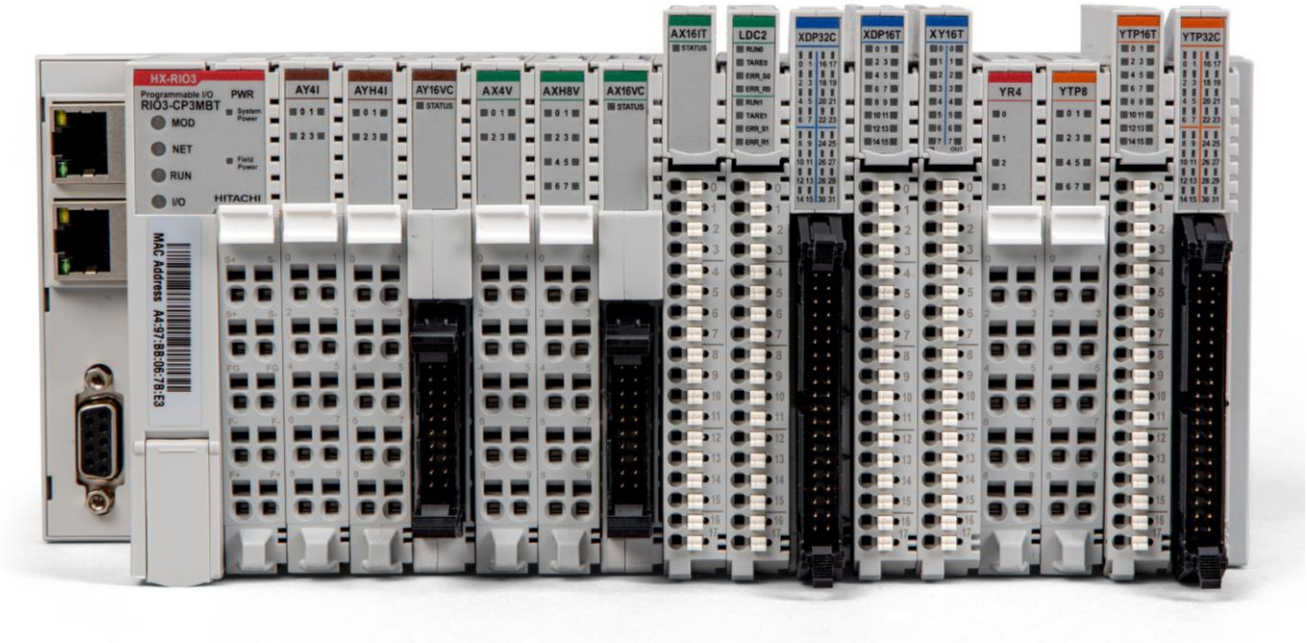
MODBUS Programmable I/O

RIO3-CP3MBT

User Manual



Version 2.06

HITACHI
Inspire the Next

DOCUMENT CHANGE SUMMARY				
REV	PAGE	REMARKS	DATE	EDITOR
1.02	All	new	Jan 21	Faber
1.02	62	Remove product list table and add a reference	Aug 21	Faber
1.021	all	Serial COM Port information added (p16), EAC Logo added (p7), Typo corrected	Oct 21	Faber
2.0	-	Codesys Version-up (3.5.11.3 → 3.5.17.3)		
2.02	10 11 34 12 41 48	Add Specifications (File system, RTC addition description) Changed the Ethernet connection specification and number of the max sockets. Information on RTC update options Change Non-Volatile Memory and Battery description User Management (Not supporting → Supporting) External Features library information CPU Usage	Aug 24	Lankala
2.06	12 13 65 18 12	Additional explanation for online change & update to RTC description Added Serial connection specification Modbus 7bit Setting Exception indication Reduced Non-volatile memory to 12Kbytes	Feb 25	Lankala

