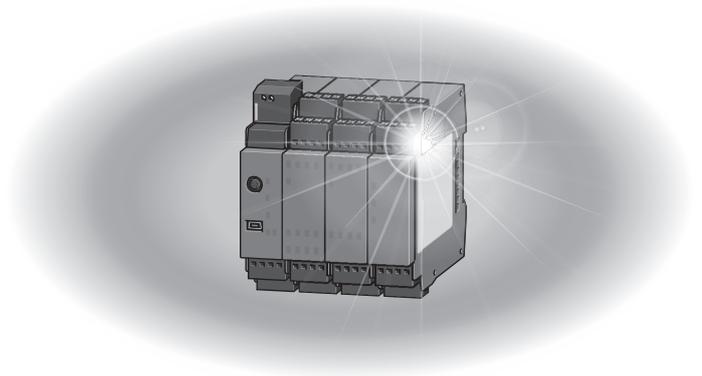


MELSEC **WS** series

# Safety Controller Ethernet Interface Module User's Manual

---

-WS0-GETH



Powered by  
**SICK**

MELSEC-WS series products were jointly developed and manufactured by Mitsubishi Electric and SICK AG, Industrial Safety Systems in Germany.  
\* Note that the warranty on MELSEC-WS series products differs from that on MELSEC-Q or MELSEC-QS series products.  
(Refer to "WARRANTY" written in this manual.)



This document is protected by the law of copyright, whereby all rights established therein remain with the company Mitsubishi Electric Corporation. Reproduction of this document or parts of this document is only permissible within the limits of the legal determination of Copyright Law. Alteration or abridgement of the document is not permitted without the explicit written approval of the company Mitsubishi Electric Corporation.

### **Precautions regarding warranty and specifications**

MELSEC-WS series products are jointly developed and manufactured by Mitsubishi and SICK AG, Industrial Safety Systems, in Germany.

Note that there are some precautions regarding warranty and specifications of MELSEC-WS series products.

#### <Warranty>

- The gratis warranty term of the product shall be for one (1) year after the date of delivery or for eighteen (18) months after manufacturing, whichever is less.
- The onerous repair term after discontinuation of production shall be for four (4) years.
- Mitsubishi shall mainly replace the product that needs a repair.
- It may take some time to respond to the problem or repair the product depending on the condition and timing.

#### <Specifications>

- General specifications of the products differ.

	<b>MELSEC-WS</b>	<b>MELSEC-Q</b>	<b>MELSEC-QS</b>
Operating ambient temperature	-25 to 55°C*1	0 to 55°C	0 to 55°C
Operating ambient humidity	10 to 95%RH	5 to 95%RH	5 to 95%RH
Storage ambient temperature	-25 to 70°C	-25 to 75°C	-40 to 75°C
Storage ambient humidity	10 to 95%RH	5 to 95%RH	5 to 95%RH

\*1 When the WS0-GCC100202 is included in the system, operating ambient temperature will be 0 to 55 °C.

- EMC standards that are applicable to the products differ.

	<b>MELSEC-WS</b>	<b>MELSEC-Q, MELSEC-QS</b>
EMC standards	EN61000-6-2, EN55011	EN61131-2

## ● SAFETY PRECAUTIONS ●

(Read these precautions before using this product.)

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

The precautions given in this manual are concerned with installation of this product and wiring with the external application only.

For the safety precautions of the MELSEC-WS safety controller system, refer to the user's manual for the CPU module used.

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

 <b>WARNING</b>	Indicates that incorrect handling may cause hazardous conditions, resulting in death or severe injury.
 <b>CAUTION</b>	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**

- **The Ethernet interface module is not suitable for operation on a safety fieldbus.**  
The Ethernet interface module only generates non-safety-related fieldbus data (status bytes) for control and diagnostics purposes.
- **Do not use non-safe data from an Ethernet interface module for safety related applications.**  
With the Ethernet interface module it is possible to integrate non-safe data into the logic editor such that the safety function of the MELSEC-WS safety controller is compromised. Never implement the Ethernet interface module into a MELSEC-WS safety controller without having this danger checked by a safety specialist.

 CAUTION

- Do not install the control lines or communication cables together with the main circuit lines or power cables. Keep a distance of 100 mm or more between them.  
Failure to do so may result in malfunction due to noise.
- **The MELSEC-WS safety controller fulfils the requirements of Class A (industrial applications) in accordance with the "Interference emission" basic specifications.**  
The MELSEC-WS safety controller is therefore only suitable for use in an industrial environment and not for private use.

[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]

### CAUTION

- Use the MELSEC-WS safety controller in an environment that meets the general specifications in the user's manual for the CPU module used. Failure to do so may result in electric shock, fire, malfunction, or damage to or deterioration of the product.
- Latch the module onto the DIN rail. Incorrect mounting may cause malfunction, failure or drop of the module.
- To ensure full electromagnetic compatibility (EMC), the DIN mounting rail has to be connected to functional earth (FE).  
Ensure that the earthing contact is positioned correctly. The earthing spring contact of the module must contact the DIN rail securely to allow electrical conductivity.  
Additionally connect all network cable shields directly at the control cabinet entrance to a common FE ground line.
- Tighten the screw within the specified torque range.  
Undertightening can cause drop of the screw, short circuit or malfunction.  
Overtightening can damage the screw and/or module, resulting in drop, short circuit, or malfunction.
- 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 damage to the product.
- Do not directly touch any conductive part of the module.  
Doing so can cause malfunction or failure of the module.
- The MELSEC-WS safety controller is only suitable for mounting in a control cabinet with at least IP 54 degree of protection.  
Failure to meet the installation method may cause the module to fail or malfunction due to the deposition of dust or the adhesion of water.
- **Make sure that the connection of the Ethernet interface module cannot lead to hazardous situations during installation.**  
Ensure that the connection of the Ethernet dedicated cables and power cables cannot lead to a hazardous situation when implementing the unit on to the MELSEC-WS safety controller and Ethernet network. Prevent unintended start-up of equipment during connection of a Ethernet interface module.  
Failure to do so may cause the system to malfunction.

## [Wiring Precautions]

### CAUTION

- Switch the entire machine/system off line.  
The system could start up unexpectedly while you are connecting the devices.
- Connectors for external devices must be crimped or pressed with the tool specified by the manufacturer, or must be correctly soldered. Incomplete connections could result in short circuit, fire, or malfunction.
- Connect the connectors to the module securely.  
Failure to do so may cause malfunction due to poor contact.
- Place the cables in a duct or clamp them.  
If not, dangling cable may swing or inadvertently be pulled, resulting in damage to the module or cables or malfunction due to poor contact.
- Tighten the terminal screw within the specified torque range.  
Undertightening can cause short circuit or malfunction.  
Overtightening can damage the screw and/or module, resulting in drop, short circuit, or malfunction.
- When disconnecting the cable from the module, do not pull the cable by the cable part.  
When disconnecting the cable with connector from the module, hold the connector part.  
When disconnecting the cable connected to a terminal block, loosen the terminal screws first.  
Failure to do so may result in damage to the module or cable or malfunction.
- Prevent foreign matter such as dust or wire chips from entering the module.  
Such foreign matter can cause a fire, failure, or malfunction.

## ● CONDITIONS OF USE FOR THE PRODUCT ●

- (1) Although Mitsubishi Electric has obtained the certification for Product's compliance to the international safety standards IEC61508, ISO13849-1 from TUV Rheinland, this fact does not guarantee that Product will be free from any malfunction or failure. The user of this Product shall comply with any and all applicable safety standard, regulation or law and take appropriate safety measures for the system in which the Product is installed or used and shall take the second or third safety measures other than the Product. Mitsubishi Electric is not liable for damages that could have been prevented by compliance with any applicable safety standard, regulation or law.
- (2) Mitsubishi Electric prohibits the use of Products with or in any application involving, and Mitsubishi Electric shall not be liable for a default, a liability for defect warranty, a quality assurance, negligence or other tort and a product liability in these applications.
  - (a) power plants,
  - (b) trains, railway systems, airplanes, airline operations, other transportation systems,
  - (c) hospitals, medical care, dialysis and life support facilities or equipment,
  - (d) amusement equipments,
  - (e) incineration and fuel devices,
  - (f) handling of nuclear or hazardous materials or chemicals,
  - (g) mining and drilling,
  - (h) and other applications where the level of risk to human life, health or property are elevated.
- (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.

# REVISIONS

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

<b>Print date</b>	<b>*Manual number</b>	<b>Revision</b>
September, 2009	SH(NA)-080857ENG-A	First edition
March, 2010	SH(NA)-080857ENG-B	Correction of errors in writing
August, 2011	SH(NA)-080857ENG-C	Correction of errors in writing
August, 2012	SH(NA)-080857ENG-D	Setting and Monitoring Tool was upgraded.
June, 2013	SH(NA)-080857ENG-E	Correction of errors in writing
August, 2016	SH(NA)-080857ENG-F	Description on the corporate logo was changed
December, 2019	SH(NA)-080857ENG-G	Adding the WS0-CPU3
September, 2024	SH(NA)-080857ENG-H	Correction of errors in writing

Japanese manual version SH-080854-H

This manual confers no industrial property rights or any rights of any other kind, nor does it confer any patent licenses. Mitsubishi Electric Corporation cannot be held responsible for any problems involving industrial property rights which may occur as a result of using the contents noted in this manual.

© 2009 MITSUBISHI ELECTRIC CORPORATION

# CONTENTS

SAFETY PRECAUTIONS .....	1
CONDITIONS OF USE FOR THE PRODUCT .....	5
REVISIONS .....	6
CONTENTS .....	7
GENERIC TERMS AND ABBREVIATIONS .....	9
1. About this document .....	10
1.1 Function of this document .....	10
1.2 MELSEC-WS manuals .....	10
1.3 Target group .....	11
1.4 Depth of information .....	12
1.5 Scope .....	12
1.6 Abbreviations used .....	12
1.7 Symbols used .....	12
2. On safety .....	13
2.1 Qualified safety personnel .....	13
2.2 Correct use .....	13
2.3 Environmental protection .....	15
2.3.1 Disposal .....	15
2.3.2 Material separation .....	15
3. Ethernet interface module .....	16
3.1 Overview .....	16
3.1.1 Interfaces and operation .....	16
3.2 Mounting/Dismantling .....	18
3.2.1 Steps for mounting the modules .....	18
3.2.2 Steps for dismantling the modules .....	20
3.3 Electrical installation .....	21
3.4 First configuration steps .....	21
3.4.1 Establishing a connection between MELSEC-WS safety controller and PC .....	21
3.4.2 Assignment of an IP address .....	23
3.4.3 Configuration of the Ethernet interface module .....	25
3.4.4 Transfer of a configuration .....	26
3.4.5 Verification of a configuration .....	27
3.4.6 Upload of a configuration .....	27
3.5 Features of the Ethernet interface module .....	28
3.5.1 TCP/IP configuration interface .....	28
3.5.2 Ethernet TCP/IP socket interface .....	32
3.5.3 TCP/IP process image example .....	40
4. Data sets .....	43
4.1 Data transmitted into the network (network input data sets) .....	44
4.1.1 Logic results .....	46
4.1.2 Module and EFI input and output values .....	46
4.1.3 Routing of data from a second network .....	46
4.1.4 Configuration checksums (CRCs) .....	47
4.1.5 Error and status information of the modules .....	47
4.2 Data received from the network (network output data sets) .....	50

4.3	Default settings for the input data set 1 .....	51
4.4	Customizing the input data set 1 (MELSEC-WS to Ethernet) .....	52
4.4.1	The toolbar .....	53
4.4.2	Available data area .....	54
4.4.3	Network Module Data area .....	55
4.4.4	Tag names area .....	55
4.5	Tag names for incoming data (Ethernet to MELSEC-WS) .....	56
4.5.1	Saving and loading a configuration .....	58
4.5.2	Importing and exporting a configuration .....	58
4.6	Monitoring the data set online .....	59
5.	Troubleshooting.....	60
6.	Technical specifications.....	62
6.1	Technical specifications of Ethernet interface module.....	62
6.2	Technical specifications, supply circuit .....	62
6.3	General technical specifications .....	62
6.4	Dimensional drawings.....	64
6.5	Module information .....	64
7.	Connecting to MELSEC programmable controllers .....	65
7.1	Features achieved by the connection.....	65
7.2	Applicable system.....	66
8.	Procedures before operation.....	67
8.1	Wiring.....	72
8.1.1	Wiring precautions .....	74
8.2	Communications with the QJ71E71-100 (QCPU) .....	75
8.2.1	System configuration .....	75
8.2.2	Polling mode (input data set reading).....	76
8.2.3	Polling mode (output data set writing) .....	91
8.2.4	Auto update mode .....	99
8.3	Communications with the Built-in Ethernet port QCPU .....	108
8.3.1	System configuration .....	108
8.3.2	Polling mode (input data set reading).....	108
8.3.3	Polling mode (output data set writing) .....	113
8.3.4	Auto update mode .....	115
8.4	Communications with the QJ71E71-100 (QSCPU).....	121
8.4.1	System configuration .....	121
8.4.2	Polling mode (input data set reading).....	121
8.4.3	Polling mode (output data set writing) .....	125
8.4.4	Auto update mode .....	128
9.	Troubleshooting.....	131
9.1	When data cannot be normally exchanged .....	131
9.2	When data cannot be exchanged in polling mode (reading/writing) .....	132
9.3	When data cannot be exchanged in auto update mode.....	134
10.	Annex .....	135
10.1	SICK contact.....	135

# GENERIC TERMS AND ABBREVIATIONS

Generic term/abbreviation	Description
WS0-CPU0	The abbreviation for the WS0-CPU000200 MELSEC-WS safety controller CPU module
WS0-CPU1	The abbreviation for the WS0-CPU130202 MELSEC-WS safety controller CPU module
WS0-CPU3	The abbreviation for the WS0-CPU320202 MELSEC-WS safety controller CPU module
WS0-XTIO	The abbreviation for the WS0-XTIO84202 MELSEC-WS safety controller safety I/O combined module
WS0-XTDI	The abbreviation for the WS0-XTDI80202 MELSEC-WS safety controller safety input module
WS0-4RO	The abbreviation for the WS0-4RO4002 MELSEC-WS safety controller safety relay output module
WS0-GETH	The abbreviation for the WS0-GETH00200 MELSEC-WS safety controller Ethernet interface module
CPU module	A generic term for the WS0-CPU0, WS0-CPU1, and WS0-CPU3
Safety I/O module	A generic term for the WS0-XTIO and WS0-XTDI
Ethernet interface module	Another name for the WS0-GETH
Network module	A generic term for the WS0-GETH and WS0-GCC1
Q series programmable controller	The abbreviation for the Mitsubishi Electric MELSEC-Q series programmable controller
QJ71E71-100	The abbreviation for the MELSEC QJ71E71-100 Ethernet module
Built-in Ethernet port QCPU	A generic term for the Q03UDEHCPU, Q04UDEHCPU, Q06UDEHCPU, Q10UDEHCPU, Q13UDEHCPU, Q20UDEHCPU, Q26UDEHCPU, Q50UDEHCPU, and Q100UDEHCPU
GX Developer	The product name of the software package for the MELSEC programmable controllers
GX Works2	The product name of the software package for the MELSEC programmable controllers
Programming tool	A generic term for GX Works2 and GX Developer

# 1. About this document

Please read the SAFETY PRECAUTIONS, Chapter 1, and Chapter 2 carefully before working with this documentation and the MELSEC-WS safety controller Ethernet interface module.

## 1.1 Function of this document

This manual only applies in conjunction with the other MELSEC-WS manuals (see Section 1.2) and instructs the technical staff of the machine manufacturer and/or of the machine operator on safe mounting, adjustment, electrical installation, commissioning as well as operation and maintenance of the Ethernet interface module.

The manual does not provide instructions for operating the machine in which the MELSEC-WS safety controller and Ethernet interface module are, or will be, integrated. Information of this kind will be found in the manuals for the machine.

## 1.2 MELSEC-WS manuals

For the MELSEC-WS safety controller there are three manuals with clearly distinguished fields of application as well as user's manuals (hardware) for each module.

- This manual describes the Ethernet interface module and its functions in detail.
- The user's manuals (hardware) are enclosed with each MELSEC-WS module. They inform on the basic technical specifications of the modules and contain simple mounting instructions. Use the user's manuals (hardware) when mounting MELSEC-WS safety controllers.
- The Safety Controller User's Manual describes all the MELSEC-WS modules and their functions in detail. Use the manual in particular to configure MELSEC-WS safety controller.
- The Safety Controller Setting and Monitoring Tool Operating Manual describes the software-supported configuration and parameterization of the MELSEC-WS safety controller. In addition, the manual contains the description of the diagnostics functions that are important for operation and detailed information for the identification and elimination of errors. Use the manual in particular for the configuration, commissioning and operation of MELSEC-WS safety controllers.

The following shows the relevant manuals.

Table 1:  
Overview of the  
MELSEC-WS manuals

Title	Number
Safety Controller User's Manual	WS-CPU-U-E
Safety Controller Ethernet Interface Module User's Manual	WS-ET-U-E
Safety Controller CC-Link Interface Module User's Manual	WS-CC-U-E
Safety Controller Setting and Monitoring Tool Operating Manual	SW1DNN-WS0ADR-B-O-E
Safety Controller CPU Module User's Manual (Hardware)	WS-CPU-U-HW
Safety Controller Safety I/O Module User's Manual (Hardware)	WS-IO-U-HW
Safety Controller Safety Relay Output Module User's Manual (Hardware)	WS-SR-U-HW
Safety Controller Ethernet Interface Module User's Manual (Hardware)	WS-ET-U-HW
Safety Controller CC-Link Interface Module User's Manual (Hardware)	WS-CC-U-HW

### 1.3 Target group

This manual is addressed to planning engineers, machine designers and the operators of systems in which a MELSEC-WS safety controller is integrated and who want to exchange data with a fieldbus (a controller) via a network module.

It also addresses people who integrate the network module into a machine, commission it initially or who are in charge of servicing and maintaining the unit.

## 1.4 Depth of information

This manual contains information on the Ethernet interface module in the following subjects:

- mounting
- implementation into a network
- configuration via Setting and Monitoring Tool
- data transfer to and from the network
- status information, planning and related mapping
- part numbers




---

### Warning!

Pay attention to the safety notes and safety measures on the Ethernet interface module!

---

**Note** For the acquisition of Setting and Monitoring Tool, please contact your local Mitsubishi representative.

## 1.5 Scope

This manual applies to the WS0-GETH Ethernet interface module.

This document is the original manual.

## 1.6 Abbreviations used

**EFI** Enhanced Function Interface

## 1.7 Symbols used

**Note** Notes provide special information on the device.

●Red, ✨Red, ○Green LED symbols describe the state of a diagnostics LED. Examples:

●Red The red LED is illuminated constantly.

✨Red The red LED is flashing.

○Green The green LED is off.

⇒ **Action** Instructions for taking action are shown by an arrow. Read carefully and follow the instructions for action.




---

### ATTENTION!

An "ATTENTION" indicates an actual or potential risk or health hazard. They are designed to help you to prevent accidents.

Read carefully and follow the attention notices!

---

## 2. On safety

This chapter deals with your own safety and the safety of the equipment operators.

- Please read this chapter carefully before working with an Ethernet interface module.

### 2.1 Qualified safety personnel

The Ethernet interface module may only be installed, commissioned and serviced by qualified safety personnel.

Qualified safety personnel are defined as persons who...

- have undergone the appropriate technical training  
**and**
- have been instructed by the responsible machine operator in the operation of the machine and the current valid safety guidelines  
**and**
- have access to the manuals of the Ethernet interface module and safety controller and have read and familiarized themselves with them.

### 2.2 Correct use

The Ethernet interface module can only be operated with a MELSEC-WS safety controller.

The Ethernet interface module does not have a dedicated voltage supply.



---

**The Ethernet interface module is not suitable for operation on a safety fieldbus!**

The Ethernet interface module only generates non-safety-related fieldbus data (status bytes) for control and diagnostics purposes.

**Do not use non-safe data from an Ethernet interface module for safety related applications!**

With the Ethernet interface module it is possible to integrate non-safe data into the logic editor such that the safety function of the MELSEC-WS safety controller is compromised. Never implement the Ethernet interface module into a MELSEC-WS safety controller without having this danger checked by a safety specialist.

---

The Ethernet interface module may only be used by qualified safety personnel and only on the machine where they have been installed and initialized by qualified safety personnel in accordance with the manual.



---

**Pay attention to the safety notes and safety measures on the Ethernet interface module!**

If the device is used for any other purposes or modified in any way - also during mounting and installation - any warranty claim against Mitsubishi Electric Corporation shall become void.

---

- Note**
- When mounting, installing and using the Ethernet interface module, observe the standards and directives applicable in your country.
  - The national/international rules and regulations apply to the installation, commissioning, use and periodic technical inspection of the MELSEC-WS safety controller, in particular:
    - EMC directive 2004/108/EC,
    - Provision and Use of Work Equipment Directive 89/655/EC,
    - the work safety regulations/safety rules.
  - This manual must be made available to the operator of the machine where the MELSEC-WS safety controller is used. The machine operator is to be instructed in the use of the device by qualified safety personnel and must be instructed to read the manual.
  - Since an Ethernet interface module is powered from the CPU module, connect a Class 2 power supply or a Class 2 transformer in accordance with UL310 or UL1585 to the CPU module.



---

**The MELSEC-WS safety controller fulfils the requirements of Class A (industrial applications) in accordance with the “Interference emission” basic specifications.**

The MELSEC-WS safety controller is therefore only suitable for use in an industrial environment and not for private use.

---

## 2.3 Environmental protection

The Ethernet interface module has been designed to minimize environmental impact. It uses only a minimum of power and natural resources.

- At work, always act in an environmentally responsible manner.

### 2.3.1 Disposal

Disposal of unusable or irreparable devices should always occur in accordance with the applicable country-specific waste-disposal regulations (e.g. European Waste Code 16 02 14).

### 2.3.2 Material separation



**Material separation may only be performed by qualified safety personnel!**

Exercise care when disassembling the devices. The danger of injury is present.

Before you can turn over the devices for environmental-friendly recycling, you must separate the different materials of the Ethernet interface module from one another.

- Separate the housing from the remaining components (especially the PCB).
- Send the separated components to the corresponding recycling centers (see the following table).

Component	Disposal
Product Housing PCBs, cables, plugs and electrical connection pieces	Electronics recycling
Packaging Cardboard, paper	Paper/cardboard recycling

Table 2:  
Overview of disposal by  
components

## 3. Ethernet interface module

This chapter describes the Ethernet interface module (WS0-GETH).

### 3.1 Overview

The Ethernet interface module is connected to Setting and Monitoring Tool, a MELSEC programmable controller, and PC via TCP/IP. The module can:

- Exchange configuration data and programs of a MELSEC-WS safety controller by connecting to Setting and Monitoring Tool via TCP/IP.
- Exchange non-safety related data by connecting to an external application (e.g. MELSEC programmable controller, PC) via TCP/IP.

#### 3.1.1 Interfaces and operation

The Ethernet interface module is equipped with an integrated three-port switch for connection with the Ethernet network. Two RJ45 sockets are available for the connection. The switch functionality allows the Ethernet interface module to be used for connection to another Ethernet component (e.g. connection to a notebook) without having to interrupt the Ethernet connection to the network. (One IP address can be configured.)

Figure 1:  
Interfaces and display  
elements of the WS0-  
GETH

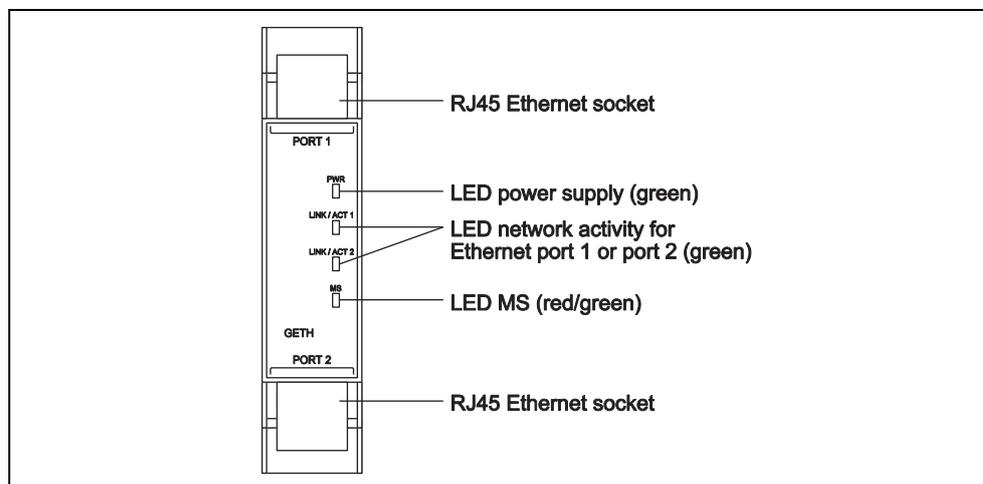


Table 3:  
Meaning of the LED  
displays

LED		Meaning
PWR	○	No power supply
	● Green	Power supply switched on
LINK/ACT 1	○	No Ethernet connection
LINK/ACT 2	● Green	Ethernet connection active, no data transmission
	☀ Green	Ethernet connection active, data transmission
MS	○	No power supply or immediately after the module start
	● Green	Executing (live process data from/to CPU)
	☀ Green	Idle
	☀ Red	1 Hz: Configuring/configuration required 2 Hz: Critical fault on Ethernet interface module
	● Red	Critical fault on another module
	☀ Red/Green	Executing, but Ethernet communication not established or faulty

**Note** Error elimination is described in Chapter 5.

#### Power-up sequence

On power up, the following LED test sequence is performed:

- LED MS ○ **Off** for 6 s.
- LED MS ● **Red** for 0.25 s.
- LED MS ● **Green** for 0.25 s.
- LED MS ○ **Off**

## 3.2 Mounting/Dismantling

This chapter describes the mounting of the Ethernet interface module.



---

**Make sure that the connection of the Ethernet interface module cannot lead to hazardous situations during installation!**

Ensure that connecting an Ethernet interface module cannot lead to a hazardous situation when implementing the unit on to the MELSEC-WS safety controller and Ethernet network. Prevent unintended start-up of equipment during connection of an Ethernet interface module.

---

### 3.2.1 Steps for mounting the modules



---

**The MELSEC-WS safety controller is only suitable for mounting in a control cabinet with at least IP 54 degree of protection.**

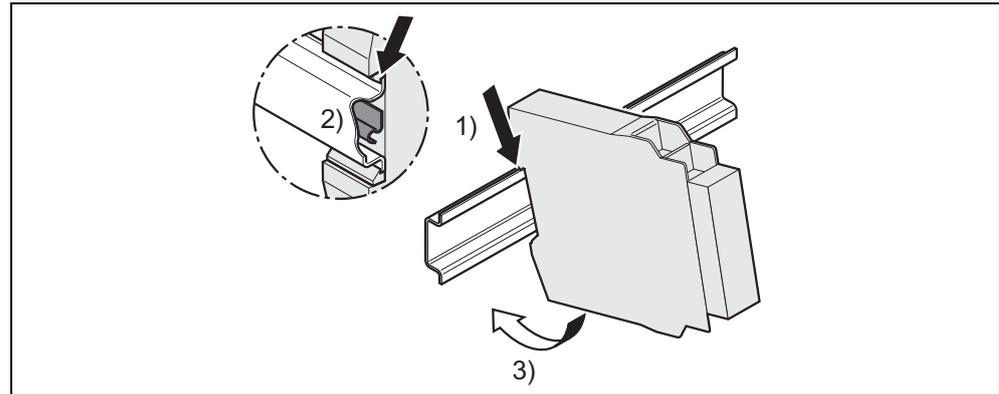
**While supply voltage is applied, modules must not be plugged to nor be removed from the MELSEC-WS safety controller.**

**To ensure full electromagnetic compatibility (EMC), the DIN mounting rail must be connected to functional earth (FE). Additionally connect all network cable shields directly at the control cabinet entrance to a common FE ground line.**

---

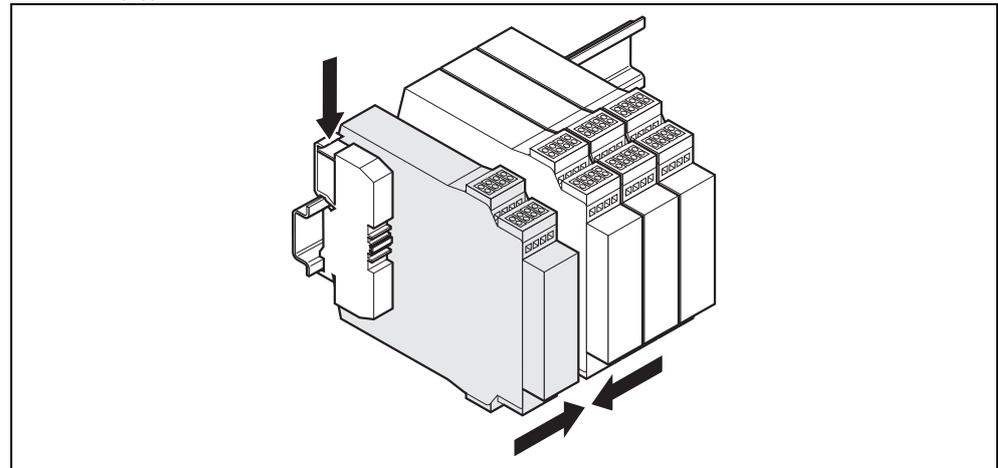
- In a MELSEC-WS safety controller, the WS0-CPU0 or WS0-CPU1 module is positioned at the extreme left.
- The two optional network modules follow directly to the right of the CPU module.
- Connect further MELSEC-WS safety I/O modules (e.g. WS0-XTIO or WS0-XTDI) onto the right side of the network modules and any additional safety relay output modules (WS0-4RO) to the extreme right of the entire MELSEC-WS safety controller.
- Mount the modules in accordance with EN 50274.
- The modules are located in a 22.5 mm wide modular system for 35 mm DIN rails according to EN 60715.
- Ensure that suitable ESD protective measures are taken during mounting. Otherwise the FLEXBUS+ backplane bus may be damaged.
- The connection between the modules is effected by means of the plug connection integrated in the housing. Take into account that, when replacing a module, the MELSEC-WS modules have to be pushed approx. 10 mm apart before the corresponding module can be removed from the DIN rail.
- Take suitable measures to ensure that foreign matter does not penetrate the connector openings, in particular that of the memory plug.

Figure 2:  
Mounting the module  
onto the DIN rail



- ⇒ Make sure that the voltage supply of the MELSEC-WS safety controller is switched off.
- ⇒ Hang the device onto the DIN rail (1)).
- ⇒ Connect the Ethernet interface module directly onto the right side of the CPU module of the MELSEC-WS safety controller. Up to two network modules per system are possible.
- ⇒ Ensure that the earthing spring contact is positioned correctly (2)). The earthing spring contact of the module must contact the DIN rail securely to allow electrical conductivity.
- ⇒ Latch the module onto the DIN rail by pressing it lightly in the direction of the arrow (3)).

Figure 3:  
Installing the end clips



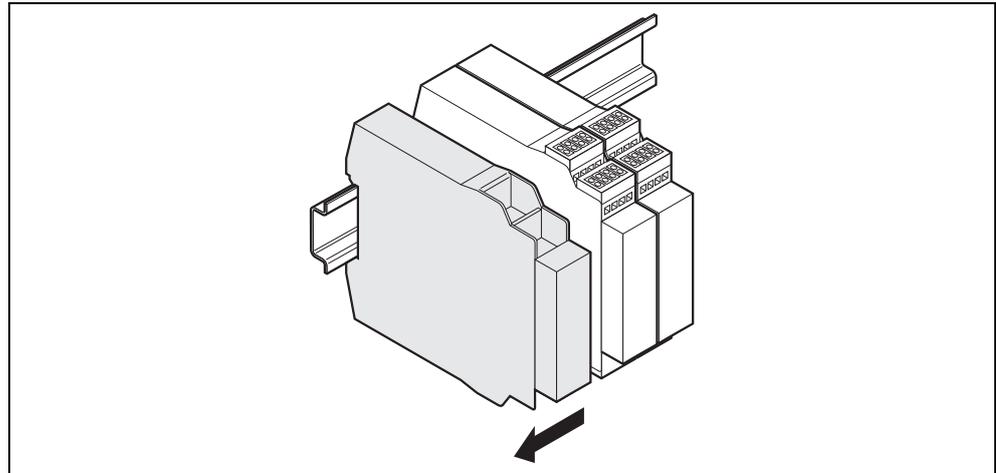
- ⇒ If there are several modules, slide the modules together individually in the direction of the arrow until the side plug connection latches in.
- ⇒ Install the end clips on the right and left.

The following steps are necessary after mounting:

- Complete the electrical connections (See Section 3.3.)
- Configuration (See Section 3.4 and the Safety Controller Setting and Monitoring Tool Operating Manual.)
- Checking the installation (see the Safety Controller User's Manual.)

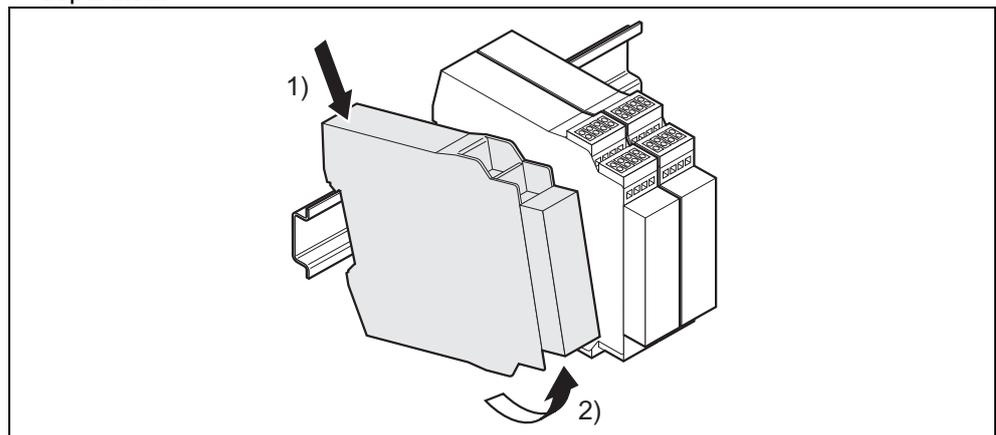
### 3.2.2 Steps for dismantling the modules

Figure 4:  
Disconnecting the plug  
connections



- ⇒ Disconnect the cables and remove the end clips.
- ⇒ If there are several modules, slide the modules away from each other individually in the direction of the arrow until the side plug connection is separated.

Figure 5:  
Removing modules  
from the DIN rail



- ⇒ Press the module downwards at the rear (1)) and remove it from the DIN rail in the direction of the arrow while keeping it pressed down (2)).

### 3.3 Electrical installation




---

#### Switch the entire machine/system off line!

The system could start up unexpectedly while you are connecting the devices.

---

- Note**
- The Ethernet interface module fulfills the EMC requirements in accordance with the basic specification EN 61000-6-2 for industrial use.
  - To ensure full electromagnetic compatibility (EMC), the mounting rail has to be connected to functional earth (FE).
  - The control cabinet or assembly casing of the MELSEC-WS safety controller must comply at least with enclosure rating IP 54.
  - Mount the modules in accordance with EN 50274.
  - Electrical installation in accordance with EN 60204-1
  - The voltage supply of the devices must be capable of buffering brief mains voltage failures of 20 ms as specified in EN 60204-1.
  - The voltage supply has to fulfill the regulations for extra-low voltages with safe separation (SELV, PELV) in accordance with EN 60664 and DIN 50178 (equipment of electrical power installation with electronic devices).
  - Ensure that all the modules of the MELSEC-WS safety controller, the connected protective devices as well as the voltage supplies are connected with the same earth (GND). The GND of the RS-232 interface is connected internally to the GND of the supply of the CPU module (A2).
  - Connect all fieldbus and Ethernet cable shields directly at the control cabinet entrance to the functional earth (FE).

