10000 times), the written trigger setting data cannot be assured. To decrease the number of writes to the flash ROM, write trigger settings to the execution memory for adjustment. After the adjustment is completed, write the trigger setting to the flash ROM.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Cumulative number of trigger setting writes	128, 129

Analog input module mounting slot number setting

Set the slot number of the analog input module on the same base to be used for the A/D conversion value logging function.

■Setting range

The setting range is from 0 to 11. Set the slot number where the target analog input module is mounted.

If hardware logic control is started with the following settings, an analog input module mounting slot number setting error (error code: 10E1H) occurs.

- An empty slot number in this area or the slot number of a module that is not the target analog input module is set.
- Enable (1) is set in 'No.☐ ADCValue logging enable/disable' (Un\G150, Un\G160).

■Default value

The default value is 0.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Analog input module mounting slot number setting	140

Analog input module mounting slot number setting monitor

The setting status of 'Analog input module mounting slot number setting' (Un\G140) can be monitored.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Analog input module mounting slot number setting monitor	145

No.☐ ADCValue logging enable/disable

Set whether to enable or disable A/D conversion value logging.

This area is set using a different buffer memory depending on whether the signal is No.1 or No.2.

Setting value	Setting details
0	Disable
1	Enable

If a value other than the above is set, the value is regarded as Enable (1).

■Default value

The default value is Disable (0).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging enable/disable	150
No.2 ADCValue logging enable/disable	160

No.☐ ADCValue logging counter setting

Specify the multi function counter block to be used for the A/D conversion value logging function.
This area is set using a different buffer memory depending on whether the signal is No.1 or No.2.

■Setting range

The setting range is from 0 to 7. Set the number of the multi function counter block to be used.
If a value outside the setting range is set and hardware logic control is started, a No.☐ ADCValue logging counter setting error (error code: 110☐H) occurs. The A/D conversion value logging function for the No. where the error occurred is not executed.

■Default value

The default value is Counter 0 (0).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging counter setting	151
No.2 ADCValue logging counter setting	161

No.☐ ADCValue logging channel setting

Set the channel number of the analog input module to be used for the A/D conversion value logging function.
This area is set using a different buffer memory depending on whether the signal is No.1 or No.2.

■Setting range

The setting range varies depending on the target analog input module.

Analog input module	Setting range	Channel number corresponding to setting value
R60AD4, R60ADH4	0 to 3	0: CH1 1: CH2 2: CH3 3: CH4 4: CH5 5: CH6 6: CH7 7: CH8
R60ADV8, R60ADI8	0 to 7	

If a value outside the above setting range is set and hardware logic control is started, a No.☐ ADCValue logging channel setting error (error code: 111☐H) occurs. The A/D conversion value logging function for the No. where the error occurred is not executed.

■Default value

The default value is CH1 (0).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging channel setting	152
No.2 ADCValue logging channel setting	162

No.☐ ADCValue logging target setting

Set the logging target data of the analog input module to be used for the A/D conversion value logging function.
This area is set using a different buffer memory depending on whether the signal is No.1 or No.2.

Setting value	Setting details
0	Digital output value
1	Digital operation value

If a value other than the above is set, the value is regarded as Digital operation value (1).
Depending on the analog input module used, digital output values may not be output. Even if A/D conversion value logging is executed in this state, no error occurs. The value for A/D conversion value logging is always 0.

■Default value

The default value is Digital output value (0).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging target setting	153
No.2 ADCValue logging target setting	163

No.☐ ADCValue logging area switching method setting

Set the switching method of the ADCValue logging data storage area (A side/B side) to be used for the A/D conversion value logging function.
This area is set using a different buffer memory depending on whether the signal is No.1 or No.2.

Setting value	Setting details
0	Automatic switching
1	Trigger switching

If a value other than the above is set, the value is regarded as Trigger switching (1).
For details on the switching methods, refer to the following.
☞ Page 37 Operation during A/D conversion value logging

■Default value

The default value is Automatic switching (0).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging area switching method setting	154
No.2 ADCValue logging area switching method setting	164

No.☐ ADCValue logging area switching condition setting

If Trigger switching (1) is set to 'No.☐ ADCValue logging area switching method setting' (Un\G154, Un\G164), set the switching condition for the ADCValue logging data storage area.
This area is set using a different buffer memory depending on whether the signal is No.1 or No.2.

Setting value	Setting details
0	Disable external input
1	Enable external input

- If a value other than the above is set, the value is regarded as Enable external input (1).
- By setting Enable external input (1) in this area the ADCValue logging data storage area can be switched by turning on the external input block. Specification of the external input block to be used is set with 'No.☐ ADCValue logging area switching external input setting' (Un\G156, Un\G166).
- By setting Disable external input (0) in this area the ADCValue logging data storage area is switched by turning on 'No.☐ ADCValue logging area switching request' (YD, YE).

■Default value

The default value is Disable external input (0).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging area switching condition setting	155
No.2 ADCValue logging area switching condition setting	165

No.☐ ADCValue logging area switching external input setting

When Enable external input (1) is set for 'No.☐ ADCValue logging area switching condition setting' (Un\G155, Un\G165), set the external input block to be used for switching.
This area is set using a different buffer memory depending on whether the signal is No.1 or No.2.

■Setting range

The setting range is from 0H to BH. Set the number of the external input block to be used.
If a value outside the setting range is set and hardware logic control is started, a No.☐ ADCValue logging area switching external input setting error (error code: 112☐H) occurs. However, the setting of 'No.☐ ADCValue logging area switching condition setting' (Un\G155, Un\G165) for the No. where the error occurred is regarded as Disable external input (0).

■Default value

The default value is IN 0 (0H).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging area switching external input setting	156
No.2 ADCValue logging area switching external input setting	166

No.☐ ADCValue logging enable/disable monitor

During hardware logic control, the setting status of 'No.☐ ADCValue logging enable/disable' (Un\G150, Un\G160) can be monitored.

This area is set using a different buffer memory depending on whether the signal is No.1 or No.2.

Stored value	Description
0	Disabled
1	Enabled

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging enable/disable monitor	170
No.2 ADCValue logging enable/disable monitor	180

No.☐ ADCValue logging counter setting monitor

During hardware logic control, the setting status of 'No.☐ ADCValue logging counter setting' (Un\G151, Un\G161) can be monitored. If Disable (0) is set in 'No.☐ ADCValue logging enable/disable' (Un\G150, Un\G160), 0 is stored.

Stored value	Description
0	Counter_0
1	Counter_1
2	Counter_2
3	Counter_3
4	Counter_4
5	Counter_5
6	Counter_6
7	Counter_7

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging counter setting monitor	171
No.2 ADCValue logging counter setting monitor	181

No.☐ ADCValue logging channel setting monitor

During hardware logic control, the setting status of 'No.☐ ADCValue logging channel setting' (Un\G152, Un\G162) can be monitored. If Disable (0) is set in 'No.☐ ADCValue logging enable/disable' (Un\G150, Un\G160), 0 is stored.

Stored value	Description
0	CH1
1	CH2
2	CH3
3	CH4
4	CH5
5	CH6
6	CH7
7	CH8

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging channel setting monitor	172
No.2 ADCValue logging channel setting monitor	182

No.☐ ADCValue logging target setting monitor

During hardware logic control, the setting status of 'No.☐ ADCValue logging target setting' (Un\G153, Un\G163) can be monitored. If Disable (0) is set in 'No.☐ ADCValue logging enable/disable' (Un\G150, Un\G160), 0 is stored.

Stored value	Description
0	Digital output value
1	Digital operation value

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging target setting monitor	173
No.2 ADCValue logging target setting monitor	183

No.☐ ADCValue logging area switching method setting monitor

During hardware logic control, the setting status of 'No.☐ ADCValue logging area switching method setting' (Un\G154, Un\G164) can be monitored. If Disable (0) is set in 'No.☐ ADCValue logging enable/disable' (Un\G150, Un\G160), 0 is stored.

Stored value	Description
0	Automatic switching
1	Trigger switching

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging area switching method setting monitor	174
No.2 ADCValue logging area switching method setting monitor	184

No.☐ ADCValue logging area switching condition setting monitor

During hardware logic control, the setting status of 'No.☐ ADCValue logging area switching condition setting' (Un\G155, Un\G165) can be monitored. If Disable (0) is set in 'No.☐ ADCValue logging enable/disable' (Un\G150, Un\G160), 0 is stored.

Stored value	Description
0	Disable external input
1	Enable external input

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging area switching condition setting monitor	175
No.2 ADCValue logging area switching condition setting monitor	185

No.☐ ADCValue logging area switching external input setting monitor

During hardware logic control, the setting status of 'No.☐ ADCValue logging area switching external input setting' (Un\G156, Un\G166) can be monitored.

Stored value	Description
0H	IN 0
1H	IN 1
2H	IN 2
3H	IN 3
4H	IN 4
5H	IN 5
6H	IN 6
7H	IN 7
8H	IN 8
9H	IN 9
AH	IN A
BH	IN B

In the following cases, 0 is stored.

- Disable (0) is set in 'No.☐ ADCValue logging enable/disable' (Un\G150, Un\G160).
- Disable external input (0) is set in 'No.☐ ADCValue logging area switching condition setting' (Un\G155, Un\G165).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging area switching external input setting monitor	176
No.2 ADCValue logging area switching external input setting monitor	186

No.☐ ADCValue logging data storage monitor

Whether the A/D conversion logging in progress is storing on A side or B side can be checked.

This area is monitored using a different buffer memory depending on whether the signal is No.1 or No.2.

Stored value	Description
0	Logging stopped (hardware logic control stopped)
1	A side storage in progress
2	B side storage in progress

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging data storage monitor	190
No.2 ADCValue logging data storage monitor	200

No.☐ ADCValue logging data storage flag

During A/D conversion value logging, whether the storage of logging data to the ADCValue logging data storage area has been completed can be checked.