Contents

1. Important Notes 7

1.1. Safety Instruction 8

1.1.1. Symbols 8

1.1.2. Safety Notes 8

1.1.3. Certifications 8

2. RIO3-Series System 9

2.1. Electrical Interface 9

2.2. I/O Process Image Map 10

3. Specification 11

3.1. General Specification 11

3.2. Environment Specification 11

3.3. Programmable Specification 12

3.4. Interface Specification 13

3.5. Ethernet Connection Specification 13

3.6. Serial Connection Specification 13

4. Module Description 14

4.1. Wiring Diagram 14

4.2. LED Indicator 16

4.2.1. MOD (Module Status LED) 16

4.2.2. NET (Network Status LED) 16

4.2.3. RUN (PLC Run/Stop Status LED) 17

4.2.4. I/O (Extension Module Status LED) 17
4.2.5. Field Power Status LED 17
4.2.6. Exception indication 18
4.3. RJ-45 Socket , RS232/485 Port 18
4.4. Toggle Switch and Push Button 19
4.5. RTB Terminal Block 19
4.6. Pin Description 20
4.7. Dimension (mm) 20
5. Mechanical Setup 21
5.1. How to mount on Din-Rail 21
5.2. How to dismount on Din-Rail 21
5.3. Removable Terminal Block (RTB) 22
5.4. Method of Wiring 22
6. HX-RIO3 Series PIO (Programmable I/O) Functions 23
6.1. Connection to IOGuidePro by Modbus RTU (RS-232 or RS-485) 23
6.2. Connection to IOGuidePro by Modbus TCP 25
6.3. Confirmation of Network Information 25
6.4. BOOTP / DHCP 28
6.5. Setup IP Address 30
6.6. Serial Communication Settings 33
6.7. Memory Reset 35
6.8. RTC(Real Time Clock) Function 36
6.9. Webserver 38

6.10. IP Default Setting 40

7. External Feature Library 41

7.1. Battery State of charge 41

7.2. Read / Write Special Register (PIO Parameter Access) 42

7.2.1. Read Special Register 42

7.2.2. Write Special Register 42

7.2.3. Example to change Special Register 43

7.3. Read / Write Slot Parameter 44

7.3.1. Read Slot Parameter 44

7.3.2. Write Slot Parameter 44

7.3.3. Example to access the extended I/O module parameter 45

8. Upgrade Firmware 46

8.1. Using IAP over Ethernet 46

9. CPU Usage 48

10. Troubleshooting 49

10.1. How to diagnose by LED indicator 49

10.2 . How to diagnose when device couldn't communicate 50

11. MODBUS INTERFACE 51

11.1. MODBUS Interface Register / Bit Map 51

11.1.1. Register Map 51

11.1.2. Bit Map 51

11.2. MODBUS Transmission Mode 51

11.2.1. RTU Transmission Mode 51
11.2.2. ASCII Transmission Mode 52
11.3. Supported MODBUS Function Codes 52
11.3.1. 1 (0x01) Read Coils 52
11.3.2. 2 (0x02) Read Discrete Inputs 53
11.3.3. 3 (0x03) Read Holding Registers 54
11.3.4. 4 (0x04) Read Input Registers 55
11.3.5. 5 (0x05) Write Single Coil 56
11.3.6. 6 (0x06) Write Single Register 57
11.3.7. 8 (0x08) Diagnostics 58
11.3.8. 15 (0x0F) Write Multiple Coils 60
11.3.9. 16 (0x10) Write Multiple Registers 61
11.3.10. 23 (0x17) Read/Write Multiple Registers 62
11.4. MODBUS Special Register Map 63
11.4.1. Adapter Register Mapping 63
11.4.2. Adapter Identification Special Register (0x1000, 4096) 63
11.4.4. Adapter Information Special Register (0x1100, 4352) 64
11.4.5. Adapter Setting Special Register (0x1600, 5632) 65
11.4.6. Expansion Slot Information Special Register (0x2000, 8192) 66
11.5. I/O Process Image Map 69
11.5.1. Example of Input Process Image (Input Register) Map 69
11.5.2. Example of Output Process Image (Output Register) Map 72
11.6. MODBUS Reference 73
A. APPENDIX A - Product List 73

1. Important Notes

Solid state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will HITACHI be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment. The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, HITACHI cannot assume responsibility or liability for actual use based on the examples and diagrams.