### 3.4 First configuration steps

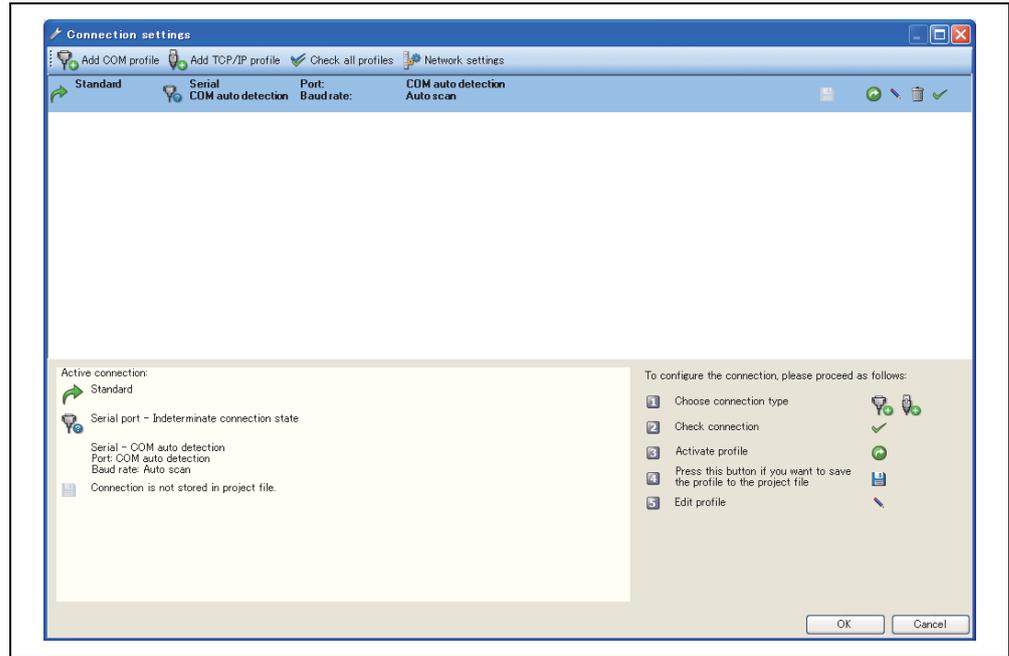
This chapter describes the basic steps you have to perform for the configuration of the Ethernet interface module:

- Establish a first connection between the MELSEC-WS safety controller and a PC or notebook.
- Upload or transfer of a configuration
- Verification of a configuration

#### 3.4.1 Establishing a connection between MELSEC-WS safety controller and PC

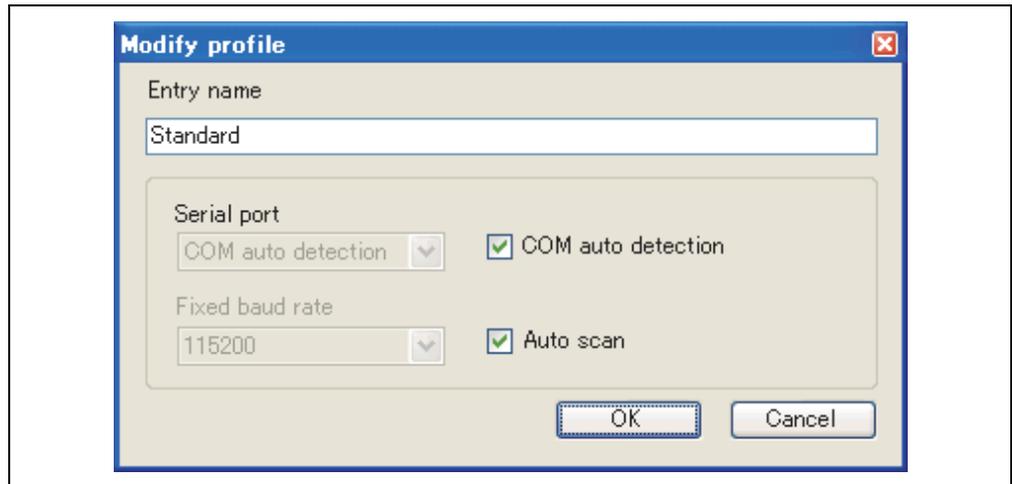
- ⇒ Connect a PC or notebook to the RS-232 interface of the CPU module.
- ⇒ Power on the MELSEC-WS safety controller.
- ⇒ Open the Setting and Monitoring Tool installed on the PC.
- ⇒ Click on **Com settings** to ensure the correct communication interface has been selected. The following dialog appears:

Figure 6:  
Connection settings  
dialog



⇒ To edit the settings click on the pencil icon to the right. The following dialog appears:

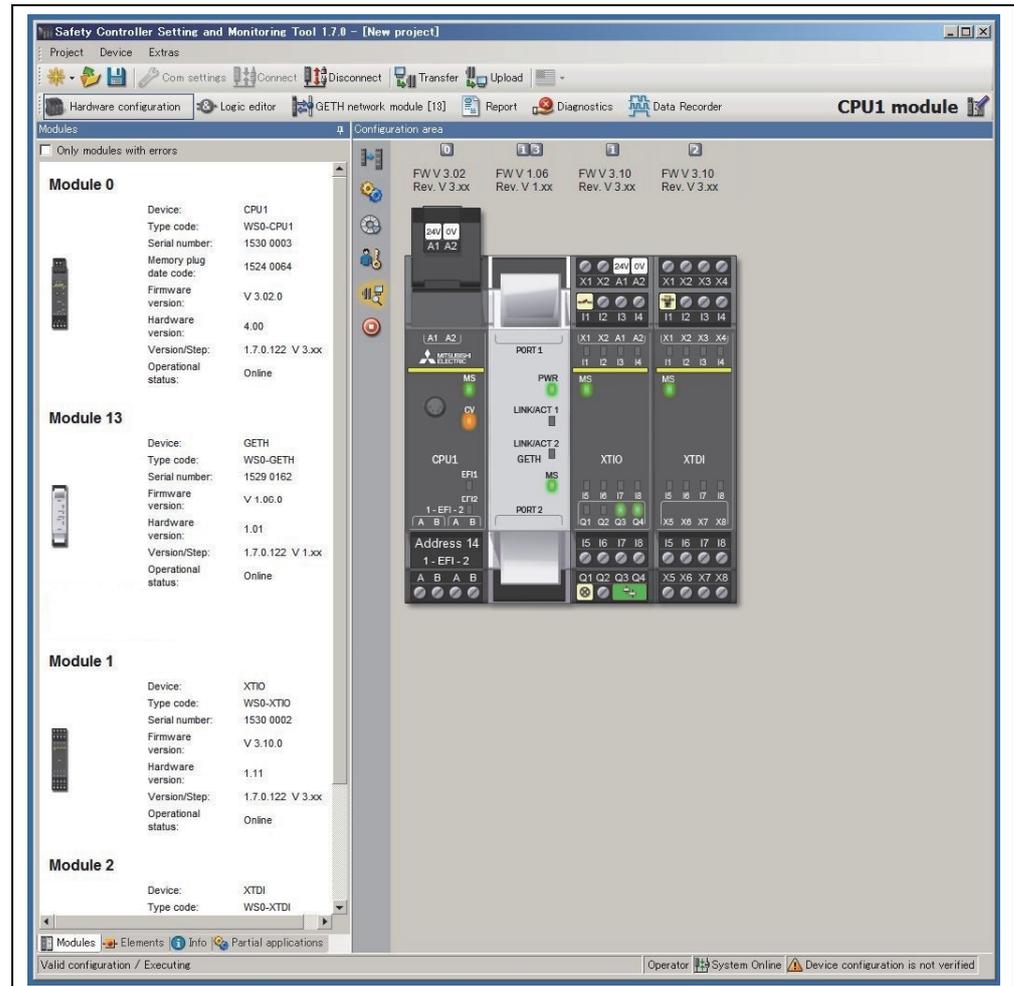
Figure 7:  
Modify profile dialog



- ⇒ Modify the settings if required.
- ⇒ Click **OK**. The dialog closes.
- ⇒ Click on **Connect**. The Setting and Monitoring Tool will search for connected MELSEC-WS safety controller and load the hardware configuration into the **Hardware configuration** dialog. Once all modules have been identified correctly, the Setting and Monitoring Tool will ask whether the configuration shall be uploaded.
- ⇒ Click **Yes** to upload the configuration.

As an example, the following hardware configuration may appear:

Figure 8:  
Hardware configuration  
dialog

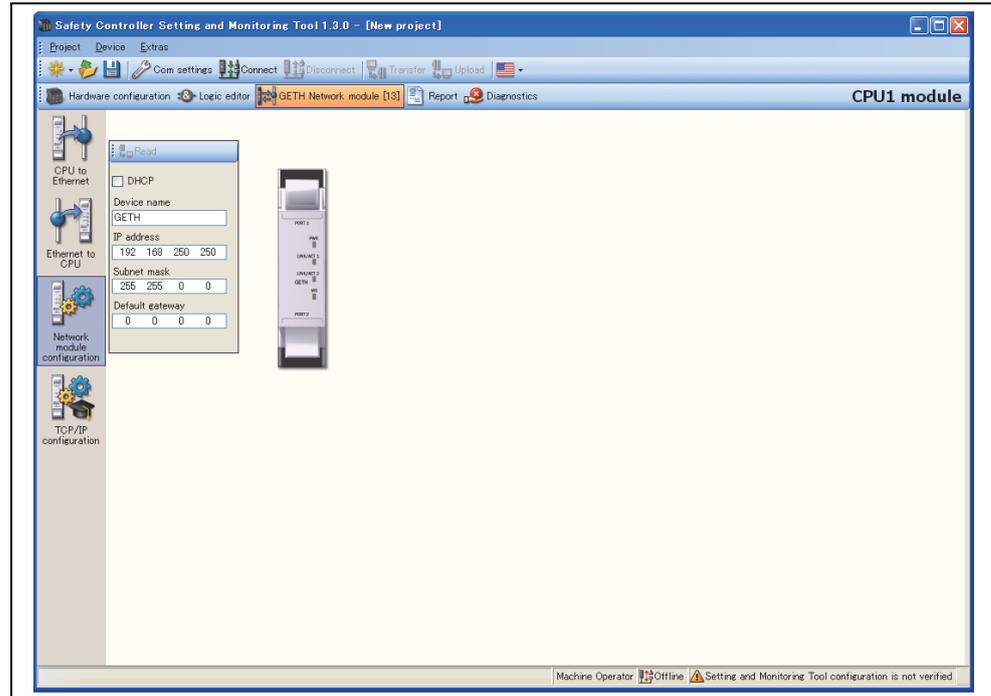


⇒ Click **Disconnect** to go into the offline mode if you want to change the configuration of the MELSEC-WS modules.

### 3.4.2 Assignment of an IP address

- ⇒ If your project is online, click on the **Disconnect** button to go offline.
- ⇒ Click on the **GETH Network module [13]** button (**Network Modules** button when two network modules are used) above the main window and select the desired Ethernet interface module.
- ⇒ Click on **Network module configuration** on the left hand menu. The following dialog appears:

Figure 9:  
Network module  
configuration dialog



On the left side of the dialog you will find the area for the Ethernet interface module IP configuration.

- ⇒ If desired, enter a **Device name** for the Ethernet interface module.
- ⇒ Enter a valid **IP address**, for the Ethernet interface module, and if required a valid **Subnet mask** and a valid IP address for a **Default gateway**.

**Or:**

- ⇒ If your network uses a DHCP server, activate the **DHCP** checkbox.

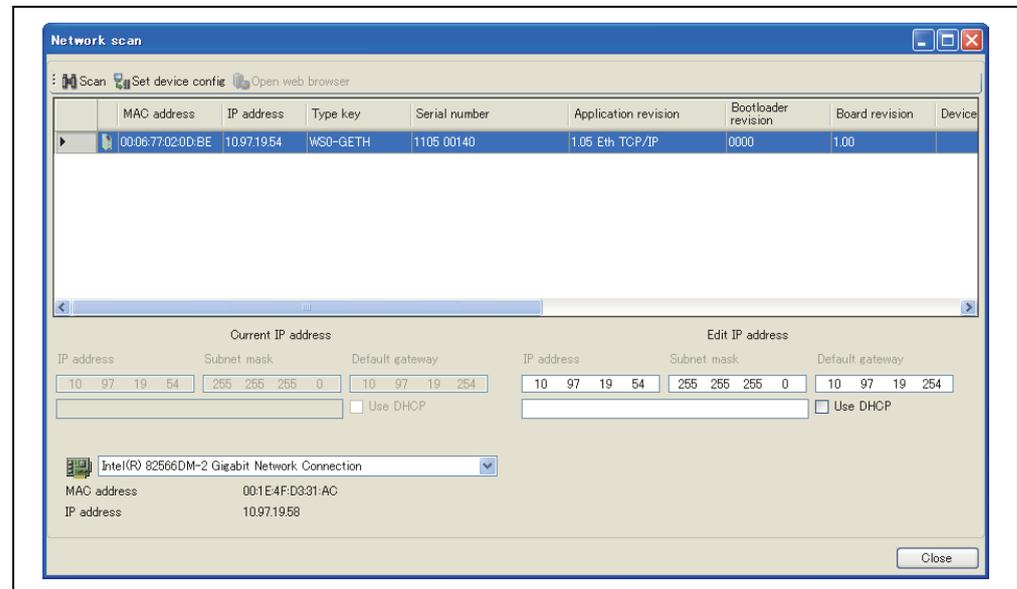
**Note**

- If your project is online, you can use the **Read** button at the upper left corner of the Ethernet interface module IP configuration area to retrieve the current IP settings of the Ethernet interface module.
- The out-of-the-box default IP address of the Ethernet interface module is 192.168.250.250.

You can assign an IP address from **Network settings** in the **Connection settings** dialog.

- ⇒ Click on the **Network settings** button. The **Network scan** dialog is opened.
- ⇒ Click on the **Scan** button. Your network is scanned for connected network modules and the network modules found are displayed in the list.

Figure 10:  
Found network modules  
in the Network scan  
dialog



- ⇒ Click on the Ethernet interface module you want to edit.
- ⇒ Enter the new settings in the **Edit IP address** area.
- ⇒ Click on the **Set device config** button to transfer the new settings to the Ethernet interface module.

- Note**
- If multiple Ethernet interface modules are connected to the PC, network scanning may fail. In that case, disable the unused modules.
  - The **Open web browser** button is unavailable for Ethernet interface modules.

### 3.4.3 Configuration of the Ethernet interface module

For the configuration of the Ethernet interface module, please refer to Section 3.5. For the configuration of the input data set 1 (data transfer from and to the network), please refer to Section 4.4 and Section 4.5.

### 3.4.4 Transfer of a configuration

Once you have finished the configuration, you have to transfer the configuration to your MELSEC-WS safety controller. In order to transfer a configuration, perform the following steps:

- ⇒ Click **Connect** to go online. The Setting and Monitoring Tool connects to the MELSEC-WS safety controller.
- ⇒ Click **Transfer** to transfer the configuration to the MELSEC-WS safety controller.

**Note** Depending on your current user level, you will be prompted to log on as Administrator to be able to transfer a configuration. For details please see the Safety Controller Setting and Monitoring Tool Operating Manual.

- ⇒ Once the transfer has been completed, you will be asked whether you want to run the CPU module. Depending on your choice, click **Yes** or **No** to leave the dialog.

**Note** You can also start and stop the application in the **Hardware configuration** view using the **Run application** or **Stop application** buttons while the project is online. More information can be found in the Safety Controller Setting and Monitoring Tool Operating Manual.

### 3.4.5 Verification of a configuration

After the configuration has been transferred successfully, the MELSEC-WS safety controller can be verified. To this purpose, the downloaded configuration data are read back out from the safety controller and compared with the project data. If they match, the data are displayed in a report. If the user confirms that they are correct, the safety controller is considered to be verified.

- ⇒ In the **Hardware configuration** view, click the icon **Upload and verify configuration** and compare the configuration. The **Upload and verify result** window is opened.
- ⇒ Click **Yes** below at the question **Set device to verified?** if the displayed configuration is the expected configuration. The safety controller is then considered to be verified.

- Note**
- You have to be logged in as an Administrator in order to mark the device as “verified”.
  - If the verification is completed successfully, a “Read in and compare” report that provides the most important project information is created subsequently. You can print out or store this report.
  - The status verified/not verified is indicated in the lower right-hand corner of the Setting and Monitoring Tool and by the CV LED of the CPU module turning on.
  - Only if the device and the corresponding configuration have been marked as verified, the Auto RUN Mode is active in the configuration of the CPU module. If the configuration is not set to verified, the safety controller stays in Idle mode (CV LED on the CPU module flashing) after power up and needs to be set to **Run Mode** using the Setting and Monitoring Tool.
  - If differences between the project data and the read-back configuration data are detected, a corresponding message including information about possible actions is displayed. Verification of the configuration is not possible then. Observe the information in the error message for the further procedure. Terminate the dialog box by clicking **Close**.
  - If you change a verified configuration, the status is reset to “not verified”.  
**Exception:** If you make only non safety-related changes such as modifying the Ethernet interface module name, the Ethernet interface module’s IP address or the port number for a TCP/IP socket connection, the configuration status remains “verified”.

More information can be found in the Safety Controller Setting and Monitoring Tool Operating Manual.

### 3.4.6 Upload of a configuration

When in online mode, you can upload a configuration from the connected MELSEC-WS safety controller:

- ⇒ Click on **Upload**. The current configuration of the safety controller will be loaded into the Setting and Monitoring Tool and can be edited after going offline.

## 3.5 Features of the Ethernet interface module

### 3.5.1 TCP/IP configuration interface

The Ethernet interface module offers a TCP/IP configuration interface which allows the configuration of the MELSEC-WS safety controller over Ethernet TCP/IP. This runs parallel to the Ethernet TCP/IP.



---

**Do not connect to the MELSEC-WS safety controller via the RS-232 and the Ethernet interface at the same time!**

The MELSEC-WS safety controller can only communicate with one instance of the Setting and Monitoring Tool at one time. Connecting to the safety controller using multiple instances of the Setting and Monitoring Tool, either on a single PC or multiple PCs, may result in inconsistencies of the configuration and the diagnostics as well as in operational errors. This applies to both RS-232 and Ethernet connections equally.

---

In order to configure an interface for TCP/IP configuration for the first time, perform the following steps:

**Step 1: Assign an IP address**

- ⇒ Connect a PC or notebook to the RS-232 interface of the CPU module.
- ⇒ Power on the MELSEC-WS safety controller.
- ⇒ Open the Setting and Monitoring Tool installed on the PC and load the hardware configuration including the Ethernet interface module.
- ⇒ If your project is online, click on the **Disconnect** button to go offline.
- ⇒ Assign an IP address. For details, see Section 3.4.2.
- ⇒ Click on the **Connect** button to go online and transfer the new settings to the MELSEC-WS safety controller.

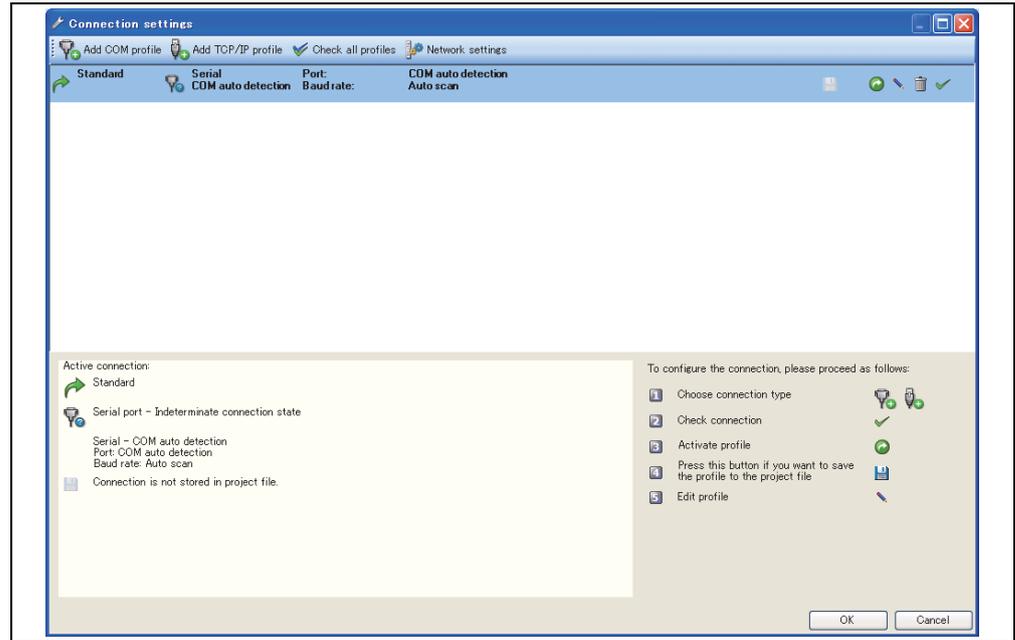
**Step 2: Add a TCP/IP profile to your project**

- ⇒ Connect one of the two Ethernet ports of the Ethernet interface module with your Ethernet network using a shielded Ethernet cable.
- ⇒ Connect a PC (or notebook) to the same Ethernet network. Ensure the IP address settings of the PC match the network setup.

**Note** You can also connect your PC directly to one of the two Ethernet ports of the Ethernet interface module. In this case, you can either adapt the IP address settings of your PC or the IP address settings of the Ethernet interface module to match the other device's IP setup.

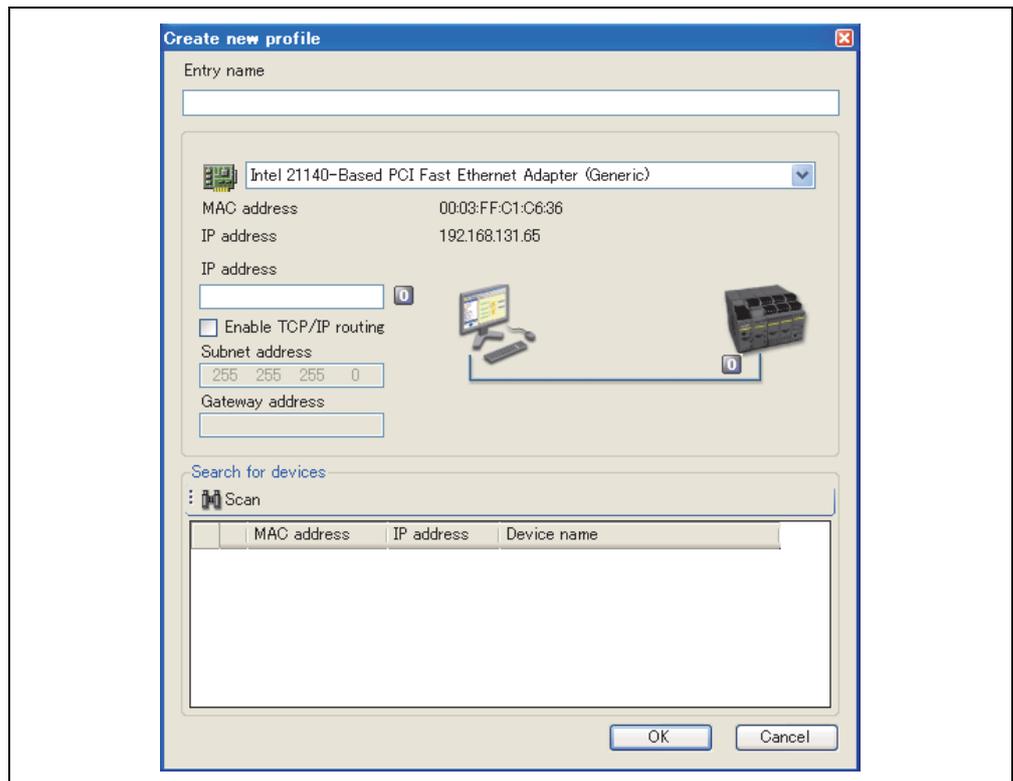
- ⇒ Open the Setting and Monitoring Tool installed on the PC and load the hardware configuration including the Ethernet interface module.
- ⇒ If your project is online, click on the **Disconnect** button to go offline.
- ⇒ Click on **Com settings**. The following dialog appears:

Figure 11:  
Com settings dialog



⇒ Click on **Add TCP/IP profile**. The following dialog appears:

Figure 12:  
Create new TCP/IP  
profile dialog

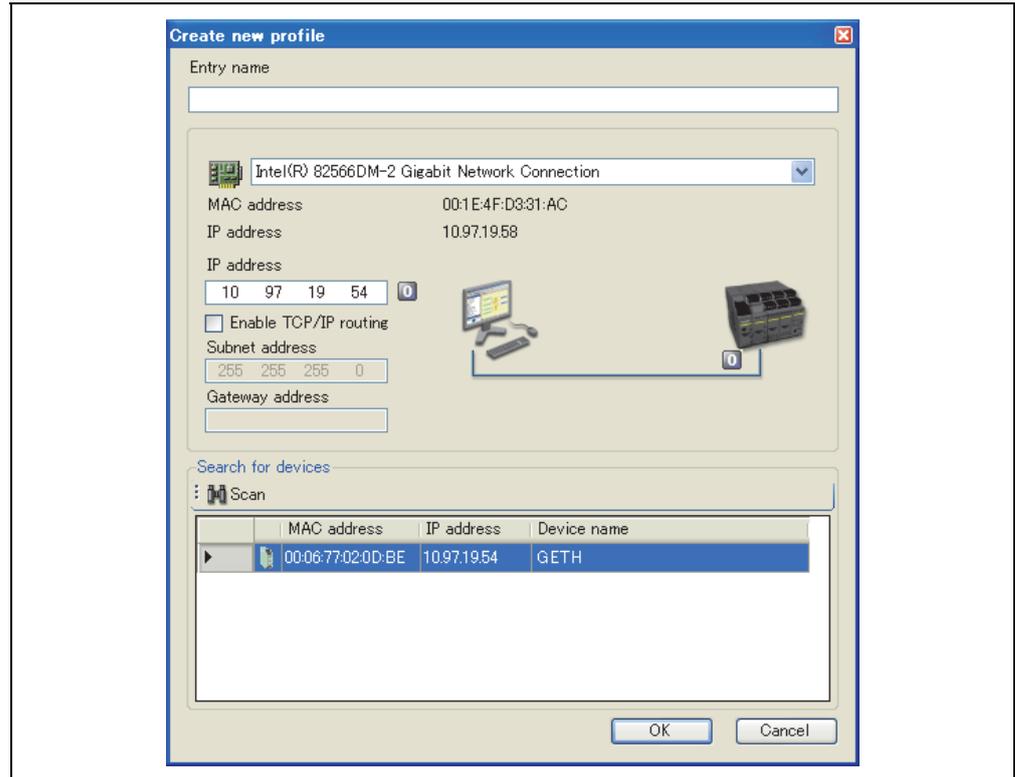


⇒ Select your Ethernet network adaptor from the drop down list.

⇒ Click on **Scan** to search for Ethernet interface modules on your Ethernet network. Ethernet interface modules located will be displayed as shown in the dialog below. The IP address will be displayed as well as MAC address and device name.

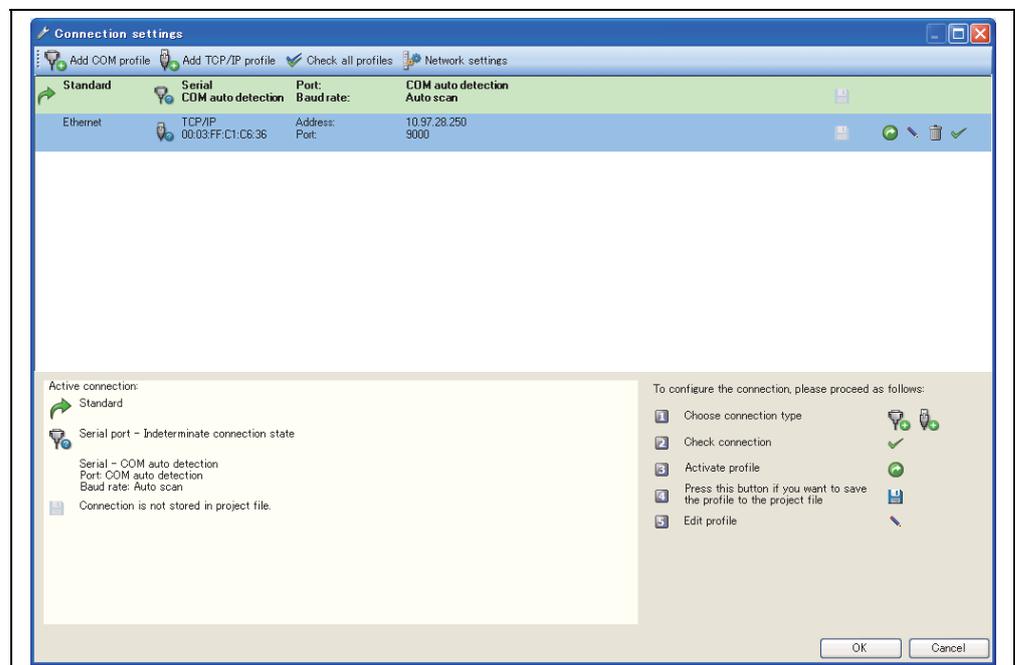
**Note** If multiple Ethernet interface modules are connected to the PC, network scanning may fail. In that case, disable the unused modules.

Figure 13:  
Create new TCP/IP  
profile dialog after scan  
has been performed



- ⇒ Select the Ethernet interface module that you want to use for the new profile.
- ⇒ Enter a name for the entry point to the **Entry name** edit field.
- ⇒ Click **OK**. The profile has now been created and is shown in the connection dialog:

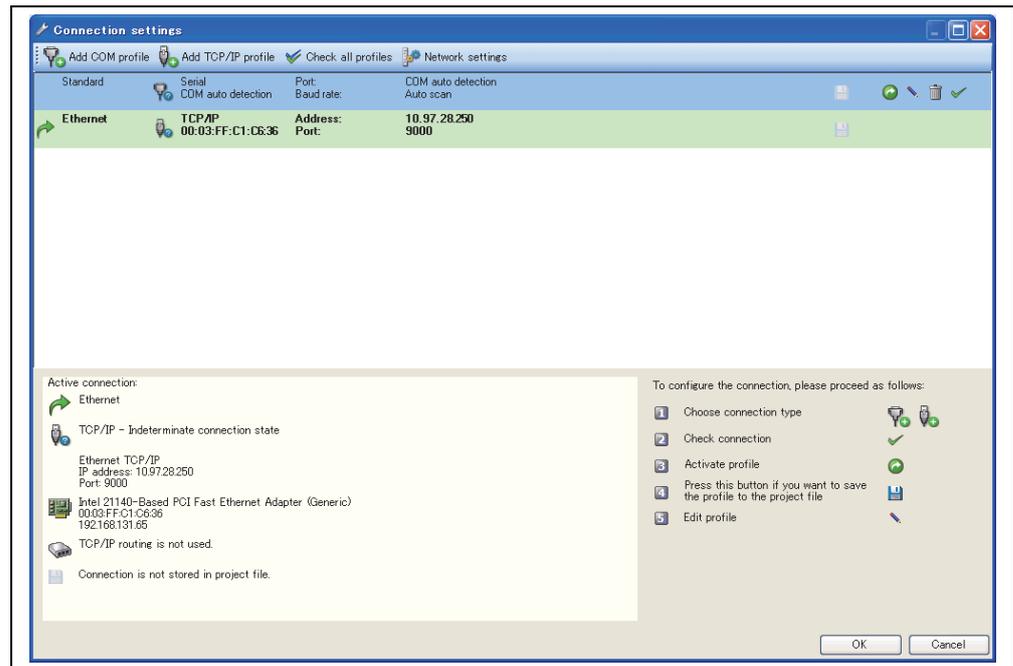
Figure 14:  
Connection settings  
dialog with new TCP/IP  
profile



In order to use this profile, it needs to be activated.

- ⇒ Click on the **Activate profile** icon (white arrow in green circle) on the far right. The entry point will then be activated and marked as such:

Figure 15:  
Connection settings  
dialog with new TCP/IP  
profile activated



⇒ Click **OK**. All communication to the MELSEC-WS safety controller will now happen via TCP/IP. In order to use the entry point via the serial interface again, you will have to re-activate it.

**Note** The port number for the TCP/IP configuration interface is pre-set to port 9000 and can not be changed.

### Step 3: Connect via TCP/IP

⇒ Click on the **Connect** button to go online.

**Note** While Setting and Monitoring Tool is connected via TCP/IP, do not transfer the parameters in which the IP address of the connected Ethernet interface module has been changed. Doing so will result in disconnection of communication. In that case, click on the **Disconnect** button to go offline, make the correct configuration in the **Connection settings** dialog, and then retry the connection.

### 3.5.2 Ethernet TCP/IP socket interface

Each Ethernet interface module supports a total number of four TCP/IP socket interfaces. This allows up to four different applications to communicate with the Ethernet interface module at the same time over Ethernet TCP/IP.



#### Do not use the same output data set number for two different programmable controller connections or TCP/IP sockets!

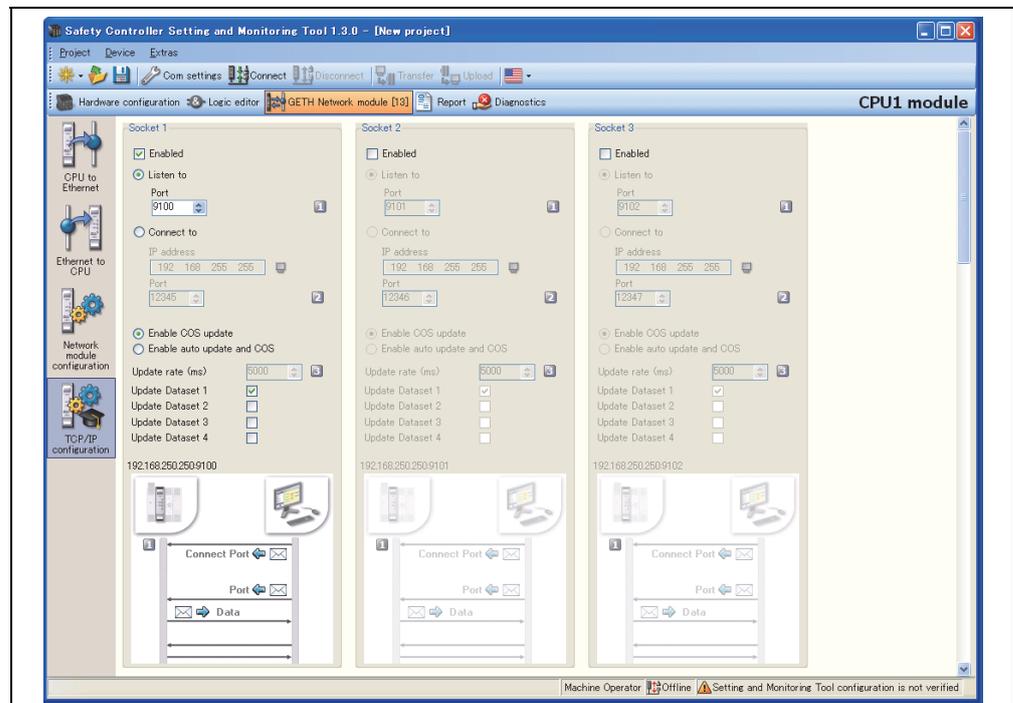
The output data set can be written to the Ethernet interface modules in parallel by all communication interfaces or TCP/IP sockets, if they use the same output data set number. In that case the last message overrides data received earlier.

The Ethernet interface module processes the data of a MELSEC-WS safety controller and makes it available in different compilations, the data sets. These data sets are available over the TCP/IP interface. For a detailed description of the data sets please refer to Chapter 4.

In order to configure the Ethernet TCP/IP socket interface, perform the following steps:

- ⇒ Open the Setting and Monitoring Tool and load the hardware configuration including the Ethernet interface module.
- ⇒ Click on the **GETH Network module [13]** button (**Network Modules** button when two network modules are used) above the main window and select the desired network module or double click on the desired network module in the hardware configuration view to open the network module configuration dialog.
- ⇒ Click on **TCP/IP configuration** on the left hand menu. The following dialog appears:

Figure 16:  
TCP/IP configuration  
dialog



### Configuration of the TCP/IP interface - who establishes the connection

If the Ethernet interface module shall connect to the external application, perform the following configuration steps:

- ⇒ Activate the **Connect to** radio button.
- ⇒ Set **IP address** to the IP address of the computer the application is running on.
- ⇒ Enter the **Port** number for the application.

**Note** The configuration is considered faulty if either the connect socket port and/or the connect IP address is zero when in **Connect** mode.

If the external application shall connect to the Ethernet interface module, perform the following configuration steps:

- ⇒ Activate the **Listen to** radio button.
- ⇒ Enter the **Port** number for the application.

**Note**

- Suggested port numbers are 9100 to 9103 (default values).
- Port 0 and port 9000 are reserved and can not be used (faulty configuration).
- Port numbers 0 to 1023 are managed by the Internet Assigned Numbers Authority (IANA) and can not be used. See [www.iana.org/assignments/port-numbers](http://www.iana.org/assignments/port-numbers)

Finally, determine how the data is transferred. Follow the steps outlined in the following section.

### Data transfer method - how the data is transferred

Whenever the TCP/IP socket connection has been established (either by an application on a PC or by the Ethernet interface module itself), there are two possible methods how the data sets can be transferred:

- The application requests the data set(s) per command message (Application requests (Polling) mode),
- or**
- the Ethernet interface module auto-updates the data sets as per configuration (Ethernet interface module writes to Address/Port (Auto update) mode).

In the auto update mode, there are two update modes how the Ethernet interface module update the data:

- Update COS (Change of state): when any data of the input data set change status.
- Heartbeat interval: data will be sent automatically according to the configured **Heartbeat rate** in ms.

**Note** If Heartbeat interval is enabled, a change of state will trigger an immediate update of the data as well, regardless of the set heartbeat rate. (I.e. COS is always active.)

For both methods the following structure of messages applies.

**General telegram structure**

The request/response message (e.g. telegram) is structured as shown below:

0	1	...	...	...	...	...	...	...	...	...	...	...	...	n
<b>Command</b>	<b>Parameter(s)</b> (content depends on type of command)											<b>Data</b>		

Each element in the request/response messages is transferred in Big Endian format.

Table 4:  
Telegram structure

Parameter	Length	Description
Command	WORD	0000H = Undefined (no command) <b>Polling mode (input data set reading)</b> 00F1H = Input data set(s) request message 001FH = Input data set(s) response message <b>Polling mode (output data set writing)</b> 00F2H = Write output data set settings 002FH = Response to write output data set settings <b>Auto update mode</b> 00E1H = Auto update control 001EH = Auto update control response 002EH = Auto update input data set(s) message
Parameter(s)	Length determined by command	As defined in specific command
Data	Length determined by command	As defined in specific command

**Error response to invalid messages**

The Ethernet interface module will set the most significant bit of the command word in the event that an invalid or improperly formatted message is received.

Table 5:  
Error response message

Parameter	Length	Description
Command	WORD	Bit 15 of received command will be set. (i.e. command of 00F2H would become 80F2H.)
Following data	Length determined by command	Unchanged. Returned as it was received.

**Application requests (Polling) mode**

In this mode the Ethernet interface module will only send any data upon request (e.g. polling). Therefore the application shall send request telegrams as per definition below and the Ethernet interface module will respond with telegrams structured as per definition below.

**Input data set reading**

The request message is sent by an application to the Ethernet interface module. The request message telegram shall be structured as shown below:

Table 6:  
Input data set read  
request

Parameter	Length	Value
Command	WORD	00F1 <sub>H</sub> = Data set(s) request message
Request data set 1	WORD	0 = Do not send data set 1. 1 = Send data set 1.
Request data set 2	WORD	0 = Do not send data set 2. 1 = Send data set 2.
Request data set 3	WORD	0 = Do not send data set 3. 1 = Send data set 3.
Request data set 4	WORD	0 = Do not send data set 4. 1 = Send data set 4.

The response message is returned to the application by the Ethernet interface module. The response message telegram will be structured as shown below:

Table 7:  
Input data set read  
response

Parameter	Length	Value
Command	WORD	001F <sub>H</sub> = Data set(s) response message
Data set 1 length	WORD	0 = Data set not returned in data set(s) data field Non-zero = Length of data set
Data set 2 length	WORD	0 = Data set not returned in data set(s) data field Non-zero = Length of data set
Data set 3 length	WORD	0 = Data set not returned in data set(s) data field Non-zero = Length of data set
Data set 4 length	WORD	0 = Data set not returned in data set(s) data field Non-zero = Length of data set
Data set(s) data	Array of bytes	Data set(s) information

### Output data set writing

The following command message is sent by the application to the Ethernet interface module to write to the output data sets:

Table 8:  
Output data set write  
request

Parameter	Length	Value
Command	WORD	00F2 <sub>H</sub> = Set output data set(s) command message
Output data set 1 length	WORD	0 = Output data set not included in data set(s) data field Non-zero = Length of data set (10 bytes)
Output data set 2 length	WORD	0 = Output data set not included in data set(s) data field Non-zero = Length of data set (10 bytes)
Output data set 3 length	WORD	0 = Output data set not included in data set(s) data field Non-zero = Length of data set (10 bytes)
Output data set 4 length	WORD	0 = Output data set not included in data set(s) data field Non-zero = Length of data set (10 bytes)
Output data set 5 length	WORD	0 = Output data set not included in data set(s) data field Non-zero = Length of data set (10 bytes)
Data set(s) data	Array of bytes	Data set(s) information

**Note**

Each output data set length of the Ethernet interface module is 10 bytes long.