This area has a flag for checking that data is stored in the first 4096 points (A side) and a flag to check that data is stored in the last 4096 points (B side) of the ADCValue logging data storage area.

Stored value	Description
0	Storage not completed
1	Storage completed

The operation of this area differs depending on the setting of 'No.☐ ADCValue logging area switching method setting' (Un\G154, Un\G164).

- If Automatic switching (0) is set, then Storage completed (1) is stored every time logging data for the first 4096 points (A side) or the last 4096 points (B side) has been stored.
- If Trigger switching (1) is set, storage completed (1) is stored every time the storage location is switched.
- When logging data storage in the file register of the CPU module is completed, set Storage not completed (0) in either of the areas to accept the next storage flag.

■Default value

The default value is Storage not completed (0).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging data A side storage flag	191
No.1 ADCValue logging data B side storage flag	192
No.2 ADCValue logging data A side storage flag	201
No.2 ADCValue logging data B side storage flag	202

No.☐ ADCValue logging data storage count value

The number of data logged in the ADCValue logging data storage area during the execution of A/D conversion logging is counted for each logging side.

■Count timing

- Each time A/D conversion value logging is executed, 1 is added to this area.
- If automatic switching is set as the ADCValue logging data storage area switching method, the value of this switching destination area becomes 1 when the first A/D conversion value logging is executed after the logging side is switched.

■Reset timing

The stored data count value is reset to 0 under the following conditions.

- Upon transitioning from hardware logic control stop to control start
- When 'No.☐ ADCValue logging reset request' (Un\G195, Un\G205) is set from No request (0) to Reset request (1)
- When the logging side is switched after trigger switching has been set as the ADCValue logging data storage area switching method

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging data A side storage count value	193
No.1 ADCValue logging data B side storage count value	194
No.2 ADCValue logging data A side storage count value	203
No.2 ADCValue logging data B side storage count value	204

No.☐ ADCValue logging reset request

The status of A/D conversion value logging can be reset without stopping hardware logic control.

If No request (0) is changed to Reset request (1) for this area, A/D conversion value logging restarts from the beginning of ADCValue logging data storage area A side.

In addition, 'No.☐ ADCValue logging data A side storage count value' (Un\G193, Un\G203) and 'No.☐ ADCValue logging data B side storage count value' (Un\G194, Un\G204) are reset to 0.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging reset request	195
No.2 ADCValue logging reset request	205

Synchronization latch count value

This area stores the latch count value of the counter timer for each inter-module synchronization cycle. When this area is referred to in an inter-module synchronous interrupt program, the latch count value synchronized among multiple modules can be acquired.

When the inter-module synchronization function is not used, the default value (0) is stored.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory address	Description	
	Multi function counter block	Bit arrangement
700	Counter_0	Lower 16 bits
701		Upper 16 bits
702	Counter_1	Lower 16 bits
703		Upper 16 bits
704	Counter_2	Lower 16 bits
705		Upper 16 bits
706	Counter_3	Lower 16 bits
707		Upper 16 bits
708	Counter_4	Lower 16 bits
709		Upper 16 bits
710	Counter_5	Lower 16 bits
711		Upper 16 bits
712	Counter_6	Lower 16 bits
713		Upper 16 bits
714	Counter_7	Lower 16 bits
715		Upper 16 bits

Hardware logic area

The monitor items and setting items of the hardware logic can be assigned to buffer memory addresses.

There are two types of areas for the buffer memory areas: High speed area and low speed area. For each area, the addresses are assigned as follows.

Area	Buffer memory address	Read/write cycle	
		Input terminal and parameter	Monitor
High speed area	Un\G1000 to Un\G1029	• 100μs cycle (Normal operation mode) • Immediately after the completion of a synchronous interrupt program (Inter-module synchronous mode)	• 100μs cycle (Normal operation mode) • 1ms cycle (Inter-module synchronous mode)
Low speed area	Un\G1030 to Un\G1099	1ms	1ms

These items are also read or written when an interrupt signal is sent to the CPU module.

■Assignable monitor items and setting items

The following items can be assigned to these areas.

Block	Type	Variable name	R/W*1	Setting value/ Stored value	Description
Counter timer block	Input terminal	RUN/STOP	R/W	0	Both the "RUN" and "STOP" terminals turn to Low.
				1	The "RUN" terminal turns to High and the "STOP" terminal turns to Low.
				2	The "RUN" terminal turns to Low and the "STOP" terminal turns to High.
				3	Both the "RUN" and "STOP" terminals turn to High.
		PRESET	R/W	0	The "PRESET" terminal turns to Low.
				1	The "PRESET" terminal turns to High.
	Parameter	Count Enable	R/W	0	Count Enable is off.
				1	Count Enable is on.
		Upper Limit*2	R/W	*3	Set the upper limit value.
		Lower Limit*2	R/W	*3	Set the lower limit value.
		Add Value*2	R/W	*3	Set the addition value.
		Preset Value*2	R/W	*3	Set the preset value.
	Monitor	Count Value*2	R	*3	A count value is stored.
		Latch Value*2	R	*3	A latch value is stored.
		Overflow*4	R	0	No overflow occurs.
				1	An overflow has occurred.
		Underflow*4	R	0	No underflow occurs.
				1	An underflow has occurred.
Cam switch block	Parameter	Step No.0 ⋮ Step No.15*2	R/W	*3	Step No.0 ⋮ Step No.15
Comparison block	Parameter	Compare Value	R/W	*3	Value compared with a count value of the counter timer
External output block	Parameter	Enable Forced Output	R/W	0	The forced output is disabled.
				1	The forced output is enabled.
	Parameter	Forced Output	R/W	0	The forced output status of the external terminal is off (0: OFF).
				1	The forced output status of the external terminal is on (1: ON).
	Monitor	External terminal monitor	R	0	The output status (0: OFF) of the external output terminal is stored.
				1	The output status (1: ON) of the external output terminal is stored.

*1 The output values of each block in the hardware logic are "R" and the input values of each block in the hardware logic are "R/W".

*2 Assign a parameter of two words (32 bits) to an even address.

*3 The setting and storage ranges are determined depending on the type of a counter timer block.

*4 The item cannot be assigned in a 16-bit counter timer block.

Latest address of error history

This area stores the buffer memory address which has the latest error code among the addresses of Error history No. ☐ (Un\G8010 to Un\G8169).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Latest address of error history	8000

Clear setting of error history

Set whether to clear an error history when 'Error clear request' (YF) is turned on.

Clear setting of error history	Setting value
Do not clear the history.	0
Clear the history.	1

■Enabling the setting

The setting is enabled immediately after a value is set in the buffer memory area.

■Default value

The default value is Do not clear the history. (0).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Clear setting of error history	8002

Error history

This area stores up to 16 errors that have occurred in the module.

For details on the error history function, refer to the following.

☞ Page 76 Error History Function

	b15 to b8	b7 to b0
Un\G8010	Error code	
Un\G8011	First two digits of the year	Last two digits of the year
Un\G8012	Month	Day
Un\G8013	Hour	Minute
Un\G8014	Second	Day of the week
Un\G8015	Millisecond (higher-order digits)	Millisecond (lower-order digits)
Un\G8016 to Un\G8019	System area	

Item	Description	Storage example*1
First two digits of the year/last two digits of the year	Stored in a BCD code.	2016H
Month/day		1031H
Hour/minute		1234H
Second		56H
Day of the week	For each day of the week, one of the following values is stored in a BCD code. Sunday: 0H, Monday: 1H, Tuesday: 2H, Wednesday: 3H, Thursday: 4H, Friday: 5H, Saturday: 6H	1H
Millisecond (upper)/millisecond (lower)	Stored in a BCD code.	0789H
System area	—	—

*1 Value of when a reset error during hardware logic control (error code: 1020H) occurred at 12:34:56.789 on Monday, October 31st, 2016

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Error history No. ☐	8010 to 8169

RUN LED status monitor

This area stores the current status of the RUN LED.

RUN LED status	Stored value	Description
Off	0	The RUN LED is off.
On	1	The RUN LED is on.
Flashing (1s cycles)	2	The RUN LED is flashing (1s cycles).
Flashing (400ms cycles)	3	The RUN LED is flashing (400ms cycles).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
RUN LED status monitor	8170

ERR LED status monitor

This area stores the current status of the ERR LED.

ERR LED status	Stored value	Description
Off	0	The ERR LED is off.
On	1	The ERR LED is on.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
ERR LED status monitor	8171

Synchronization status monitor

Whether this module has been set for a target or non-target of the inter-module synchronization function can be checked. The inter-module synchronization status (during synchronization or synchronization suspended) can also be checked.

Any of the following values is stored depending on the parameter setting of GX Works3 and the operating status of the flexible high-speed I/O control module.

Target or non-target of inter-module synchronization	Operating status of the flexible high-speed I/O control module	Stored value	Description
Non-target	—	0	Not the inter-module synchronization target
Target	During hardware logic control stop	1	Inter-module synchronization target (synchronization suspended)
	During hardware logic control	2	Inter-module synchronization target (during synchronization)

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Synchronization status monitor	8173

Continuous logging cycle setting

Set the logging cycle for the continuous logging function.

Setting value	Setting details
0	1μs
1	10μs
2	100μs
3	1000μs

If a value other than the above is set, the continuous logging cycle setting range error (error code: 10D6H) occurs and the continuous logging does not start.

■Default value

The default value is 1μs (0).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Continuous logging cycle setting	15008

Continuous logging start/stop request

Set whether to start or stop continuous logging.

Setting value	Setting details
0	Stop request
1	Start request

- Changing 'Continuous logging start/stop request' (Un\G15009) from Stop request (0) to Start request (1) starts the continuous logging.
- Changing 'Continuous logging start/stop request' (Un\G15009) from Start request (1) to Stop request (0) stops the continuous logging.
- If a value other than the above is set, the request is ignored and the operation of continuous logging does not change.

■Default value

The default value is Stop request (0).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Continuous logging start/stop request	15009

Continuous logging status monitor

The execution status of the continuous logging function is stored.