Warning!

- **If you don't follow the directions, it could cause a personal injury, damage to the equipment or explosion**
- Do not assemble the products and wire with power applied to the system. Else it may cause an electric arc, which can result into unexpected and potentially dangerous action by field devices. Arching is explosion risk in hazardous locations. Be sure that the area is non-hazardous or remove system power appropriately before assembling or wiring the modules.
- Do not touch any terminal blocks or IO modules when system is running. Else it may cause the unit to an electric shock or malfunction.
- Keep away from the strange metallic materials not related to the unit and wiring works should be controlled by the electric expert engineer. Else it may cause the unit to a fire, electric shock or malfunction.

Caution!

- **If you disobey the instructions, there may be possibility of personal injury, damage to equipment or explosion. Please follow below Instructions.**
- Check the rated voltage and terminal array before wiring. Avoid the circumstances over 60℃ of temperature. Avoid placing it directly in the sunlight.
- Avoid the place under circumstances over 90% of humidity.
- Do not place Modules near by the inflammable material. Else it may cause a fire.
- Do not permit any vibration approaching it directly.
- Go through module specification carefully, ensure inputs, output connections are made with the specifications. Use standard cables for wiring.
- Use Product under pollution degree 2 environment.

1.1. Safety Instruction

1.1.1. Symbols

DANGER	Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death property damage or economic loss.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.
ATTENTION	Identifies information about practices or circumstances that can lead to personal injury, property damage, or economic loss. Attentions help you to identity a hazard, avoid a hazard, and recognize the consequences.