The response message is returned to the application by the Ethernet interface module. The response message telegram is structured as shown below:

Table 9:  
Output data set write  
response

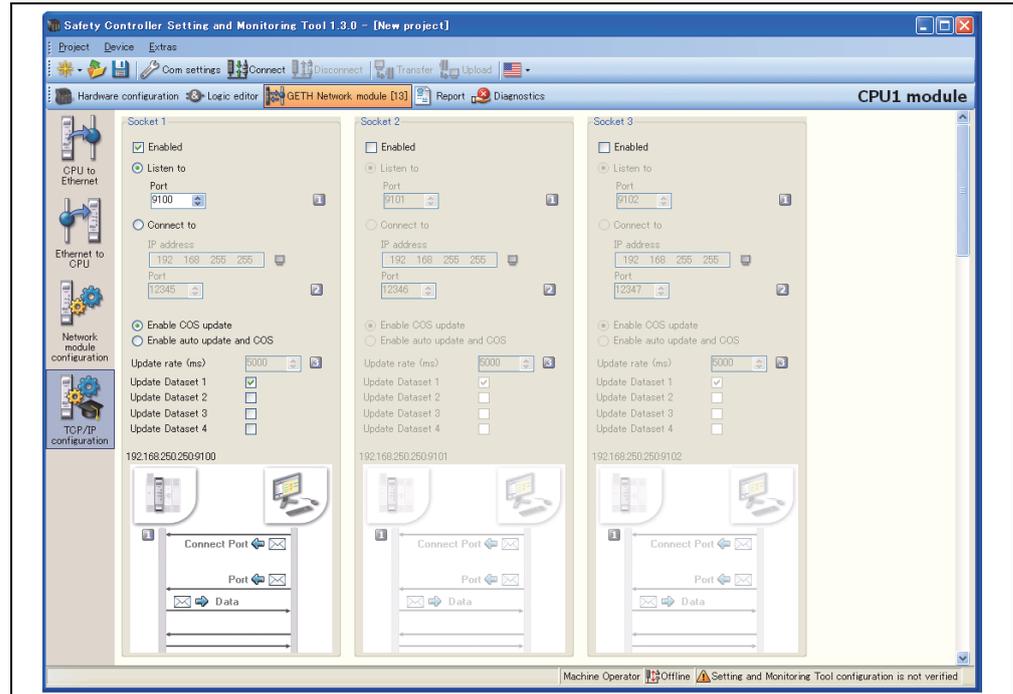
Parameter	Length	Value
Command	WORD	002FH = Response to write output data set settings message
Status	WORD	0 = Success. Output data sets written correctly. 1 = Error - Can not write output data sets due to any of: <ul style="list-style-type: none"> <li>- Stop state of the CPU module</li> <li>- Loss of backplane communication</li> <li>- Incorrect routing information</li> </ul>

**Configuration via Setting and Monitoring Tool**

In order to configure the polling mode of the Ethernet interface module via the Setting and Monitoring Tool, perform the following steps:

- ⇒ Open the Setting and Monitoring Tool and load the hardware configuration including the Ethernet interface module.
- ⇒ Click on the **GETH Network module [13]** button (**Network Modules** button when two network modules are used) above the main window and select the desired Ethernet interface module or double click the desired Ethernet interface module in the hardware configuration view to open the network module configuration dialog.
- ⇒ Click on **TCP/IP configuration** on the left hand menu. The following dialog appears:

Figure 17:  
TCP/IP configuration for  
polling mode



- ⇒ Check the **Listen to** checkbox.
- ⇒ Enter the **Port** number on which the application will connect.
- ⇒ Select the update mode: **Enable COS update** or **Enable auto update and COS**.
- ⇒ If you have selected **Enable auto update and COS**, select the **Update rate** in ms. (Configurable range: 40 to 65535)
- ⇒ Select which data sets shall be updated: Check the **Update Dataset n** checkbox.

### Ethernet interface module writes to Address/Port (Auto update) mode

The Ethernet interface module can be configured to automatically update the data set information (i.e. the application does not need to send any request messages as it would do in polling mode) once the connection to the application has been made.

The configuration settings are available via the Setting and Monitoring Tool or via the TCP/IP interface itself. Using one interface does not disable the other: The auto update mode could be enabled via Setting and Monitoring Tool and disabled via TCP/IP command, for example.

### Configuration via TCP/IP interface

This command message is sent by an application to the Ethernet interface module to configure the auto update mode. This message can be used to either disable or enable the auto update mode directly through the TCP/IP interface.

Table 10:  
Auto update mode  
configuration command

Parameter	Length	Value
Command	WORD	00E1 <sub>H</sub> = Auto update control
Request data set 1	WORD	0 = Do not send data set 1. 1 = Send data set 1.
Request data set 2	WORD	0 = Do not send data set 2. 1 = Send data set 2.
Request data set 3	WORD	0 = Do not send data set 3. 1 = Send data set 3.
Request data set 4	WORD	0 = Do not send data set 4. 1 = Send data set 4.
Auto update mode update rate	WORD	0 = Disable auto update messages. Non-zero = Enable auto update message at specified rate in ms. (If the value is 40 or less, data sets are sent at 40ms interval.) Minimum = 40 ms, Maximum = 65535 ms

**Note** Auto update is disabled if all Request Input Data Set flags are set to zero.

The response message returned to the application by the Ethernet interface module:

Table 11:  
Auto update mode  
configuration response

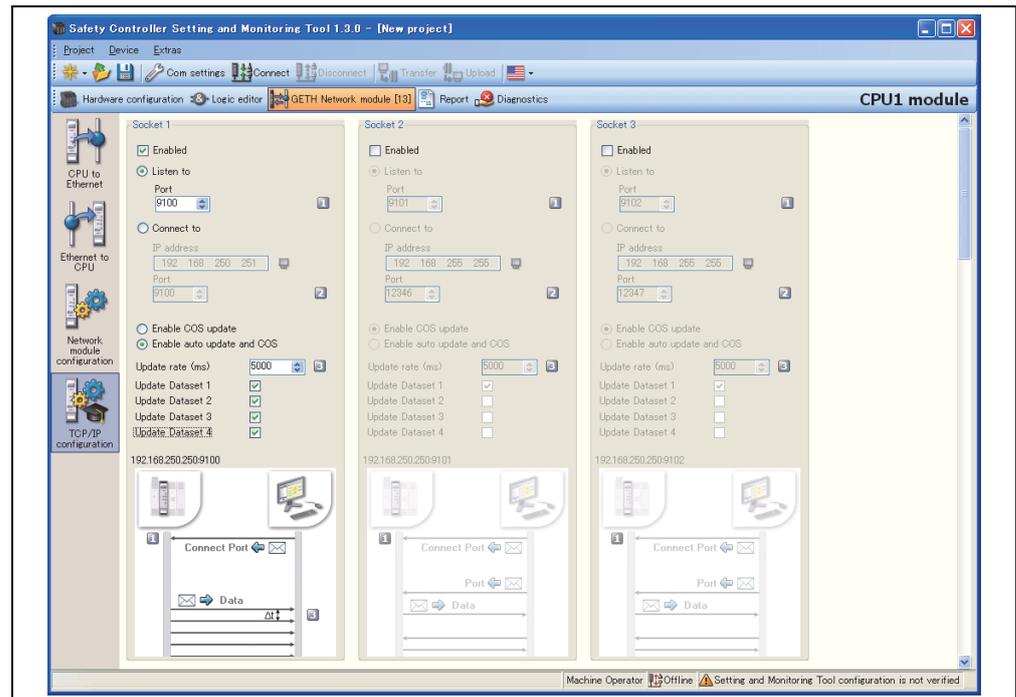
Parameter	Length	Value
Command	WORD	001E <sub>H</sub> = Response to the auto update control message

### Configuration via Setting and Monitoring Tool

In order to configure the auto update mode of the Ethernet interface module via the Setting and Monitoring Tool, perform the following steps:

- ⇒ Open the Setting and Monitoring Tool and load the hardware configuration including the Ethernet interface module.
- ⇒ Click on the **GETH Network module [13]** button (**Network Modules** button when two network modules are used) above the main window and select the desired network module or double click on the desired network module in the hardware configuration view to open the network module configuration dialog.
- ⇒ Click on **TCP/IP configuration** on the left hand menu. The following dialog appears:

Figure 18:  
TCP/IP configuration for  
auto update



- ⇒ Check the **Connect** checkbox.
- ⇒ Enter the **IP address** and the **Port** number the Ethernet interface module shall write to.
- ⇒ Select the update mode: **Enable COS update** or **Enable auto update and COS**.
- ⇒ If you have selected **Enable auto update and COS**, select the **Update rate** in ms. (Configurable range: 40 to 65535)
- ⇒ Select which data sets shall be updated: Check the **Update Dataset n** checkbox.

**Normal operation**

The following message is sent from the Ethernet interface module to the application while operating in auto update mode.

Table 12:  
Auto update mode  
normal operation  
message

Parameter	Length	Value
Command	WORD	002EH = Auto update data set(s) message
Data set 1 length	WORD	0 = Data set not returned in data set(s) data field Non-zero = Length of data set
Data set 2 length	WORD	0 = Data set not returned in data set(s) data field Non-zero = Length of data set
Data set 3 length	WORD	0 = Data set not returned in data set(s) data field Non-zero = Length of data set
Data set 4 length	WORD	0 = Data set not returned in data set(s) data field Non-zero = Length of data set
Data set(s) data	Array of bytes (length dependent on set definition)	Data set(s) information. Details see Section 4.1 and Section 4.3.

**3.5.3 TCP/IP process image example**

The following example shows a possible process image sent by an Ethernet interface module via TCP/IP in auto update mode:

Table 13:  
TCP/IP process image  
example

Byte values (hex)	Part of message	Meaning
00 2E	Command	Auto update data sets (see Table 12)
00 32	Command parameters	Length of data set 1: 50 bytes
00 20		Length of data set 2: 32 bytes
00 3C		Length of data set 3: 60 bytes
00 3C		Length of data set 4: 60 bytes
03 FF 03 03	Data set 1 (default byte assignments, see Table 15)	Logic results 0-3
C0		Input values module 1: C0 = 11000000 = Inputs I8 and I7 <b>Active</b>
03		Input values module 2: 03 = 00000011 = Inputs I2 and I1 <b>Active</b>
3F 05		Input values module 3-12
05 05 00 00		Output values module 1-12
00 00 00 00		
00 00 00 00		Not assigned
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00		

Byte values (hex)	Part of message	Meaning
52 A1 10 4C	Data set 2 (see Table 15)	Overall CRC (same as system CRC)
52 A1 10 4C		System CRC
00 00 00 00		Reserved
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
FF FF FF FF	Data set 3 (see Table 15 and Table 16) Data set 3 data is transferred in Big Endian format, i.e. in 32 bit double word format with the most significant byte (byte 3) placed in the leftmost position.	Status module 0 (WS0-CPU0/WS0-CPU1): OK
FF FF FF FF		Status module 1 (e.g. WS0-XTDI): OK
FD FB FF FF		Status module 2 (e.g. WS0-XTIO): Byte 0: FF = 11111111: No errors Byte 1: FF = 11111111: No errors Byte 2: FB = 11111011: Input 3 external test signal failure Byte 3: FD = 11111101: Output 1 stuck-at-low error
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF		Status modules 3-6: OK
FF FF FF FF FF FF FF FF		Status modules 7-12 (no modules present)
FF FF FF FF		Status module 13 (e.g. WS0-GETH): OK
FF FF FF FF		Status module 14 (no module present)

Byte values (hex)	Part of message	Meaning
00 00 00 00	Data set 4	Reserved
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		
00 00 00 00		

## 4. Data sets

The Ethernet interface module allows the MELSEC-WS safety controller to send and receive non-safety related data to and from the external fieldbus system for control and diagnostics purposes.

**Note** In this manual, the data exchanged between the MELSEC-WS safety controller and the respective network will be considered always from the network master (programmable controller) point of view. Therefore data sent from the MELSEC-WS safety controller into the network will be referred to as input data while data received from the network will be referred to as output data.



---

**Do not operate an Ethernet interface module on a safety fieldbus!**

The Ethernet interface module is not suitable for operation on a safety fieldbus. It does not support any safety mechanism, which would be mandatory to communicate within a safety network.

---

Configuration of the Ethernet interface module is performed using the Setting and Monitoring Tool on a PC or notebook connected to the CPU module over RS-232 interface or connected to the Ethernet interface module over Ethernet TCP/IP. Configuration data can be exported, imported, and saved.

The safety relevant logic of the MELSEC-WS safety controller operates independently from the Ethernet interface module. If however the MELSEC-WS safety controller has been configured to integrate non-safe information from the fieldbus into the logic editor, a decoupling of the Ethernet interface module can result in availability problems.

An Ethernet interface module can only be operated on a MELSEC-WS safety controller. It does not have a dedicated voltage supply. It is possible to use two network modules per system.

The Ethernet interface module is fitted in a 22.5 mm wide housing for 35 mm rails in accordance with EN 60715.

## 4.1 Data transmitted into the network (network input data sets)

### Available data

The Ethernet interface module can provide the following data:

- Operational data
  - **Logic results** from the CPU module (see Section 4.1.1)
  - **Input values** (Active/Inactive) for all safety I/O modules in the system and EFI devices connected (see Section 4.1.2)
  - **Output values** (Active/Inactive) for all safety I/O modules and EFI devices connected (see Section 4.1.2)
  - **Output data** from another network, i.e. data received by a second network module in the safety controller (see Section 4.1.3)
- Diagnostics
  - **Checksums** (CRCs) (see Section 4.1.4)
  - **Error and status information** for all modules except the safety relay output module (see Section 4.1.5)

### Data sets

Input data sets to be sent from the Ethernet interface module to the network consist of logic results, I/O state data, diagnostics data (e.g. module status, CRC), and other data. These data are organized into four input data sets.

Table 14:  
Description of input  
data sets 1 to 4

Input data set	Description	Size	Customization
1	<ul style="list-style-type: none"> <li>● Logic result</li> <li>● Module and EFI input and output values</li> <li>● Data set of the second network module</li> </ul>	50 bytes	Available
2	CRC	32 bytes	N/A
3	Error and status information of the module	60 bytes	N/A
4	Reserved	60 bytes	N/A

For details of data to be sent to the network, see Table 15 and Table 16.

You can customize the contents of input data set 1 in units of 1 byte. In the delivery status, the input data set 1 is pre-configured (default settings). (See Section 4.3 and Section 4.4.)

Table 15 shows the overview of data sets available for the Ethernet interface module.

Table 15:  
Overview input data  
sets 1 to 3

	Data set 1 (default)	Data set 2	Data set 3	
<b>Byte 0</b>	Logic result 0	Overall CRC	Module status module 0. Module 0 is always the CPU. For detailed information about the module status see Table 16.	
<b>Byte 1</b>	Logic result 1			
<b>Byte 2</b>	Logic result 2			
<b>Byte 3</b>	Logic result 3			
<b>Byte 4</b>	Input values module 1	System CRC (SCID)	Module status module 1	
<b>Byte 5</b>	Input values module 2			
<b>Byte 6</b>	Input values module 3			
<b>Byte 7</b>	Input values module 4			
<b>Byte 8</b>	Input values module 5	Reserved	Module status module 2	
<b>Byte 9</b>	Input values module 6			
<b>Byte 10</b>	Input values module 7			
<b>Byte 11</b>	Input values module 8			
<b>Byte 12</b>	Input values module 9	Verified configuration CRC	Module status module 3	
<b>Byte 13</b>	Input values module 10			
<b>Byte 14</b>	Input values module 11			
<b>Byte 15</b>	Input values module 12			
<b>Byte 16</b>	Output values module 1	Reserved	Module status module 4	
<b>Byte 17</b>	Output values module 2			
<b>Byte 18</b>	Output values module 3			
<b>Byte 19</b>	Output values module 4			
<b>Byte 20</b>	Output values module 5		Module status module 5	
<b>Byte 21</b>	Output values module 6			
<b>Byte 22</b>	Output values module 7			
<b>Byte 23</b>	Output values module 8			
<b>Byte 24</b>	Output values module 9		Module status module 6	
<b>Byte 25</b>	Output values module 10			
<b>Byte 26</b>	Output values module 11			
<b>Byte 27</b>	Output values module 12			
<b>Byte 28</b>	Not assigned		Module status module 7	
<b>Byte 29</b>	Not assigned			
<b>Byte 30</b>	Not assigned			
<b>Byte 31</b>	Not assigned			
<b>Byte ...</b>	Not assigned	...		
<b>Byte 49</b>	Not assigned			
<b>Byte ...</b>	No data			
<b>Byte 56</b>				
<b>Byte 57</b>				
<b>Byte 58</b>				
<b>Byte 59</b>		Module status module 14. Module 13 and 14 are always the network modules.		
<b>Length</b>	<b>50 bytes</b>		<b>32 bytes</b>	<b>60 bytes</b>

**Note** If a dual-channel element is connected to a safety I/O module, the low-order bit of the redundant I/O signals shows the active I/O state (ON/OFF) of the element.

#### 4.1.1 Logic results

Logic results generated by the logic editor of the CPU module can be made available to the network. Up to 20 bytes are available where each bit represents one logic result from the logic editor.

Data set 1 containing the logic results can be customized. For detailed information see Section 4.4.

#### 4.1.2 Module and EFI input and output values

The Ethernet interface module can transmit all input and output states of all MELSEC-WS modules and EFI devices connected to the MELSEC-WS safety controller into the network. Data set 1 containing the input and output values and the EFI information can be customized. For detailed information see Section 4.4.

##### Module input and output states

The input and output states of the modules are transmitted using one byte for each module's inputs and one byte for each module's outputs where each bit represents the state of one input or output (on/off).

##### EFI system information

The WS0-CPU1 and WS0-CPU3 have 2 EFI interfaces. An EFI interface is a safe communication interface between SICK devices. It allows to

- read out information from the safety devices (e.g. SICK C4000, S3000)
- transfer commands to the safety devices

The Ethernet interface module allows these EFI devices connected to the WS0-CPU1 and WS0-CPU3 to transmit their data into the network.

**Note** It is only possible to select the EFI data in byte arrays. 4 byte arrays for each connected EFI device are available. Some of the data contents are reserved and can not be used at the programmable controller.

Further information about the properties, functions and benefits of the EFI interfaces can be found in the Safety Controller User's Manual.

#### 4.1.3 Routing of data from a second network

If your MELSEC-WS safety controller contains two network modules, it is possible to route information received by the first network module from one network (e.g. from a programmable controller) into a second network via the second network module (e.g. to an HMI) and vice versa.

#### 4.1.4 Configuration checksums (CRCs)

Data set 2 contains the following configuration CRCs for the MELSEC-WS safety controller:

- Overall CRC (same as system CRC). The overall CRC is the checksum displayed in the Setting and Monitoring Tool report.
- System CRC (SCID)

Each checksum is four bytes long. Data set 2 can not be customized.

#### 4.1.5 Error and status information of the modules

Data set 3 contains the module status information transferred to the network.

Four bytes are used for each module (e.g. WS0-XTIO). These four bytes are being transferred in Big Endian format, i.e. in 16 bit word format with the first byte placed in the least significant, or rightmost byte of the integer and the second byte placed in the most significant, or leftmost byte of the integer:

Data set 3 can not be customized.

#### Meaning of the module status bits

Table 16: Meaning of the module status bits

Byte	Bit	Safety I/O modules (e.g. WS0-XTIO, WS0-XTDI)	CPU modules (e.g. WS0-CPUx)	Network modules (e.g. WS0-GETH)
0	0	Module operating state 1 = Executing 0 = Any other state	Module operating state 1 = Executing 0 = Any other state	Module operating state 1 = Executing 0 = Any other state
	1	Internal error: Internal tests failed 1 = No error 0 = Error	Internal error: Internal tests failed 1 = No error 0 = Error	Internal error: Internal tests failed 1 = No error 0 = Error
	2	External error: Summary of bit 0.5..3.7 1 = No error 0 = Error	Power supply out of range or EFI communication failure 1 = No error 0 = Error	External error: Summary of bit 0.5..3.7 1 = No error 0 = Error/inactive
	3	Reserved	Reserved	Reserved
	4	Configuration is incompatible or invalid. 1 = Configuration valid 0 = Conf. invalid or unknown	Configuration of a module within the system is incompatible or invalid. 1 = Configuration valid 0 = Conf. invalid or unknown	Configuration is incompatible or invalid. 1 = Configuration valid 0 = Conf. invalid or unknown
	5	Output power supply out of range. 1 = Power supply o.k. 0 = Power supply out of range	Power supply out of specified range 1 = Power supply o.k. 0 = Power supply out of range	Input status changed to invalid 1 = Valid network communication 0 = Invalid or no communication

Byte	Bit	Safety I/O modules (e.g. WS0-XTIO, WS0-XTDI)	CPU modules (e.g. WS0-CPUx)	Network modules (e.g. WS0-GETH)
0	6	Output fast shut off logic control time out 1 = No error 0 = Time out exceed because the control signal from CPU logic has not switched off within 1s	EFI 1 communication failure 1 = No error 0 = Error	Output status 1 = Valid network communication 0 = Invalid or no communication
	7	Reserved	EFI 2 communication failure 1 = No error 0 = Error	Reserved
1	0	Input 1-2 dual channel input evaluation error 1 = No error 0 = Error	Reserved	Reserved
	1	Input 3-4 dual channel input evaluation error 1 = No error 0 = Error		
	2	Input 5-6 dual channel input evaluation error 1 = No error 0 = Error		
	3	Input 7-8 dual channel input evaluation error 1 = No error 0 = Error		
	4	Reserved		
	5	Reserved		
	6	Reserved		
	7	Reserved		
2	0	Input 1 external test signal failure. 1 = No error 0 = Error	Reserved	Reserved
	1	Input 2 external test signal failure. 1 = No error 0 = Error		
	2	Input 3 external test signal failure. 1 = No error 0 = Error		

Byte	Bit	Safety I/O modules (e.g. WS0-XTIO, WS0-XTDI)	CPU modules (e.g. WS0-CPUx)	Network modules (e.g. WS0-GETH)
2	3	Input 4 external test signal failure. 1 = No error 0 = Error	Reserved	Reserved
	4	Input 5 external test signal failure. 1 = No error 0 = Error		
	5	Input 6 external test signal failure. 1 = No error 0 = Error		
	6	Input 7 external test signal failure. 1 = No error 0 = Error		
	7	Input 8 external test signal failure. 1 = No error 0 = Error		
3	0	Output 1 stuck-at-high error. 1 = No error 0 = Error		
	1	Output 1 stuck-at-low error. 1 = No error 0 = Error		
	2	Output 2 stuck-at-high error. 1 = No error 0 = Error		
	3	Output 2 stuck-at-low error. 1 = No error 0 = Error		
	4	Output 3 stuck-at-high error. 1 = No error 0 = Error		
	5	Output 3 stuck-at-low error. 1 = No error 0 = Error		
	6	Output 4 stuck-at-high error. 1 = No error 0 = Error		
	7	Output 4 stuck-at-low error. 1 = No error 0 = Error 0 = Error		

- Note**
- Reserved (for future use) = static 1 (no status change)
  - If no module is present, all values including the reserved values are set to logical 1.

If no module is present, all values including the reserved values are set to logical 1. In input data set 3, the status value will be "0" only when an error occurs in the corresponding module.

You will find an example process image in Section 3.5.3.

## 4.2 Data received from the network (network output data sets)

The data received from the network are organized in output data sets (max. 50 bytes). These data have been subdivided in five data blocks holding 10 bytes each for the Ethernet interface module.

Table 17:  
Output data blocks 1 to 5 for the Ethernet interface module

	Output data block 1 size	Output data block 2 size	Output data block 3 size	Output data block 4 size	Output data block 5 size
<b>WS0-GETH</b>	10 bytes				

The contents of the output data blocks can be used within the Setting and Monitoring Tool logic editor and can also be made available to another network via a second network module in the MELSEC-WS safety controller.

- Note**
- In order to make the data from the network available in the logic editor or as input to another network, you will have to define a tag name for each bit that shall be used.
  - Bits without a tag name will not be available in the logic editor nor for routing via another network module. For detailed information on how to define tag names for the data received please see Section 4.5.
  - The status of the communication to and from the network can be monitored in the logic editor using the module input status bit for data from the network and the module output status bit for data to the network. When the Ethernet interface module detects an invalid communication, the contents of the data sets will be set to zero (logical 0) and the corresponding module status bit will also be set to zero (logical 0).
  - In case the communication is dropped, the data of the output data sets will be set to zero (logical 0) and the module input status bit will also be set to zero (logical 0).

These modules may only be used by qualified safety personnel and only on the machine where they have been installed and initialized by qualified safety personnel in accordance with the manual.




---

### **Do not use the same output data set number for two different interface connections or TCP/IP sockets!**

The output data set can be written to the Ethernet interface modules in parallel by all communication interfaces or TCP/IP sockets, if they use the same output data set number. In that case the last message overrides data received earlier.

---

### 4.3 Default settings for the input data set 1

In the delivery status, the input data set 1 of the Ethernet interface module is pre-configured.

The following table gives an overview which bytes are assigned to the default configuration and how the data are modularized.

Table 18:  
Default configuration for  
the input data set 1  
transmitted into the  
network

Byte	Default assignment	Input data set
0	Logic result 0	#1 (50 bytes)
1	Logic result 1	
2	Logic result 2	
3	Logic result 3	
4	Module 1 input	
5	Module 2 input	
6	Module 3 input	
7	Module 4 input	
8	Module 5 input	
9	Module 6 input	
10	Module 7 input	
11	Module 8 input	
12	Module 9 input	
13	Module 10 input	
14	Module 11 input	
15	Module 12 input	
16	Module 1 output	
17	Module 2 output	
18	Module 3 output	
19	Module 4 output	
20	Module 5 output	
21	Module 6 output	
22	Module 7 output	
23	Module 8 output	
24	Module 9 output	
25	Module 10 output	
26	Module 11 output	
27	Module 12 output	
28 to 35	Not assigned	
36 to 47	Not assigned	
48 to 49	Not assigned	

The default byte assignment can be freely customized as will be described in the following section.

## 4.4 Customizing the input data set 1 (MELSEC-WS to Ethernet)

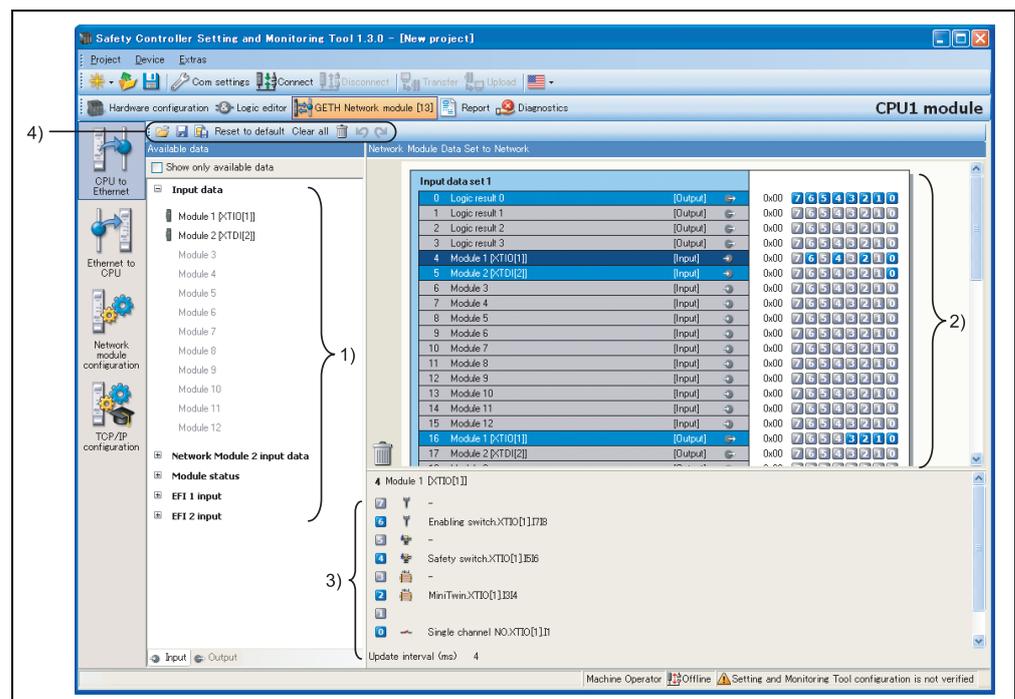
This section outlines briefly how you can customize the input data set 1 that the Ethernet interface module transmits to the network.

In the delivery status, the data routing configuration of the Ethernet interface module is shown in the network module configuration dialog.

- ⇒ Click on the **GETH Network module [13]** button (**Network Modules** button when two network modules are used) above the main window and select the desired Ethernet interface module or double click the desired Ethernet interface module in the hardware configuration to open the network module configuration dialog.
- ⇒ Click on the **MELSEC-WS to Ethernet** tab on the left hand menu to display the network module configuration dialog.

The default setting is as follows:

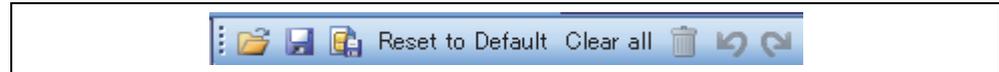
Figure 19:  
MELSEC-WS to  
Ethernet dialog of the  
WS0-GETH



Basically this dialog is divided into three areas: **Available data (1)**, **Network Module Data (2)** and **Tag names (3)**. The upper left corner of the dialog holds the toolbar (4).

#### 4.4.1 The toolbar

Figure 20:  
Toolbar for the routing  
configuration



The toolbar contains buttons for the following actions (from left to right):

- The **Load user configuration** and **Save user configuration** buttons allow you to load and/or save a configuration including the used tag names in XML format. If you load a configuration, all previously made changes that have not been saved will be lost. You can not undo this action.
- With the **Import** and **Export** buttons you can import and export the used tag names as a CSV (comma separated values) file. This allows you to import and use the assigned tag names in a programmable controller program. To edit the CSV file, use a text editor.

**Note** When importing the exported network module data after editing the data in a text editor such as Notepad, select **CSV Files - Shift-JIS encoded** for **Files of type** in the **Open** dialog box if the data is saved in ANSI format. If not, characters in the data may get garbled.

The **Import** button is only available for the Ethernet to MELSEC-WS routing configuration.

- **Reset to default** restores the default routing configuration. You will be prompted for confirmation. If you click **Yes**, all previously made changes that have not been saved will be lost. You can not undo this action.
- **Clear all** clears the configuration, i.e. deletes all assigned bytes in the **Network Module Data** area. You will be prompted for confirmation.
- **Delete routing** deletes the selected byte from the **Network Module Data** area.
- The **Undo** and **Redo** buttons allow you to undo or redo changes you made to your configuration.

#### 4.4.2 Available data area

This area offers all sources from which data may be routed into the network. It is divided in two views holding the available **Input types** and **Output** data. You can switch between these views using the file cards at the bottom.

- The **Input types** view contains the input values for the connected MELSEC-WS modules and EFI devices. If your MELSEC-WS safety controller contains a second network module, the input data of this network module (i.e. data received from the network the second network module is connected to) will be available here as well.
- The **Output** view offers the output values for the connected MELSEC-WS modules and EFI devices as well as the **Logic results** from the logic editor.

All sources supported by the current configuration are displayed in black:

- connected MELSEC-WS modules
- connected EFI devices
- configured logic results\*<sup>1</sup>
- input data available from another network module in the system

\*<sup>1</sup> In the default configuration, only the first logic result byte (Logic Result 0) is active and available. You can activate more logic result output bits in the logic editor.

Sources currently not configured will be displayed in grey. Activating the **Show only available data** checkbox in the upper left corner hides the unused sources from the view.

Sources that offer “live” data are marked with a little icon next to the text.

#### How to add a data byte to the routing table:

⇒ Drag and drop an element (i.e. byte) from the **Available data** area to a free slot in the **Network Module Data** area. If the desired position is not free, you will have to clear it first by deleting or moving the byte currently assigned to it.

**Note** It is possible to use the same byte several times in the routing table.

#### 4.4.3 Network Module Data area

This area contains the routing table. It shows the current content of the Ethernet interface module's input data modules. Bytes and bits highlighted blue will hold "live" data from the system since the hardware configuration does support the source. Bytes highlighted grey actually do not have data associated with them since the hardware configuration does not support the sources.

##### How to delete a data byte from the routing table:

⇒ Drag and drop the byte you want to delete to the trashcan icon in the bottom left corner of the **Network Module Data** area.

**Or:**

⇒ Select the byte you want to delete by clicking it with the left mouse button. Then, click on the **Delete routing** button in the toolbar.

**Or:**

⇒ Call up the context menu by clicking the respective byte with the right mouse button. In the context menu, select the **Delete routing** command.

##### How to move a data byte to another place in the routing table:

⇒ Drag and drop the byte you want to move to the desired position. If the desired position is not free, you will have to clear it first by deleting or moving the byte currently assigned to it.

#### 4.4.4 Tag names area

This area shows the tag names associated with each bit of the byte currently selected in the **Available data** or the **Network Module Data** area. You can enter these tag names in the tag name editor.

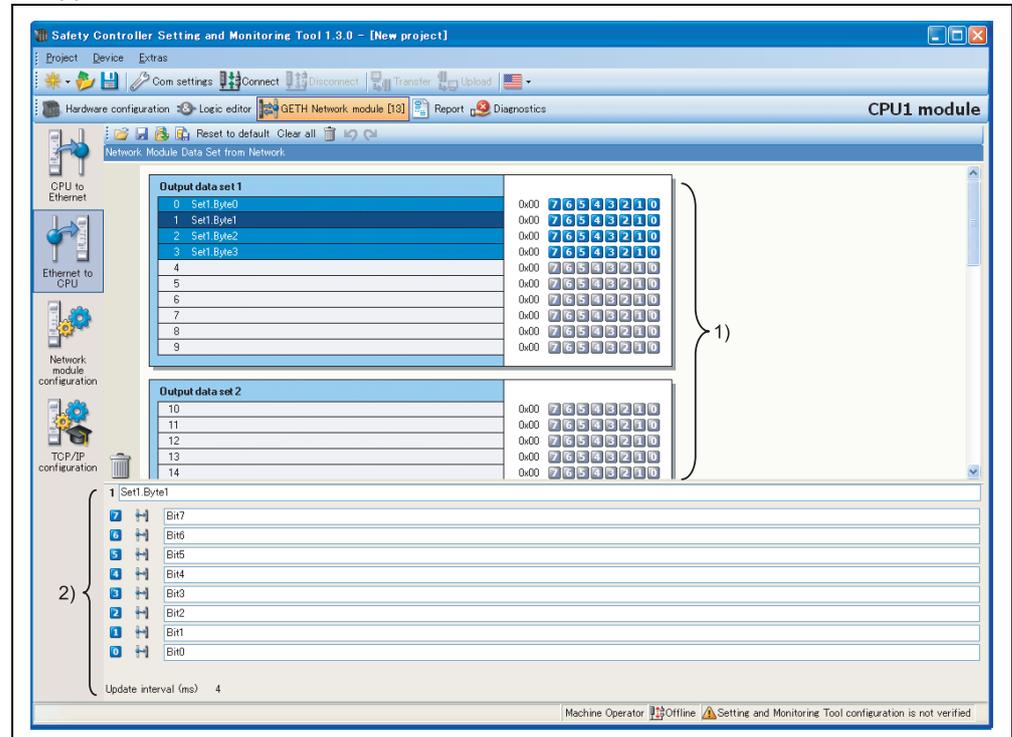
In the **Tag names** area of the **CPU to Ethernet** configuration dialog, you can only edit tag names for unassigned data sets by activating the **Use direct editing** checkbox (with CPU module firmware version V2.00.0 or higher).

## 4.5 Tag names for incoming data (Ethernet to MELSEC-WS)

In order to enable incoming data bits:

⇒ Click on **Ethernet to MELSEC-WS** on the left hand menu. The following dialog appears:

Figure 21:  
Ethernet to MELSEC-WS dialog of the WS0-GETH



Basically this dialog is divided into two areas: **Network Module Data (1)** and **Tag names (2)**:

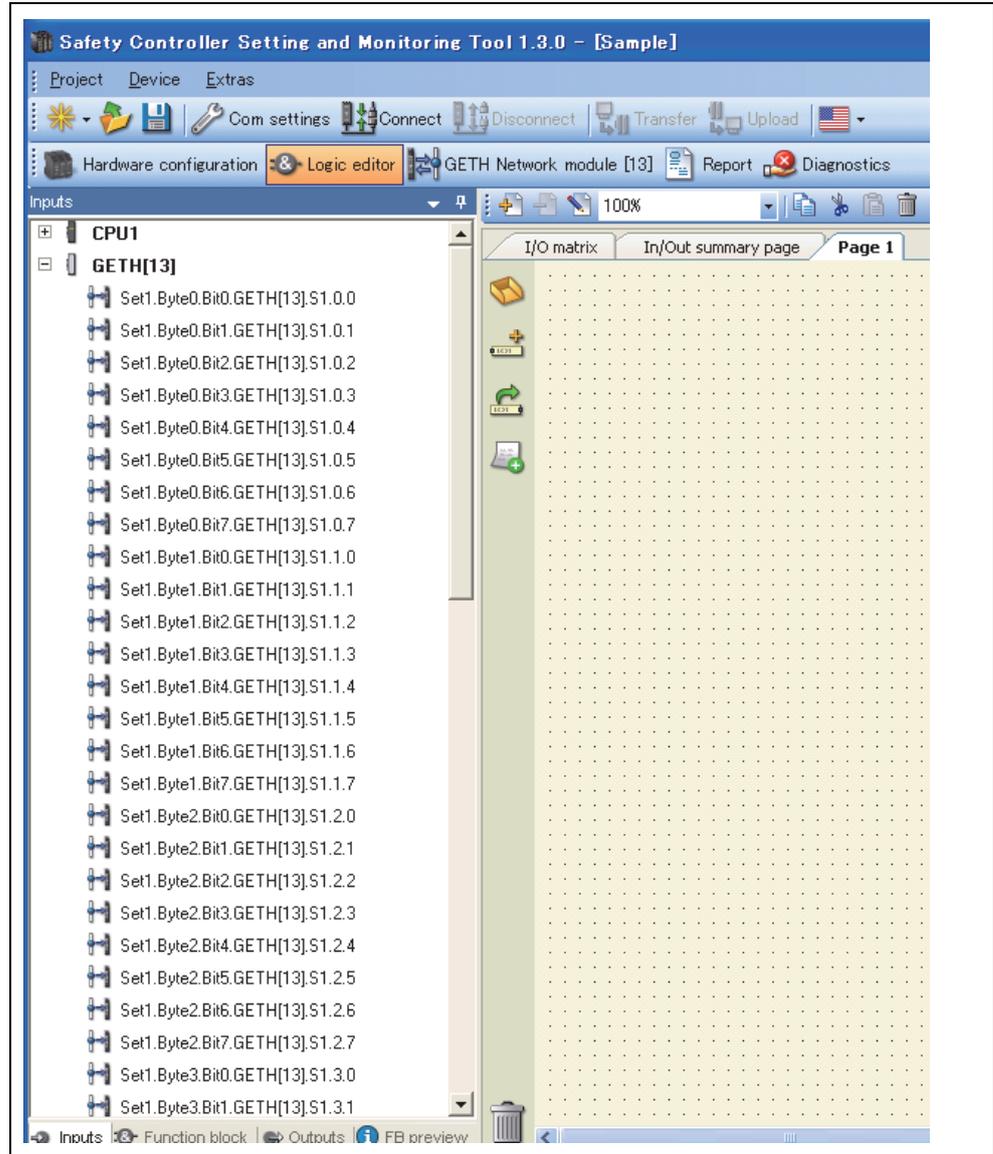
The **Network Module Data** area shows the current configuration of the output modules.

The **Tag names** area shows the tag names associated to the byte selected in the **Network Module Data** area.

- ⇒ Select a byte in the **Network Module Data** area.
- ⇒ For each bit of the selected byte that you wish to use, enter a tag name in the **Tag names** area.

Each bit you enter a tag name for here will be available within the logic editor or for the process image of a second network module:

Figure 22:  
Tag names of incoming bits of the WS0-GETH in the logic editor dialog



#### 4.5.1 Saving and loading a configuration

Using the buttons **Load user configuration** and **Save user configuration** you can save and load your configuration in XML format. If you load a configuration, all previously made changes that have not been saved will be lost. You can not undo this action.