Stored value	Description
0	Disabled
1	Start request waiting
2	In progress

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Continuous logging status monitor	15010

Continuous logging cycle monitor (μs)

The logging cycle (unit: μs) for the continuous logging function is stored.

Ex.

When 1000μs (3) is set in 'Continuous logging cycle setting' (Un\G15008), 1000 is stored in this area.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Continuous logging cycle monitor (μs)	15011

Continuous logging data storage flag

Use this buffer memory area to check that data of 5120 points is stored in continuous logging data storage areas during continuous logging.

These areas consist of 'Continuous logging data A side storage flag' (Un\G15012) for checking that data is stored in the first 5120 points area (A side) and 'Continuous logging data B side storage flag' (Un\G15013) for checking that data is stored in the last 5120 points area (B side).

Setting value	Setting details
0	Storage not completed
1	Storage completed

- Storage completed (1) is stored every time logging data for the first 5120 points (A side) or the last 5120 points (B side) has been stored.
- When logging data storage in the file register of the CPU module is completed, set Storage not completed (0) in either of the areas to accept the next storage flag.
- When 'Continuous logging start/stop request' (Un\G15009) is changed from Stop request (0) to Start request (1), these areas are cleared to 0.

■Default value

The default value is Storage not completed (0).

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Continuous logging data A side storage flag	15012
Continuous logging data B side storage flag	15013

Continuous logging data points

The number of logging data points stored in continuous logging data storage areas can be checked during continuous logging.

- After continuous logging is started, 'Continuous logging data points' (Un\G15014, Un\G15015) increases by 64 every time logging data of 64 points is stored. When the data count reaches 3600000000, 'Continuous logging data points' (Un\G15014, Un\G15015) returns to 0 and then increases by 64 again.
- When 'Continuous logging start/stop request' (Un\G15009) is changed from Stop request (0) to Start request (1), these areas are cleared to 0.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Continuous logging data points	15014, 15015

Continuous logging data storage area

These areas store the data logged with the continuous logging function. Up to 10240 points of data can be stored.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
Continuous logging data storage area (A side)	15020 to 20139
Continuous logging data storage area (B side)	20140 to 25259

No.☐ ADCValue logging data storage area

These areas store the data logged with the A/D conversion value logging function. Up to 4096 points of data can be stored in each of A side and B side.

This area uses different buffer memory depending on whether the signal is No.1 or No.2.

■Buffer memory address

The following shows the buffer memory address of this area.

Buffer memory name	Buffer memory address
No.1 ADCValue logging data storage area (A side)	30000 to 34095
No.1 ADCValue logging data storage area (B side)	34096 to 38191
No.2 ADCValue logging data storage area (A side)	38192 to 42287
No.2 ADCValue logging data storage area (B side)	42288 to 46383

Appendix 4 Logic Analyzer Function (Executed with a Program)

The logic analyzer function verifies the hardware logic control in the flexible high-speed I/O control module during the system operation. Users can check that the hardware logic written to the module operates properly by using the signals input to the external input terminals of the flexible high-speed I/O control module.
If a trigger condition is set and the trigger start is executed with the configuration tool, the logic analyzer function can be used.
The trigger start can be controlled with a program. Users can start acquisition of sampling data (trigger start) or stop acquisition of sampling data (trigger stop) at a desired timing by performing the control with a program.
Users can acquire sampling data repeatedly by using module function blocks.

Items that can be controlled with a program

The following table lists the items that can be controlled with a program.

Item	Overview
Trigger start	Enables the trigger condition set with the configuration tool. If the trigger condition is satisfied, sampling data is generated in the module.
Trigger stop	Disables the trigger condition enabled by the trigger start.

Buffer memory areas to be used

The following table lists the buffer memory areas to be used for the control of the logic analyzer function.

Buffer memory address	Name	Data type
Un\G120	Trigger start request	Control
Un\G121	Trigger stop request	Control
Un\G122	Trigger setting status monitor	Monitor
Un\G123	Trigger status monitor	Monitor
Un\G124	Sampling data acquired flag	Monitor

For details on the buffer memory areas, refer to the following.
☞ Page 271 Details of buffer memory areas

Setting method

Set the trigger condition in the "Logic Analyzer" window of the configuration tool in advance. For details, refer to the following.
☞ Page 131 Logic analyzer function

■Trigger start request

When Start request (1) is set in 'Trigger start request' (Un\G120) while the hardware logic control is executed and Trigger stop (0) is stored in 'Trigger status monitor' (Un\G123), a trigger start request is issued.
If Trigger ready (1) is stored in 'Trigger status monitor' (Un\G123) after the trigger start request is issued, sampling data generation starts.

Point

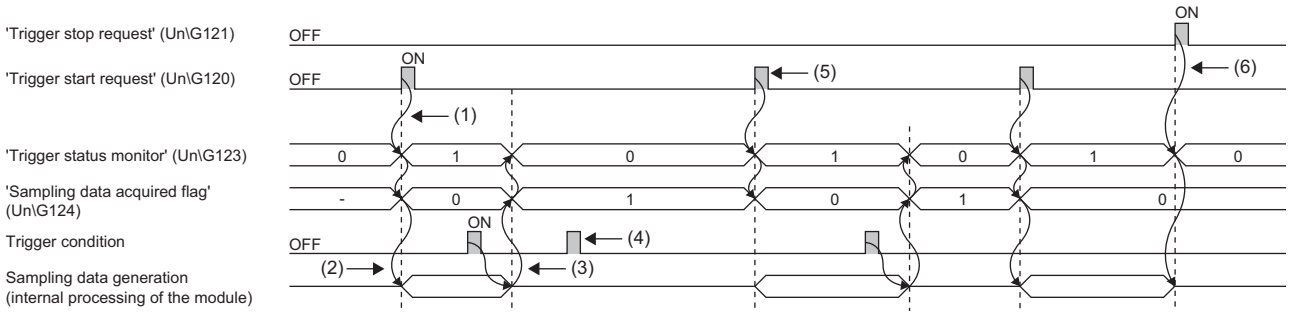
When the auto trigger is enabled, a trigger start request is issued automatically at the same time as when the hardware logic control is started. Thus, issuing a trigger start request is not required.
When the auto trigger is disabled or sampling data is acquired again after acquisition of sampling data resulting from the trigger condition satisfied after the trigger start, issue a trigger start request.

■Trigger stop request

When Stop request (1) is set in 'Trigger stop request' (Un\G121) while Trigger ready (1) is stored in 'Trigger status monitor' (Un\G123), a trigger stop request is issued.
After the trigger stop, sampling data is not generated in the module even if the trigger condition is satisfied before a trigger start request is issued.

Operation

The following shows the operations of the logic analyzer function.



(1) When a trigger start request is issued, the trigger status is switched to the trigger ready. (☞ Page 294 Trigger start request)
(2) Sampling data is generated at the same time as when the trigger status is switched to the trigger ready.
(3) After the trigger condition is satisfied, acquisition of the sampling data is completed, and then the trigger status is switched to the trigger stop. (☞ Page 295 Sampling data acquisition completion)
(4) Even if the trigger condition is satisfied during the trigger stop, sampling data is not generated.
(5) When executing sampling repeatedly, issue a trigger start request again.
(6) When a trigger stop request is issued in the trigger ready, acquisition of sampling data is stopped. (☞ Page 295 Trigger stop request)

■Trigger start request

When Start request (1) is set in 'Trigger start request' (Un\G120) while the hardware logic control is being executed and Trigger stop (0) is stored in 'Trigger status monitor' (Un\G123), a trigger start request is issued.
After the trigger start request is received, Trigger ready (1) is stored in 'Trigger status monitor' (Un\G123) and Sampling data is not acquired (0) is stored in 'Sampling data acquired flag' (Un\G124). Sampling data is generated in a module according to the sampling period set with the configuration tool.

Point

- When Start request (1) is set in 'Trigger start request' (Un\G120) before a trigger condition is set, a trigger start error at trigger setting disabled (error code: 10C0H) occurs.
- When the hardware logic control is stopped in the trigger ready, the trigger status is switched to the trigger stop.
- When Start request (1) is set in 'Trigger start request' (Un\G120) during simulation, a trigger start error during simulation execution (error code: 10C1H) occurs. In addition, the trigger status is not switched to the trigger ready.

■Sampling data acquisition completion

When the trigger condition is satisfied while Trigger ready (1) is stored in 'Trigger status monitor' (Un\G123), sampling data is generated according to the trigger setting. Trigger stop (0) is stored in 'Trigger status monitor' (Un\G123) and Sampling data is acquired (1) is stored in 'Sampling data acquired flag' (Un\G124).
Sampling data can be read only when Sampling data is acquired (1) is stored in 'Sampling data acquired flag' (Un\G124).

Point

Sampling data generated before the trigger start is discarded by a trigger start request or execution of the simulation function.
When any of the following operations is performed, Sampling data is not acquired (0) is stored in 'Sampling data acquired flag' (Un\G124).

- Executing "Write to Module (execution memory)" with the configuration tool
- Executing "Write to Module (execution + flash ROM)" with the configuration tool
- Turning on 'Operating condition settings batch-reset command' (Y3)

If sampling data is acquired repeatedly, read the sampling data, and then issue a trigger start request.

■Trigger stop request

When Stop request (1) is set in 'Trigger stop request' (Un\G121) while Trigger ready (1) is stored in 'Trigger status monitor' (Un\G123), a trigger stop request is issued.
After Trigger stop request is received, Trigger stop (0) is stored in 'Trigger status monitor' (Un\G123). The sampling data generation stops in a module.
After the trigger stop, sampling data is not generated until a trigger start request is issued again even if the trigger condition is satisfied.