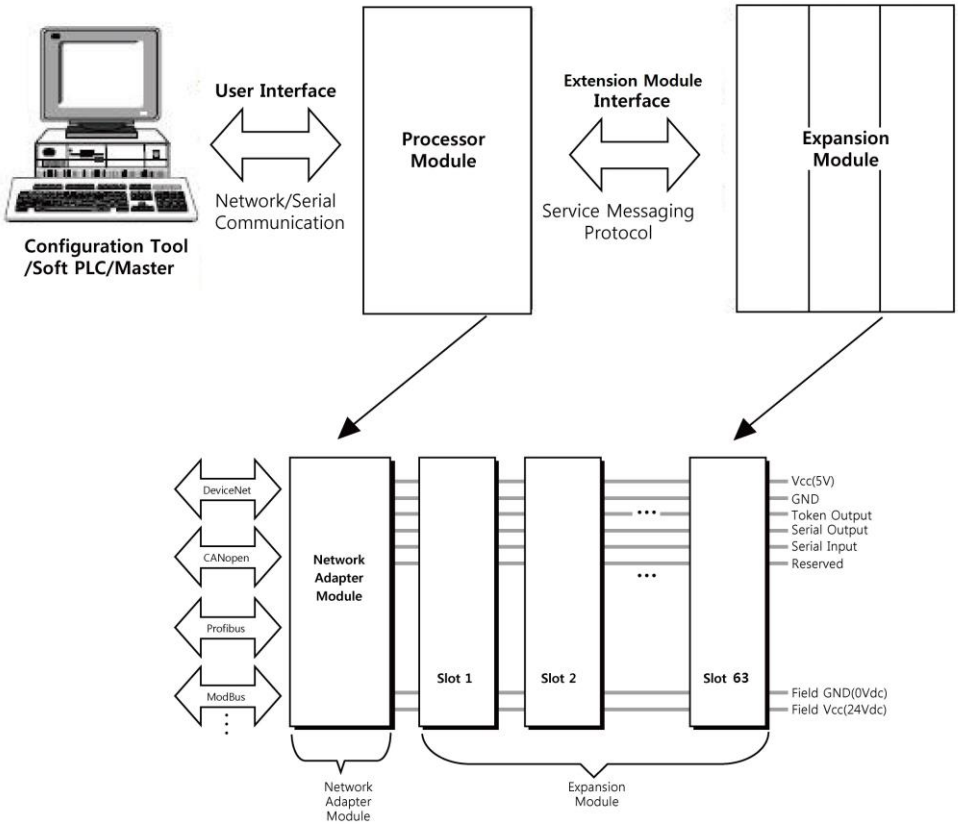
1.1.2. Safety Notes

DANGER	The modules are equipped with electronic components that may be destroyed by electrostatic discharge. When handling the modules, ensure that the environment (persons, workplace and packing) is well grounded. Avoid touching conductive components.

1.1.3. Certifications

2. RIO3-Series System

2.1. Electrical Interface



Network Adapter Module

The Network Adapter Module forms the link between the field bus and the field devices with the Expansion Modules. The connection to different field bus systems can be established by each of the corresponding Network Adapter Module, e.g. for MODBUS TCP, Ethernet IP, EtherCAT, PROFINET, PROFIBUS, CANopen, MODBUS/Serial etc.

Expansion Module

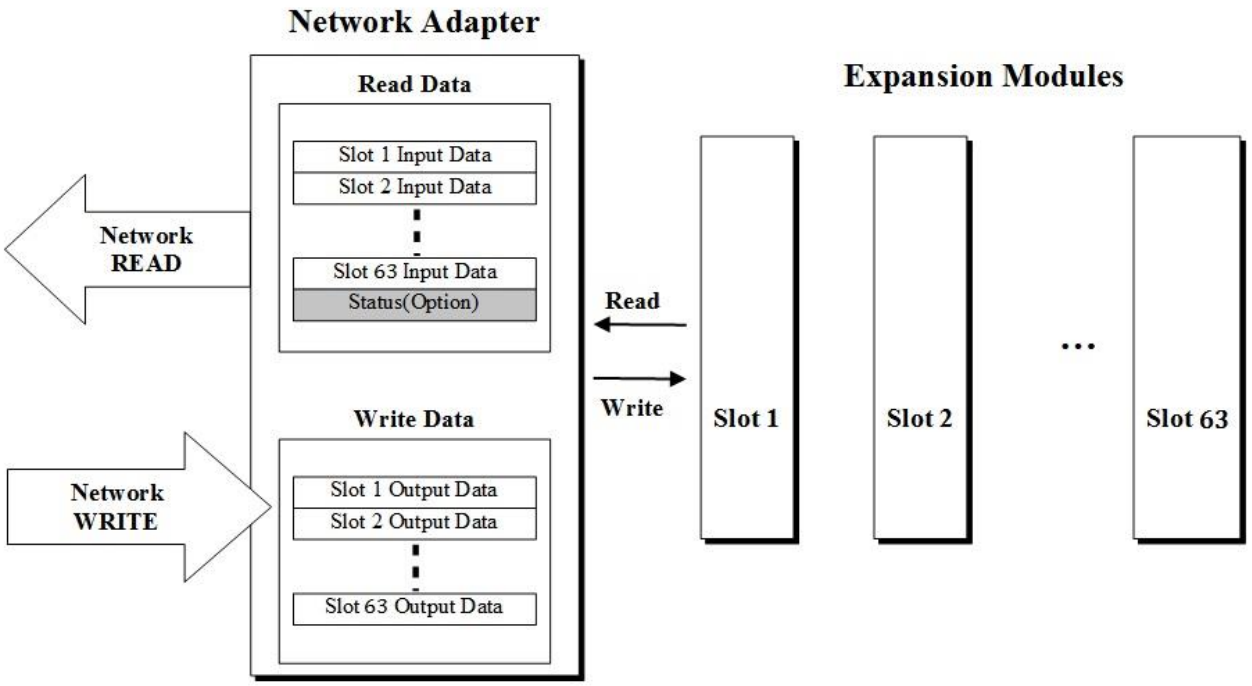
The Expansion Modules are supported a variety of input and output field devices. There are digital and analog input/output modules and special function modules.