#### 4.5.2 Importing and exporting a configuration

With the **Import** and **Export** buttons you can import and export a configuration including the tag names used as a CSV (comma separated values) file. This allows you to import and use tag names you have assigned in the MELSEC-WS safety controller project in another programmable controller program and vice versa. To edit the CSV file, use a text editor.

If you import a configuration, all previously made changes that have not been saved will be lost. You can not undo this action.

**Note** The **Import** button is only available for the Ethernet to MELSEC-WS routing configuration.

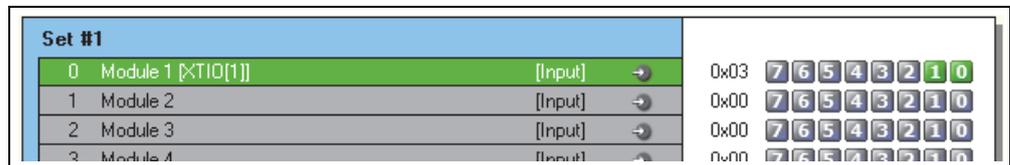
### 4.6 Monitoring the data set online

When the MELSEC-WS safety controller is online and running, you can monitor the data set online in the network module configuration dialog.

- ⇒ Click on the **GETH Network module [13]** button (**Network Modules** button when two network modules are used) above the main window and select the desired Ethernet interface module or double click the desired Ethernet interface module in the hardware configuration to open the network module configuration dialog.
- ⇒ Click on the **MELSEC-WS to Ethernet** or the **Ethernet to MELSEC-WS** tab on the left hand menu to display the routing view for the input or output data you want to monitor.

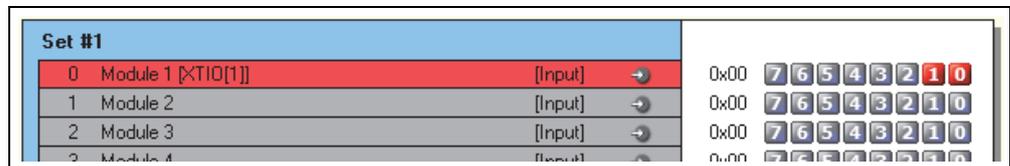
For both directions, **MELSEC-WS to Ethernet** as well as **Ethernet to MELSEC-WS**, inactive bits are displayed grey while active bits are highlighted green:

Figure 23:  
Active and inactive bits in the online process image



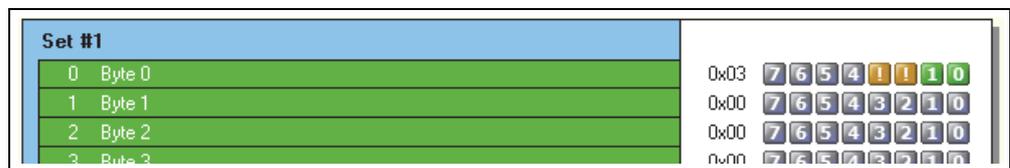
In the **MELSEC-WS to Ethernet** view, bits that are inactive due to an error are displayed red. This could be the case e.g. for the outputs of a WS0-XTIO module if the power supply of this module is faulty:

Figure 24:  
Inactive Network input bits as a result of an error



In the **Ethernet to MELSEC-WS** view, bits that have no tag name assigned (so that they can not be used in the logic editor) but which are included in the process image that the Ethernet interface module receives from the programmable controller, are highlighted yellow:

Figure 25:  
Network output bits with no tag names assigned in the online process image



**Note** The Ethernet interface module always reflects the actual physical status of the inputs and outputs on the connected modules and devices. This means that even when Force mode is active and inputs that are physically **Inactive (Low)** are forced **Active (High)** (or vice versa) the actual physical status of these inputs will be transmitted to the programmable controller instead of the (virtual) forced status. If, however, as a result of forcing of one or several inputs, one or several outputs change their status, the changed status of these outputs will be transmitted to the programmable controller since the actual physical status of the outputs on the devices has changed.

## 5. Troubleshooting

For information how to perform diagnostics on the MELSEC-WS safety controller please refer to the Safety Controller Setting and Monitoring Tool Operating Manual.

**Note** Check the LEDs of the module during troubleshooting.

Table 19: Troubleshooting for the WS0-GETH

Error	Cause	Possible remedy
The Setting and Monitoring Tool does not connect to the Ethernet interface module.	WS0-GETH has no power supply. WS0-GETH is not in the same physical network as the PC. The PC is configured to another subnet mask in the TCP/IP settings. WS0-GETH has already been configured once and has a fixed set IP address or an IP address assigned by a DHCP server that is not recognized.	Establish the power supply. Check the Ethernet wiring and network settings on the PC and correct if necessary. Set the subnet mask on the PC to 255.255.0.0 (factory setting of the WS0-GETH). Check the communication settings in the Setting and Monitoring Tool.
WS0-GETH does not supply any data. LED PWR ● Green LED LINK/ACT ●/☀ Green LED MS ☀ Red/Green	WS0-GETH is configured for data transfer to programmable controller, but Ethernet communication is not yet established or faulty. Duplicate IP address detected. Another device on the network has the same IP address.	Minimum one Ethernet connection needs to be established. Set up Ethernet connection on programmable controller side, check Ethernet cabling, check Ethernet connection settings on programmable controller and in the Setting and Monitoring Tool. If no Ethernet communication is required, disable the Ethernet connections/PLC interfaces on the WS0-GETH. Adjust IP address and power cycle device.
WS0-GETH does not supply any data. LED PWR ● Green LED LINK/ACT ●/☀ Green LED MS ☀ Red (1 Hz)	Configuration required. Configuration download is not completed.	Configure the WS0-GETH and download the configuration to the safety controller. Wait until the configuration download has been completed.
WS0-GETH does not supply any data. LED PWR ● Green LED LINK/ACT ●/☀ Green LED MS ● Green	No data set is activated. No Ethernet communication interface is enabled.	Activate at least one data set for communication in auto update mode.
WS0-GETH does not supply any data. LED PWR ● Green LED LINK/ACT ●/☀ Green LED MS ☀ Green (1Hz)	WS0-GETH is in Idle mode.	CPU/application is stopped. Start CPU (change into Run mode).

Error	Cause	Possible remedy
WS0-GETH functioned correctly after configuration, but suddenly no longer supplies data. LED PWR ● Green LED LINK/ACT ●/☀ Green LED MS ☀ Red/Green	WS0-GETH is operated in slave mode, the IP address is assigned from a DHCP server. After the WS0-GETH or the DHCP server has been restarted, a different IP address that is unknown to the programmable controller has been assigned to the WS0-GETH.	Either assign a fixed IP address to the WS0-GETH, or reserve a fixed IP address for the WS0-GETH in the DHCP server (manual assignment by means of the MAC address of the WS0-GETH).
WS0-GETH/safety controller is in Critical fault mode. LED PWR ● Green LED LINK/ACT ☀ Green LED MS ● Red	WS0-GETH is not plugged properly into the other MELSEC-WS module. Module connection plug is soiled or damaged. Other MELSEC-WS module has internal critical error. Ethernet communication with the WS0-GETH is under excessive load.	Plug the WS0-GETH in correctly. Clean the connecting socket/plug. Repower the safety controller. Check the error MELSEC-WS module and remove the error cause. Check the Ethernet line status. If a load is excessively placed to the line, reduce the load using a device such as a router.
WS0-GETH is in Critical fault mode. LED PWR ● Green LED LINK/ACT ●/☀ Green LED MS ☀ Red (2 Hz)	WS0-GETH internal device error	Switch off the power supply of the safety controller and switch it on again. Check if the WS0-GETH is mounted in correct position. Check the diagnostics messages with the Setting and Monitoring Tool. If the error remains, replace the Ethernet interface module.

## 6. Technical specifications

### 6.1 Technical specifications of Ethernet interface module

Table 20:  
Technical specifications  
of Ethernet interface  
module

Interface	
<b>Fieldbus</b>	Ethernet TCP/IP
<b>Integrated switch</b>	3-port layer-2 managed switch with Auto-MDI-X for automatic detection of crossed Ethernet cable
<b>Connection technique</b>	RJ45 socket
<b>Transfer rate</b>	10 Mbps (10Base-T) or 100 Mbps (100Base-TX), autosensing
<b>Update rate (heartbeat rate)</b>	Configurable from 40 to 65535 ms
<b>Change of state (COS) update rate</b>	10 ms
<b>Addressing factory setting</b>	IP address: 192.168.250.250 Subnet mask: 255.255.0.0 Default gateway: 0.0.0.0
<b>MAC address</b>	Printed on type label, example: 00:06:77:02:00:A7

### 6.2 Technical specifications, supply circuit

Table 21:  
Technical  
specifications, supply  
circuit

Item	Specifications
<b>Supply voltage</b>	24 V DC (16.8 to 30 V DC)
<b>Power consumption</b>	Max. 2.4 W

### 6.3 General technical specifications

Table 22:  
General technical  
specifications

Terminals	
<b>Fieldbus</b>	Ethernet (TCP/IP)
<b>FLEXBUS+</b>	10-pin connector for internal safety bus (plug)

Climatic conditions	
<b>Ambient operating temperature <math>T_A</math></b>	-25 °C to +55 °C
<b>Storage temperature</b>	-25 °C to +70 °C
<b>Relative humidity</b>	10 % to 95 %, non-condensing
<b>Climatic conditions</b>	In accordance with EN61131-2

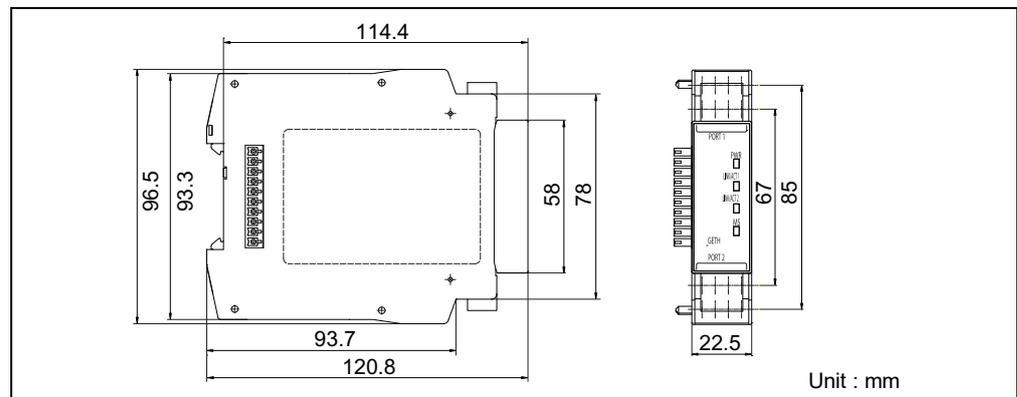
<b>Mechanical strength</b>	
<b>Vibration resistance, shock resistance</b>	Tested in accordance with IEC 61161-2.

<b>Electrical safety (see WS0-CPUx)</b>	
<b>Protection class</b>	III
<b>Electromagnetic compatibility</b>	IEC 61000-6-2/EN 55011 Class A

<b>Mechanical and assembly</b>	
<b>Housing material</b>	Polycarbonate
<b>Housing type</b>	Device for control cabinet installation
<b>Housing enclosure rating/terminals</b>	IP 40/IP 20
<b>Housing color</b>	Light grey
<b>Weight</b>	125 g
<b>Data interface</b>	FLEXBUS+ backplane bus
<b>Ethernet interface</b>	One connector on the top and one connector on the bottom
<b>Mounting rail</b>	Mounting rail according to EN 60715

### 6.4 Dimensional drawings

Figure 26:  
Dimensional drawings  
of WS0-GETH (mm)



### 6.5 Module information

Table 23:  
Module information

Type	Module name	Abbreviation
<b>WS0-GETH00200</b>	MELSEC-WS safety controller Ethernet interface module	WS0-GETH

## 7. Connecting to MELSEC programmable controllers

This chapter describes how to exchange non-safety related data (input/output data) between the MELSEC-WS safety controller and MELSEC programmable controllers via TCP/IP.

In this or later chapters, each Ethernet interface module in a MELSEC-WS safety controller system and MELSEC programmable controller system is referred to as follows:

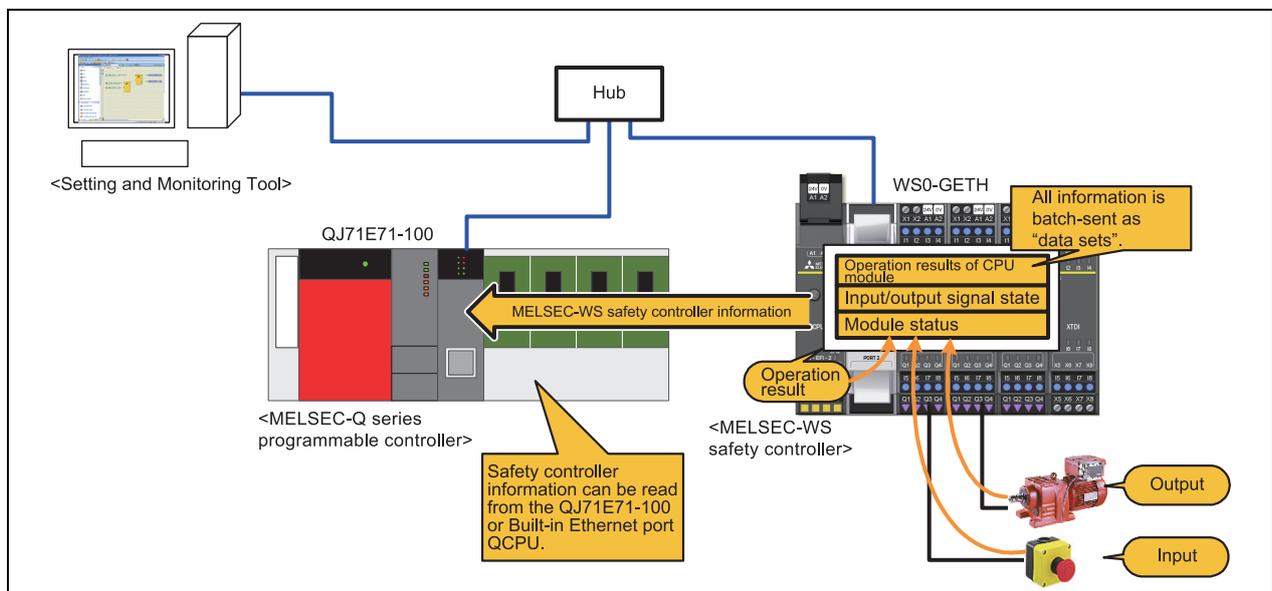
- Ethernet interface module in a MELSEC-WS safety controller system → WS0-GETH
- MELSEC QJ71E71-100 Ethernet interface module in a MELSEC programmable controller system → QJ71E71-100

### 7.1 Features achieved by the connection

**MELSEC programmable controllers can acquire MELSEC-WS safety controller information over Ethernet.**

The QJ71E71-100 connected to a MELSEC programmable controller or the Built-in Ethernet port QCPU can read MELSEC-WS safety controller information (e.g. operation results of the CPU module, safety input/output data) over Ethernet. Two communication modes are available: polling mode and auto update mode.

Figure 27: Acquiring MELSEC-WS safety controller information

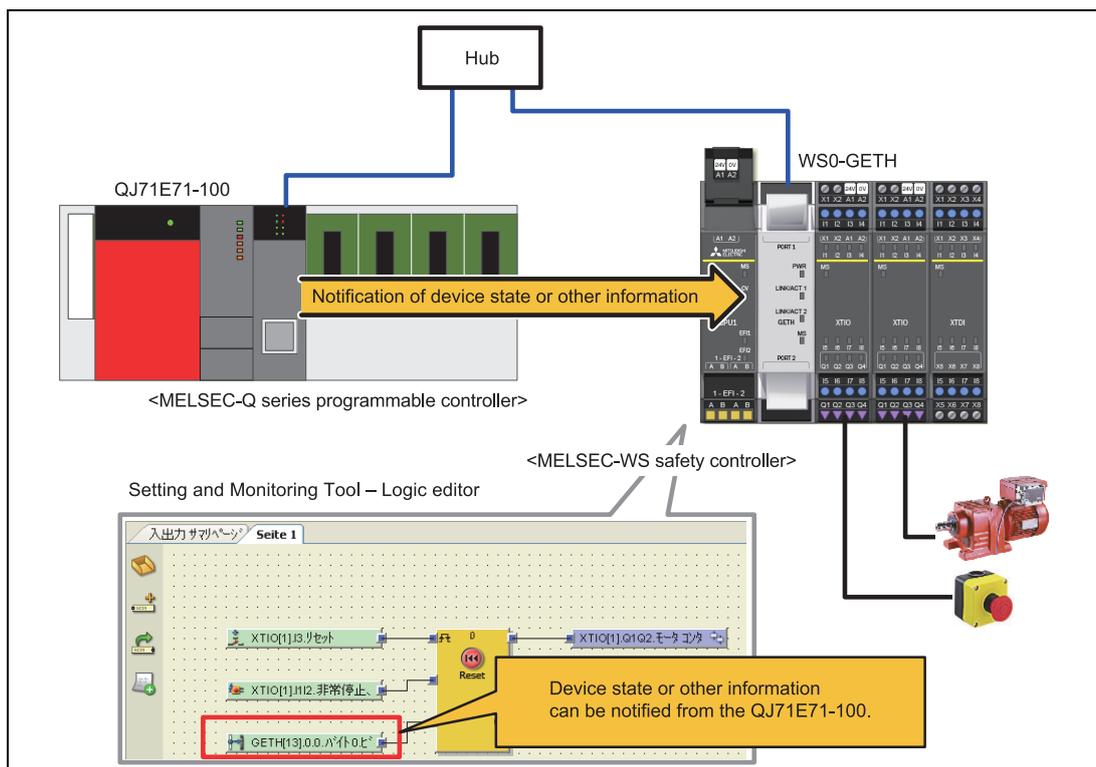


**MELSEC programmable controllers can send data to the MELSEC-WS safety controller over Ethernet.**

The QJ71E71-100 connected to a MELSEC programmable controller or the Built-in Ethernet port QCPU can send data (e.g. device state or other information of external devices) to the MELSEC-WS safety controller over Ethernet.

The send data can be used as input data for a function block in the logic editor of Setting and Monitoring Tool.

Figure 28: Sending data to the MELSEC-WS safety controller



## 7.2 Applicable system

For configurations of MELSEC programmable controller systems, refer to the following.

- QnUCPU User's Manual (Communication via Built-in Ethernet Port)
- Q Corresponding Ethernet Interface Module User's Manual (Basic)

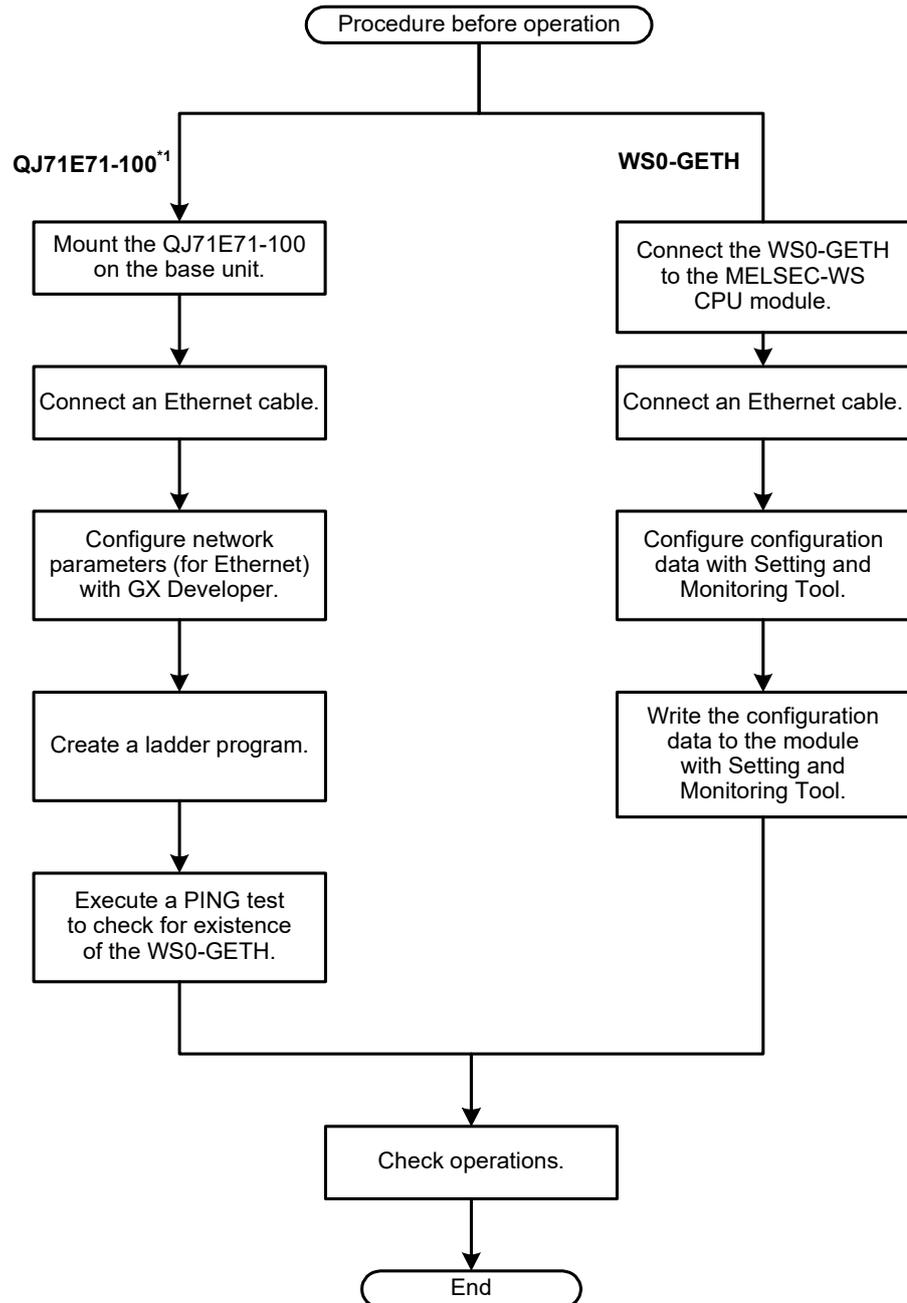
## 8. Procedures before operation

(1) Procedures before operation

(a) Communications with the QJ71E71-100

The following is a procedure to establish communications between the WS0-GETH and the QJ71E71-100.

Figure 29:  
Procedure before operation  
(communications with the QJ71E71-100)

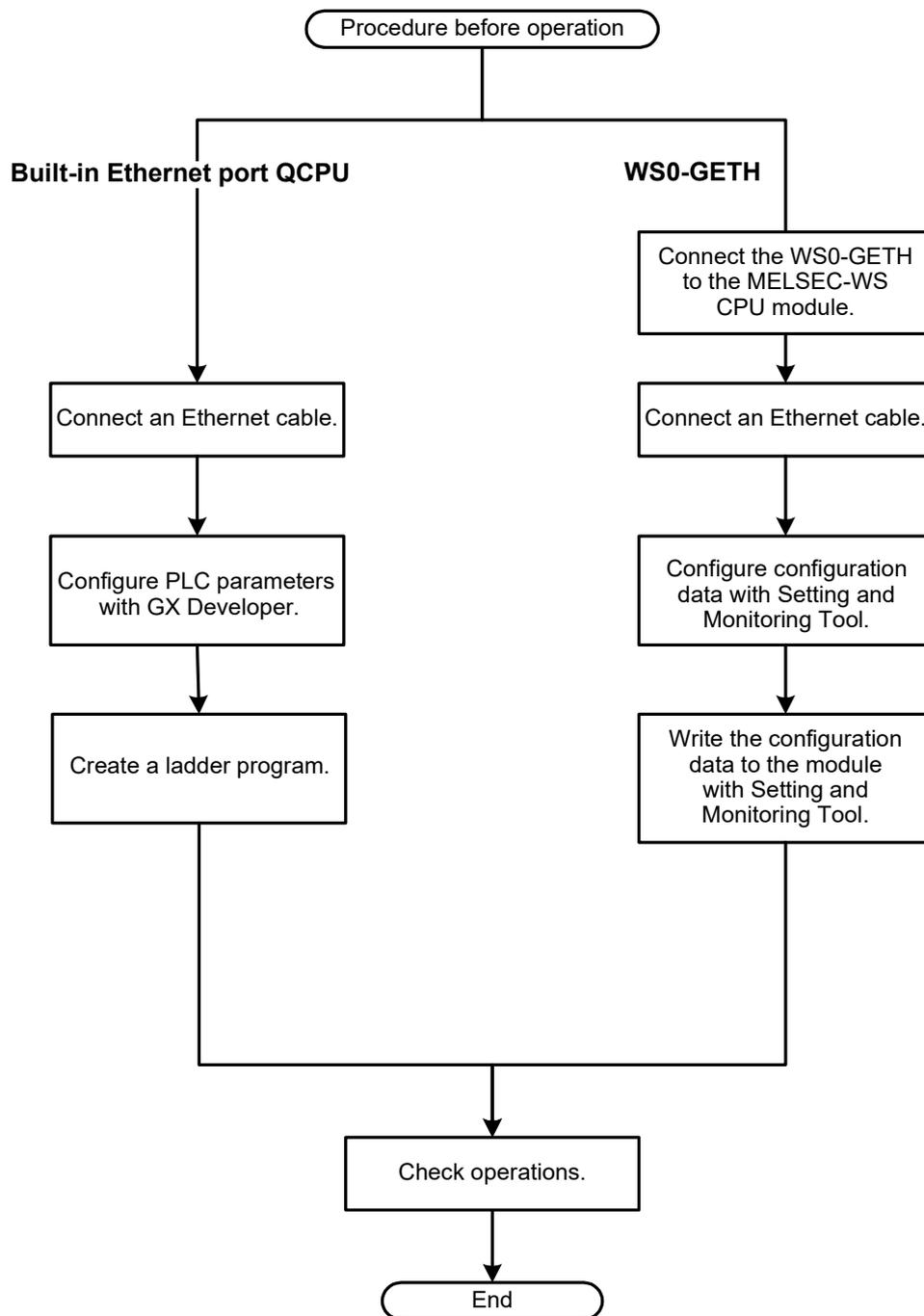


\*1 For details, refer to the Q Corresponding Ethernet Interface Module User's Manual (Basic).

(b) Communications with the Built-in Ethernet port QCPU

The following is a procedure to establish communications between the WS0-GETH and the Built-in Ethernet port QCPU.

Figure 30:  
Procedure before  
operation  
(communications with  
the Built-in Ethernet  
port QCPU)



## (2) Communication modes

This part describes communication modes used between the WS0-GETH and the Qj71E71-100 or Built-in Ethernet port QCPU.

Two communication modes are available: polling mode and auto update mode. For system configuration, refer to the section for the communication-target module.

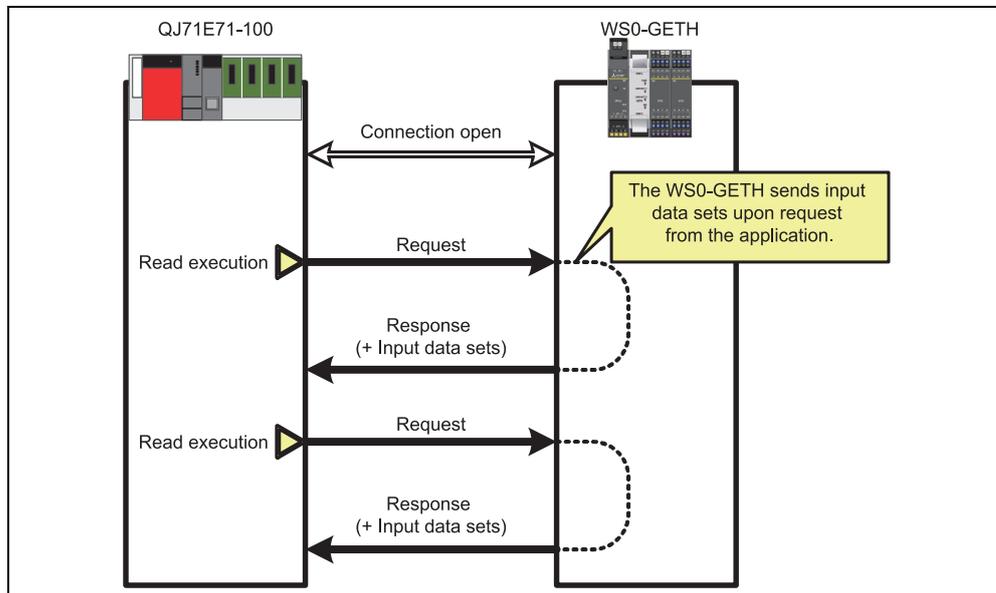
For the acquisition of sample programs provided in this manual, please contact your local Mitsubishi representative.

**Polling mode (Example: QJ71E71-100)**

Polling mode (input data set reading) (command: 00F1/001F)

The QJ71E71-100 sends the Input data set request message command to the MELSEC-WS safety controller to read input data sets (MELSEC-WS safety controller information) from the WS0-GETH.

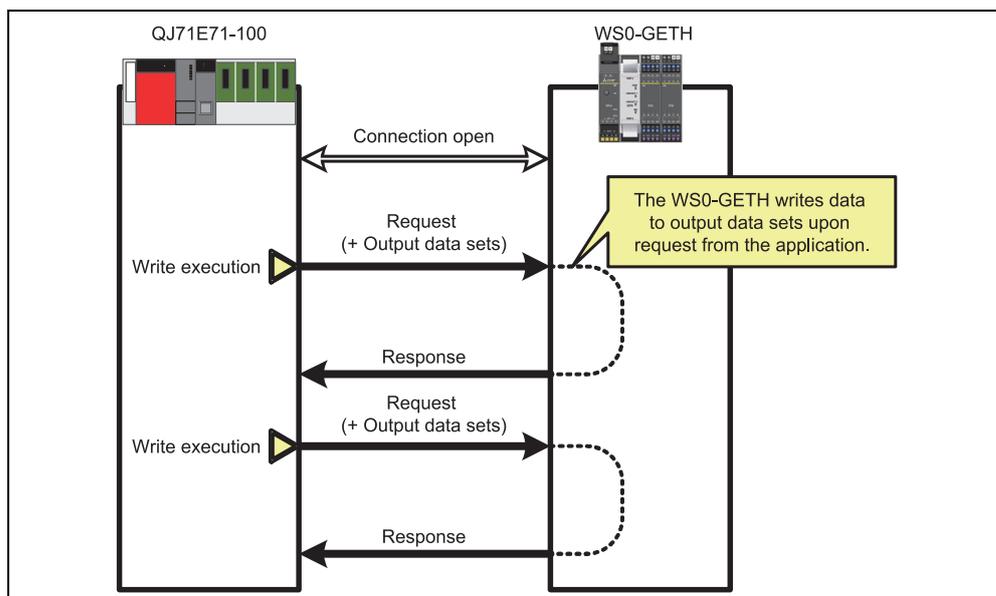
Figure 31:  
Polling mode (input data set reading)



Polling mode (output data set writing) (command: 00F2/002F)

The QJ71E71-100 sends the Write output data set settings command to the MELSEC-WS safety controller to write data to the output data sets in the WS0-GETH.

Figure 32:  
Polling mode (output data set writing)



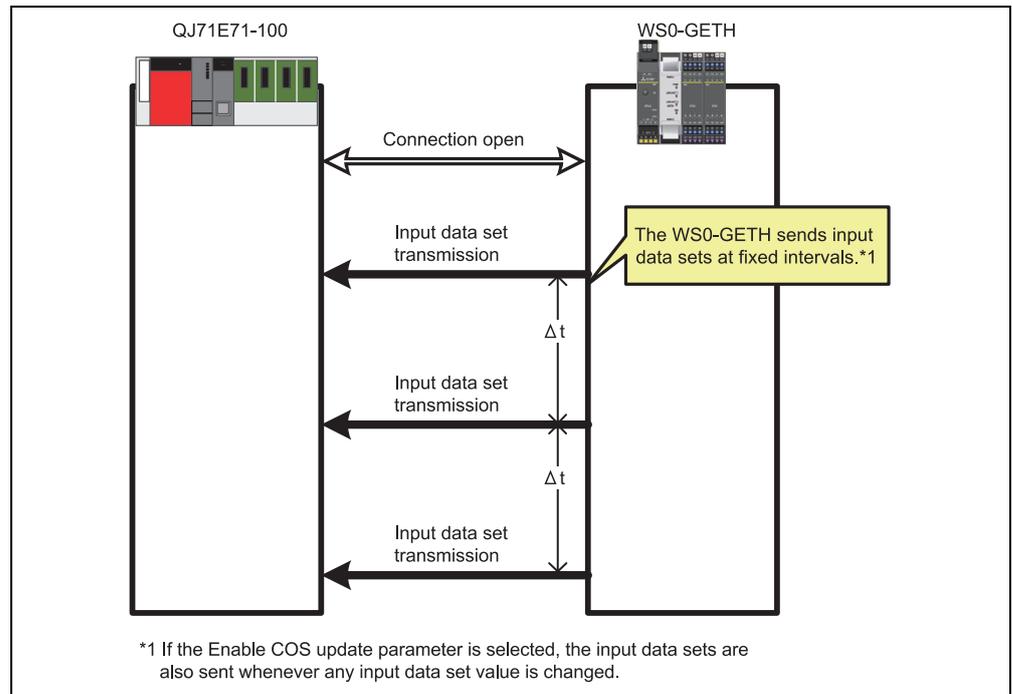
**Auto update mode (Example: QJ71E71-100)**

Input data set auto update (command: 002E)

The WS0-GETH automatically sends input data sets (MELSEC-WS safety controller information) to the QJ71E71-100 at the following timing.

- At fixed intervals
- Upon change of any input data set value

Figure 33:  
Auto update mode



### 8.1 Wiring

This section describes the wiring of the WS0-GETH.

For the wiring of the communication-target module, refer to the following.

- Q Corresponding Ethernet Interface Module User's Manual (Basic)
- QnUCPU User's Manual (Communication via Built-in Ethernet Port)

(1) Cables and connectors

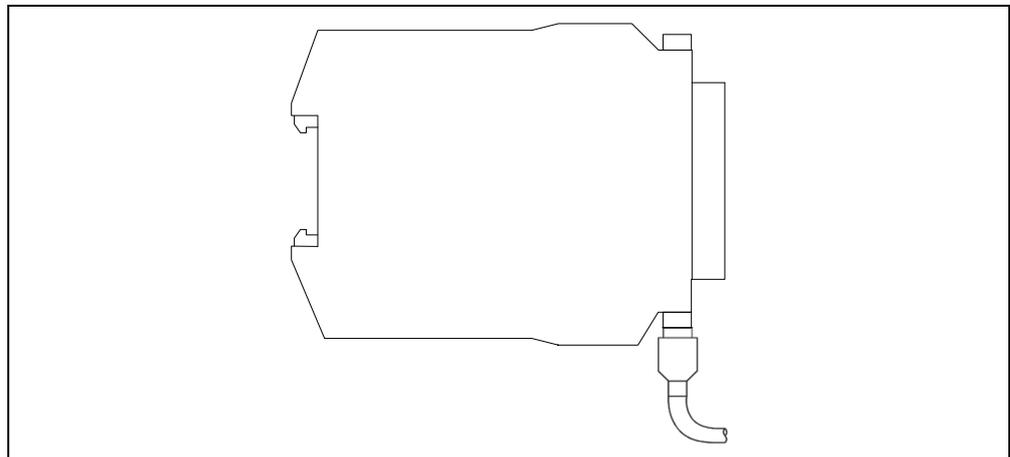
Prepare the cable shown in the following table.

Table 24:  
Ethernet cable

Part	Name	Remarks
Cable	Shielded twisted pair cable (STP) (with RJ45 jack)	Straight cable of Category 5 or higher

The following figure shows connection of the cable and the connector.

Figure 34:  
Cable connection diagram



(2) Network topologies

Two network topologies using Ethernet cables are available: line topology and star topology (using a hub).

Figure 35:  
Wiring example: line topology

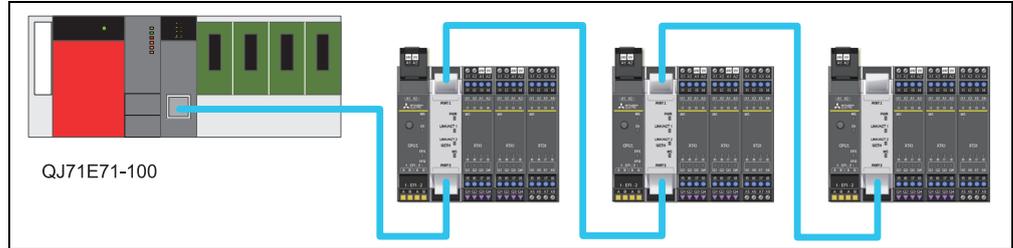


Figure 36:  
Wiring example: star topology

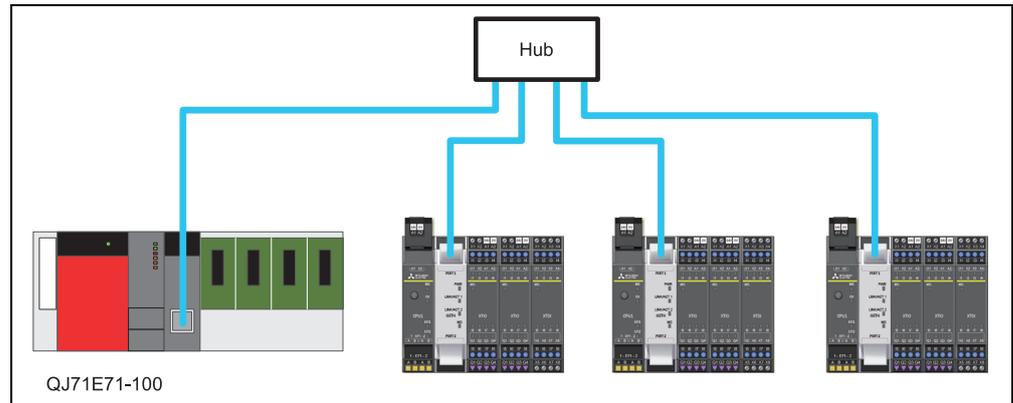
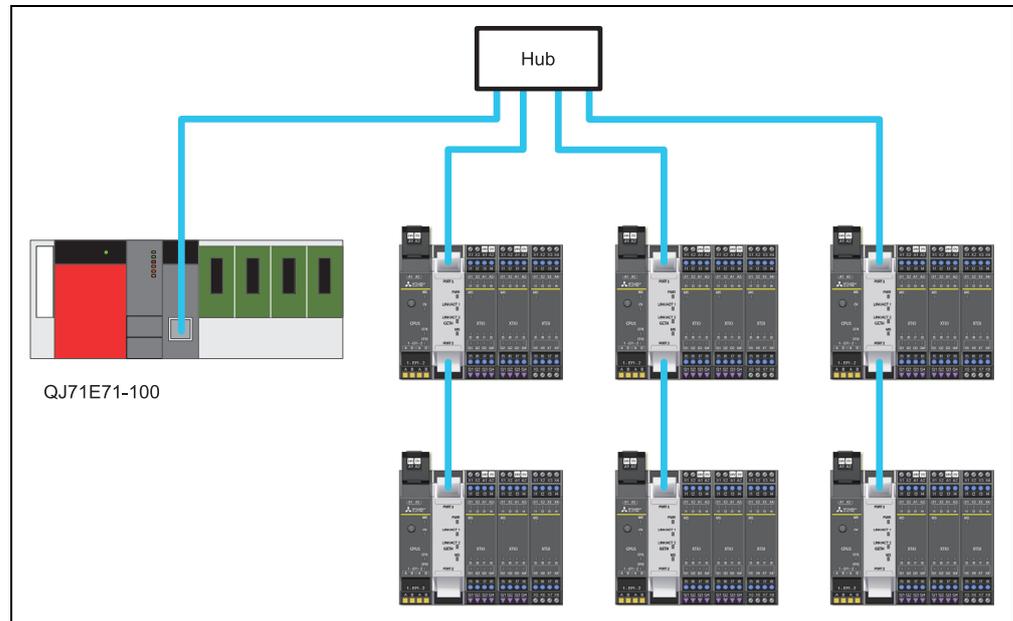


Figure 37:  
Wiring example: line topology and star topology



### 8.1.1 Wiring precautions

Observe the following wiring precautions.