Point

- When simulation is executed with the configuration tool while the logic analyzer function is operating (while Trigger ready (1) is stored in 'Trigger status monitor' (Un\G123)), the logic analyzer function stops automatically. At a logic analyzer function stop, Trigger stop (0) is stored in 'Trigger status monitor' (Un\G123). The logic analyzer function will not restart automatically after the simulation completion. To restart the logic analyzer function, issue a trigger start request again.
- When "Write to Module (execution memory)" or "Write to Module (execution + flash ROM)" is executed with the configuration tool while the logic analyzer function is operating (while Trigger ready (1) is stored in 'Trigger status monitor' (Un\G123)), the hardware logic control is suspended, and thus the logic analyzer stops. When the auto trigger is enabled, the logic analyzer restarts after writing data into the module is completed.

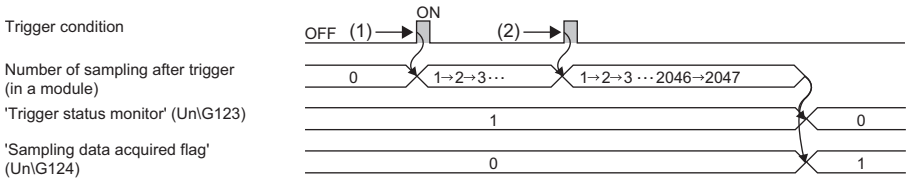
When the trigger condition is satisfied twice or more times

If the trigger condition is satisfied again while the sampling for the specified points is being performed with the previous trigger as the starting point, next sampling is performed for the points specified in "After Trigger Point" with the subsequent trigger as the starting point.

Ex.

The figure below shows an example when the second trigger condition is satisfied while the sampling with the first trigger is being performed. Sampling is performed with the following settings.

- "Sampling Point": 2048
- "Before Trigger Point": 1
- "After Trigger Point": 2047



(1) The sampling starts at the timing when the first trigger condition is satisfied.
(2) The next sampling is performed for the points specified in "After Trigger Point" with the second trigger.

Appendix 5 Troubleshooting When Using the Configuration Tool

This section describes the errors which may occur when using the configuration tool and corrective actions for the errors.

Symptom	Check item	Action
In Windows 10, text sometimes cannot be input using a keyboard.	For Windows 10 (version 2004 or later)*1, check the settings of IME mode.	❶ Click [Time & Language] ⇨ [Language] in [SETTINGS] of Windows. ❷ In the right pane on the window, click [Japanese] in "Preferred languages" in the [Language] window, and click the [Options] button. ❸ In the [Language Options: Japanese] window, click [Microsoft IME] in "Keybords", and click the [Options] button. ❹ Click [General] in the [Microsoft IME] window. Scroll down to "Compatibility" in the [General] window, and change the switch for the [Use previous version of Microsoft IME] to "On". ❺ When the [Change IME version] window is displayed, click the [OK] button.

*1 Check the Windows version as follows.
Press Windows key + R. Or go to the start of Windows, select [Windows System] and click [Run].
Enter "winver" in the [Run] window.
Check the version in the displayed window.

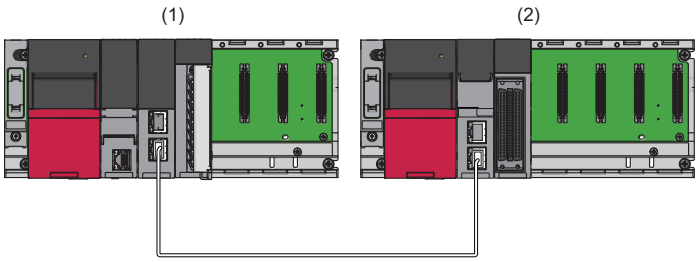
A

Appendix 6 Operation Examples of When the Remote Head Module Is Mounted

This section describes operation examples of when the remote head module is mounted

System configuration example

The following system configuration is used to explain an example of operation.



(1) Master station (Network number 1, station number 0)
- Power supply module: R61P
- CPU module: R04CPU
- Master/local module: RJ71GF11-T2 (Start I/O number: 0000H to 001FH)
- Input module: RX10 (Start I/O number: 0020H to 002FH)

(2) Intelligent device station (Network number 1, station number 1)
- Power supply module: R61P
- Remote head module: RJ72GF15-T2
- Flexible high-speed I/O control module: RD40PD01 (Start I/O number: 0000H to 001FH*1)

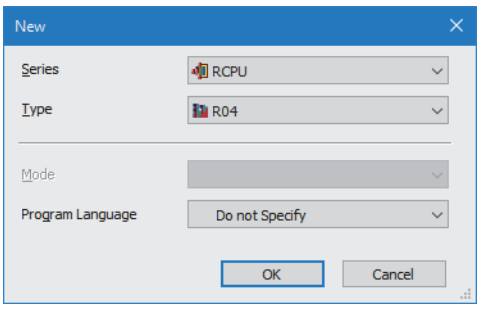
*1 In the RX/RY setting of the master station, set 1000H to 101FH as the start I/O number of the flexible high-speed I/O control module.

Setting in the master station

Connect the engineering tool to the CPU module of the master station and set parameters.

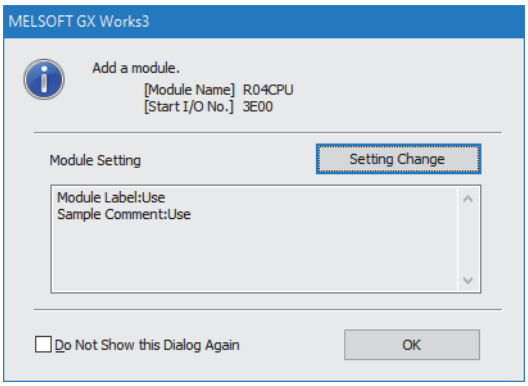
1. Create the project with the following settings.

[Project] ⇨ [New]



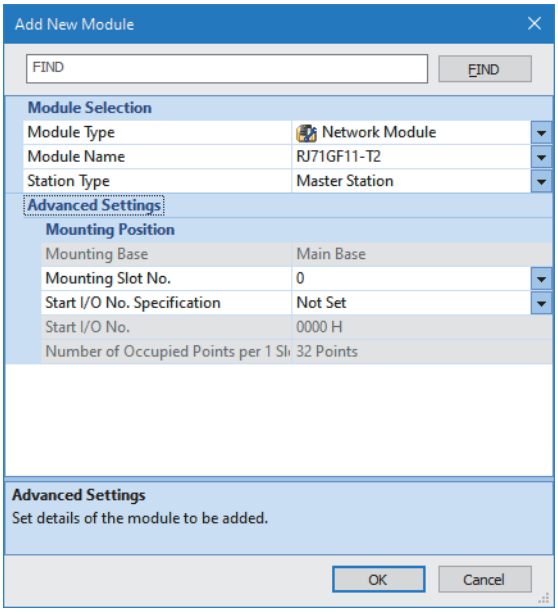
2. Click the [Setting Change] button and set the module to use the module labels.

3. Click the [OK] button in the following window to add the module labels of the CPU module.



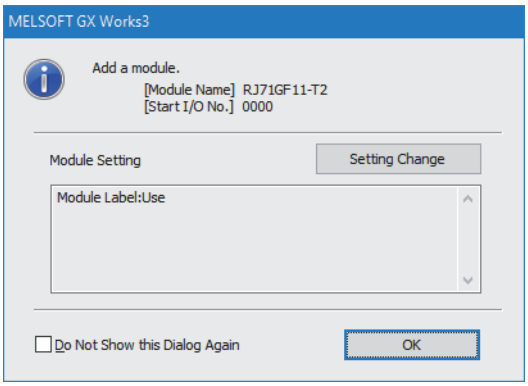
4. Add the master/local module with the following settings.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ Right-click ⇨ [Add New Module]



A

5. Configure the setting to use the module labels and add the module labels of the master/local module.



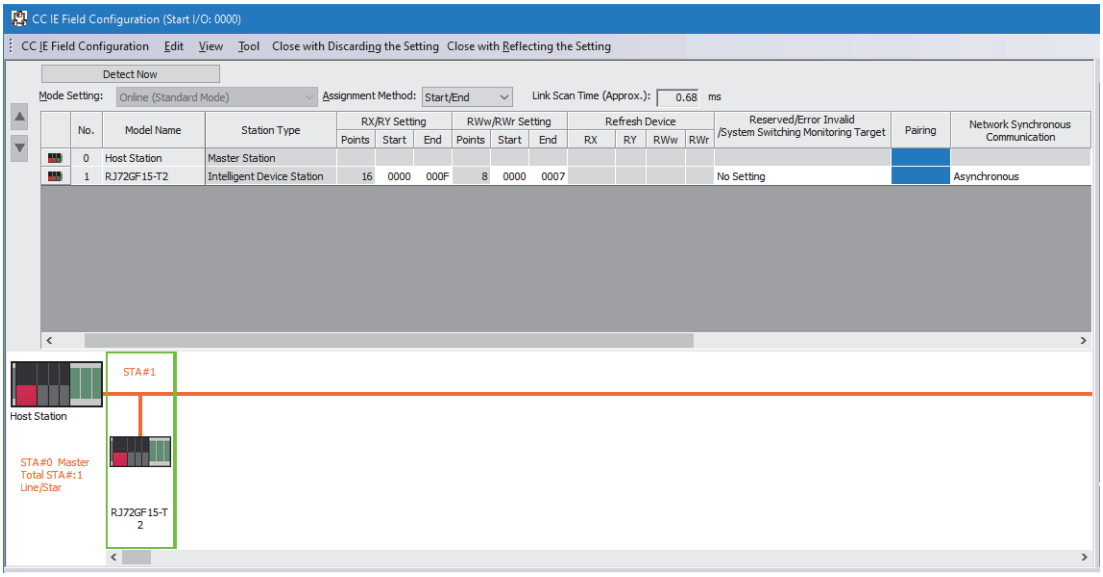
6. Set "Required Settings" of the module parameter of the master/local module as shown below.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71GF11-T2] ⇨ [Required Settings]

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

7. Set "Network Configuration Settings" of the module parameter of the master/local module as shown below.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71GF11-T2] ⇨ [Basic Settings] ⇨ [Network Configuration Settings]



8. Set "Refresh Settings" of the module parameter of the master/local module as shown below.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71GF11-T2] ⇨ [Basic Settings] ⇨ [Refresh Settings]

No.	Link Side					CPU Side				
	Device Name	Points	Start	End		Target	Device Name	Points	Start	End
-	SB	512	00000	001FF		Module Label				
-	SW	512	00000	001FF		Module Label				
1	RX	256	00000	000FF		Specify Device	X	256	01000	010FF
2	RY	256	00000	000FF		Specify Device	Y	256	01000	010FF
3	RWw	256	00000	000FF		Specify Device	W	256	00000	000FF
4	RWr	256	00000	000FF		Specify Device	W	256	01000	010FF
5										

9. Write the set parameters to the CPU module on the master station. Then reset the CPU module or power off and on the system.