Two types of Message

Service Messaging / I/O Messaging

2.2. I/O Process Image Map

An expansion module may have 3 types of data as I/O data, configuration parameter and memory register. The data exchange between network adapter and expansion modules is done via an I/O process image data by internal-protocol. The following figure shows the data flow of process image between network adapter and expansion modules.



3. Specification

3.1. General Specification

General Specification	
UL System Power	Supply Voltage : 24Vdc nominal, Class 2
System Power	Supply Voltage : 24Vdc nominal Supply Voltage range : 15~30Vdc Protection : Output current limit (Min. 1.5A) Reverse polarity protection
Power Dissipation	110mA typical @ 24Vdc
Current for I/O Module	1.5A @ 5Vdc
Isolation	System Power to internal logic : Non-Isolation System Power I/O driver : Isolation
Field Power	Supply Voltage : 24Vdc typical (Max. 30Vdc) *Field Power Range is different depending on IO Module series. Refer to IO Module`s Specification.
Max. Current Field Power Contact	DC 10A Max.
Wiring	I/O Cable Max. 2.0mm2(AWG 14)
Weight	<167g
Module Size	54mm x 99mm x 70mm

3.2. Environment Specification

Environment Condition	
Operating Temperature	-20℃ ~ 60℃
UL Temperature	-20℃ ~ 60℃
Storage Temperature	-40℃ ~ 85℃
Relative Humidity	5% ~ 90% non-condensing
Mounting	DIN rail
Shock Operating	IEC 60068-2-27
Vibration/shock resistance	Based on IEC 60068-2-6, 4g Sine Vibration - 10 ~ 25 Hz : 0.5mm - 50 ~ 150 Hz : 5g - 150 ~ 1000 Hz : 2g - Sweep Rate : 1 Oct/min, 50 cycles Sine Vibration - 10 ~ 25 Hz : 0.03 g²/Hz - 25 ~ 50 Hz : 0.05 g²/Hz - 50 ~ 150 Hz : 0.15 g²/Hz - 150 ~ 1000 Hz : 0.01 g²/Hz
EMC resistance burst/ESD	EN 61000-6-2 : 2019 EN 61000-6-4/ALL : 2011
Installation Pos. / Protect. Class	Variable / IP20
Product Certifications	UL, CE

3.3. Programmable Specification

Programmable Specification	
Module list	RIO3-CP3MBT
Programming	CODESYS V3.5.11.3 (FW REV 2.xxx) CODESYS V3.5.17.0 (FW REV 3.xxx)
Run-Time System	Multiple PLC Tasks
Program Languages	IEC 61131-3 (LD, IL, ST, FBD, SFC)
Program Memory	16 MByte
Data Memory	16 MByte
	I/O Input: %IW0 ~ %IW2047 (2048 words)
	I/O Output: %QW0 ~ %QW2047 (2048 words)
	Memory: %MW0 ~ %MW8191 (8192 words)
Non-Volatile Memory[1)]	12 Kbytes
	Retain: 6 KByte
	Persistent: 6 KByte
MQTT[2)]	✓
MQTT Sparkplug B	✓
User management[3)]	✓
OPC	OPC DA Server, OPC UA Server / Client
Online Change[4*)]	✓
Source Upload/Download	✓
File transmit	✓
Breakpoint	✓
Web Visualization[5)]	✓
Max. Task	10
Max. Cycle Task	10
Max. Status Task	10
Process Time	0.1645usec
RTC[6)]	Retain Time : < 15 days (Accuracy : < 2min per a month) Recommendation : Battery should be fully recharged at 25℃.