- (1) Use either port 1 or port 2 for star connection.
- (2) Use Ethernet cables specified in Section 8.1 (1).
- (3) The Ethernet cable has restrictions on the bend radius.  
The maximum bend radius is four times of the cable outside diameter.
- (4) Place the cables in a duct or clamp them.  
If not, dangling cable may swing or inadvertently be pulled, resulting in damage to the module or cables or malfunction due to poor contact.
- (5) When laying the cables, do not touch the contact surfaces of the connector of the cable and the connector of the module nor let dirt or dust collect on the contact surfaces.  
If oil from the hands, dirt, or dust adheres to the cores, the transmission loss may increase, resulting in communication failure.
- (6) When connecting/disconnecting the cable to/from the module, hold the connector part.
- (7) Connect the connectors of the cable and the module until they click.

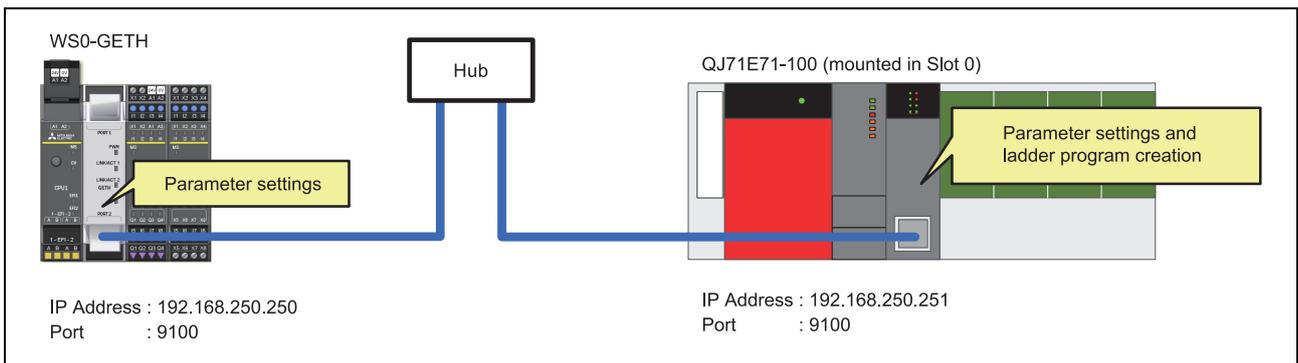
## 8.2 Communications with the QJ71E71-100 (QCPU)

This section describes parameter configuration examples and sample programs for communications between the WS0-GETH and the QJ71E71-100 connected to the QCPU.

### 8.2.1 System configuration

The following is the system configuration used for sample programs.

Figure 38: System configuration example



### 8.2.2 Polling mode (input data set reading)

This section describes an example of communications in polling mode.

The QCPU reads MELSEC-WS safety controller information (e.g. logic result, CRC, and each module status) from the safety controller through communications between the WS0-GETH and the QJ71E71-100.

#### Read-target data

- Logic result 0 (input data set 1)
- CRC (input data set 2)
- Each module status (input data set 3)

#### (1) Parameter configuration

##### (a) Configuration required for the WS0-GETH

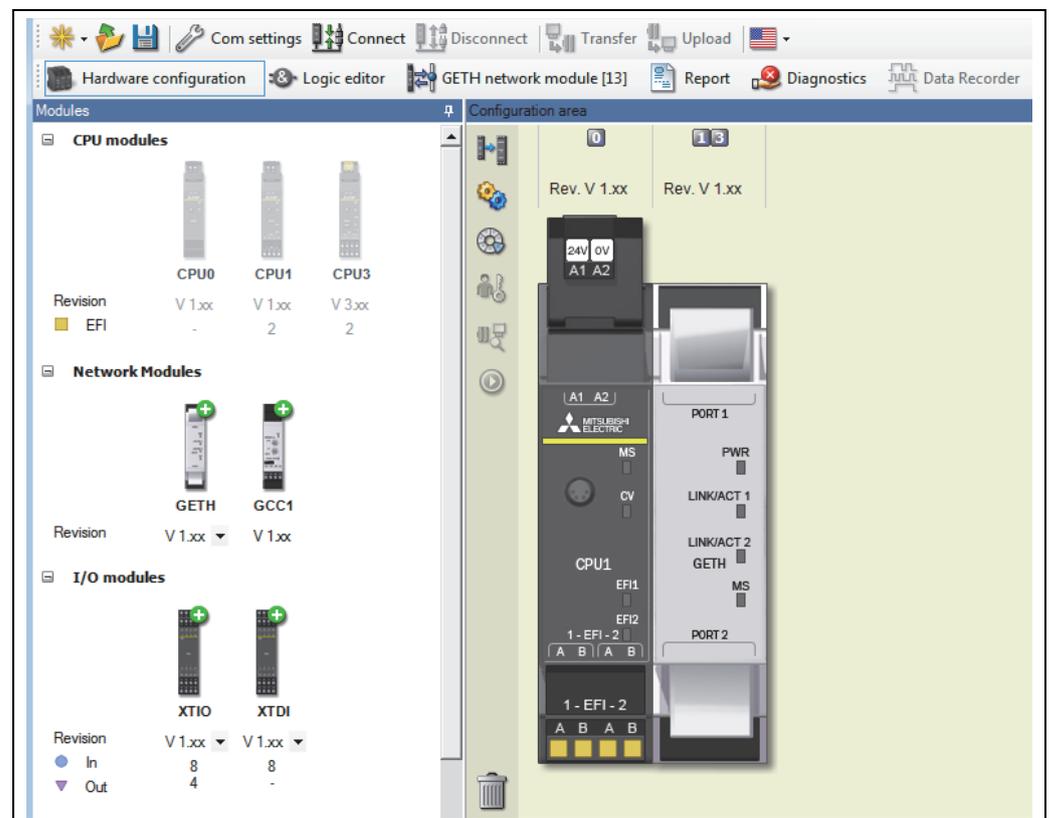
Configuration examples of network parameters required for communications in polling mode are shown below. Configure these parameters using Setting and Monitoring Tool.

The WS0-GETH requires no program for communications.

##### 1) Creating a hardware configuration [for reading/writing]

Read the hardware configuration data from the actual MELSEC-WS safety controller to Setting and Monitoring Tool or place modules in Setting and Monitoring Tool according to the actual configuration.

Figure 39:  
Hardware configuration dialog



2) Assigning an IP address [for reading/writing]

Assign an IP address of the WS0-GETH in the **Network module configuration dialog**.

Figure 40:  
Network module  
configuration dialog

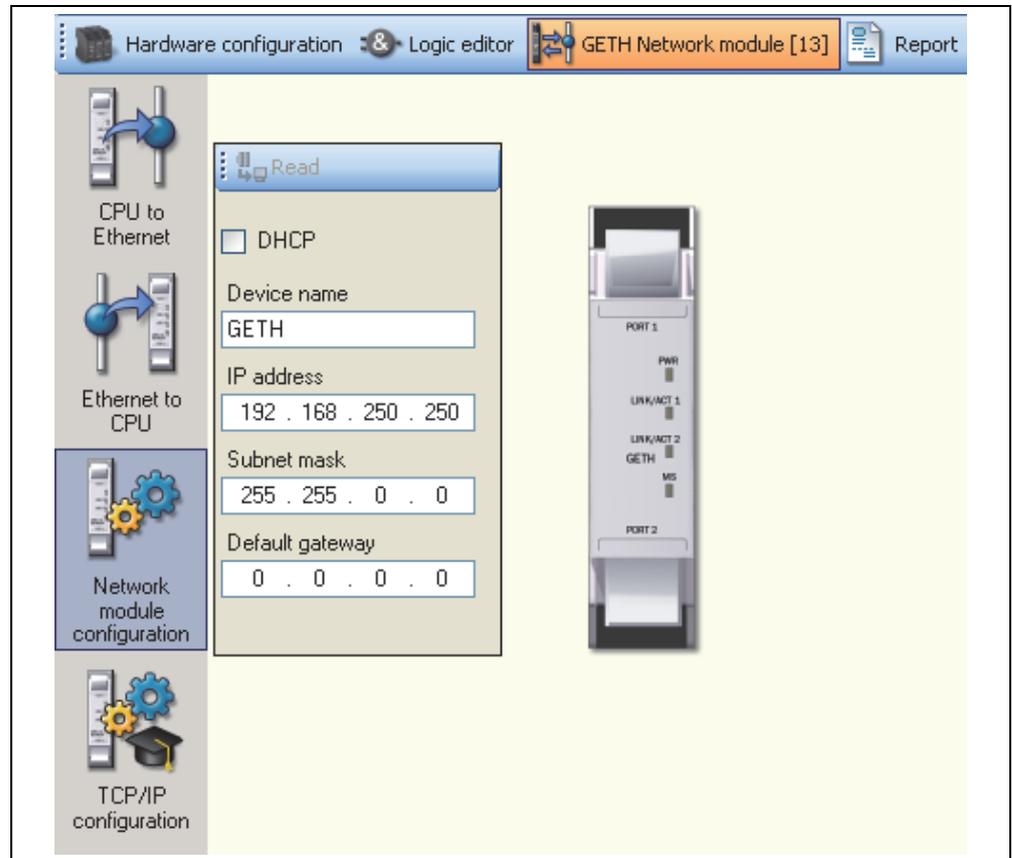


Table 25:  
Network module  
configuration  
parameters

Option	Configuration value
DHCP	Unchecked
Device name	GETH (any given name)
IP address	192.168.250.250 (default)
Subnet mask	255.255.0.0 (default)
Default gateway	0.0.0.0 (default)

3) Configuring TCP/IP [for reading/writing]

Configure the communication mode (polling mode) in the **TCP/IP configuration** dialog.

Select **Enable COS update** and **Listen to** for socket 1.

(In the sample program, the WS0-GETH is configured as slave.)

Figure 41:  
TCP/IP configuration dialog

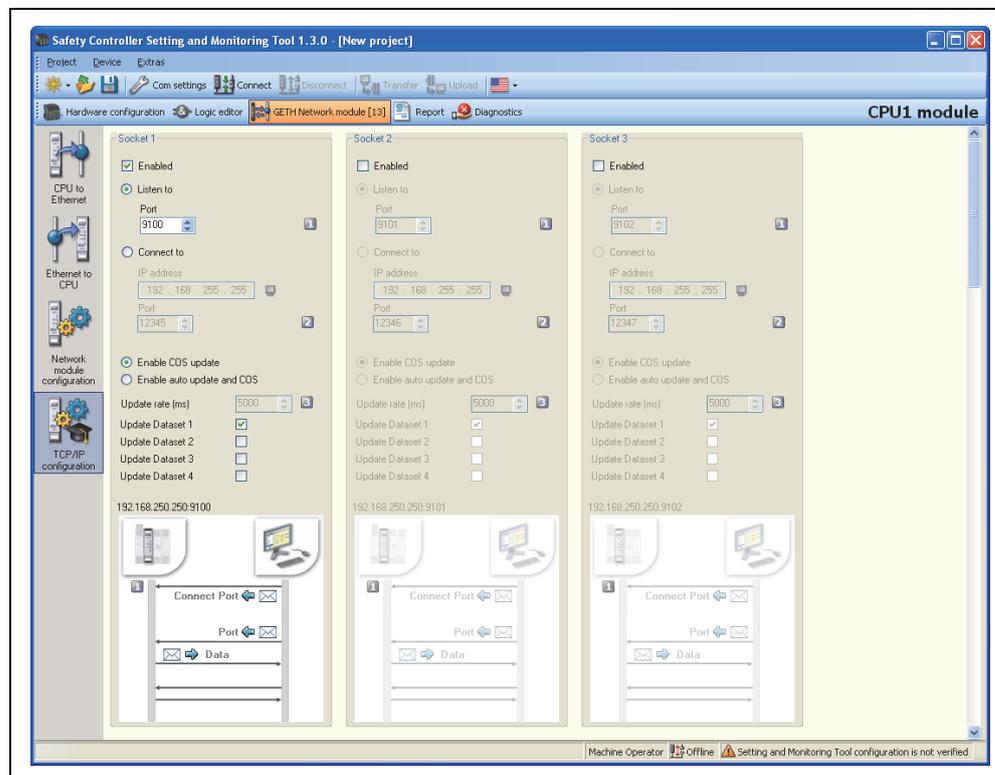


Table 26:  
TCP/IP configuration parameters

Option	Configuration value (Listen to)	Configuration value (Connect to)
Enabled	Checked	
Listen to	Selected	Deselected
Port	9100 (238C <sub>H</sub> ) (default)	-
Connect to	Deselected	Selected
IP address	-	192.168.250.251
Port	-	9100 (238C <sub>H</sub> )
Enable COS update/Enable auto update and COS	Any (All Update Dataset checkboxes need to be unchecked.)	
Update rate (ms)	-	
Update Dataset 1	Unchecked	
Update Dataset 2	Unchecked	
Update Dataset 3	Unchecked	
Update Dataset 4	Unchecked	

4) Editing data sets

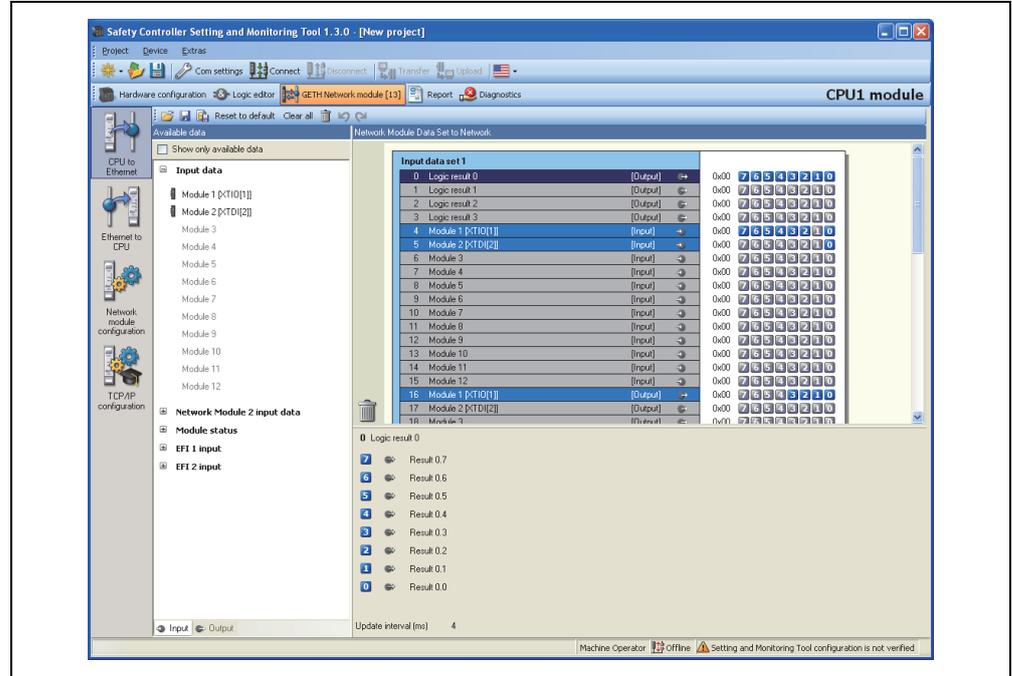
Editing input data set [for reading]

Open the **CPU to Ethernet** dialog and edit input data set 1.

In the sample program, the default settings are used.

(Input data sets 2 to 4 cannot be edited since their data formats are fixed.)

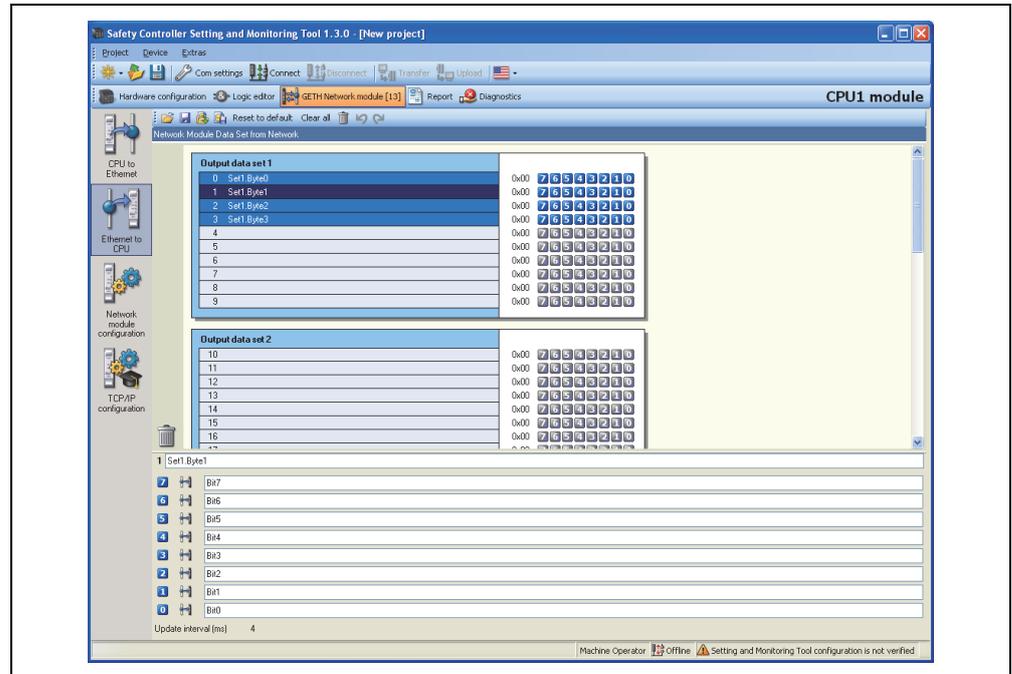
Figure 42:  
CPU to Ethernet dialog



Editing output data sets [for writing]

Open the **Ethernet to CPU** dialog and configure tag names for each output data set as required.

Figure 43:  
Ethernet to CPU dialog



(b) Configuration required for the QJ71E71-100

Configuration examples of network parameters required for communications in polling mode are shown below. Configure these parameters using a programming tool.

1) Configuring network parameters [for reading/writing]

Select **Ethernet/CC IE/MELSECNET** to open the network parameter dialog and configure the following options.

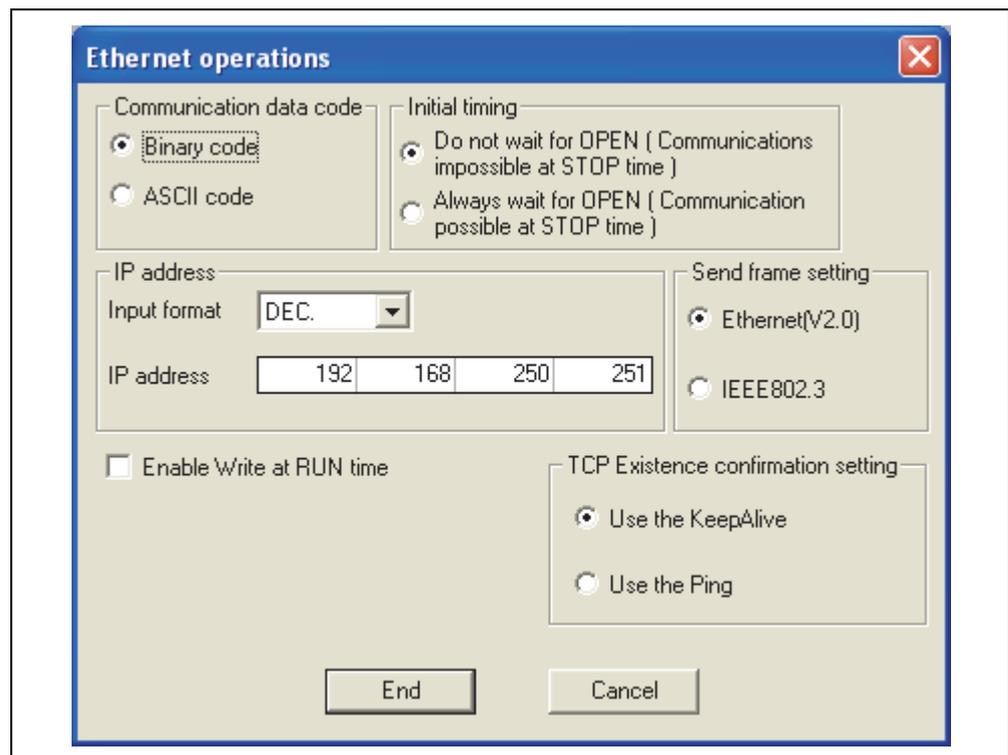
Table 27:  
Network parameter configuration for Ethernet

Option	Configuration value
Network type	Ethernet
Starting I/O No.	0000
Network No.	1
Group No.	1
Station No.	2

2) Network parameter – **Operational settings** [for reading/writing]

Click the **Operational settings** button to open the configuration dialog and configure the options as follows.

Figure 44:  
Network parameter - Operational settings



3) Network parameter– **Open settings** [for reading/writing]

Click the **Open settings** button to open the configuration dialog and configure the options as follows.

(In the sample program, the QJ71E71-100 is configured as master.)

Figure 45:  
Network parameter -  
Open settings

	Protocol	Open system	Fixed buffer	Fixed buffer communication procedure	Pairing open	Existence confirmation	Host station Port No.	Transmission target device IP address	Transmission target device Port No.
1	TCP	Active	Receive	No procedure	Enable	Confirm	238C	192.168.250.250	238C
2	TCP	Active	Send	No procedure	Enable	Confirm	238C	192.168.250.250	238C

Table 28:  
Network parameter -  
Open settings

Option	Configuration value (Active)	Configuration value (Unpassive)
Protocol	TCP	
Open system	Active	Unpassive
Fixed buffer communication procedure	No procedure	
Pairing open	Enable	
Existence confirmation	Confirm	
Host station Port No. (hex)	238CH (9100)	238CH (9100)
Transmission target device IP address	192.168.250.250	-
Transmission target device Port No. (hex)	238CH (9100)	-

4) Network parameter – **Initial settings** [for reading/writing]

Click the **Initial settings** button to open the configuration dialog and configure “120” for **Destination existence confirmation starting interval**.

The QJ71E71-100 can quickly detect a communication stop due to cable disconnection by configuring a smaller value for this option.

Figure 46:  
Network parameter -  
Initial settings

	Setting value	Default value	In units
TCP ULP timer		60	×500ms
TCP zero window timer		20	×500ms
TCP resend timer		20	×500ms
TCP end timer		40	×500ms
IP assembly timer		10	×500ms
Response monitoring timer		60	×500ms
Destination existence confirmation starting interval	120	1200	×500ms
Destination existence confirmation interval timer		20	×500ms
Destination existence confirmation resend		3	Times

(2) Sample program

1) Device assignment [for reading/writing]

The following tables list devices used in the sample program.

Table 29: Devices used in the sample program

W device	
W0 to W18	Input data set 1 storage area
W20 to W2F	Input data set 2 storage area
W30 to W4D	Input data set 3 storage area

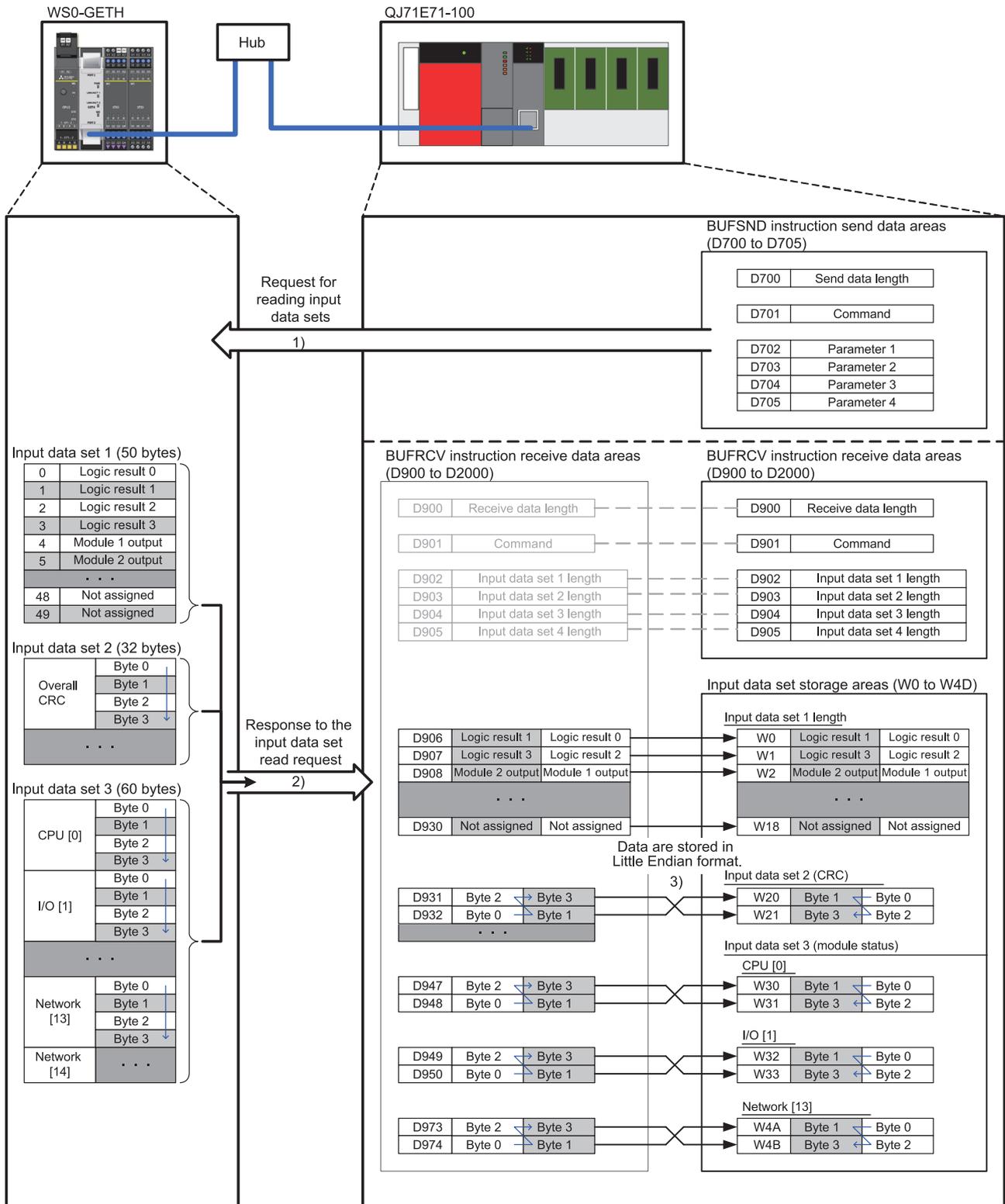
M device		D device	
M0 to M15	Open complete signal	D0	OPEN instruction abnormal completion error code
M20 to M35	Open request signal	D1	BUFSND instruction abnormal completion error code
M40 to M55	Fixed buffer reception status signal	D2	BUFRCV instruction abnormal completion error code
M300	Open execution	D3	CLOSE instruction abnormal completion error code
M301	OPEN instruction success device	D300 to D309	OPEN instruction control data
M302	OPEN instruction failure device	D400 and D401	BUFSND instruction control data
M310	OPEN instruction normal completion	D500 and D501	BUFRCV instruction control data
M311	OPEN instruction abnormal completion	D520	Response command check device (reading)
M400	Send execution	D521	Response command check device (writing)
M401	BUFSND instruction success device	D600 and D601	CLOSE instruction control data
M402	BUFSND instruction failure device	D700 to D705	BUFSND instruction send data area (reading)
M410	BUFSND instruction normal completion	D800 to D831	BUFSND instruction send data area (writing)
M411	BUFSND instruction abnormal completion	D900 to D976	BUFRCV instruction receive data area (reading and writing)
M420	Command in execution	<b>X device</b>	
M500	Receive execution	X19	Initial processing normal completion signal
M501	BUFRCV instruction success device	-	-
M502	BUFRCV instruction failure device	-	-
M510	BUFRCV instruction normal completion	-	-
M511	BUFRCV instruction abnormal completion	-	-
M520	Response command verification normal completion	-	-
M521	Data size error device	-	-
M601	CLOSE instruction success device	-	-
M602	CLOSE instruction failure device	-	-
M610	CLOSE instruction normal completion	-	-
M611	CLOSE instruction abnormal completion	-	-

<b>C device*1</b>		
C0	Open instruction	Number of executions
C1		Number of successes
C2		Number of failures
C10	BUFSND instruction	Number of executions
C11		Number of successes
C12		Number of failures
C20	BUFRCV instruction	Number of executions
C21		Number of successes
C22		Number of failures
C31	Verification check	Number of successes
C32		Number of failures
C33		Number of response errors (writing)
C40	Close instruction	Number of executions
C41		Number of successes
C42		Number of failures

\*1 C device is used to check the operation of the communication program.

The following shows details of data transmission and reception processing in a program for reading input data sets in polling mode.

Figure 47:  
Data transmission and reception processing in polling mode (input data set reading)



- The command and parameters for reading input data sets are stored to the BUFSND instruction send data areas (D700 to D705), and the BUFSND instruction is executed (refer to 1) in Figure 47).
- The WS0-GETH responds to the command. The response data (input data sets) are stored to the BUFRCV instruction receive data areas (D900 to D976) (refer to 2) in Figure 47). (The BUFRCV instruction is executed automatically upon data reception.)
- The byte order of the receive data<sup>\*1</sup> is changed, and input data sets are stored to the corresponding input data set storage areas (W0 to W4D) (refer to 3) in Figure 47).

<sup>\*1</sup> Since the QJ71E71-100 treats application data in Little Endian format, the byte-order needs to be swapped. Note that the byte order of the input data set 1 and output data set 1 does not need to be changed because these data are one byte long.

<Data need to be swapped>

- Command and parameters (word data (16 bits))
- Input data set 2 and 3 (double-word data (32 bits))

The following table lists data to be sent at reading.

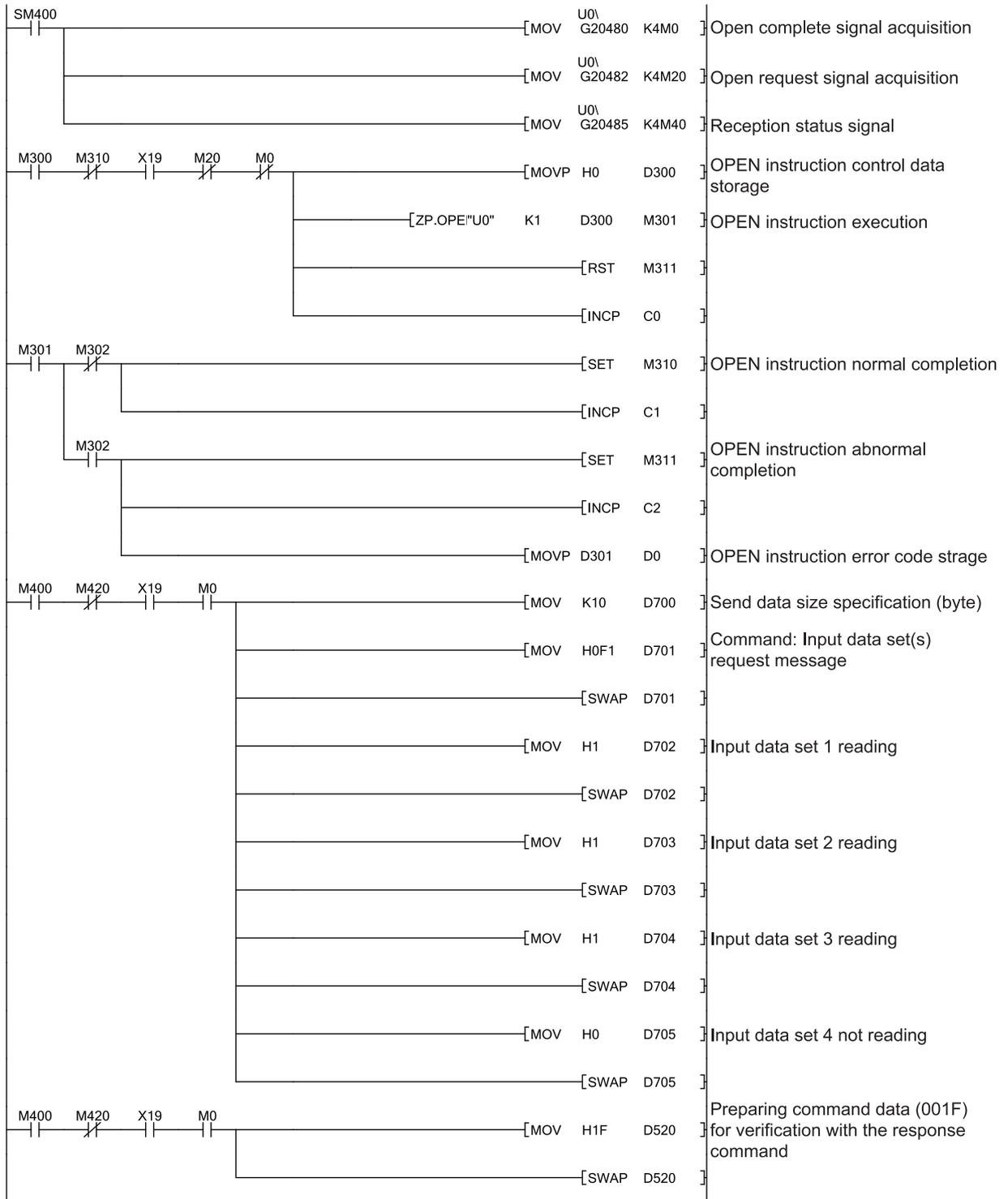
Table 30:  
Data to be sent at  
reading

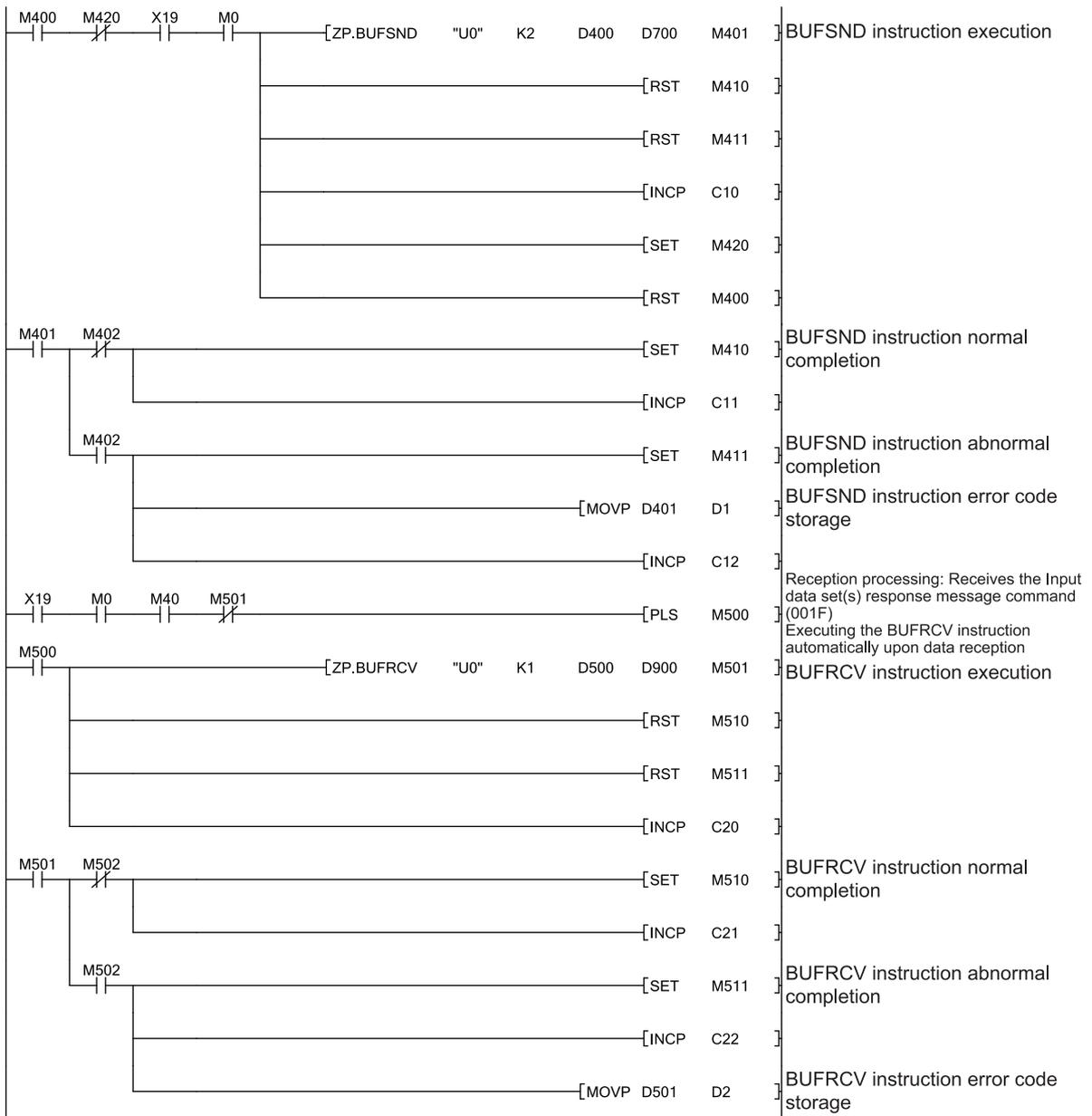
Device	Configuration value (hex)	Byte swapping <sup>*1</sup>	Description
D700	000A <sub>H</sub>	Not required	Send data length (10 bytes)
D701	00F1 <sub>H</sub>	Required	Command
D702	0001 <sub>H</sub>		Read data specification (Input data set 1 reading)
D703	0001 <sub>H</sub>		Read data specification (Input data set 2 reading)
D704	0001 <sub>H</sub>		Read data specification (Input data set 3 reading)
D705	0000 <sub>H</sub>		Read data specification (Input data set 4 not reading)

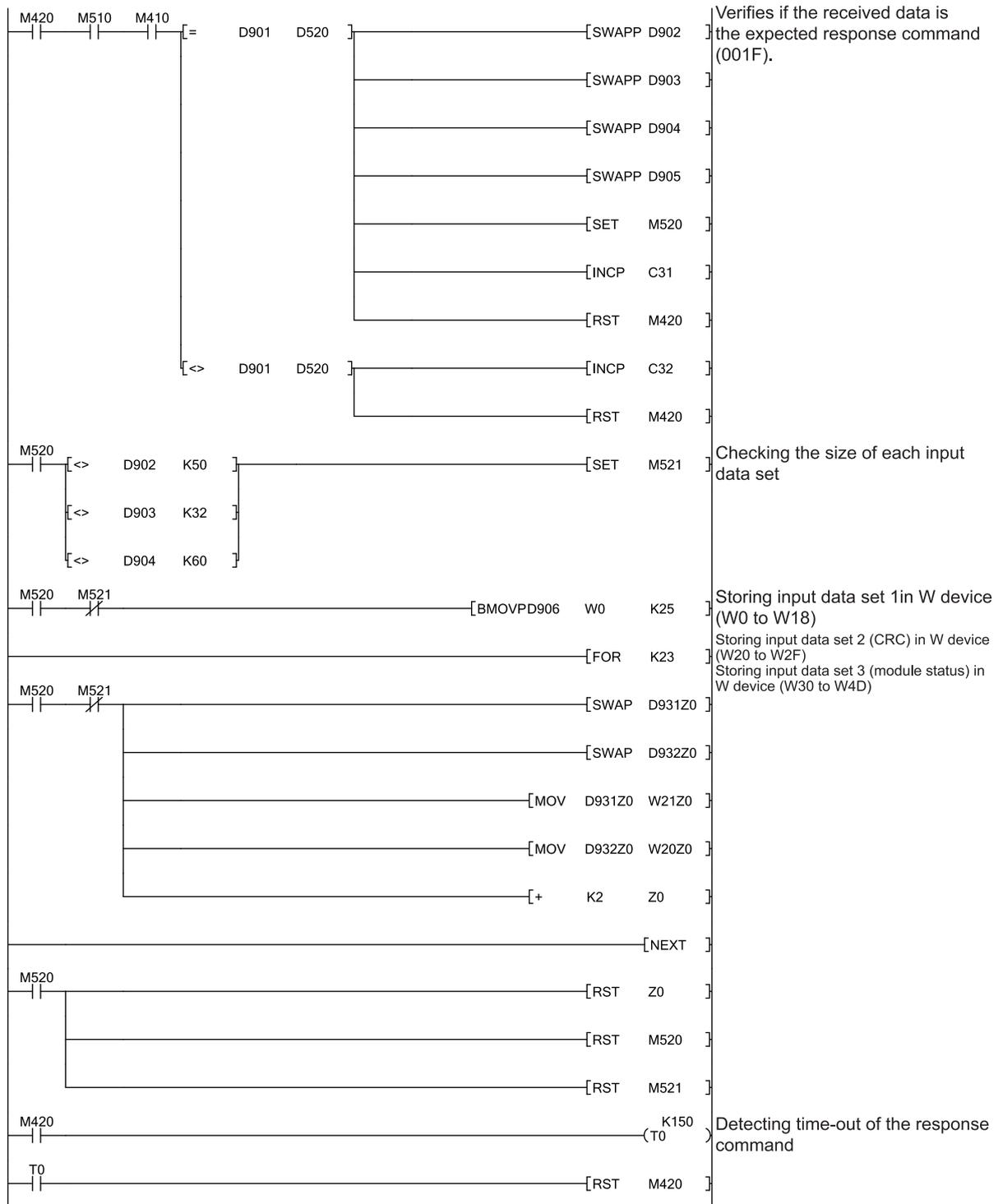
<sup>\*1</sup> Since the QJ71E71-100 treats application data in Little Endian format, the byte-order needs to be swapped.