[Online] ⇨ [Write to PLC]

Point

For parameters of the master/local module which are not described in this procedure, set default values. For details on parameters of the master/local module, refer to the following.

MELSEC iQ-R CC-Link IE Field Network User's Manual (Application)

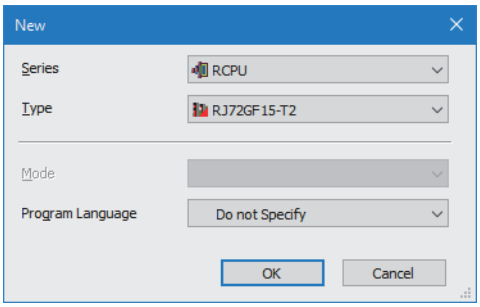
Setting in the intelligent device station

Connect the engineering tool to the remote head module of the intelligent device station and set parameters. Write the hardware logic to the flexible high-speed I/O control module with the configuration tool.

Parameter

1. Create the project with the following settings.

[Project] ⇨ [New]



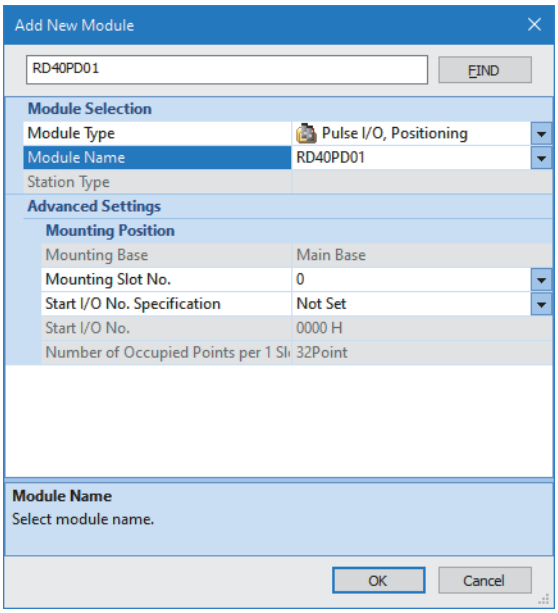
2. Set "Network Required Setting" of "CPU Parameter" of the remote head module as shown below.

[Navigation window] ⇨ [Parameter] ⇨ [RJ72GF15-T2] ⇨ [CPU Parameter] ⇨ [Network Required Setting]

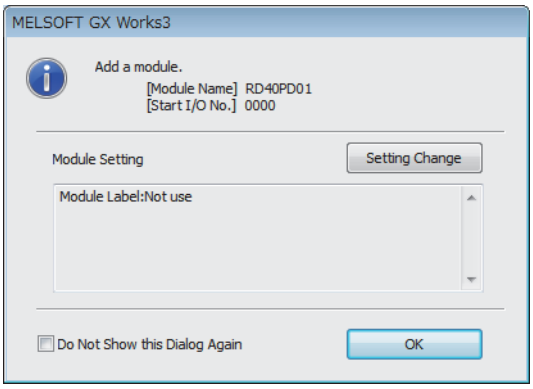
Item	Setting
Network Number	
Network Number	1
Station Number	
Station No.	1

3. Add the flexible high-speed I/O control module with the following settings.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ Right-click ⇨ [Add New Module]



4. Configure the setting not to use the module labels.



5. Set "Application setting" of the module parameter of the flexible high-speed I/O control module as shown below.

Item	Setting Value
Inter-module synchronization function	Hardware logic control function in accordance with the inter-module synchronization cycle.
Hardware logic control selection during synchronization error occurrence	Hardware logic control continue
Online module change	The module can be changed without the system being stopped.
Hardware logic control auto restoration executed/unexecuted	Enable
CC-Link IE Field Network head module installation setting	Set action when CC-Link IE Field Network head module is installed and using.
Hardware logic control at disconnection enable/disable	Disable
CPU error output mode setting	Clear
Continuous logging interrupt setting	Set wheter to send interruption requests when continuous logging data are stored.
Continuous logging data storage interrupt enable/disable	Disable

When the own station is disconnected, the operations of the flexible high-speed I/O control module differ depending on the settings of "Hardware logic control at disconnection enable/disable" and "CPU error output mode setting". For details, refer to the following.

☞ Page 310 Operations of the flexible high-speed I/O control module when the remote head module is mounted

6. Set "Refresh Timing" of the module parameter of the flexible high-speed I/O control module as shown below.

Item	Setting Value
Refresh at the set timing.	
Transfer to the CPU.	Transfer the buffer memory data to the specified device.
Latest error code	W10
Latest address of error history	
Refresh Timing	Set refresh timing.
Refresh Timing	-
Refresh Group [n](n: 1-64)	1
Refresh Timing (I/O)	Specify the timing which transfers the I/O device data.
Refresh Timing	Based on Refresh Timing (Buffer Memory)
Inter-module synchronous interrupt function	
Transfer to CPU	Transfer the buffer memory data to the specified device.
Synchronization latch count value 0_0	
Synchronization latch count value 0_1	
Synchronization latch count value 1_0	
Synchronization latch count value 1_1	
Synchronization latch count value 2_0	
Synchronization latch count value 2_1	
Synchronization latch count value 3_0	
Synchronization latch count value 3_1	
Synchronization latch count value 4_0	
Synchronization latch count value 4_1	
Synchronization latch count value 5_0	
Synchronization latch count value 5_1	
Synchronization latch count value 6_0	
Synchronization latch count value 6_1	
Synchronization latch count value 7_0	
Synchronization latch count value 7_1	
Synchronization status monitor	
Refresh Timing (Synchronous Interrupt)	Specify the timing to transfer the data of buffer memory for the Inter-module synchronous interrupt.
Refresh Timing	When Inter-module Synchronous Interrupt Program Executed

7. Write the set parameters to the remote head module on the intelligent device station. Then reset the remote head module or power off and on the system.

[Online] ⇨ [Write to PLC]

Point

For parameters of the remote head module which are not described in this procedure, set default values. For details on parameters of the remote head module, refer to the following.

- 📖 MELSEC iQ-R CC-Link IE Field Network Remote Head Module User's Manual (Application)

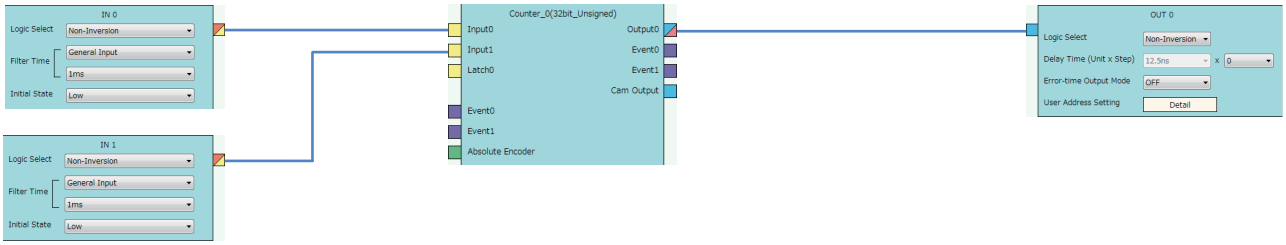
Creating a hardware logic

Create a hardware logic with the configuration tool.

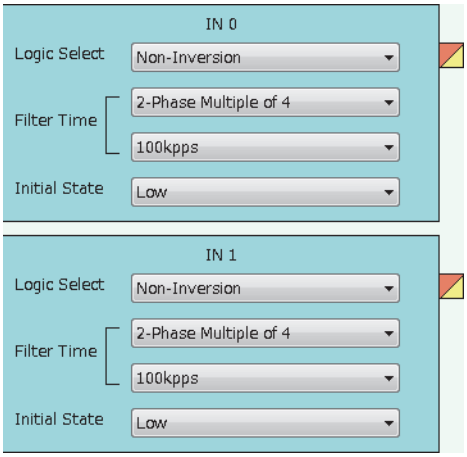
■Link and setting in the Hardware logic outline window

1. Arrange a 32-bit unsigned multi function counter block in the hardware logic outline window, and link it with an external input block.

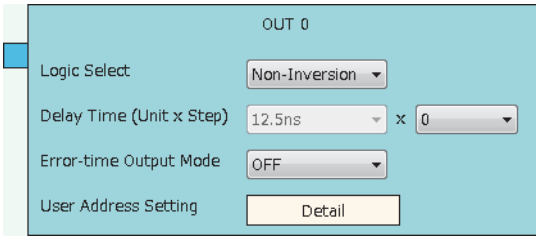
- To output signals externally according to the counting result, link the multi function counter block with the external output block.