1) A phenomenon was discovered where data was not recorded properly in the retain memory area, so to improve the problem by adding the ECC function that checks the NAND flash status, the storage time has increased. Therefore, it is difficult to store all data, so the memory size has been reduced.
2) MQTT can't support the TLS.
3) The functionality is available in a limited form.
4) Online Change: re-downloads only the changed parts of an application that is already running on the controller without initializing variables.
   *Precautions for Online Changes
   Due to product characteristics, performing a re-download may impact PLC logic execution, causing delays. Therefore, please proceed with "online change" only when the equipment is stable and in a safe state.
5) Web Visualization may not be supported in Internet Explorer.
6) RTC (at 25℃)

Battery charging time	Retain time at 25℃	*** RTC Warning
4 hours	Up to 2 days	There will be operating problems when the battery is discharged. - RTC data will not be stored and will be the initial value. - Retain time may vary depending on temperature and environment.
12 hours	Up to 12 days	
16 hours	Up to 15 days	

- Recommend charging for at least 16 hours when the battery is discharged.
- The battery is one of components on PCB in the plastic housing. It means that it can't be checked visually but can be charged by supplying the voltage to the system power.

3.4. Interface Specification

Interface Specification			
Adapter Type	Master & Slave Node (Modbus TCP, Modbus RTU)		
Max. Expansion	63 Slots		
Max. Data Size	Max. 128Byte each slot (Input + Output)		
Max. Nodes	Limited by Ethernet Specification		
Baud rate	10/100Mbps, Auto-negotiation, Full Duplex		
Ethernet Interface	RJ-45 socket (x 2ports)		
Ethernet Protocol**	Modbus TCP, Modbus UDP, SNTP, SNMP, MQTT HTTP (Web Visualization*, Web-Server), DHCP/BOOTP, OPC-Server		
Max. Socket	UDP: 16, TCP: 64		
Serial Interface	RS232/RS485 (x 1port)		
Serial Protocol	Modbus RTU		
Interface connector	D-Sub 9Pin Connector (x 1port)		
Baud Rate	2400~115200 bps (Default: 115200 bps)		
Indicator	6 LEDs		
	1 Green/Red	Module Status	(MOD)
	1 Green/Red	Network Status	(NET)
	1 Green/Red	PLC Run/Stop Status	(RUN)
	1 Green/Red	Expansion I/O Module Status	(IOS)
	1 Green	System Power Status	
	1 Green	Field Power Status	

3.5. Ethernet Connection Specification

Function*	Model	Max. number of concurrent communications
Web Visualization	RIO3-CP3MBT	One for each functions are available at the same time.
ARTI (OPC-server)		
CODESYS link		
Network-Variable		
Modbus/TCP Master		64 Modbus/TCP Slaves can be connected
Modbus/TCP Slave		64 Modbus/TCP Masters can be connected
Web-Server		64 clients can be opened