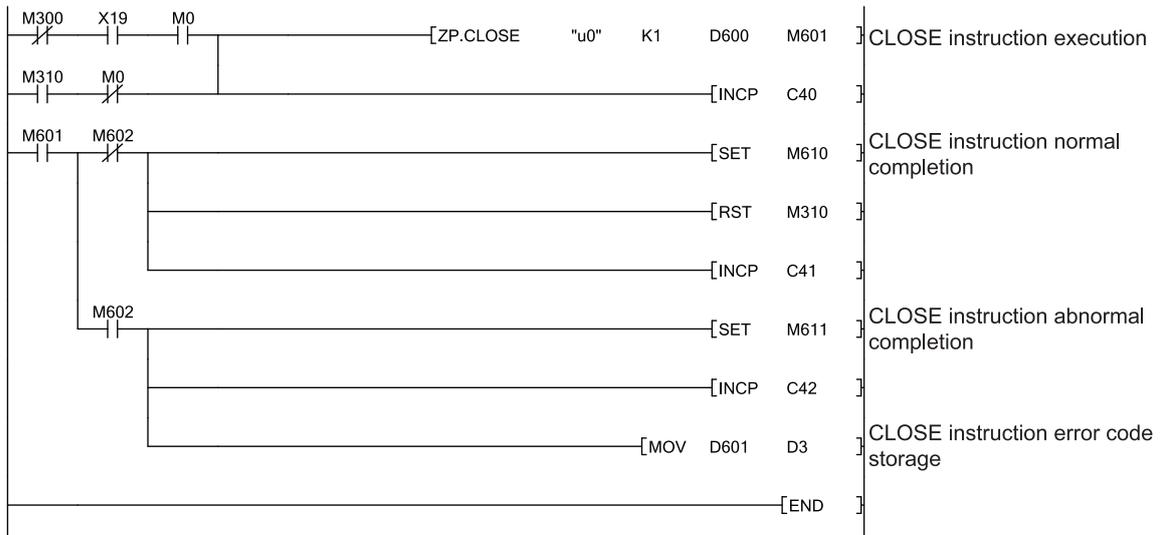
**Sample program: polling mode (input data set reading)**

Figure 48: Sample program: polling mode (input data set reading)









(3) Monitoring data with a programming tool

The read input data sets can be monitored using the device monitor function of a programming tool. The following windows show the execution results of the program for reading input data sets in polling mode.

Example of a logic result (input data set 1) (The bit 0 of the logic result 0 is **Active** (High).)

Figure 49:  
Input data set 1  
(a logic result)

Device	+F E D C	+B A 9 8	+7 6 5 4	+3 2 1 0	
W0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 1	0001
W1	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0000

Example of overall CRC (input data set 2) (overall CRC = 0 x 47B97013)

Figure 50:  
Input data set 2  
(overall CRC)

Device	+F E D C	+B A 9 8	+7 6 5 4	+3 2 1 0	
W20	0 1 1 1	0 0 0 0	0 0 0 1	0 0 1 1	47B97013
W21	0 1 0 0	0 1 1 1	1 0 1 1	1 0 0 1	

Example of CPU module [0] status (input data set 3) (in RUN status)

Figure 51:  
Input data set 3  
(CPU module status)

Device	+F E D C	+B A 9 8	+7 6 5 4	+3 2 1 0	
W30	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	FFFF
W31	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	FFFF

(Reference: When the CPU module is in the STOP state, the value will be "FFFF FFFE".)

Example of network module [13] status (input data set 3)  
(Input status = **Inactive** (Low))

Figure 52:  
Input data set 3  
(network module status)

Device	+F E D C	+B A 9 8	+7 6 5 4	+3 2 1 0	
W4A	1 1 1 1	1 1 1 1	1 1 0 1	1 1 1 1	FFDF
W4B	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	FFFF

(Reference: The input status becomes **Active** (High) after a write command is executed to the WS0-GETH.)

### 8.2.3 Polling mode (output data set writing)

This section describes an example of communications in polling mode.

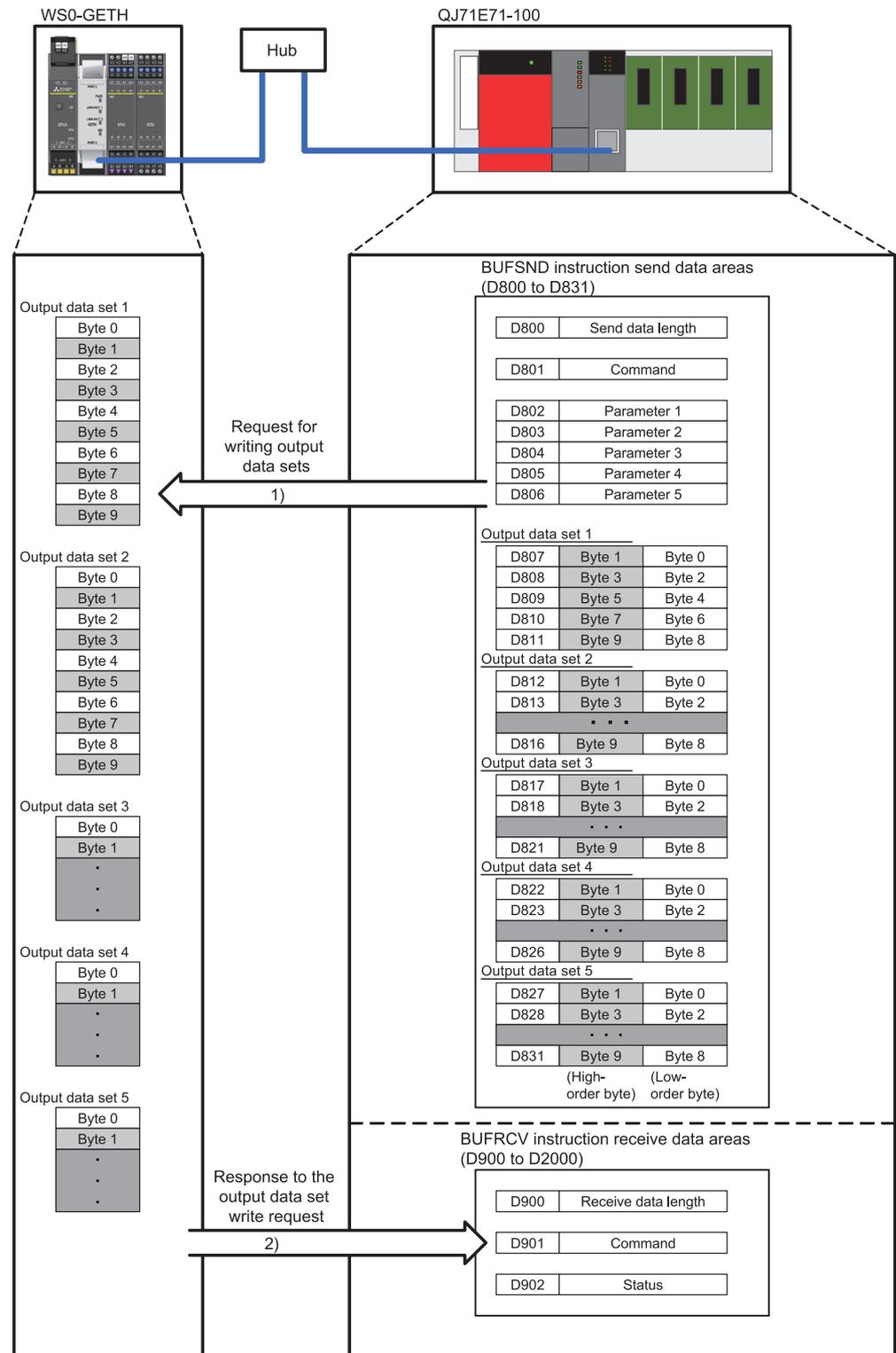
The QCPU writes MELSEC-WS safety controller information (e.g. logic result, CRC, and each module status) to the safety controller through communications between the WS0-GETH and the QJ71E71-100.

#### Write-target data

- Output data sets 1 to 5  
(In the sample program, bit 0 of Byte 0, bit 1 of Byte 1, bit 2 of Byte 2, and bit 3 of Byte 3 in output data set 1 are set to Active (High).)
- (1) Parameter configuration
    - (a) Configuration required for the WS0-GETH  
Refer to Section 8.2.2 (1).
    - (b) Configuration required for the QJ71E71-100  
Refer to Section 8.2.2 (1).
  - (2) Sample program
    - 1) Device assignment  
Refer to Section 8.2.2 (2).

The following shows details of data transmission and reception processing in a program for writing output data sets in polling mode.

Figure 53:  
Data transmission and reception processing in polling mode (output data set writing)



- The command and parameters for writing output data sets and the output data set information are stored to the BUFSND instruction send data areas (D800 to D831), and the BUFSND instruction is executed (refer to 1) in Figure 53).
- The WS0-GETH responds to the command. The response data are stored to the BUFRCV instruction receive data areas (D900 to D976) (refer to 2) in Figure 53). (The BUFRCV instruction is executed automatically upon data reception.)

The following table lists data to be sent at writing.

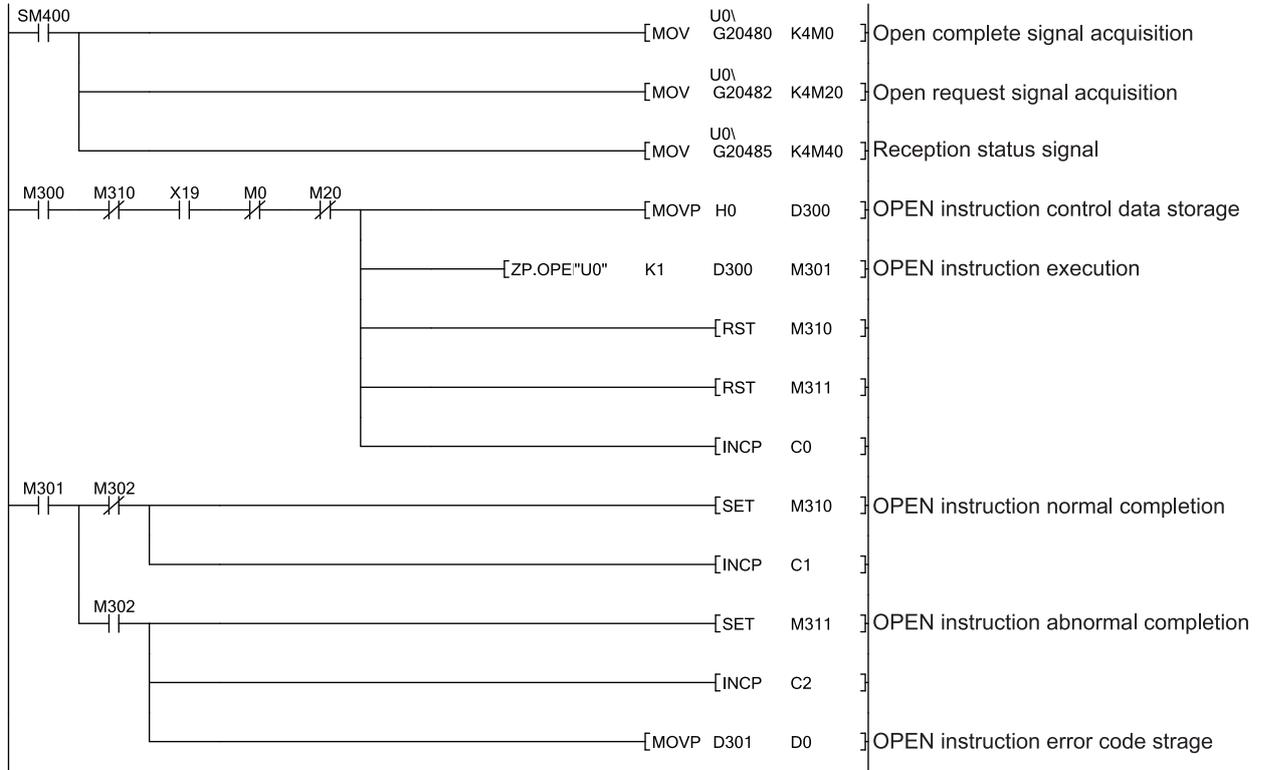
Table 31:  
Data to be sent at  
writing

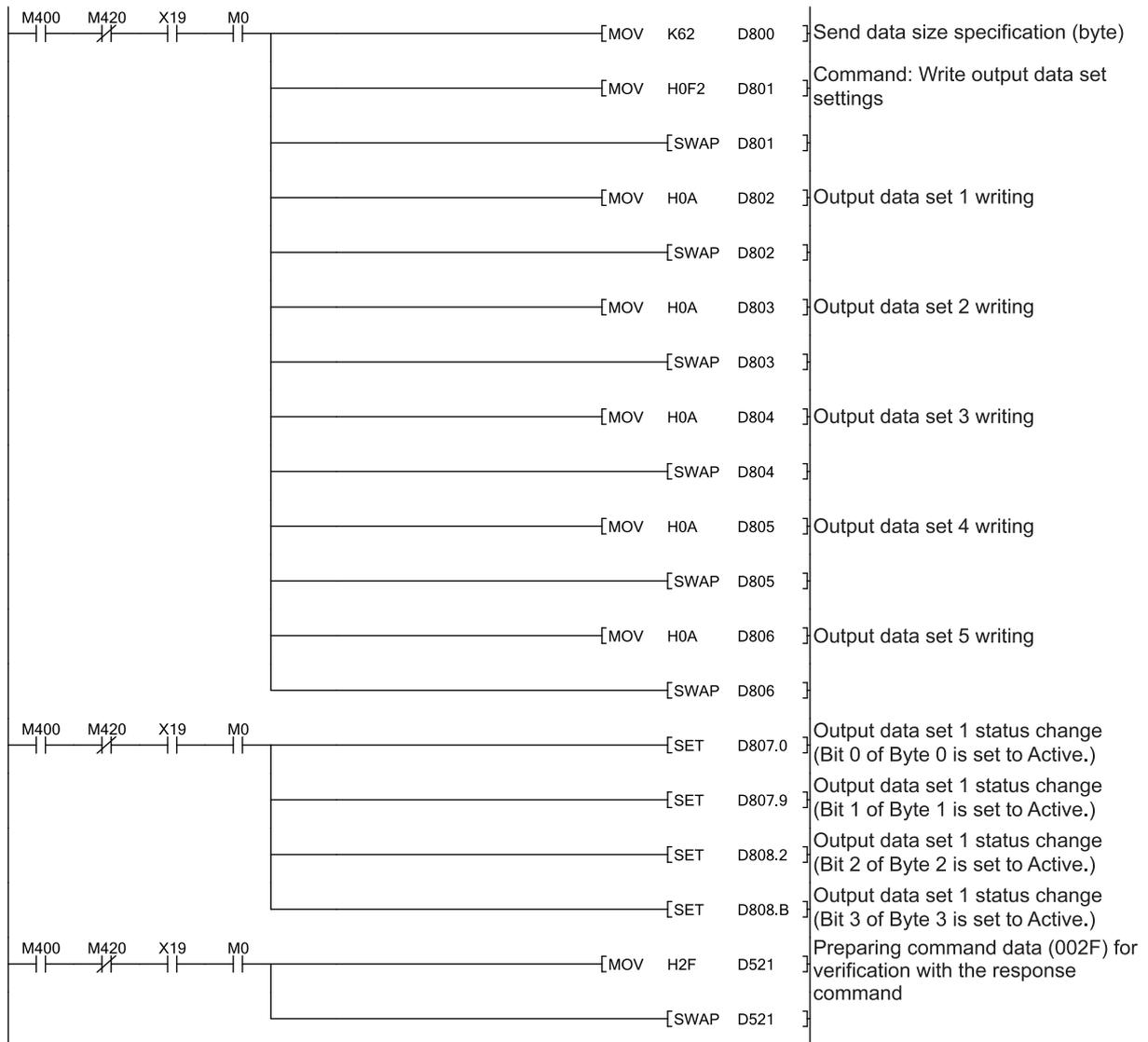
Device	Configuration value (hex)	Byte swapping*1	Description
D800	003E <sub>H</sub>	Not required	Send data length (62 bytes)
D801	00F2 <sub>H</sub>	Required	Command
D802	000A <sub>H</sub>		Write data specification (Output data set 1 writing (10 bytes))
D803	000A <sub>H</sub>		Write data specification (Output data set 2 writing (10 bytes))
D804	000A <sub>H</sub>		Write data specification (Output data set 3 writing (10 bytes))
D805	000A <sub>H</sub>		Write data specification (Output data set 4 writing (10 bytes))
D806	000A <sub>H</sub>		Write data specification (Output data set 5 writing (10 bytes))
D807	Any	Not required	Output data set 1: Byte 0 and Byte 1
D808			Output data set 1: Byte 2 and Byte 3
D809			Output data set 1: Byte 4 and Byte 5
D810			Output data set 1: Byte 6 and Byte 7
D811			Output data set 1: Byte 8 and Byte 9
D812 to D816	Any	Not required	Output data set 2
D817 to D821	Any	Not required	Output data set 3
D822 to D826	Any	Not required	Output data set 4
D827 to D831	Any	Not required	Output data set 5

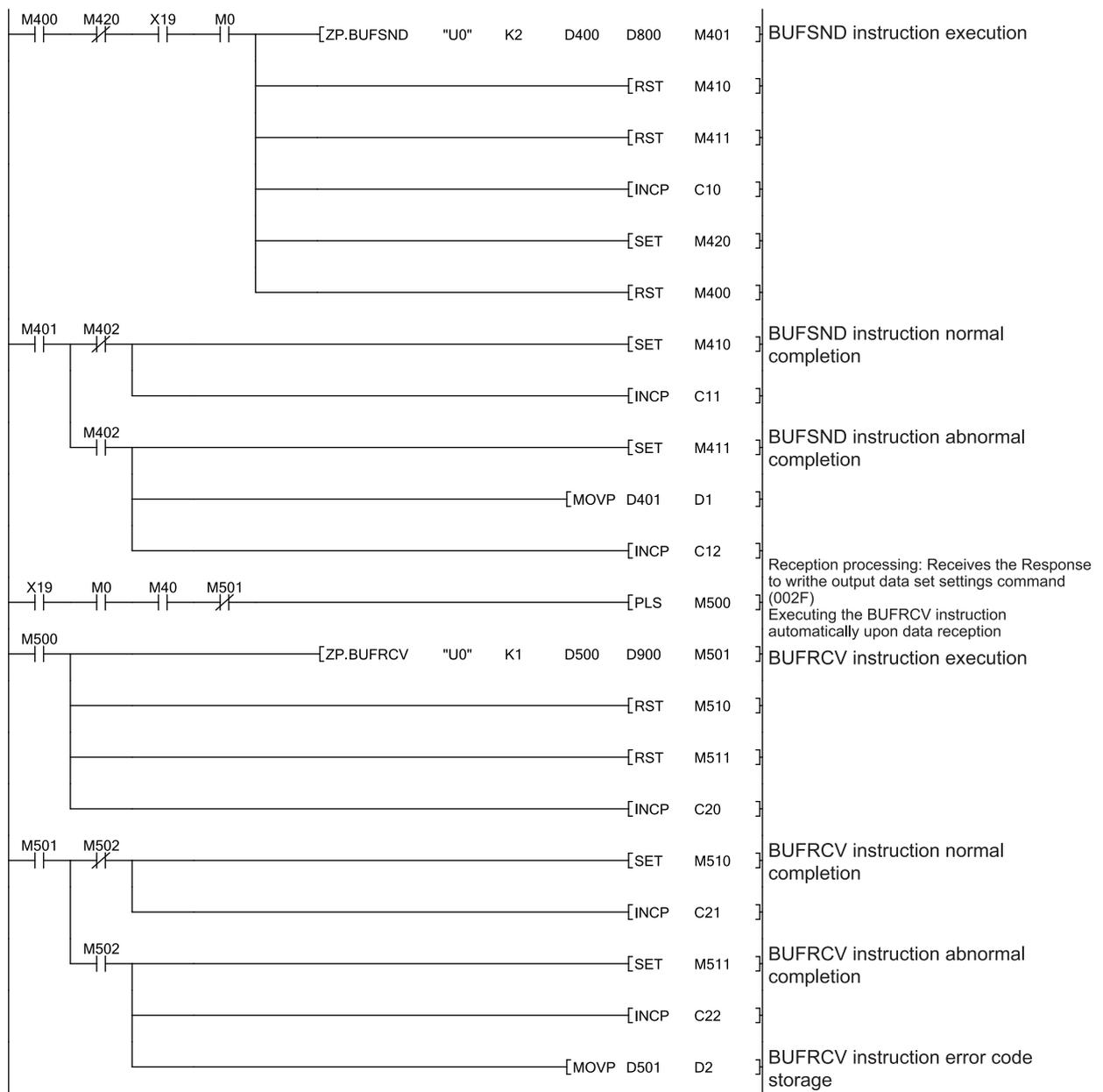
\*1 Since the QJ71E71-100 treats application data in Little Endian format, the byte-order needs to be swapped.

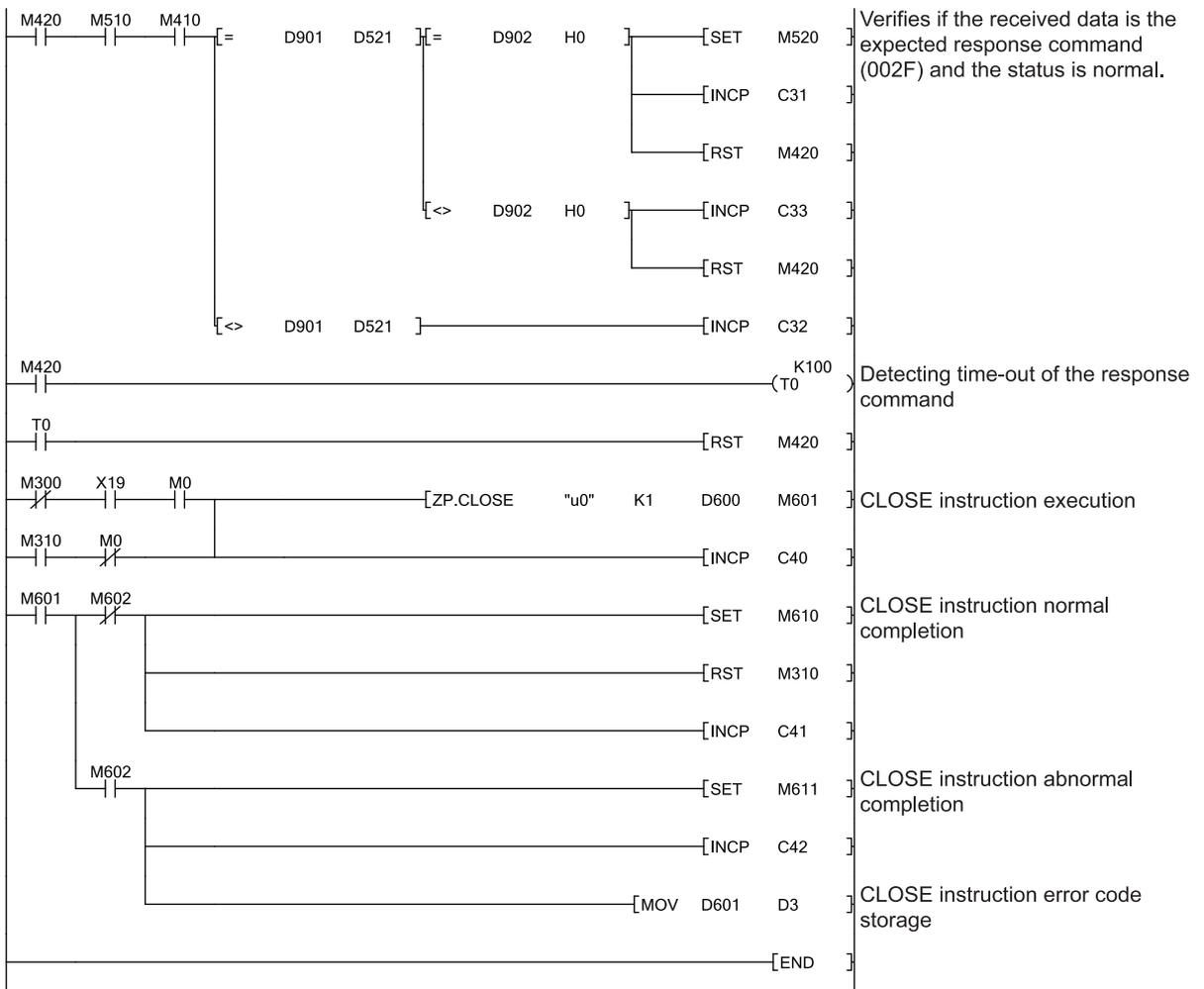
**Sample program: polling mode (output data set writing)**

Figure 54: Sample program: polling mode (output data set writing)









The following table shows an example of the execution result of the program for writing output data sets in polling mode.

Table 32:  
Execution result  
example in polling  
mode (output data set  
writing)

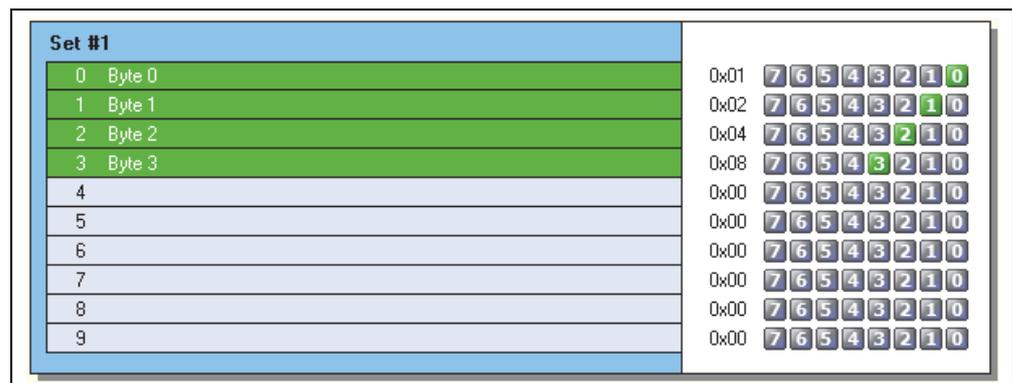
Device	Storage value (hex)	Byte swapping*1	Description
D900	000A <sub>H</sub>	Not required	Receive data length
D901	0004 <sub>H</sub>	Required	Response command
D902	0000 <sub>H</sub>		Status 0: Successful 1: Error

\*1 Since the QJ71E71-100 treats application data in Little Endian format, the byte-order needs to be swapped.

### (3) Monitoring data with Setting and Monitoring Tool

The written output data sets can be monitored in the **Ethernet to CPU** dialog of Setting and Monitoring Tool. The following window shows the execution result of the program for writing output data set in polling mode.

Figure 55:  
Monitoring result in the  
Ethernet to CPU dialog



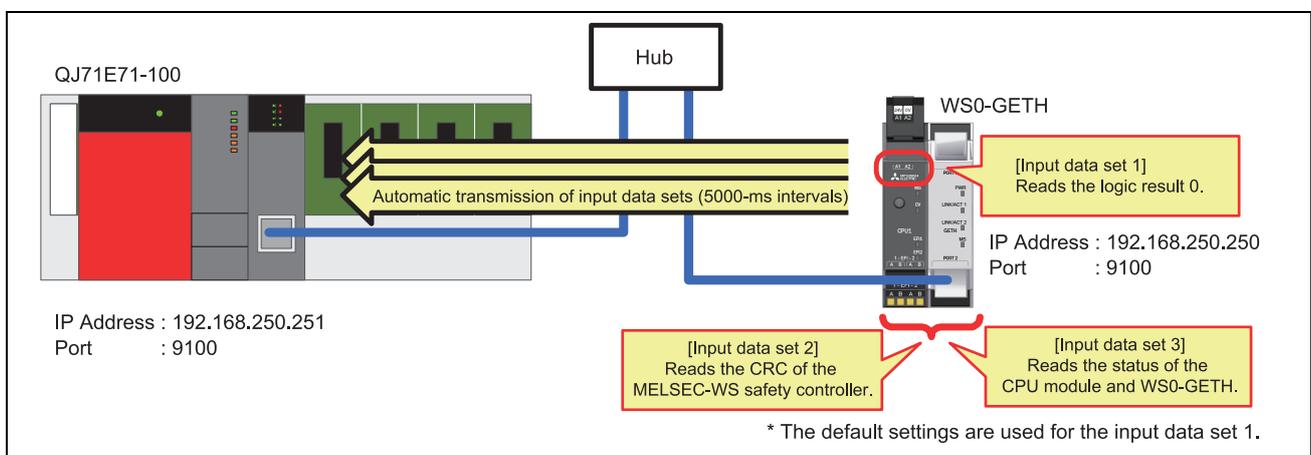
### 8.2.4 Auto update mode

This section describes an example of communications in auto update mode. The QCPU reads MELSEC-WS safety controller information (e.g. logic result, CRC, and each module status) from the safety controller through communications between the WS0-GETH and the QJ71E71-100. The WS0-GETH automatically sends input data sets to the QJ71E71-100 at 5000-ms intervals.

#### Read-target data

- Logic result 0 (input data set 1)
- CRC (input data set 2)
- Each module status (input data set 3)

Figure 56: System configuration example



#### (1) Parameter configuration

##### (a) Configuration required for the WS0-GETH

Configuration examples of network parameters required for communications in auto update mode are shown below. Configure these parameters using Setting and Monitoring Tool.

The WS0-GETH requires no program for communications.

##### 1) Creating a hardware configuration

Refer to Section 8.2.2 (1).

##### 2) Assigning an IP address

Refer to Section 8.2.2 (1).

3) Configuring TCP/IP

Configure the communication mode (auto update mode) in the **TCP/IP configuration** dialog.

Select **Enable auto update and COS** and **Connect to** for socket 1.  
(In the sample program, the WS0-GETH is configured as master.)

Figure 57:  
TCP/IP configuration dialog

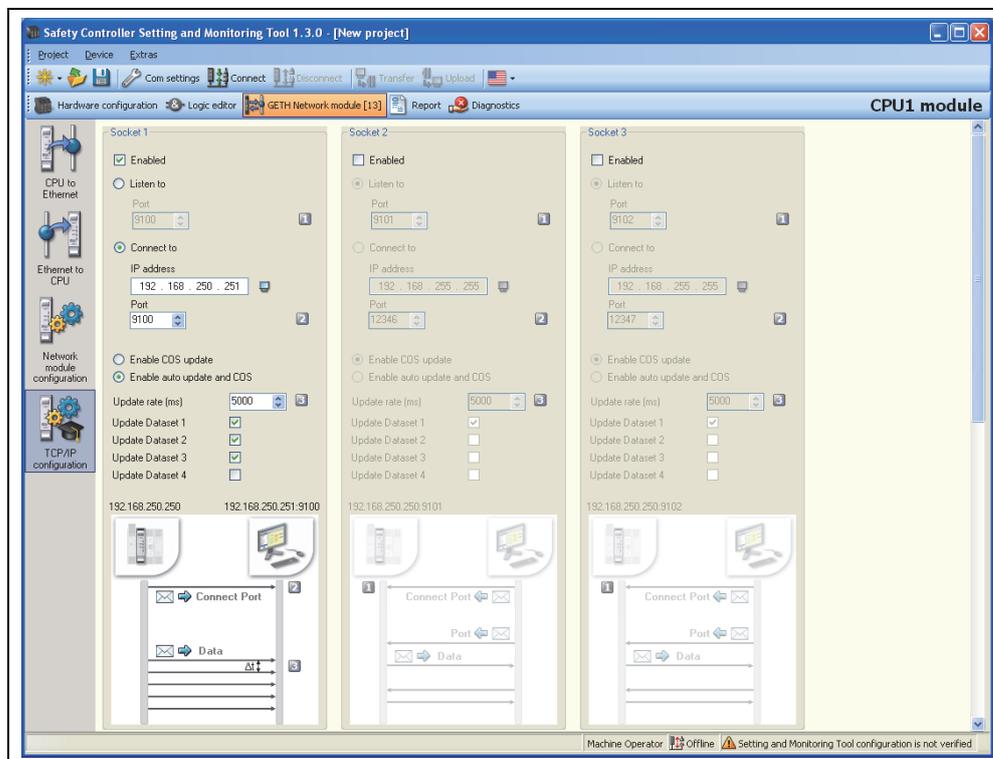


Table 33:  
TCP/IP configuration parameters

Option	Configuration value (Listen to)	Configuration value (Connect to)
Enabled	Checked	
Listen to	Selected	Deselected
Port	9100 (238CH) (default)	-
Connect to	Deselected	Selected
IP address	-	192.168.250.251
Port	-	9100 (238CH)
Enable COS update/Enable auto update and COS	Enable auto update and COS	
Update rate (ms)	5000 (default)	
Update Dataset 1	Checked	
Update Dataset 2	Checked	
Update Dataset 3	Checked	
Update Dataset 4	Unchecked	

To switch the communication mode to the polling mode, activate the **Enable COS update** radio button.

4) Editing data sets

Refer to Section 8.2.2 (1).

## (b) Configuration required for the QJ71E71-100

Configuration examples of network parameters required for communications in auto update mode are shown below. Configure these parameters using a programming tool.

## 1) Configuring network parameters

Refer to Section 8.2.2 (1).

2) Network parameter – **Operational settings**

Click the **Operational settings** button to open the configuration dialog and configure the options as follows.

Figure 58:  
Network parameter -  
Operational settings

**Ethernet operations**

Communication data code

Binary code

ASCII code

Initial timing

Do not wait for OPEN ( Communications impossible at STOP time )

Always wait for OPEN ( Communication possible at STOP time )

IP address

Input format: DEC.

IP address: 192 168 250 251

Send frame setting

Ethernet(V2.0)

IEEE802.3

Enable Write at RUN time

TCP Existence confirmation setting

Use the KeepAlive

Use the Ping

End Cancel

3) Network parameter – **Open settings**

Click the **Open settings** button to open the configuration dialog and configure the options as follows.

(In the sample program, the QJ71E71-100 is configured as slave.)

Figure 59:  
Network parameter -  
Open settings

	Protocol	Open system	Fixed buffer	Fixed buffer communication procedure	Pairing open	Existence confirmation	Host station Port No.	Transmission target device IP address	Transmission target device Port No.
1	TCP	Unpassive	Receive	No procedure	Disable	Confirm	238C		
2									

Table 34:  
Network parameter -  
Open settings

Option	Configuration value (Active)	Configuration value (Unpassive)
Protocol	TCP	
Open system	Active	Unpassive
Fixed buffer communication procedure	No procedure	
Pairing open	Disable	
Existence confirmation	Confirm	
Host station Port No. (hex)	238C <sub>H</sub> (9100)	238C <sub>H</sub> (9100)
Transmission target device IP address	192.168.250.250	-
Transmission target device Port No. (hex)	238C <sub>H</sub> (9100)	-

4) Network parameter – **Initial settings**

Click the **Initial settings** button to open the configuration dialog and configure “120” for **Destination existence confirmation starting interval**.

The QJ71E71-100 can quickly detect a communication stop due to cable disconnection by configuring a smaller value for this option.

Figure 60:  
Network parameter -  
Initial settings

Timer setting

Module will operate with default values if setting is left blank

	Setting value	Default value	In units
TCP ULP timer		60	×500ms
TCP zero window timer		20	×500ms
TCP resend timer		20	×500ms
TCP end timer		40	×500ms
IP assembly timer		10	×500ms
Response monitoring timer		60	×500ms
Destination existence confirmation starting interval	120	1200	×500ms
Destination existence confirmation interval timer		20	×500ms
Destination existence confirmation resend		3	Times

## (2) Sample program

## 1) Device assignment

The following tables list devices used in the sample program.