2. Set the external input blocks according to the external input signal specifications.

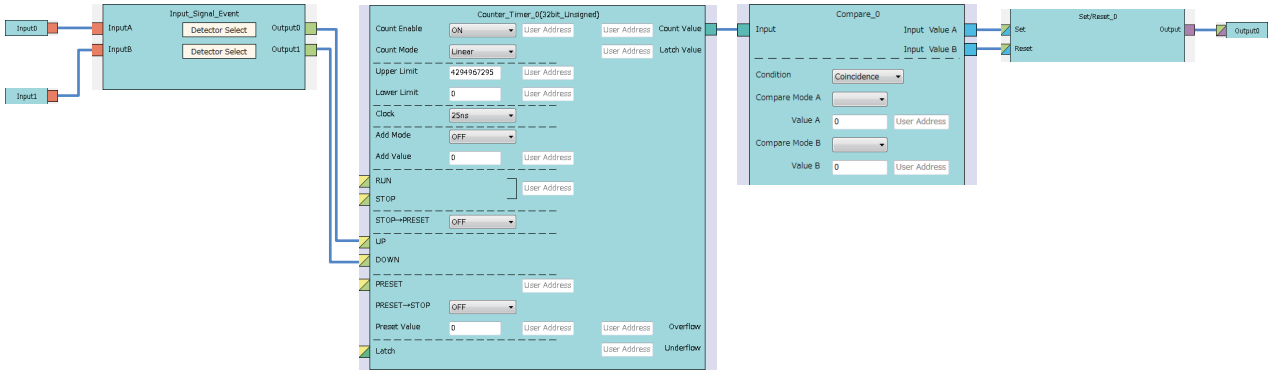


3. When using the external output block, set it according to the external output signal specifications and output timing.

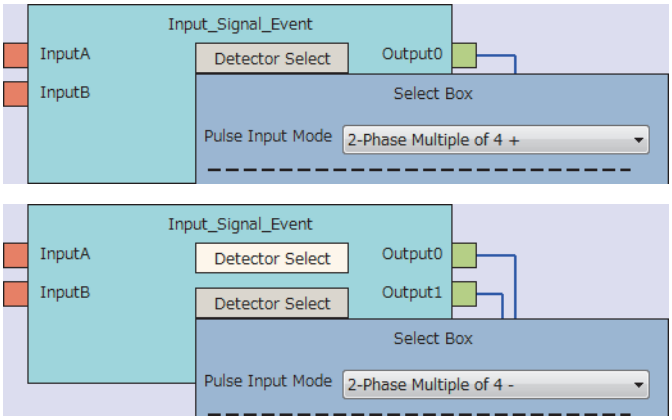


■Link and setting in the Multi function counter block detail window

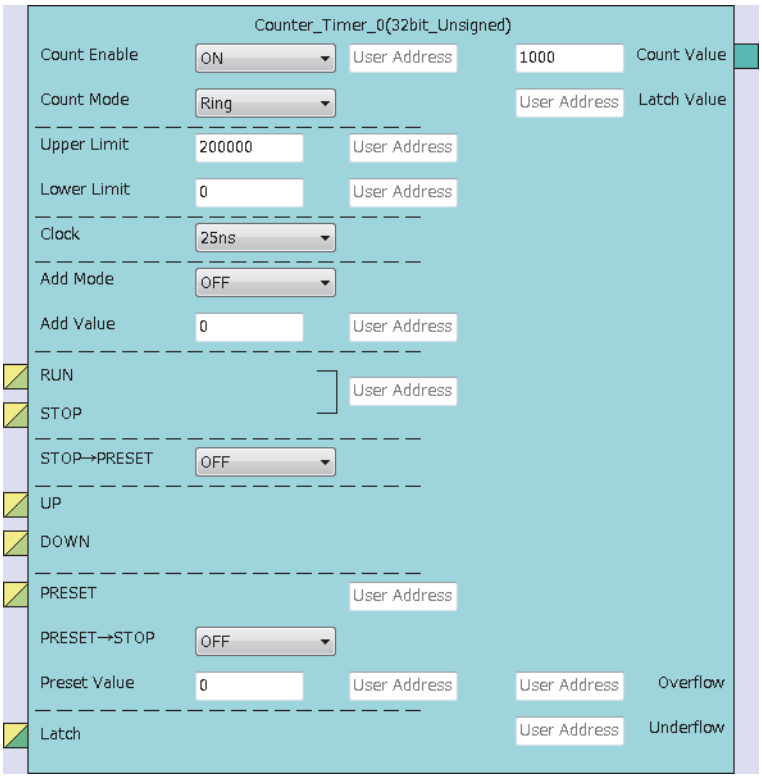
1. Link blocks in the Multi function counter block detail window as follows.



2. Set the setting values of the input signal event detection blocks as follows.

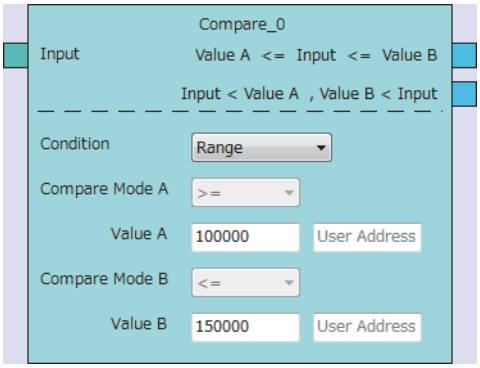


3. Set the setting values of the counter timer block as follows.



A

4. Set the setting values of the comparison block as follows.



■Writing data to the module

Write the hardware logic to the flexible high-speed I/O control module with the configuration tool.

Checking the network status

After setting parameters to the master station and the intelligent device station, check whether data link is normally performed between the master station and the intelligent device station. Check the network status using the CC-Link IE Field Network diagnostics of the engineering tool.

For how to perform the CC-Link IE Field Network diagnostics from the master station, refer to the following.

MELSEC iQ-R CC-Link IE Field Network User's Manual (Application)

Program examples

For the program examples, the module labels of the master/local module are used.

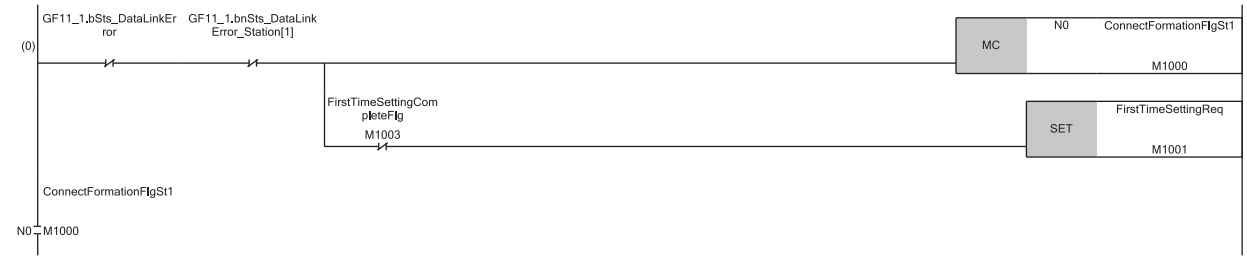
Write the programs to the CPU module on the master station.

Classification	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 number 1)	SW00B0.0
Label to be defined	Define global labels as shown below:		

Label Name	Data Type		Class	Assign (Device/Label)
LatestErrorCode	Word [Signed]	...	VAR_GLOBAL	D100
CumulativeNumberOfWriteAccessesToFlashROM	Double Word [Unsigned]/Bit String [32-bit]	...	VAR_GLOBAL	D200
ClearSettingOfErrorHistory	Word [Signed]	...	VAR_GLOBAL	D1000
CountValue	Word [Unsigned]/Bit String [16-bit]	...	VAR_GLOBAL	D1010
ParamWriteError	Bit	...	VAR_GLOBAL	F200
CountValueReadError	Bit	...	VAR_GLOBAL	F210
NumberOfWritesToFlashROMReadError	Bit	...	VAR_GLOBAL	F220
SettingStartReq	Bit	...	VAR_GLOBAL	M100
HardwareLogicControlStartReq	Bit	...	VAR_GLOBAL	M110
HardwareLogicControlStopReq	Bit	...	VAR_GLOBAL	M111
MonitorDataReadReq	Bit	...	VAR_GLOBAL	M112
CountValueReadReq	Bit	...	VAR_GLOBAL	M112
HardwareLogicResetReq	Bit	...	VAR_GLOBAL	M113
HardwareLogicControlStopFlagClearReq	Bit	...	VAR_GLOBAL	M114
ErrorResetReq	Bit	...	VAR_GLOBAL	M120
WriteCompleteFlg	Bit(0..1)	...	VAR_GLOBAL	M200
ReadCompleteFlg1	Bit(0..1)	...	VAR_GLOBAL	M210
ReadCompleteFlg2	Bit(0..1)	...	VAR_GLOBAL	M220
CountValueReadFlag	Bit	...	VAR_GLOBAL	M300
NumberOfWritesToFlashROMReadFlag	Bit	...	VAR_GLOBAL	M310
LatestErrorCodeReadFlag	Bit	...	VAR_GLOBAL	M320
ConnectFormationFlgSt1	Bit	...	VAR_GLOBAL	M1000
FirstTimeSettingReq	Bit	...	VAR_GLOBAL	M1001
SettingReqSig	Bit	...	VAR_GLOBAL	M1002
FirstTimeSettingCompleteFlg	Bit	...	VAR_GLOBAL	M1003
ModuleREADY	Bit	...	VAR_GLOBAL	X1000
OperationConditionSettingBatchResetCompleteFlag	Bit	...	VAR_GLOBAL	X1003
HardwareLogicControlFlag	Bit	...	VAR_GLOBAL	X1004
HardwareLogicControlStopFlagAtDisconnection	Bit	...	VAR_GLOBAL	X1007
ErrorFlag	Bit	...	VAR_GLOBAL	X100F
OperationConditionSettingBatchResetCommand	Bit	...	VAR_GLOBAL	Y1003
HardwareLogicControlStartRequest	Bit	...	VAR_GLOBAL	Y1004
HardwareLogicControlStopRequest	Bit	...	VAR_GLOBAL	Y1005
HardwareLogicControlStopSignalAtDisconnection	Bit	...	VAR_GLOBAL	Y1006
HardwareLogicControlStopFlagClearRequestAtDisconnection	Bit	...	VAR_GLOBAL	Y1007
ErrorClearRequest	Bit	...	VAR_GLOBAL	Y100F

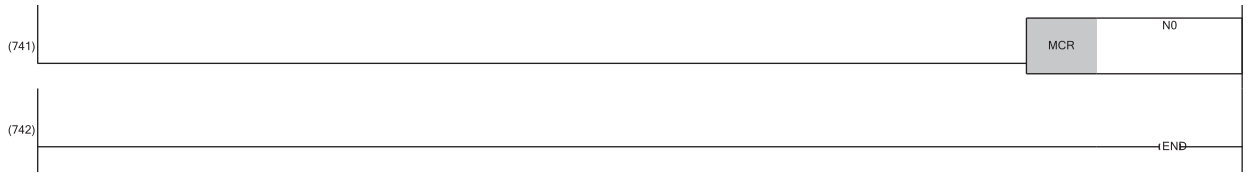
Common program

The following figure shows an example of the program to check the data link status of the remote head module (station number 1).