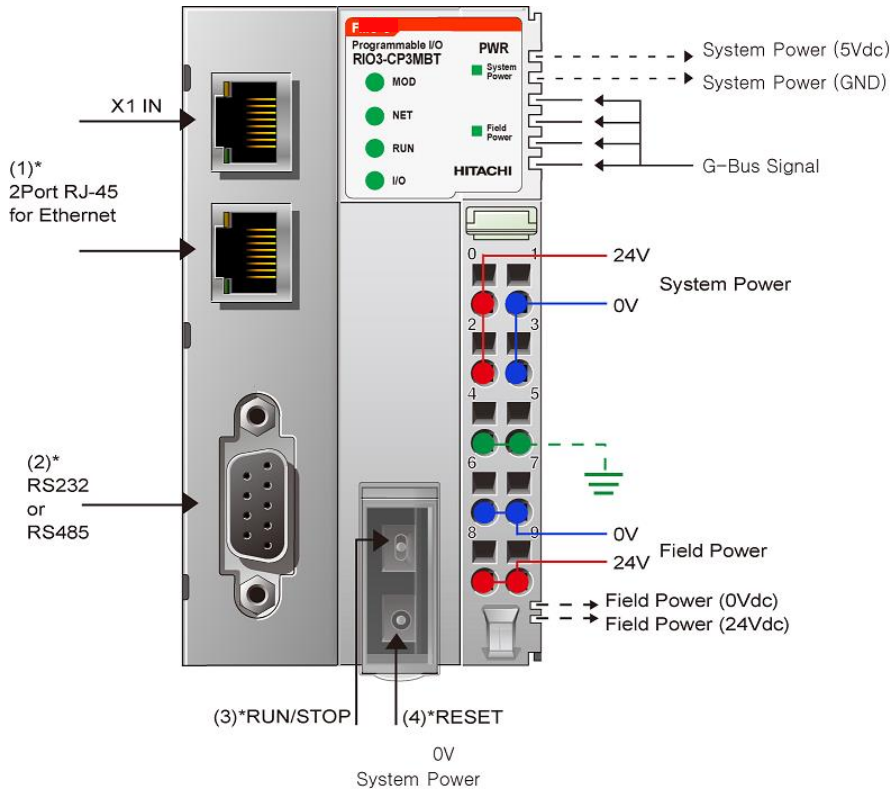
* While using these functions, maximum number of sockets are available at the same time.
* RIO3-CP3MBT four functions below can work at the same time.
(Web visualization, OPC Server, Network-Variable, and CODESYS link)

3.6. Serial Connection Specification

Function*	Model	Max. number of concurrent communications
Modbus RTU Master	RIO3-CP3MBT	RS232: 1 Slaves can be connected RS485: 31 Slaves can be connected