Table 35: Devices used in the sample program

W device	
W0 to W18	Input data set 1 storage area
W20 to W2F	Input data set 2 storage area
W30 to W4D	Input data set 3 storage area

M device		D device	
M0 to M15	Open complete signal	D2	BUFRCV instruction failure error code
M20 to M35	Open request signal	D500 and D501	BUFRCV instruction control data
M40 to M55	Fixed buffer reception status signal	D520	Response command check device
M500	Receive execution	D900 to D976	BUFRCV instruction receive data area
M501	BUFRCV instruction success device	X device	
M502	BUFRCV instruction failure device	X19	Initial processing normal completion signal
M510	BUFRCV instruction normal completion	-	-
M511	BUFRCV instruction abnormal completion	-	-
M520	Response command verification normal completion		
M521	Data size error device		

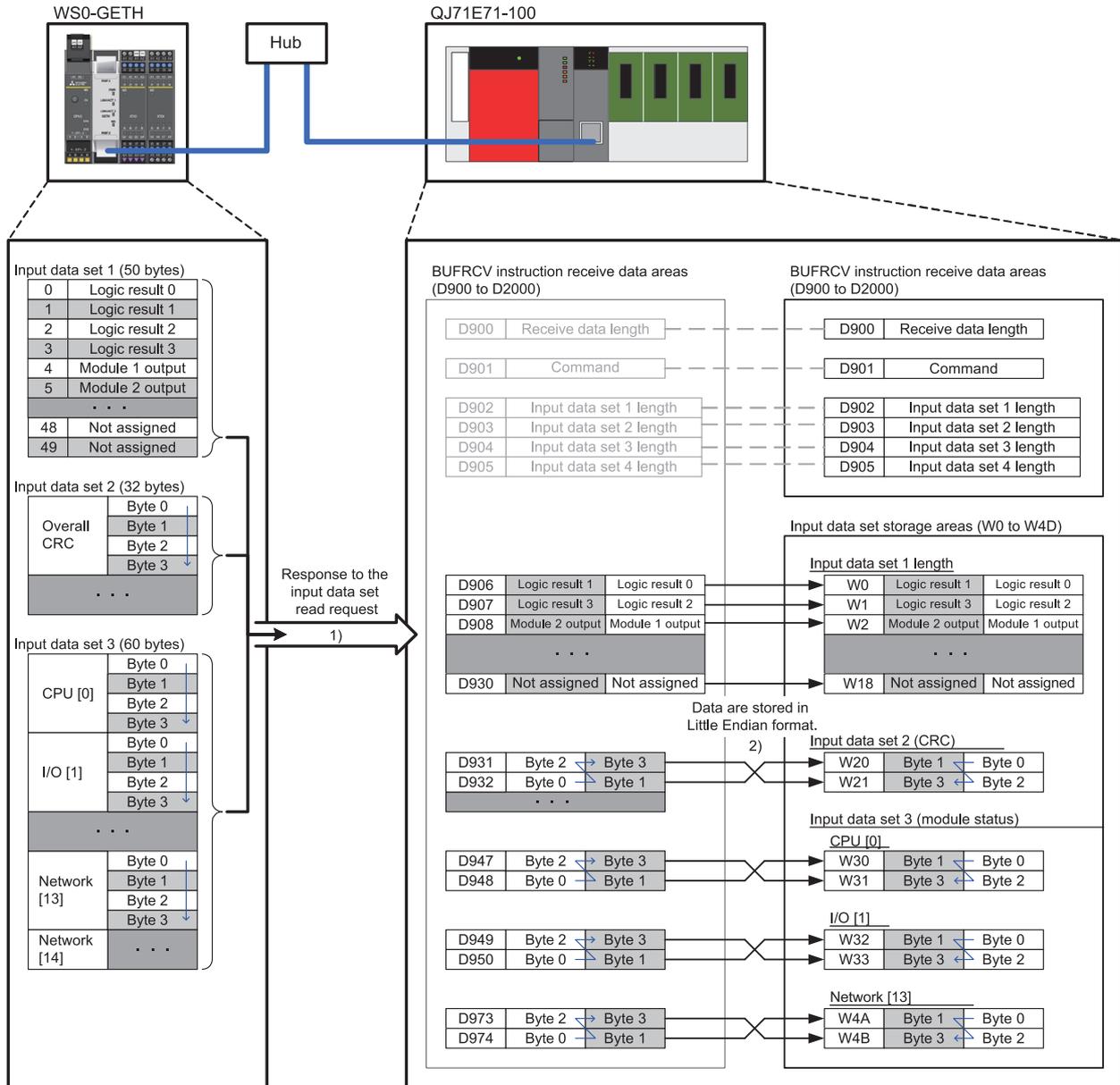
C device*1		
C20	BUFRCV instruction	Number of executions
C21		Number of successes
C22		Number of failures
C31	Verification check	Number of successes
C32		Number of failures

\*1 C device is used to check the operation of the communication program.

- Note**
- If the QCPU connected to the QJ71E71-100 stops during communications in auto update mode, the BUFRCV instruction will not be executed even when data are transmitted from the WS0-GETH. This may cause the receive buffer of the QJ71E71-100 to overflow and data transmission of the WS0-GETH to stop.
  - If the scan time of the QCPU connected to the QJ71E71-100 is longer than the transmission interval of the WS0-GETH, update of received data delays. For this reason, configure the auto update interval much longer than the scan time.

The following shows details of data reception processing in a program for receiving input data sets in auto update mode.

Figure 61: Data reception processing in auto update mode



- The WS0-GETH automatically sends input data sets. The send data are stored to the BUFRCV instruction receive data areas (D900 to D976) (refer to 1) in Figure 61). (The BUFRCV instruction is executed automatically upon data reception.)
- The byte order of the receive data\*1 is changed, and input data sets are stored to the corresponding input data set storage areas (W0 to W4D) (refer to 2) in Figure 61).

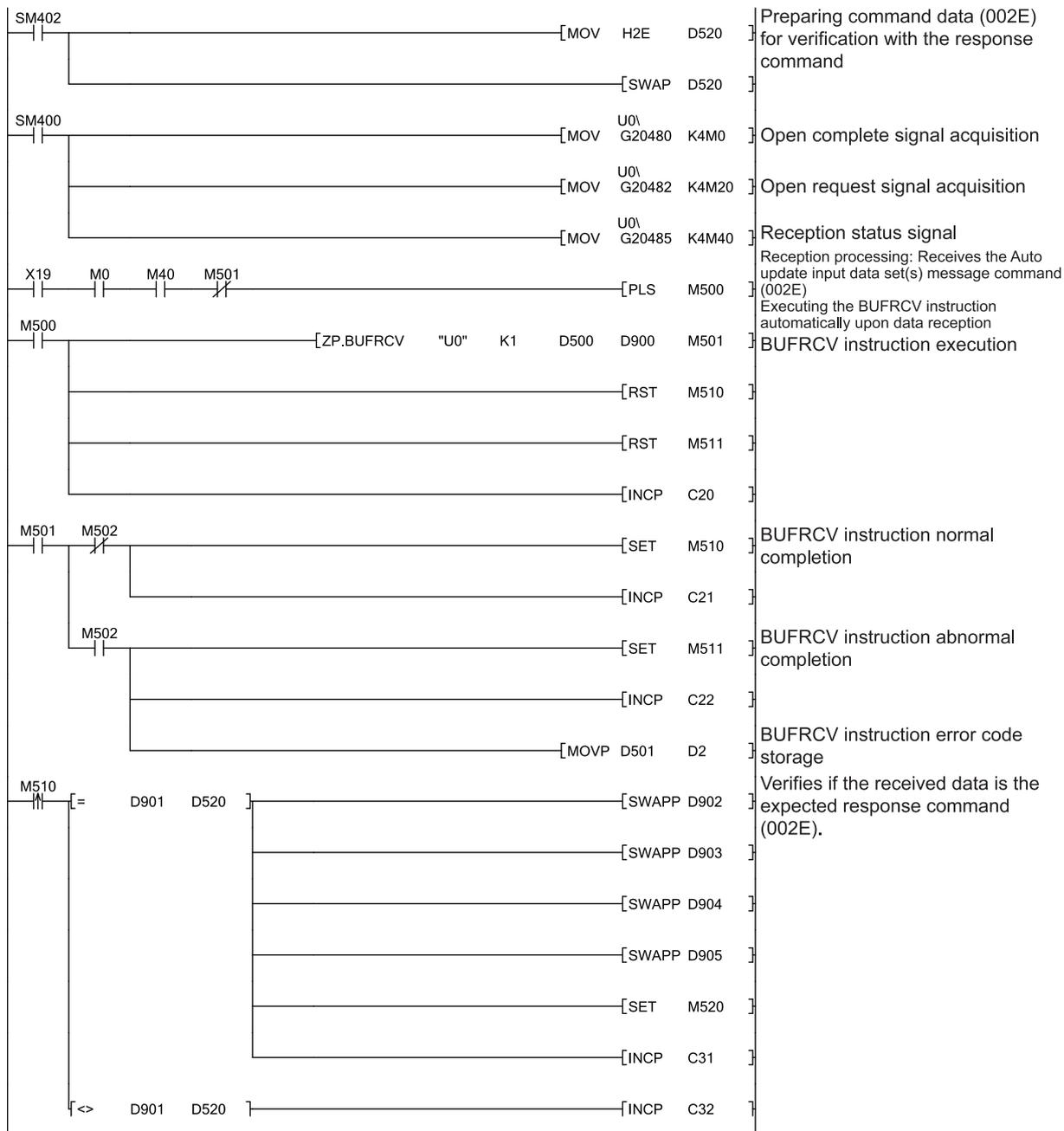
\*1 Since the QJ71E71-100 treats application data in Little Endian format, the byte-order needs to be swapped. Note that the byte order of the input data set 1 and output data set 1 does not need to be changed because these data are one byte long.

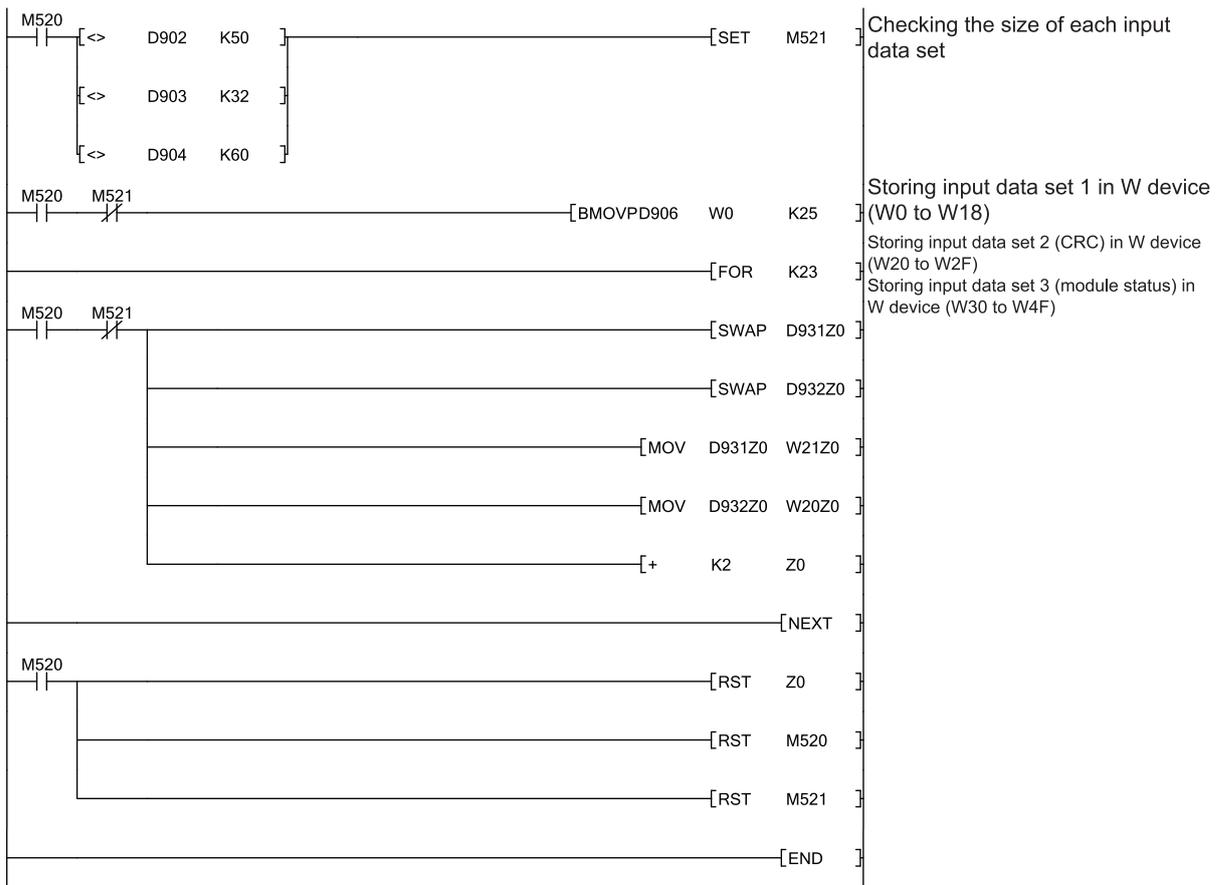
<Data need to be swapped>

- Command and parameters (word data (16 bits))
- Input data sets 2 and 3 (double-word data (32 bits))

Sample program: auto update mode

Figure 62: Sample program: auto update mode





(3) Monitoring data with a programming tool

The read input data sets can be monitored using the device monitor function of a programming tool. For the execution results of the program for auto update mode, refer to Section 8.2.2 (3).

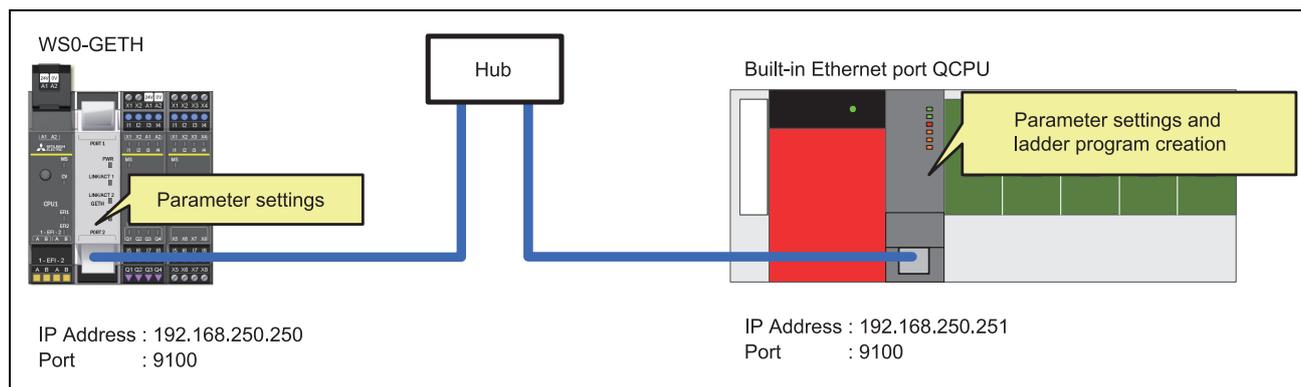
## 8.3 Communications with the Built-in Ethernet port QCPU

This section describes parameter configuration examples and sample programs for communications between the WS0-GETH and the Built-in Ethernet port QCPU.

### 8.3.1 System configuration

The following is the system configuration used for sample programs.

Figure 63: System configuration example



### 8.3.2 Polling mode (input data set reading)

This section describes an example of communications in polling mode.

The Built-in Ethernet port QCPU reads MELSEC-WS safety controller information (e.g. logic result, CRC, and each module status) from the safety controller through communications between the WS0-GETH and the Ethernet ports of Built-in Ethernet port QCPU.

#### Read-target data

- Logic result 0 (input data set 1)
- CRC (input data set 2)
- Each module status (input data set 3)

#### (1) Parameter configuration

##### (a) Configuration required for the WS0-GETH

Refer to Section 8.2.2 (1).

## (b) Configuration required for the Built-in Ethernet port QCPU

Configuration examples of network parameters required for communications in polling mode are shown below. Configure these parameters using a programming tool.

1) Configuring PLC parameters – **Built-in Ethernet port** [for reading/writing]

Click the **Built-in Ethernet port** tab to open the configuration dialog and configure the options as follows.

Figure 64:  
PLC parameter - Built-in Ethernet port

The screenshot shows the 'Built-in Ethernet port' configuration window. At the top, there is a tabbed interface with the following tabs: PLC name, PLC system, PLC file, PLC RAS(1), PLC RAS(2), Device, Program, Boot file, SFC, I/O assignment, and Built-in Ethernet port. The 'Built-in Ethernet port' tab is active.

The main configuration area is divided into two sections:

- IP address section:**
  - Input format: DEC (dropdown menu)
  - IP address: 192 | 168 | 250 | 251
  - Subnet mask pattern: 255 | 255 | 0 | 0
  - Default router IP address: 192 | 168 | 250 | 255
  - Buttons: Open settings, FTP settings, Time settings
  - Status: Set if it is needed (Default / Changed)
- Communication data code section:**
  - Binary code (selected with radio button)
  - ASCII code (radio button)

At the bottom, there are three checkboxes:

- Enable online change (FTP, MC protocol)
- Disable direct connection to MELSOFT
- Do not respond to search for CPU (Built-in Ethernet port) on network

2) PLC parameter – **Open settings** [for reading/writing]

Click the **Open settings** button to open the configuration dialog and configure the options as follows.

Figure 65:  
PLC parameter - Open settings

	Protocol	Open system	TCP connection	Host station port No.	Transmission target device IP address	Transmission target device port No.
1	TCP	Socket communication	Active	238C	192.168.250.250	238C

The **TCP connection** setting differs depending on the connection open status.

- When connection is opened by the Built-in Ethernet port QCPU, set **Active**.
- When connection is opened by the WS0-GETH, set **Unpassive**.

Table 36:  
PLC parameter - Open settings

Option	Configuration value (Active)	Configuration value (Unpassive)
Protocol	TCP	
Open system	Socket communication	
TCP connection	Active	Unpassive
Host station port No.	9100 (238CH)	
Transmission target device IP address	192.168.250.250	-
Transmission target device port No. (hex)	9100 (238CH)	-

## (2) Sample program

This manual describes only devices used in sample programs and details of the processing.

For the acquisition of sample programs, please contact your local Mitsubishi representative.

## 1) Device assignment [for reading/writing]

The following tables list devices used in the sample program.

Table 37: Devices used in the sample program

W device	
W0 to W18	Input data set 1 storage area
W20 to W2F	Input data set 2 storage area
W30 to W4D	Input data set 3 storage area

M device		D device	
M300	Open execution	D0	SOCOPEN instruction abnormal completion error code
M301	SOCOPEN instruction success device	D1	SOCSND instruction abnormal completion error code
M302	SOCOPEN instruction failure device	D2	SOCRCV instruction abnormal completion error code
M310	SOCOPEN instruction normal completion	D3	SOCCLDSE instruction abnormal completion error code
M311	SOCOPEN instruction abnormal completion	D300 to D309	SOCOPEN instruction control data
M400	Send execution	D400 and D401	SOCSND instruction control data
M401	SOCSND instruction success device	D500 and D501	SOCRCV instruction control data
M402	SOCSND instruction failure device	D520	Response command check device (reading)
M410	SOCSND instruction normal completion	D521	Response command check device (writing)
M411	SOCSND instruction abnormal completion	D600 and D601	SOCCLDSE instruction control data
M420	Command in execution	D700 to D705	SOCSND instruction send data area (reading)
M500	Receive execution	D800 to D831	SOCSND instruction send data area (writing)
M501	SOCRCV instruction success device	D900 to D976	SOCRCV instruction receive data area (reading and writing)
M502	SOCRCV instruction failure device	-	-
M510	SOCRCV instruction normal completion	-	-
M511	SOCRCV instruction abnormal completion	-	-
M520	Response command verification normal completion	-	-
M521	Data size error device	-	-
M601	SOCCLDSE instruction success device	-	-
M602	SOCCLDSE instruction failure device	-	-
M610	SOCCLDSE instruction normal completion	-	-
M611	SOCCLDSE instruction abnormal completion	-	-

C device*1		
C0	SOCOPEN instruction	Number of executions
C1		Number of successes
C2		Number of failures
C10	SOCSND instruction	Number of executions
C11		Number of successes
C12		Number of failures
C20	SOCRCV instruction	Number of executions
C21		Number of successes
C22		Number of failures
C31	Verification check	Number of successes
C32		Number of failures
C33		Number of response errors (writing)
C40	SOCCLOSE instruction	Number of executions
C41		Number of successes
C42		Number of failures

\*1 C device is used to check the operation of the communication program.

For details of data transmission and reception processing in a program for reading input data sets in polling mode, refer to Section 8.2.2 (2).

Please replace the BUFSEND and BUFRCV instructions with the SOCSND and SOCRCV instructions.

Since the Built-in Ethernet port QCPU treats application data in Little Endian format, the byte-order needs to be swapped. Note that the byte order of the input data set 1 and output data set 1 does not need to be changed because these data are one byte long.

<Data need to be swapped>

- Command and parameters (word data (16 bits))
- Input data set 2 and 3 (double-word data (32 bits))

The following table lists data to be sent at reading.

Table 38:  
Data to be sent at reading

Device	Configuration value (hex)	Byte swapping*1	Description
D700	000AH	Not required	Send data length (10 bytes)
D701	00F1H	Required	Command
D702	0001H		Read data specification (Input data set 1 reading)
D703	0001H		Read data specification (Input data set 2 reading)
D704	0001H		Read data specification (Input data set 3 reading)
D705	0000H		Read data specification (Input data set 4 not reading)

\*1 Since the QJ71E71-100 treats application data in Little Endian format, the byte-order needs to be swapped.

(3) Monitoring data with a programming tool

The read input data sets can be monitored using the device monitor function of a programming tool. For the execution results of the program, refer to Section 8.2.2 (3).

### 8.3.3 Polling mode (output data set writing)

This section describes an example of communications in polling mode. The Built-in Ethernet port QCPU writes MELSEC-WS safety controller information (e.g. logic result, CRC, and each module status) to the safety controller through communications between the WS0-GETH and the Ethernet ports of Built-in Ethernet port QCPU.

#### Write-target data

- Output data sets 1 to 5  
(In the sample program, bit 0 of Byte 0, bit 1 of Byte 1, bit 2 of Byte 2, and bit 3 of Byte 3 in output data set 1 are set to Active (High).)

(1) Parameter configuration

- (a) Configuration required for the WS0-GETH

Refer to Section 8.2.2 (1).

- (b) Configuration required for the Built-in Ethernet port QCPU

Refer to Section 8.3.2 (1).

(2) Sample program

This manual describes only devices used in sample programs and details of the processing.

For the acquisition of sample programs, please contact your local Mitsubishi representative.

- 1) Device assignment

Refer to Section 8.3.2 (2).

For details of data transmission and reception processing in a program for writing output data sets in polling mode, refer to Section 8.2.3 (2).

Please replace the BUFSND and BUFRCV instructions with the SOCSND and SOCRCV instructions.

The following table lists data to be sent at writing.

Table 39:  
Data to be sent at writing

Device	Configuration value (hex)	Byte swapping <sup>*1</sup>	Description
D800	003E <sub>H</sub>	Not required	Send data length (62 bytes)
D801	00F2 <sub>H</sub>	Required	Command
D802	000A <sub>H</sub>		Write data specification (Output data set 1 writing (10 bytes))
D803	000A <sub>H</sub>		Write data specification (Output data set 2 writing (10 bytes))
D804	000A <sub>H</sub>		Write data specification (Output data set 3 writing (10 bytes))
D805	000A <sub>H</sub>		Write data specification (Output data set 4 writing (10 bytes))
D806	000A <sub>H</sub>		Write data specification (Output data set 5 writing (10 bytes))
D807	Any	Not required	Output data set 1: Byte 0 and Byte 1
D808			Output data set 1: Byte 2 and Byte 3
D809			Output data set 1: Byte 4 and Byte 5
D810			Output data set 1: Byte 6 and Byte 7
D811			Output data set 1: Byte 8 and Byte 9
D812 to D816	Any	Not required	Output data set 2
D817 to D821	Any	Not required	Output data set 3
D822 to D826	Any	Not required	Output data set 4
D827 to D831	Any	Not required	Output data set 5

\*1 Since the QJ71E71-100 treats application data in Little Endian format, the byte-order needs to be swapped.

The following table shows an example of the execution result of the program for writing output data sets in polling mode.

Table 40:  
Execution result example in polling mode (output data set writing)

Device	Storage value (hex)	Byte swapping <sup>*1</sup>	Description
D900	0004 <sub>H</sub>	Not required	Receive data length
D901	2F00 <sub>H</sub>	Required	Response command
D902	0000 <sub>H</sub>		Status 0: Successful 1: Error

\*1 Since the QJ71E71-100 treats application data in Little Endian format, the byte-order needs to be swapped.



3) Configuring TCP/IP

Configure the communication mode (auto update mode) in the **TCP/IP configuration** dialog.

Select **Enable auto update and COS** and **Connect to** for socket 1.  
(In the sample program, the WS0-GETH is configured as master.)

Figure 67:  
TCP/IP configuration dialog

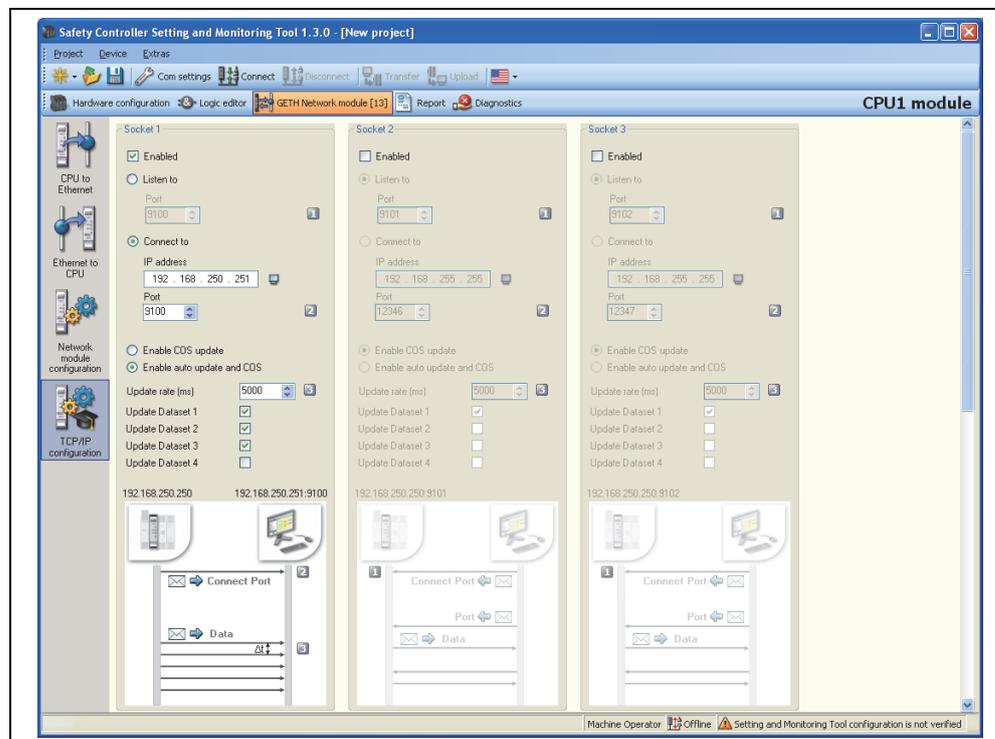


Table 41:  
TCP/IP configuration parameters

Option	Configuration value (Listen to)	Configuration value (Connect to)
Enabled	Checked	
Listen to	Selected	Deselected
Port	9100 (238C <sub>H</sub> ) (default)	-
Connect to	Deselected	Selected
IP address	-	192.168.250.251
Port	-	9100 (238C <sub>H</sub> )
Enable COS update/Enable auto update and COS	Enable auto update and COS	
Update rate (ms)	5000 (default)	
Update Dataset 1	Checked	
Update Dataset 2	Checked	
Update Dataset 3	Checked	
Update Dataset 4	Unchecked	

To switch the communication mode to the polling mode, activate the **Enable COS update** radio button.

4) Editing data sets

Refer to Section 8.2.2 (1).

## (b) Configuration required for the Built-in Ethernet port QCPU

Configuration examples of network parameters required for communications in auto update mode are shown below. Configure these parameters using a programming tool.

1) Configuring PLC parameters – **Built-in Ethernet port**

Select **Built-in Ethernet port** on the PLC parameter window and configure the following options.

Figure 68:  
Built-in Ethernet port  
dialog

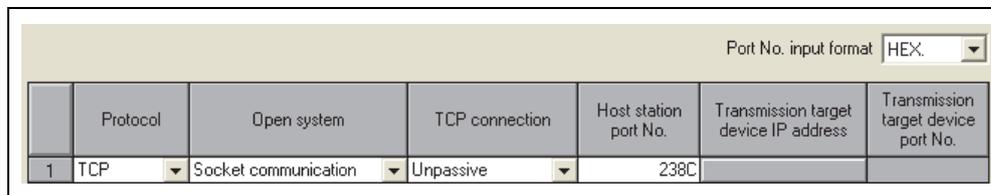
The screenshot shows the 'Built-in Ethernet port' configuration dialog. At the top, there is a tabbed menu with the following tabs: PLC name, PLC system, PLC file, PLC RAS(1), PLC RAS(2), Device, Program, Boot file, SFC, I/O assignment, and Built-in Ethernet port. The 'Built-in Ethernet port' tab is active. The main area contains the following settings:

- IP address section:**
  - Input format: DEC (dropdown)
  - IP address: 192 | 168 | 250 | 251
  - Subnet mask pattern: 255 | 255 | 0 | 0
  - Default router IP address: 192 | 168 | 250 | 255
- Buttons:** Open settings, FTP settings, Time settings.
- Status:** Set if it is needed (Default / Changed)
- Communication data code:**
  - Binary code
  - ASCII code
- Checkboxes:**
  - Enable online change (FTP, MC protocol)
  - Disable direct connection to MELSOFT
  - Do not respond to search for CPU (Built-in Ethernet port) on network

2) PLC parameter – **Open settings**

Click the **Open settings** button to open the configuration dialog and configure the options as follows.

Figure 69:  
PLC parameter – Open settings



The **TCP connection** setting differs depending on the connection open status.

- When connection is opened by the Built-in Ethernet port QCPU, set **Active**.
- When connection is opened by the WS0-GETH, set **Unpassive**.

Table 42:  
PLC parameter – Open settings

Option	Configuration value (Active)	Configuration value (Unpassive)
Protocol	TCP	
Open system	Socket communication	
TCP connection	Active	Unpassive
Host station port No.	9100 (238CH)	
Transmission target device IP address	192.168.250.250	-
Transmission target device port No. (hex)	9100 (238CH)	-

## (2) Sample program

This manual describes only devices used in sample programs and details of the processing.

For the acquisition of sample programs, please contact your local Mitsubishi representative.

## 1) Device assignment

The following tables list devices used in the sample program.

Table 43: Devices used in the sample program

W device	
W0 to W18	Input data set 1 storage area
W20 to W2F	Input data set 2 storage area
W30 to W4D	Input data set 3 storage area

M device		D device	
M500	Receive execution	D2	SOCRCV instruction failure error code
M501	SOCRCV instruction success device	D500 and D501	SOCRCV instruction control data
M502	SOCRCV instruction failure device	D520	Response command check device
M510	SOCRCV instruction normal completion	D900 to D976	SOCRCV instruction receive data area
M511	SOCRCV instruction abnormal completion	-	-
M520	Response command verification normal completion	-	-
M521	Data size error device	-	-

C device*1		
C20	SOCRCV instruction	Number of executions
C21		Number of successes
C22		Number of failures
C31	Verification check	Number of successes
C32		Number of failures

\*1 C device is used to check the operation of the communication program.

- Note**
- If the Built-in Ethernet port QCPU stops during communications in auto update mode, the SOCRCV instruction will not be executed even when data are transmitted from the WS0-GETH. This may cause the receive buffer of the Built-in Ethernet port QCPU to overflow and data transmission of the WS0-GETH to stop.
  - If the scan time of the Built-in Ethernet port QCPU is longer than the transmission interval of the WS0-GETH, update of received data delays. For this reason, configure the auto update interval much longer than the scan time.

For details of data reception processing in a program for receiving input data sets in auto update mode, refer to Section 8.2.4 (2).

Replace the BUFSND and BUFRCV instructions with the SOCSND and SOCRCV instructions.

Since the Built-in Ethernet port QCPU treats application data in Little Endian format, the byte-order needs to be swapped. Note that the byte order of the input data set 1 and output data set 1 does not need to be changed because these data are one byte long.

<Data need to be swapped>

- Command and parameters (word data (16 bits))
- Input data sets 2 and 3 (double-word data (32 bits))

### (3) Monitoring data with a programming tool

The read input data sets can be monitored using the device monitor function of a programming tool. For the execution results of the program for auto update mode, refer to Section 8.2.2 (3).

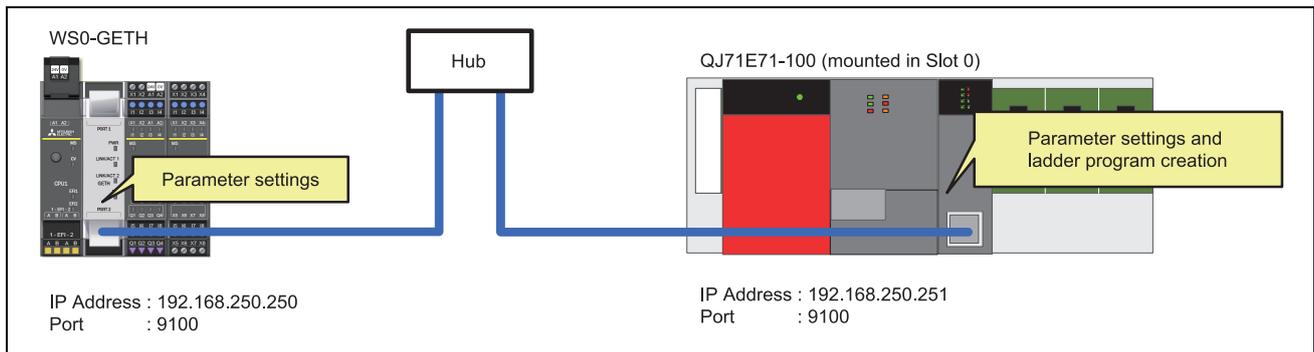
## 8.4 Communications with the QJ71E71-100 (QSCPU)

This section describes parameter configuration examples and sample programs for communications between the WS0-GETH and the QJ71E71-100 connected to the QSCPU.

### 8.4.1 System configuration

The following is the system configuration used for sample programs.

Figure 70: System configuration example



### 8.4.2 Polling mode (input data set reading)

This section describes an example of communications in polling mode.

The QSCPU reads MELSEC-WS safety controller information (e.g. logic result, CRC, and each module status) from the safety controller through communications between the WS0-GETH and the QJ71E71-100.

#### Read-target data

- Logic result 0 (input data set 1)
- CRC (input data set 2)
- Each module status (input data set 3)

#### (1) Parameter configuration

##### (a) Configuration required for the WS0-GETH

Refer to Section 8.2.2 (1).

##### (b) Configuration required for the QJ71E71-100

Refer to Section 8.2.2 (1).

(2) Sample program

This manual describes only devices used in sample programs and details of the processing.

For the acquisition of sample programs, please contact your local Mitsubishi representative.

1) Device assignment [for reading/writing]

The following tables list devices used in the sample program.

Table 44: Devices used in the sample programs

W device	
W0 to W18	Input data set 1 storage area
W20 to W2F	Input data set 2 storage area
W30 to W4D	Input data set 3 storage area

M device		D device	
M300	Open execution	D0	OPEN instruction abnormal completion error code
M301	OPEN instruction success device		
M302	OPEN instruction failure device	D1	BUFSND instruction abnormal completion error code
M310	OPEN instruction normal completion		
M311	OPEN instruction abnormal completion	D2	BUFRCV instruction abnormal completion error code
M400	Send execution		
M401	BUFSND instruction success device	D3	CLOSE instruction abnormal completion error code
M402	BUFSND instruction failure device		
M410	BUFSND instruction normal completion	D300 to D309	OPEN instruction control data
M411	BUFSND instruction abnormal completion	D400 and D401	BUFSND instruction control data
		D500 and D501	BUFRCV instruction control data
M420	Command in execution	D520	Response command check device (reading)
M500	Receive execution	D521	Response command check device (writing)
M501	BUFRCV instruction success device	D600 and D601	CLOSE instruction control data
M502	BUFRCV instruction failure device	D700 to D705	BUFSND instruction send data area (reading)
M510	BUFRCV instruction normal completion	D800 to D831	BUFSND instruction send data area (writing)
M511	BUFRCV instruction abnormal completion	D900 to D976	BUFRCV instruction receive data area (reading and writing)
M520	Response command verification normal completion	<b>X device</b>	
M521	Data size error device	X0	Fixed buffer reception status signal
M601	CLOSE instruction success device	X10	Open complete signal
M602	CLOSE instruction failure device	X19	Initial processing normal completion signal
M610	CLOSE instruction normal completion	-	-
M611	CLOSE instruction abnormal completion	-	-
M1000 to M1023	Device area used for byte swapping	-	-
M2000	OPEN execution (1PLS)	-	-
M2001	OPEN request signal	-	-

C device*1		
C0	Open instruction	Number of executions
C1		Number of successes
C2		Number of failures
C10	BUFSND instruction	Number of executions
C11		Number of successes
C12		Number of failures
C20	BUFRCV instruction	Number of executions
C21		Number of successes
C22		Number of failures
C31	Verification check	Number of successes
C32		Number of failures
C33		Number of response errors (writing)
C40	Close instruction	Number of executions
C41		Number of successes
C42		Number of failures

\*1 C device is used to check the operation of the communication program.

For details of data transmission and reception processing in a program for reading input data sets in polling mode, refer to Section 8.2.2 (2).

Since the QJ71E71-100 treats application data in Little Endian format, the byte-order needs to be swapped. Note that the byte order of the input data set 1 and output data set 1 does not need to be changed because these data are one byte long.

<Data need to be swapped>

- Command and parameters (word data (16 bits))
- Input data set 2 and 3 (double-word data (32 bits))

The following table lists data to be sent at reading.

Table 45:  
Data to be sent at reading

Device	Configuration value (hex)	Byte swapping <sup>*1</sup>	Description
D700	000A <sub>H</sub>	Not required	Send data length (10 bytes)
D701	00F1 <sub>H</sub>	Required	Command
D702	0001 <sub>H</sub>		Read data specification (Input data set 1 reading)
D703	0001 <sub>H</sub>		Read data specification (Input data set 2 reading)
D704	0001 <sub>H</sub>		Read data specification (Input data set 3 reading)
D705	0000 <sub>H</sub>		Read data specification (Input data set 4 not reading)

\*1 Since the QJ71E71-100 treats application data in Little Endian format, the byte-order needs to be swapped.

Since the QSCPU does not support the SWAP instruction, the upper and lower bytes are swapped in the following way.

- 1) Store the swap-target data into the areas (M1000 to M1015).
- 2) Move the lower-byte data in the areas (M1000 to M1007) to the other areas (M1016 to M1023).
- 3) Return the data in the areas (M1008 to M1023) to the original device.