(0) Check the data link status of the remote head module (station number 1).

Add the MCR instruction shown below to the last of the program.



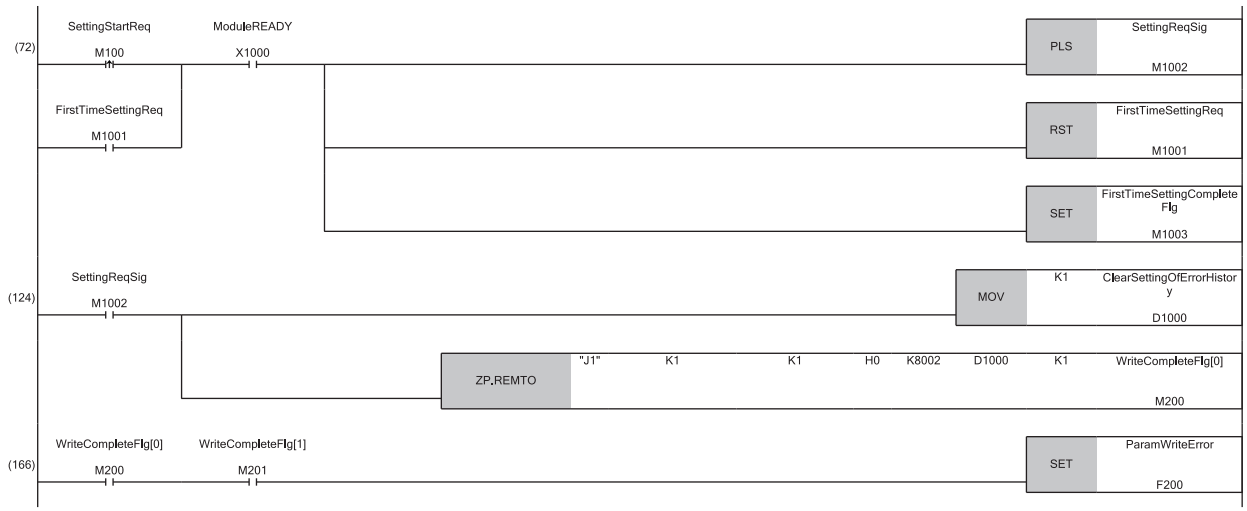
Program example 1

The following figure shows an example of the program to turn on 'Hardware logic control stop signal at disconnection' (Y1006).

(45)Turn on 'Hardware logic control stop signal at disconnection' (Y1006).

Program example 2

The following figure shows an example of the program to perform the clear setting of an error history.

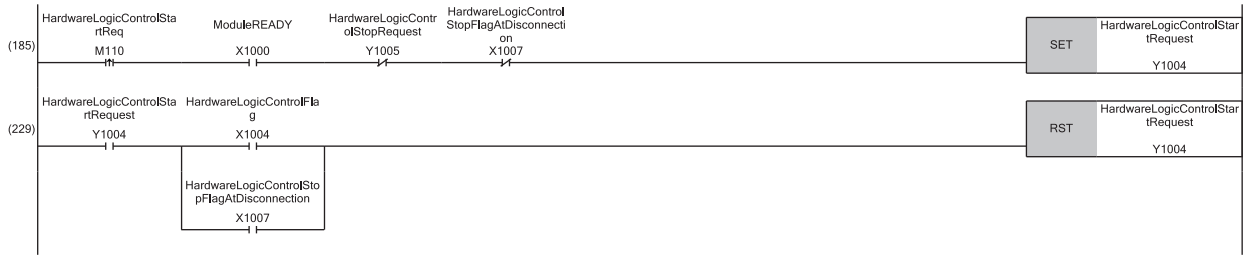


(72)Store Clear the history. (1) to 'Clear setting of error history' (Un\G8002).

A

Program example 3

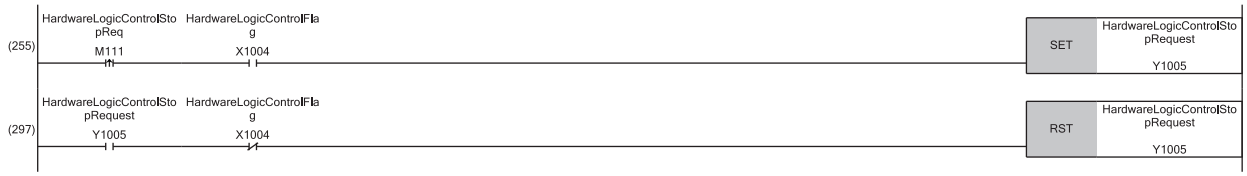
The following figure shows an example of the program to start the hardware logic control.



(185) Turn on 'Hardware logic control start request' (Y1004).
(229) Turn off 'Hardware logic control start request' (Y1004).

Program example 4

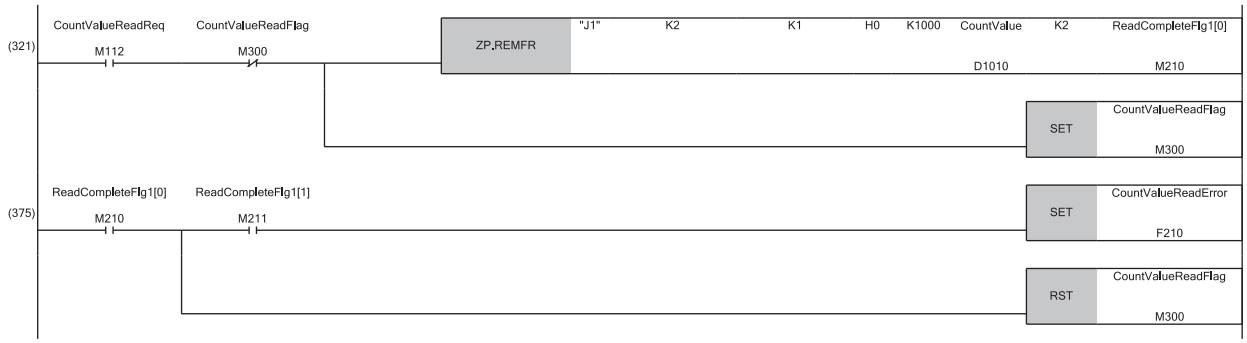
The following figure shows an example of the program to stop the hardware logic control.



(255) Turn on 'Hardware logic control stop request' (Y1005).
(297) Turn off 'Hardware logic control stop request' (Y1005).

Program example 5

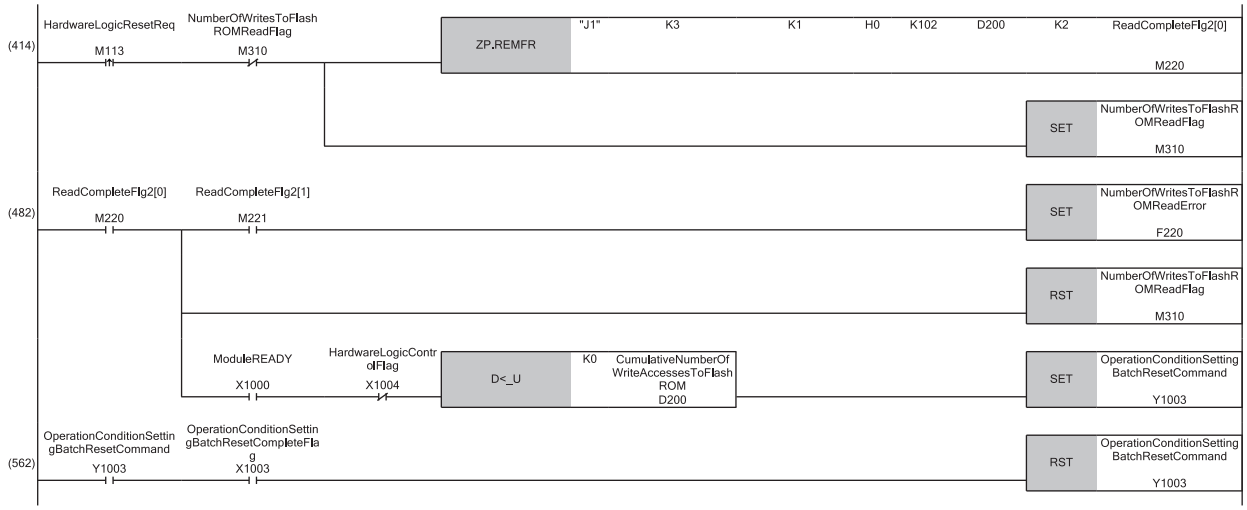
The following figure shows an example of the program to read the count value.



(321) Read the count value.

Program example 6

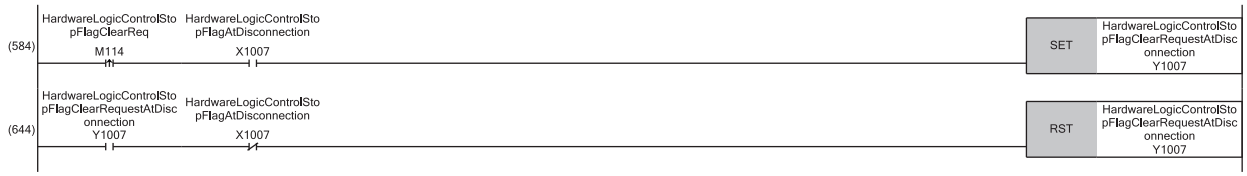
The following figure shows an example of the program to reset the operating condition settings in a batch.



(414) Turn on 'Operating condition settings batch-reset command' (Y1003).
(562) Turn off 'Operating condition settings batch-reset command' (Y1003).

Program example 7

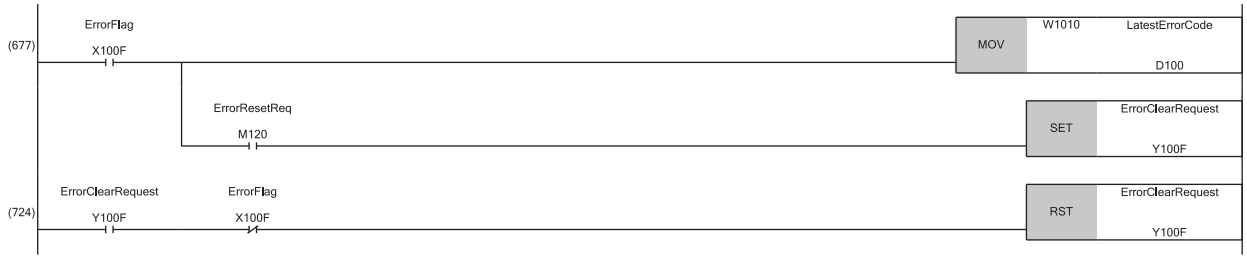
The following figure shows an example of the program to turn off 'Hardware logic control stop flag at disconnection' (X1007).



(584) Turn on 'Hardware logic control stop flag clear request at disconnection' (Y1007).
(644) Turn off 'Hardware logic control stop flag clear request at disconnection' (Y1007).

Program example 8

The following figure shows an example of the program to read the latest error code and reset the error.



(677) Read 'Latest error code' (Un\G100) and turn on 'Error clear request' (Y100F).
(724) Turn off 'Error clear request' (Y100F).

Operations of the flexible high-speed I/O control module when the remote head module is mounted

The following describes the operations of the flexible high-speed I/O control module when the own station is disconnected. The operations differ depending on the setting combination of "Hardware logic control at disconnection enable/disable" and "CPU error output mode setting" in "Application setting".

"Enable" is set for "Hardware logic control at disconnection enable/disable".

When 'Hardware logic control stop signal at disconnection' (Y6) is off, the hardware logic control is not executed. When 'Hardware logic control stop signal at disconnection' (Y6) is turned off during hardware logic control, the hardware logic control is stopped.

The following table lists the module operations depending on the setting of "CPU error output mode setting" when "Enable" is set for "Hardware logic control at disconnection enable/disable".

"CPU error output mode setting"	Operation
Clear	'Hardware logic control stop signal at disconnection' (Y6) is turned off at the remote head module disconnection, and thus the hardware logic control stops. External output at the hardware logic control stop varies depending on "Error-time Output Mode" of the external output block.
Hold	'Hardware logic control stop signal at disconnection' (Y6) is not turned off at the disconnection, and thus the hardware logic control will continue.

After the remote head module is returned, turn on 'Hardware logic control start request' (Y4) to restart the hardware logic control.

"Disable" is set for "Hardware logic control at disconnection enable/disable".

The following table lists the module operations depending on the setting of "CPU error output mode setting" when "Disable" is set for "Hardware logic control at disconnection enable/disable".

"CPU error output mode setting"	Operation
Clear	Even if the remote head module is disconnected, the hardware logic control will continue. All of the Y device terminals of the hardware logic turn to Low because Y signals turn off. If the hardware logic includes a Y device terminal, note that the module operations will change.
Hold	Even if the remote head module is disconnected, the hardware logic control will continue.

To continue the hardware logic control when the remote head module is disconnected, set "Disable" for "Hardware logic control at disconnection enable/disable".

If "Hardware logic control at disconnection enable/disable" is set to "Disable", a hardware logic control stop request cannot be issued from a program in the disconnection state.

Operations of the flexible high-speed I/O control module at disconnection

"Hardware logic control at disconnection enable/ disable"	"CPU error output mode setting"	Operations of the flexible high-speed I/O control module		
		Y signal status	Hardware logic operation	External output status
Enable	Hold	Held	Continue	Output according to the operation result of the hardware logic control
	Clear	Turn off	Stop	Output according to the setting of "Error-time Output Mode" of the external output block
Disable	Hold	Held	Continue	Output according to the operation result of the hardware logic control
	Clear	Turn off	Continue	

For details on Y signal status when the remote head module is mounted, refer to the following.

MELSEC iQ-R CC-Link IE Field Network Remote Head Module User's Manual (Application)

Appendix 7 Using the Module in the Redundant System with Redundant Extension Base Unit

This chapter describes restrictions and precautions for using the flexible high-speed I/O control module that is mounted on the extension base unit in the redundant system.

Restrictions on functions and specifications

Functions

Function	Restriction
Backing up and restoring the hardware logic	The hardware logic cannot be backed up and restored by using the module-specific backup parameter. Back up and restore the hardware logic by using the configuration tool. (☞ Page 82 When the module-specific backup parameter is not used)
A/D conversion value logging function	Cannot be used. When the function is used, proper operation cannot be guaranteed.
Interrupt function	The interrupt program cannot be executed.
Inter-module synchronization function	Cannot be used. When the function is used, proper operation cannot be guaranteed.
Logic analyzer function	
Continuous logging function	
Simulation function	

Module FBs

Any module FBs of the flexible high-speed I/O control module cannot be used. When the FB is executed, proper operation cannot be guaranteed.

Module parameter

■Application setting

Set "Hardware logic control auto restoration executed/unexecuted" to "Disable".

Precautions

Connection of the configuration tool

When using the configuration tool, connect it to the CPU module of the control system. The configuration tool cannot recognize the flexible high-speed I/O control module if it is connected to the CPU module of the standby system.

Program examples

Unless otherwise specified, program examples provided in this manual and the following manual are for when the module is used in the single CPU system or in the multiple CPU system.

MELSEC iQ-R Flexible High-Speed I/O Control Module User's Manual (Startup)

When using the module in the redundant system, refer to the following manual and observe the precautions on programming for when using the Process CPU (redundant mode).

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

Signal flow tracking setting

When applying the program examples to an actual system, set "Signal Flow Memory Tracking Setting" to "Transfer". If not, the programs may not work properly when system switching occurs.

[CPU Parameter] ⇨ [Redundant System Settings] ⇨ [Tracking Setting]

A

Appendix 8 Addition/change of a Function

This section describes an added or changed function of the flexible high-speed I/O control module and the engineering tool, and supported firmware version of the flexible high-speed I/O control module and the software version of the engineering tool.

Added/changed function	Supported firmware version of flexible high-speed I/O control module	Supported software version of engineering tool	Reference
Continuous logging function	"03" or later	"1.035M" or later	☞ Page 19 Continuous Logging Function
Watch function	"04" or later	"1.040S" or later	☞ Page 120 Watch Function
Addition of monitor items to the "Continuous logging" window	—	"1.045X" or later	☞ Page 137 Continuous logging
Change of block names and item names (from English to the language used)			—
A/D conversion value logging function	"05" or later	"1.055H" or later	☞ Page 32 A/D Conversion Value Logging Function

MEMO

INDEX

O

P

R

S

T

V

W

Y

I

REVISIONS

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

Revision date	*Manual number	Description
October 2016	SH(NA)-081647ENG-A	First edition
April 2017	SH(NA)-081647ENG-B	■Added function Continuous logging function ■Added or modified parts SAFETY PRECAUTIONS, INTRODUCTION, MANUAL PAGE ORGANIZATION, Section 1.3, 1.4, 1.5, 2.3, 2.5, 2.6, 2.8, 2.9, 3.2, 5.1 to 5.3, 6.3, 6.4, Appendix 2, 3, 5, 6
October 2017	SH(NA)-081647ENG-C	■Added function Watch function ■Added or modified parts RELEVANT MANUALS, Section 1.3, 2.1, 2.2, 2.3, 2.6, 2.8, 2.9, 2.10, 2.11, 4.1, Appendix 3, 6
July 2018	SH(NA)-081647ENG-D	■Added or modified parts Section 1.3, 1.4, 1.5, 2.2 to 2.11, 3.1, 3.2, 4.1 to 4.9, 6.4, Appendix 2, 3, 5, 6
April 2019	SH(NA)-081647ENG-E	■Added or modified parts SAFETY PRECAUTIONS, Section 1.3, 1.4, 1.5, 1.6, 2.10, 3.1, Chapter 4, Section 4.8, 4.10, 5.1, 5.3, 6.3, 6.4, Appendix 2, 3, 6
October 2020	SH(NA)-081647ENG-F	■Added or modified parts SAFETY PRECAUTIONS, CONDITIONS OF USE FOR THE PRODUCT, RELEVANT MANUALS, TERMS, GENERIC TERMS AND ABBREVIATIONS, Section 1.4, 3.1, 6.3, Appendix 3, 5, 6
August 2023	SH(NA)-081647ENG-G	■Added or modified parts SAFETY PRECAUTIONS, CONDITIONS OF USE FOR THE PRODUCT ■Structure change Appendix 5 has been added and the number of Appendix 5 to 8 have been changed.

Japanese manual number: SH-081646-G

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

Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
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.

SH(NA)-081647ENG-G(2308)MEE
MODEL: RD40PD01-U-OU-E
MODEL CODE: 13JX56

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE: TOKYO BLDG., 2-7-3, MARUNOUCHI, CHIYODA-KU, TOKYO 100-8310, JAPAN
NAGOYA WORKS: 1-14, YADA-MINAMI 5-CHOME, HIGASHI-KU, NAGOYA 461-8670, JAPAN