(3) Monitoring data with a programming tool

The read input data sets can be monitored using the device monitor function of a programming tool. For the execution results of the program, refer to Section 8.2.2 (3).

### 8.4.3 Polling mode (output data set writing)

This section describes an example of communications in polling mode.

The QSCPU writes MELSEC-WS safety controller information (e.g. logic result, CRC, and each module status) to the safety controller through communications between the WS0-GETH and the QJ71E71-100.

#### Write-target data

- Output data sets 1 to 5

(In the sample program, bit 0 of Byte 0, bit 1 of Byte 1, bit 2 of Byte 2, and bit 3 of Byte 3 in output data set 1 are set to Active (High).)

(1) Parameter configuration

(a) Configuration required for the WS0-GETH

Refer to Section 8.2.2 (1).

(b) Configuration required for the QJ71E71-100

Refer to Section 8.2.2 (1).

(2) Sample program

This manual describes only devices used in sample programs and details of the processing.

For the acquisition of sample programs, please contact your local Mitsubishi representative.

1) Device assignment

Refer to Section 8.4.2 (2).

For details of data transmission and reception processing in a program for writing output data sets in polling mode, refer to Section 8.2.3 (2).

The following table lists data to be sent at writing.

Table 46:  
Data to be sent at writing

Device	Configuration value (hex)	Byte swapping*1	Description
D800	003EH	Not required	Send data length (62 bytes)
D801	00F2H	Required	Command
D802	000AH		Write data specification (Output data set 1 writing (10 bytes))
D803	000AH		Write data specification (Output data set 2 writing (10 bytes))
D804	000AH		Write data specification (Output data set 3 writing (10 bytes))
D805	000AH		Write data specification (Output data set 4 writing (10 bytes))
D806	000AH		Write data specification (Output data set 5 writing (10 bytes))
D807	Any	Not required	Output data set 1: Byte 0 and Byte 1
D808			Output data set 1: Byte 2 and Byte 3
D809			Output data set 1: Byte 4 and Byte 5
D810			Output data set 1: Byte 6 and Byte 7
D811			Output data set 1: Byte 8 and Byte 9
D812 to D816	Any	Not required	Output data set 2
D817 to D821	Any	Not required	Output data set 3
D822 to D826	Any	Not required	Output data set 4
D827 to D831	Any	Not required	Output data set 5

\*1 Since the QJ71E71-100 treats application data in Little Endian format, the byte-order needs to be swapped.

The following table shows an example of the execution result of the program for writing output data sets in polling mode.

Table 47:  
Execution result  
example in polling  
mode (output data set  
writing)

Device	Storage value (hex)	Byte swapping <sup>*1</sup>	Description
D900	0004 <sub>H</sub>	Not required	Receive data length
D901	2F00 <sub>H</sub>	Required	Response command
D902	0000 <sub>H</sub>		Status 0: Successful 1: Error

\*1 Since the QJ71E71-100 treats application data in Little Endian format, the byte-order needs to be swapped.

### (3) Monitoring data with Setting and Monitoring Tool

The written output data sets can be monitored in the **Ethernet to CPU** dialog of Setting and Monitoring Tool. For the execution result of the program, refer to Section 8.2.3 (3).

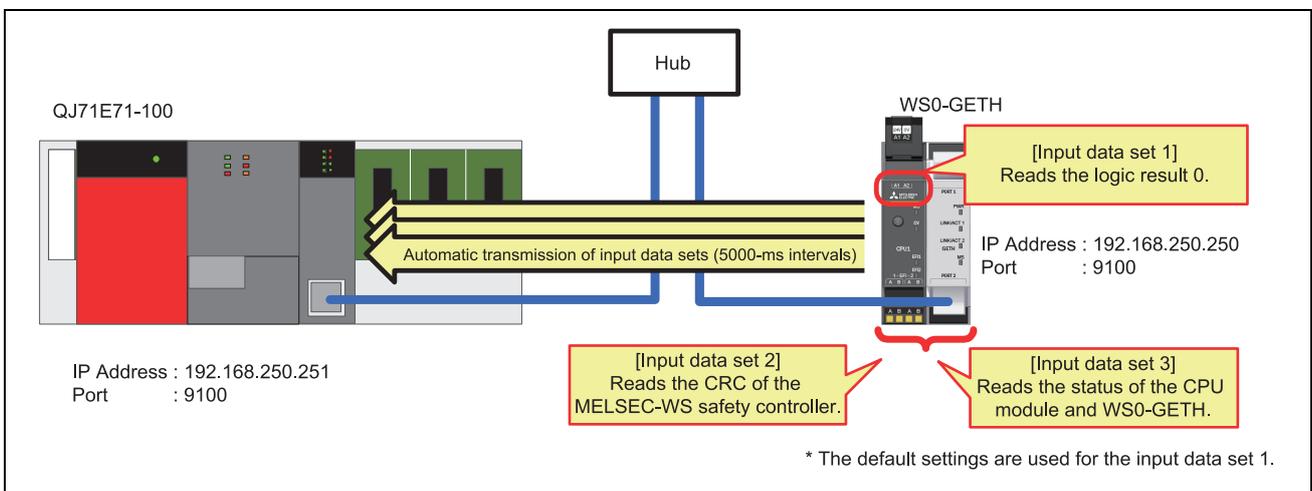
**8.4.4 Auto update mode**

This section describes an example of communications in auto update mode. The QSCPU reads MELSEC-WS safety controller information (e.g. logic result, CRC, and each module status) from the safety controller through communications between the WS0-GETH and the QJ71E71-100. The WS0-GETH automatically sends input data sets to the QJ71E71-100 at 5000-ms intervals.

Read-target data

- Logic result 0 (input data set 1)
- CRC (input data set 2)
- Each module status (input data set 3)

Figure 71: System configuration example



(1) Parameter configuration

(a) Configuration required for the WS0-GETH

Configuration examples of network parameters required for communications in auto update mode are shown below. Configure these parameters using Setting and Monitoring Tool.

The WS0-GETH requires no program for communications.

1) Creating a hardware configuration

Refer to Section 8.2.2 (1).

2) Assigning an IP address

Refer to Section 8.2.2 (1).

3) Configuring TCP/IP

Refer to Section 8.2.4 (1).

4) Editing data sets

Refer to Section 8.2.2 (1).

## (b) Configuration required for the QJ71E71-100

Configuration examples of network parameters required for communications in auto update mode are shown below. Configure these parameters using a programming tool.

## 1) Configuring network parameters

Refer to Section 8.2.2 (1).

2) Network parameter – **Operational settings**

Refer to Section 8.2.4 (1).

3) Network parameter – **Open settings**

Refer to Section 8.2.4 (1).

4) Network parameter – **Initial settings**

Refer to Section 8.2.4 (1).

## (2) Sample program

This manual describes only devices used in sample programs and details of the processing.

For the acquisition of sample programs, please contact your local Mitsubishi representative.

## 1) Device assignment

The following tables list devices used in the sample program.

Table 48: Devices used in the sample program

W device	
W0 to W18	Input data set 1 storage area
W20 to W2F	Input data set 2 storage area
W30 to W4D	Input data set 3 storage area

M device		D device	
M500	Receive execution	D2	BUFRCV instruction failure error code
M501	BUFRCV instruction success device	D500 and D501	BUFRCV instruction control data
M502	BUFRCV instruction failure device	D520	Response command check device
M510	BUFRCV instruction normal completion	D900 to D976	BUFRCV instruction receive data area
M511	BUFRCV instruction abnormal completion	X device	
M520	Response command verification normal completion	X0	Fixed buffer reception status signal
M521	Data size error device	X10	Open complete signal
M1000 to M1023	Device area used for byte swapping	X19	Initial processing normal completion signal

C device*1		
C20	BUFRCV instruction	Number of executions
C21		Number of successes
C22		Number of failures
C31	Verification check	Number of successes
C32		Number of failures

\*1 C device is used to check the operation of the communication program.

- Note**
- If the QSCPU connected to the QJ71E71-100 stops during communications in auto update mode, the BUFRCV instruction will not be executed even when data are transmitted from the WS0-GETH. This may cause the receive buffer of the QJ71E71-100 to overflow and data transmission of the WS0-GETH to stop.
  - If the scan time of the QSCPU connected to the QJ71E71-100 is longer than the transmission interval of the WS0-GETH, update of received data delays. For this reason, configure the auto update interval much longer than the scan time.

For details of data reception processing in a program for receiving input data sets in auto update mode, refer to Section 8.2.4 (2).

Since the QJ71E71-100 treats application data in Little Endian format, the byte-order needs to be swapped. Note that the byte order of the input data set 1 and output data set 1 does not need to be changed because these data are one byte long.

<Data need to be swapped>

- Command and parameters (word data (16 bits))
- Input data sets 2 and 3 (double-word data (32 bits))

### (3) Monitoring data with a programming tool

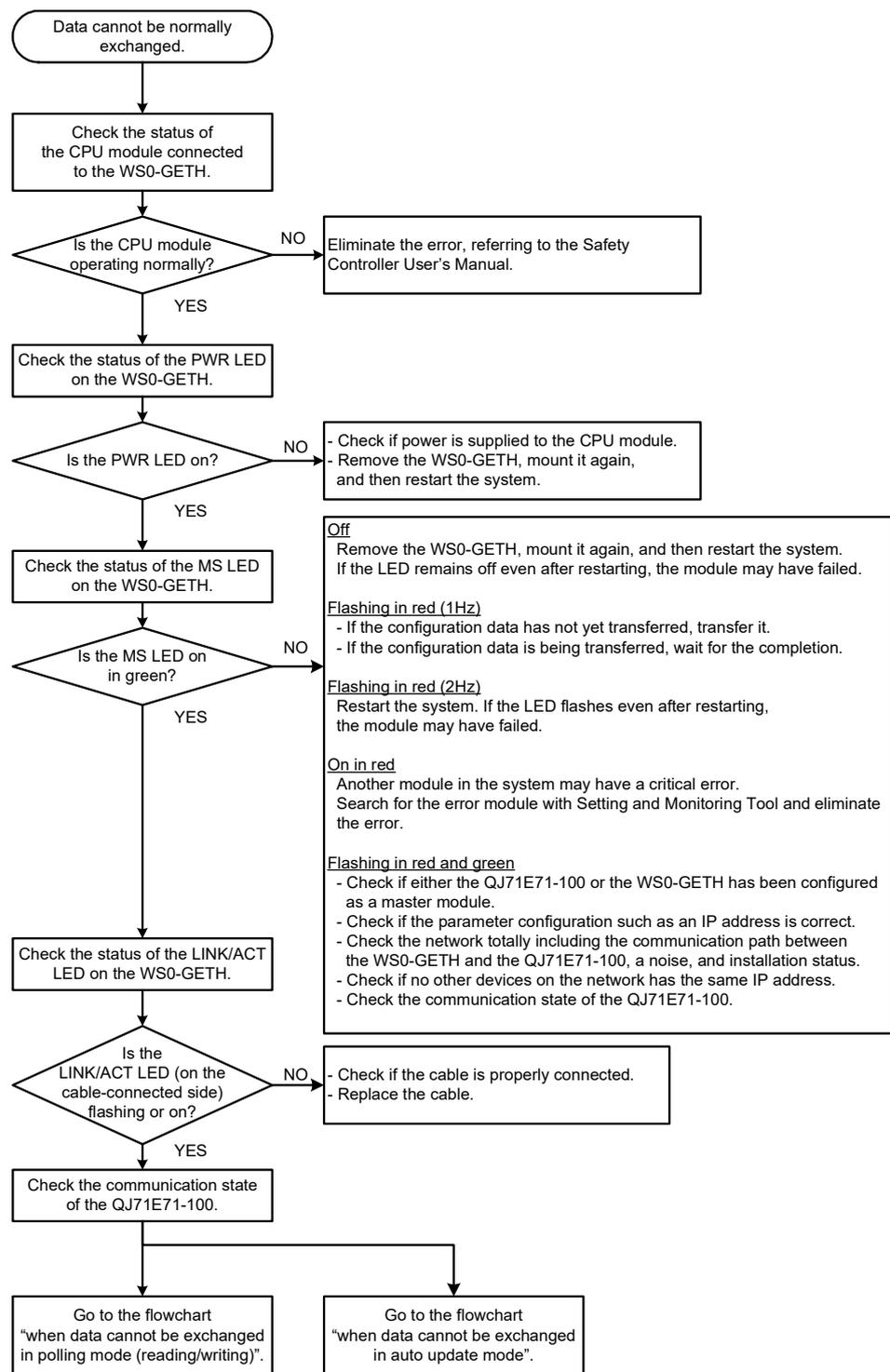
The read input data sets can be monitored using the device monitor function of a programming tool. For the execution results of the program for auto update mode, refer to Section 8.2.2 (3).

# 9. Troubleshooting

## 9.1 When data cannot be normally exchanged

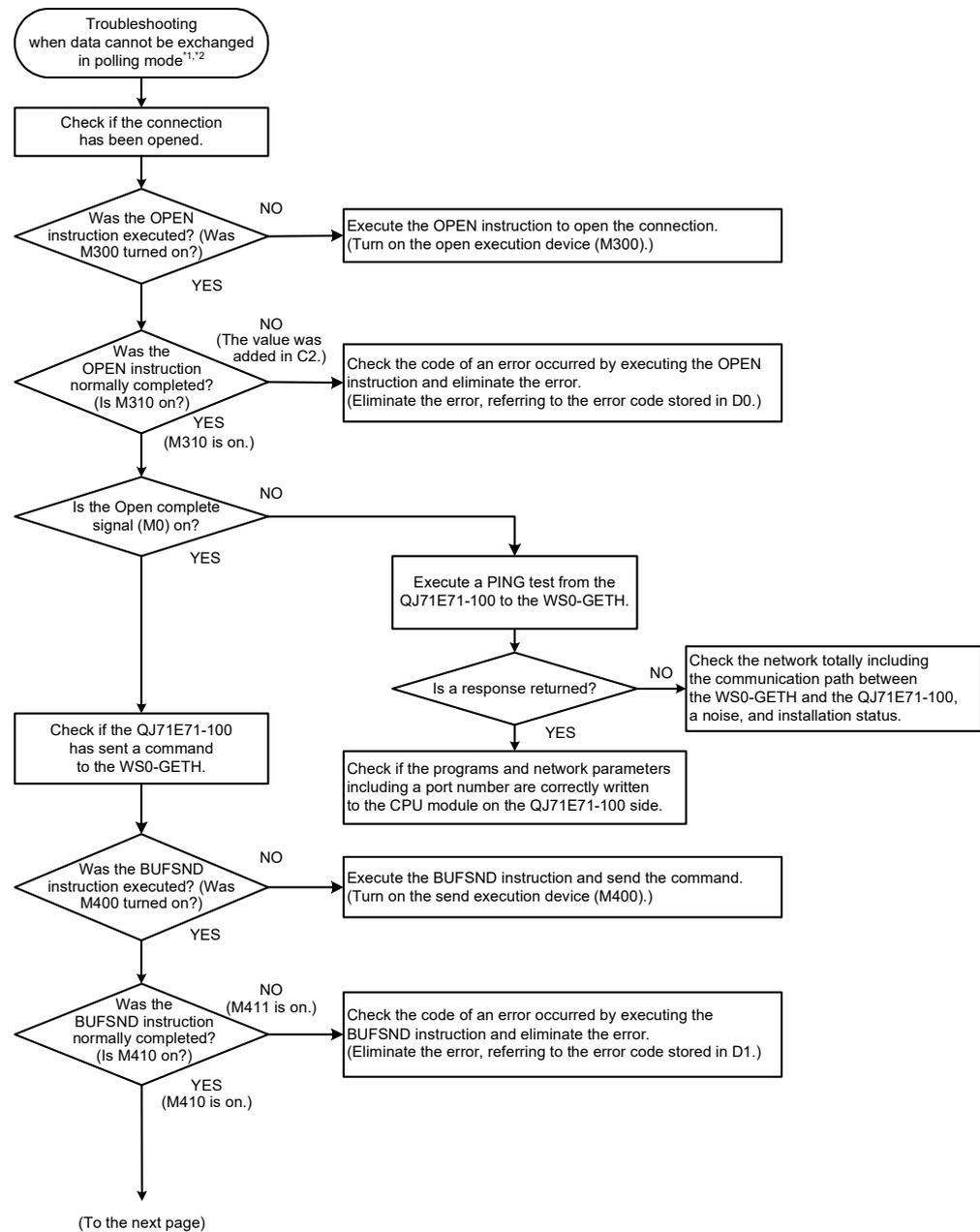
Refer to the following flowchart when the WS0-GETH cannot normally exchange data with the communication-target module.

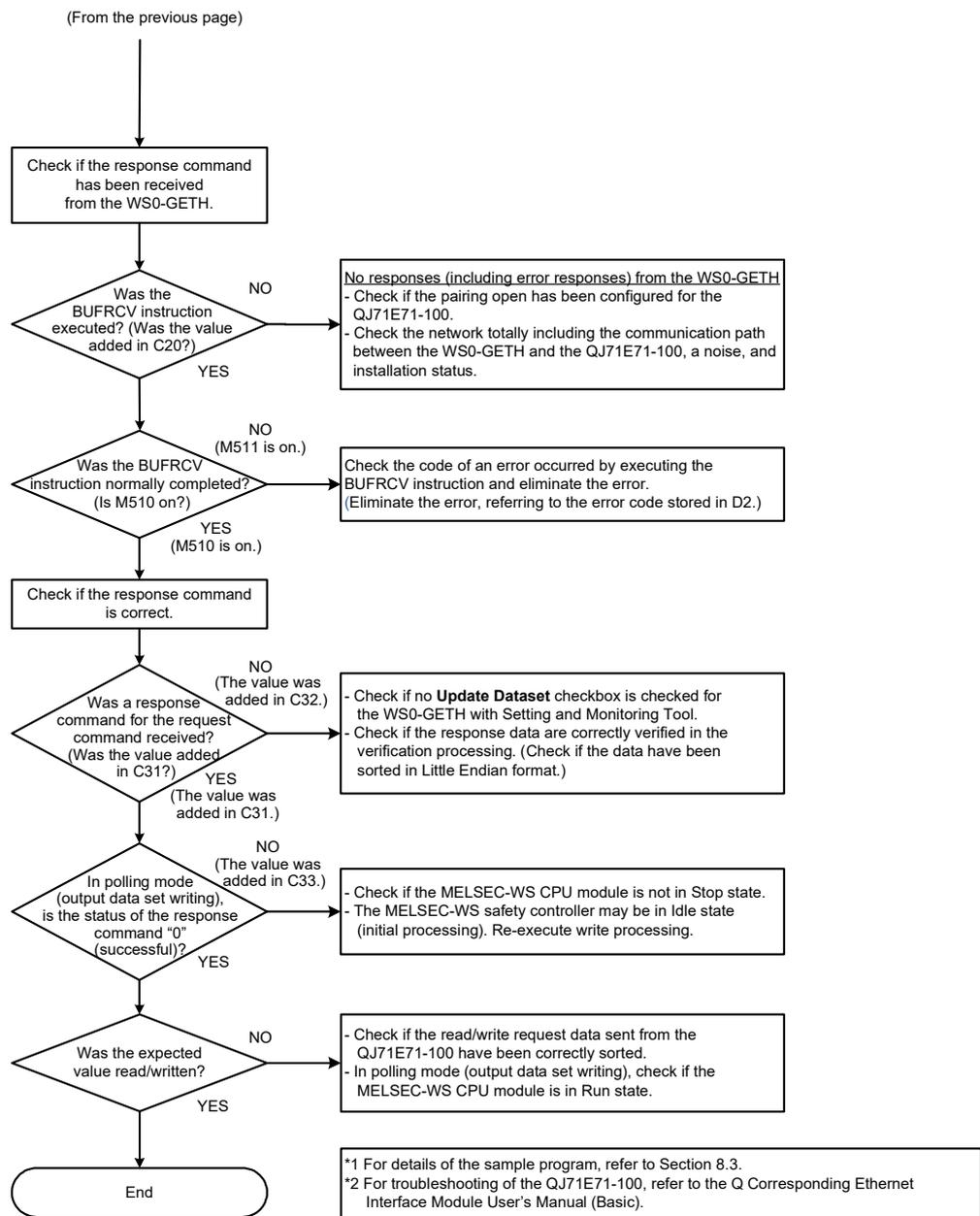
Figure 72:  
Troubleshooting  
flowchart when data  
cannot be normally  
exchanged



## 9.2 When data cannot be exchanged in polling mode (reading/writing)

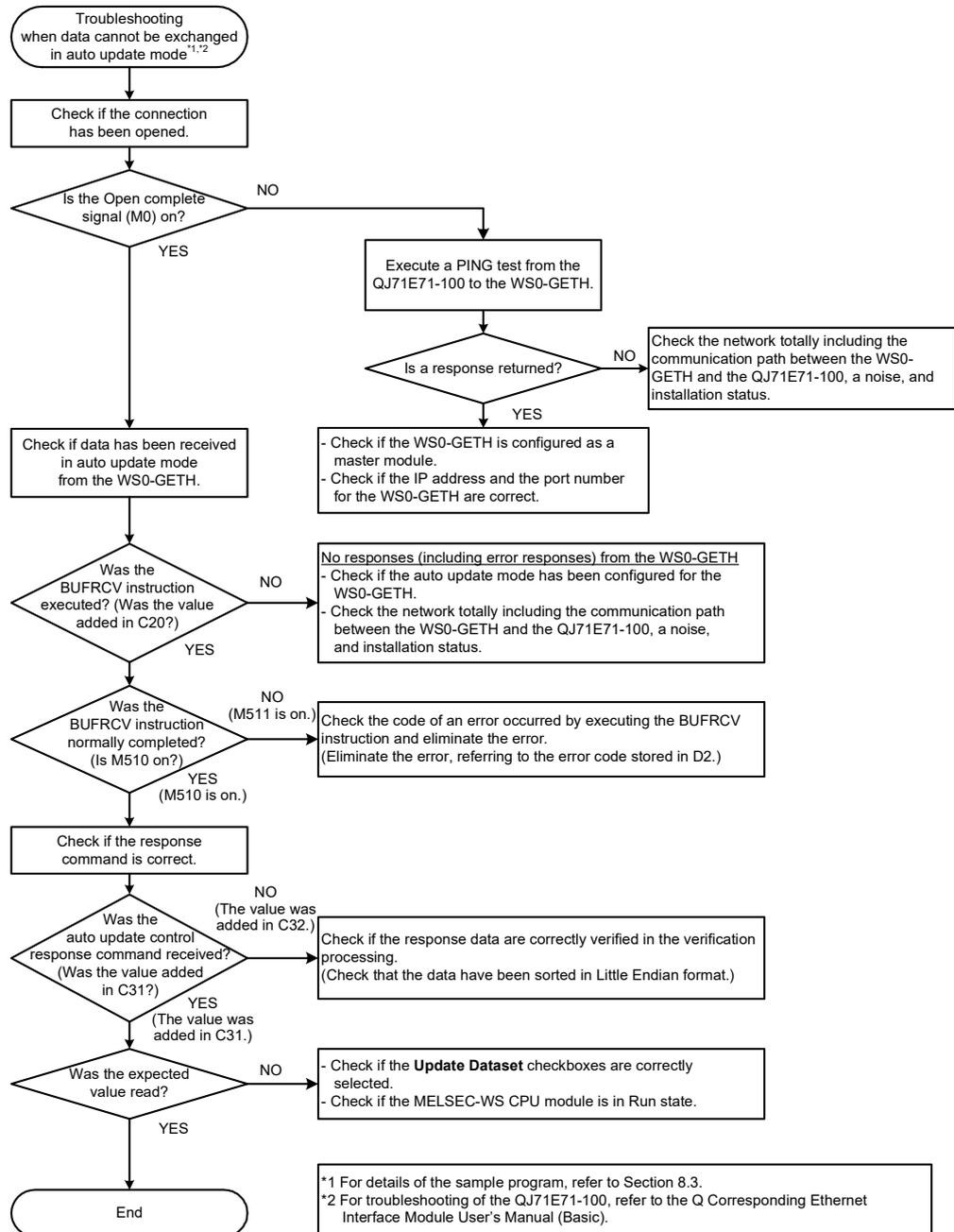
Figure 73:  
Troubleshooting  
flowchart when data  
cannot be exchanged in  
polling mode





### 9.3 When data cannot be exchanged in auto update mode

Figure 74:  
Troubleshooting  
flowchart when data  
cannot be exchanged in  
auto update mode



## 10. Annex

### 10.1 SICK contact

More representatives and agencies in all major industrial nations at [www.sick.com](http://www.sick.com)

#### **Australia**

Phone +61 3 9497 4100  
1800 33 48 02 – tollfree  
E-Mail [sales@sick.com.au](mailto:sales@sick.com.au)

#### **Belgium/Luxembourg**

Phone +32 (0)2 466 55 66  
E-Mail [info@sick.be](mailto:info@sick.be)

#### **Brasil**

Phone +55 11 3215-4900  
E-Mail [sac@sick.com.br](mailto:sac@sick.com.br)

#### **Ceská Republika**

Phone +420 2 57 91 18 50  
E-Mail [sick@sick.cz](mailto:sick@sick.cz)

#### **China**

Phone +852-2763 6966  
E-Mail [ghk@sick.com.hk](mailto:ghk@sick.com.hk)

#### **Danmark**

Phone +45 45 82 64 00  
E-Mail [sick@sick.dk](mailto:sick@sick.dk)

#### **Deutschland**

Phone +49 211 5301-260  
E-Mail [info@sick.de](mailto:info@sick.de)

#### **España**

Phone +34 93 480 31 00  
E-Mail [info@sick.es](mailto:info@sick.es)

#### **France**

Phone +33 1 64 62 35 00  
E-Mail [info@sick.fr](mailto:info@sick.fr)

#### **Great Britain**

Phone +44 (0)1727 831121  
E-Mail [info@sick.co.uk](mailto:info@sick.co.uk)

#### **India**

Phone +91-22-4033 8333  
E-Mail [info@sick-india.com](mailto:info@sick-india.com)

#### **Israel**

Phone +972-4-9990590  
E-Mail [info@sick-sensors.com](mailto:info@sick-sensors.com)

#### **Italia**

Phone +39 02 27 43 41  
E-Mail [info@sick.it](mailto:info@sick.it)

#### **Japan**

Phone +81 (0)3 3358 1341  
E-Mail [support@sick.jp](mailto:support@sick.jp)

#### **Nederlands**

Phone +31 (0)30 229 25 44  
E-Mail [info@sick.nl](mailto:info@sick.nl)

#### **Norge**

Phone +47 67 81 50 00  
E-Mail [austefjord@sick.no](mailto:austefjord@sick.no)

#### **Österreich**

Phone +43 (0)22 36 62 28 8-0  
E-Mail [office@sick.at](mailto:office@sick.at)

#### **Polska**

Phone +48 22 837 40 50  
E-Mail [info@sick.pl](mailto:info@sick.pl)

#### **Republic of Korea**

Phone +82-2 786 6321/4  
E-Mail [kang@sickkorea.net](mailto:kang@sickkorea.net)

#### **Republika Slovenija**

Phone +386 (0)1-47 69 990  
E-Mail [office@sick.si](mailto:office@sick.si)

#### **Romania**

Phone +40356171120  
E-Mail [office@sick.ro](mailto:office@sick.ro)

#### **Russia**

Phone +7-495-775-05-34  
E-Mail [info@sick-automation.ru](mailto:info@sick-automation.ru)

#### **Schweiz**

Phone +41 41 619 29 39  
E-Mail [contact@sick.ch](mailto:contact@sick.ch)

#### **Singapore**

Phone +65 6744 3732  
E-Mail [admin@sicksgp.com.sg](mailto:admin@sicksgp.com.sg)

#### **Suomi**

Phone +358-9-25 15 800  
E-Mail [sick@sick.fi](mailto:sick@sick.fi)

#### **Sverige**

Phone +46 10 110 10 00  
E-Mail [info@sick.se](mailto:info@sick.se)

#### **Taiwan**

Phone +886 2 2375 -6288  
E-Mail [sales@sick.com.tw](mailto:sales@sick.com.tw)

#### **Türkiye**

Phone +90 216 587 74 00  
E-Mail [info@sick.com.tr](mailto:info@sick.com.tr)

#### **United Arab Emirates**

Phone + 971 4 8865 878  
E-Mail [info@sick.ae](mailto:info@sick.ae)

#### **USA/Canada/México**

Phone +1(952) 941- 6780  
1800-325-7425 – tollfree  
E-Mail [info@sickusa.com](mailto:info@sickusa.com)



## **WARRANTY**

### **1. Limited Warranty and Product Support.**

- a. Mitsubishi Electric Company ("MELCO") warrants that for a period of eighteen (18) months after date of delivery from the point of manufacture or one year from date of Customer's purchase, whichever is less, Mitsubishi Safety Controller (the "Products") will be free from defects in material and workmanship.
- b. At MELCO's option, for those Products MELCO determines are not as warranted, MELCO shall either repair or replace them or issue a credit or return the purchase price paid for them.
- c. For this warranty to apply:
  - (1) Customer shall give MELCO (i) notice of a warranty claim to MELCO and the authorized dealer or distributor from whom the Products were purchased, (ii) the notice shall describe in reasonable details the warranty problem, (iii) the notice shall be provided promptly and in no event later than thirty (30) days after the Customer knows or has reason to believe that Products are not as warranted, and (iv) in any event, the notice must be given within the warranty period;
  - (2) Customer shall cooperate with MELCO and MELCO's representatives in MELCO's investigation of the warranty claim, including preserving evidence of the claim and its causes, meaningfully responding to MELCO's questions and investigation of the problem, grant MELCO access to witnesses, personnel, documents, physical evidence and records concerning the warranty problem, and allow MELCO to examine and test the Products in question offsite or at the premises where they are installed or used; and
  - (3) If MELCO requests, Customer shall remove Products it claims are defective and ship them to MELCO or MELCO's authorized representative for examination and, if found defective, for repair or replacement. The costs of removal, shipment to and from MELCO's designated examination point, and reinstallation of repaired or replaced Products shall be at Customer's expense.
  - (4) If Customer requests and MELCO agrees to effect repairs onsite at any domestic or overseas location, the Customer will pay for the costs of sending repair personnel and shipping parts. MELCO is not responsible for any re-commissioning, maintenance, or testing on-site that involves repairs or replacing of the Products.
- d. Repairs of Products located outside of Japan are accepted by MELCO's local authorized service facility centers ("FA Centers"). Terms and conditions on which each FA Center offers repair services for Products that are out of warranty or not covered by MELCO's limited warranty may vary.
- e. Subject to availability of spare parts, MELCO will offer Product repair services for (4) years after each Product model or line is discontinued, at MELCO's or its FA Centers' rates and charges and standard terms in effect at the time of repair. MELCO usually produces and retains sufficient spare parts for repairs of its Products for a period of four (4) years after production is discontinued.
- f. MELCO generally announces discontinuation of Products through MELCO's Technical Bulletins. Products discontinued and repair parts for them may not be available after their production is discontinued.

### **2. Limits of Warranties.**

- a. MELCO does not warrant or guarantee the design, specify, manufacture, construction or installation of the materials, construction criteria, functionality, use, properties or other characteristics of the equipment, systems, or production lines into which the Products may be incorporated, including any safety, fail-safe and shut down systems using the Products.
- b. MELCO is not responsible for determining the suitability of the Products for their intended purpose and use, including determining if the Products provide appropriate safety margins and redundancies for the applications, equipment or systems into which they are incorporated.
- c. Customer acknowledges that qualified and experienced personnel are required to determine the suitability, application, design, construction and proper installation and integration of the Products. MELCO does not supply such personnel.
- d. MELCO is not responsible for designing and conducting tests to determine that the Product functions appropriately and meets application standards and requirements as installed or incorporated into the end-user's equipment, production lines or systems.
- e. MELCO does not warrant any Product:
  - (1) repaired or altered by persons other than MELCO or its authorized engineers or FA Centers;
  - (2) subjected to negligence, carelessness, accident, misuse, or damage;
  - (3) improperly stored, handled, installed or maintained;
  - (4) integrated or used in connection with improperly designed, incompatible or defective hardware or software;
  - (5) that fails because consumable parts such as relay, batteries, backlights, or fuses were not tested, serviced or replaced;
  - (6) operated or used with equipment, production lines or systems that do not meet applicable and commensurate legal, safety and industry-accepted standards;
  - (7) operated or used in abnormal applications;
  - (8) installed, operated or used in contravention of instructions, precautions or warnings contained in MELCO's user, instruction and/or safety manuals, technical bulletins and guidelines for the Products;
  - (9) used with obsolete technologies or technologies not fully tested and widely accepted and in use at the time of the Product's manufacture;
  - (10) subjected to excessive heat or moisture, abnormal voltages, shock, excessive vibration, physical damage or other improper environment; or
  - (11) damaged or malfunctioning due to Acts of God, fires, acts of vandals, criminals or terrorists, communication or power failures, or any other cause or failure that results from circumstances beyond MELCO's control.
- f. All Product information and specifications contained on MELCO's website and in catalogs, manuals, or technical information materials provided by MELCO are subject to change without prior notice.
- g. The Product information and statements contained on MELCO's website and in catalogs, manuals, technical bulletins or other materials provided by MELCO are provided as a guide for Customer's use. They do not constitute warranties and are not incorporated in the contract of sale for the Products.

- h. These terms and conditions constitute the entire agreement between Customer and MELCO with respect to warranties, remedies and damages and supersede any other understandings, whether written or oral, between the parties. Customer expressly acknowledges that any representations or statements made by MELCO or others concerning the Products outside these terms are not part of the basis of the bargain between the parties and are not factored into the pricing of the Products.
- i. THE WARRANTIES AND REMEDIES SET FORTH IN THESE TERMS ARE THE EXCLUSIVE AND ONLY WARRANTIES AND REMEDIES THAT APPLY TO THE PRODUCTS.
- j. MELCO DISCLAIMS THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

### **3. Limits on Damages.**

- a. MELCO'S MAXIMUM CUMULATIVE LIABILITY BASED ON ANY CLAIMS FOR BREACH OF WARRANTY OR CONTRACT, NEGLIGENCE, STRICT TORT LIABILITY OR OTHER THEORIES OF RECOVERY REGARDING THE SALE, REPAIR, REPLACEMENT, DELIVERY, PERFORMANCE, CONDITION, SUITABILITY, COMPLIANCE, OR OTHER ASPECTS OF THE PRODUCTS OR THEIR SALE, INSTALLATION OR USE SHALL BE LIMITED TO THE PRICE PAID FOR PRODUCTS NOT AS WARRANTED.
- b. Although MELCO has obtained the certification for Product's compliance to the international safety standards IEC61508 and ISO13849-1 from TUV Rheinland, this fact does not guarantee that Product will be free from any malfunction or failure. The user of this Product shall comply with any and all applicable safety standard, regulation or law and take appropriate safety measures for the system in which the Product is installed or used and shall take the second or third safety measures other than the Product. MELCO is not liable for damages that could have been prevented by compliance with any applicable safety standard, regulation or law.
- c. MELCO prohibits the use of Products with or in any application involving power plants, trains, railway systems, airplanes, airline operations, other transportation systems, amusement equipments, hospitals, medical care, dialysis and life support facilities or equipment, incineration and fuel devices, handling of nuclear or hazardous materials or chemicals, mining and drilling, and other applications where the level of risk to human life, health or property are elevated.
- d. MELCO SHALL NOT BE LIABLE FOR SPECIAL, INCIDENTAL, CONSEQUENTIAL, INDIRECT OR PUNITIVE DAMAGES, FOR LOSS OF PROFITS, SALES, OR REVENUE, FOR INCREASED LABOR OR OVERHEAD COSTS, FOR DOWNTIME OR LOSS OF PRODUCTION, FOR COST OVERRUNS, OR FOR ENVIRONMENTAL OR POLLUTION DAMAGES OR CLEAN-UP COSTS, WHETHER THE LOSS IS BASED ON CLAIMS FOR BREACH OF CONTRACT OR WARRANTY, VIOLATION OF STATUTE, NEGLIGENCE OR OTHER TORT, STRICT LIABILITY OR OTHERWISE.
- e. In the event that any damages which are asserted against MELCO arising out of or relating to the Products or defects in them, consist of personal injury, wrongful death and/or physical property

damages as well as damages of a pecuniary nature, the disclaimers and limitations contained in these terms shall apply to all three types of damages to the fullest extent permitted by law. If, however, the personal injury, wrongful death and/or physical property damages cannot be disclaimed or limited by law or public policy to the extent provided by these terms, then in any such event the disclaimer of and limitations on pecuniary or economic consequential and incidental damages shall nevertheless be enforceable to the fullest extent allowed by law.

- f. In no event shall any cause of action arising out of breach of warranty or otherwise concerning the Products be brought by Customer more than one year after the cause of action accrues.
- g. Each of the limitations on remedies and damages set forth in these terms is separate and independently enforceable, notwithstanding the unenforceability or failure of essential purpose of any warranty, undertaking, damage limitation, other provision of these terms or other terms comprising the contract of sale between Customer and MELCO.

### **4. Delivery/Force Majeure.**

- a. Any delivery date for the Products acknowledged by MELCO is an estimated and not a promised date. MELCO will make all reasonable efforts to meet the delivery schedule set forth in Customer's order or the purchase contract but shall not be liable for failure to do so.
- b. Products stored at the request of Customer or because Customer refuses or delays shipment shall be at the risk and expense of Customer.
- c. MELCO shall not be liable for any damage to or loss of the Products or any delay in or failure to deliver, service, repair or replace the Products arising from shortage of raw materials, failure of suppliers to make timely delivery, labor difficulties of any kind, earthquake, fire, windstorm, flood, theft, criminal or terrorist acts, war, embargoes, governmental acts or rulings, loss or damage or delays in carriage, acts of God, vandals or any other circumstances reasonably beyond MELCO's control.

### **5. Choice of Law/Jurisdiction.**

These terms and any agreement or contract between Customer and MELCO shall be governed by the laws of the State of New York without regard to conflicts of laws. To the extent any action or dispute is not arbitrated, the parties consent to the exclusive jurisdiction and venue of the federal and state courts located in the Southern District of the State of New York. Any judgment there obtained may be enforced in any court of competent jurisdiction.

### **6. Arbitration.**

Any controversy or claim arising out of, or relating to or in connection with the Products, their sale or use or these terms, shall be settled by arbitration conducted in accordance with the Center for Public Resources (CPR) Rules for Non-Administered Arbitration of International Disputes, by a sole arbitrator chosen from the CPR's panels of distinguished neutrals. Judgment upon the award rendered by the Arbitrator shall be final and binding and may be entered by any court having jurisdiction thereof. The place of the arbitration shall be New York City, New York. The language of the arbitration shall be English. The neutral organization designated to perform the functions specified in Rule 6 and Rules 7.7(b), 7.8 and 7.9 shall be the CPR.

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)-080857ENG-H(2409)MEE

MODEL: WS-ET-U-E

## **MITSUBISHI ELECTRIC CORPORATION**

HEAD OFFICE : TOKYO BUILDING, 2-7-3 MARUNOUCHI, CHIYODA-KU, TOKYO 100-8310, JAPAN  
NAGOYA WORKS : 1-14, YADA-MINAMI 5-CHOME, HIGASHI-KU, NAGOYA, JAPAN

When exported from Japan, this manual does not require application to the  
Ministry of Economy, Trade and Industry for service transaction permission.

Specifications subject to change without notice.