4. Module Description

4.1. Wiring Diagram

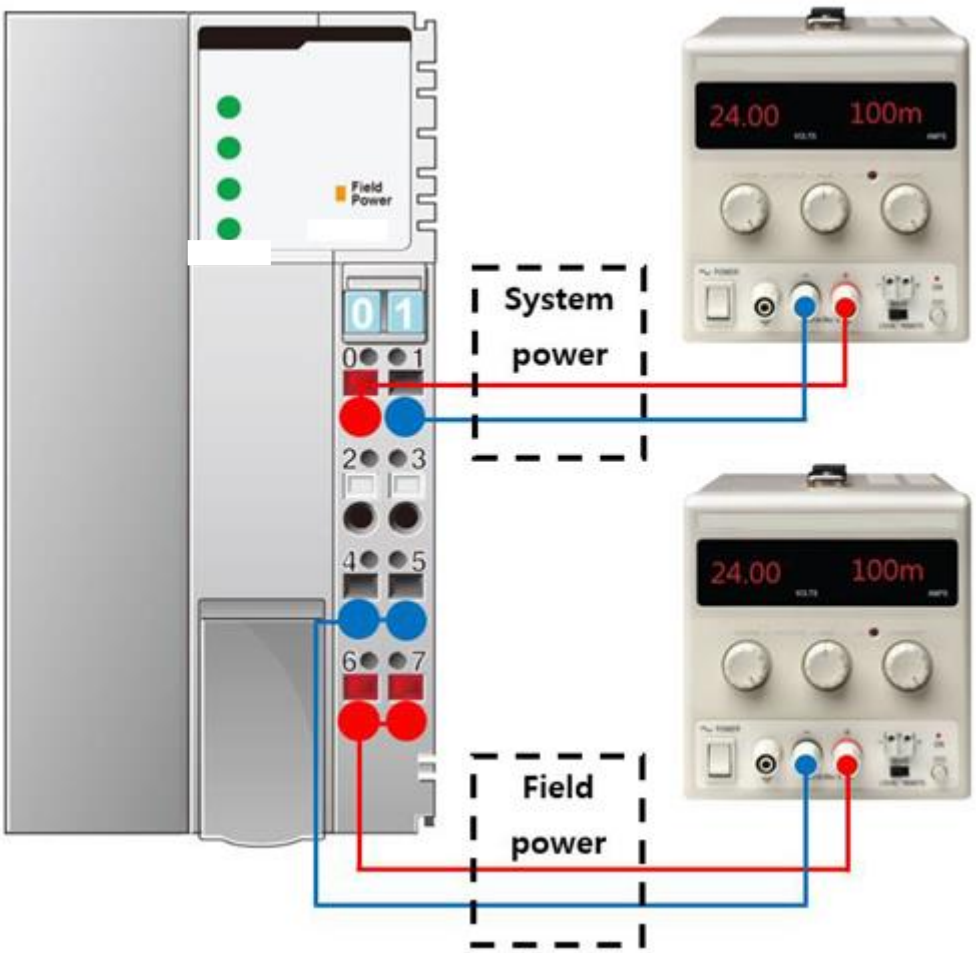


Pin No.	Signal Description	Signal Description	Pin No.
0	System Power, 24V	System Power, Ground	1
2	System Power, 24V	System Power, Ground	3
4	F.G	F.G	5
6	Field Power, Ground	Field Power 0V, Ground	7
8	Field Power, 24V	Field Power, 24V	9

System power and Field power must be supplied separately by using the different power source

ATTENTION	
	The modules are not hot swappable. It means that they should be not removed in power on condition.

How to supply the power correctly

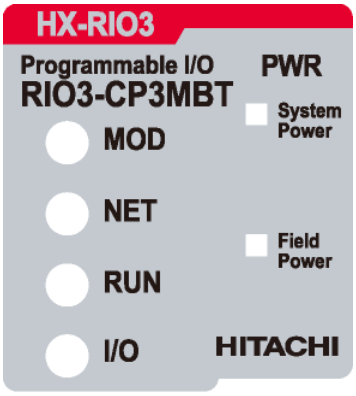


- Refer the pin map of power RTB before wiring.
- Current capacity margin should be considered when you supply the power.
- Supply voltage 24Vdc which is the recommended voltage level.

➢ System and Field power must be supplied separately as the picture above.

❖ If the power is supplied to each system and field power from same power source, it would be vulnerable to power noise.

4.2. LED Indicator



4.2.1. MOD (Module Status LED)

State	LED state	To indicate
No Power	Off	No power is supplied to the unit.
Device Operational	Green	The unit is operating in normal condition.
Device in Standby	Blinking Green	The EEPROM parameter is not initialized yet. Serial Number is zero value (0x00000000)
IAP Mode	Toggle Green/Red	IAP Mode : Available for firmware download using Firefox.
Unrecoverable Fault	Red	The unit has occurred unrecoverable fault in self-testing. - Firmware fault

*The IP address to access IAP web server during IAP Mode : 192.168.100.10 (Recommended to use Firefox)

4.2.2. NET (Network Status LED)

State	LED state	To indicate
Off-line	Off	Network Offline.
On-line (Connect)	Green	On-line Mode and network is connected.
Error	Red	Network Error
Diagnostic	Blinking Red	Diagnostic Mode

* Blinking MOD & NET LED : BOOTP/DHCP is requesting the address data for new IP address.
(You can change the IP setting mode. Please refer to this manual in the part of IP set-up.)

4.2.3. RUN (PLC Run/Stop Status LED)

State	LED state	To indicate
Not Programmed	Off	Power is not supplied or PIO is not programmed
Run	Green	PLC Run
Stop	Blinking Green	PLC Stop
Program Error	Blinking Red	User PLC Program Error

4.2.4. I/O (Extension Module Status LED)

State	LED state	To indicate
Not Powered No Expansion Module	Off	Device has no expansion module or may not be powered
On-line, Do not Exchanging I/O	Blinking Green	I/O Communication is normal but does not exchange I/O data. (Passed the expansion module configuration).
Connection, Run Exchanging IO	Green	Exchanging I/O data
Connection Fault during Exchanging IO	Blinking Red	One or more expansion module occurred in fault state. - Check the expanded module configuration. - Communication failure of the expanded module. - Overflowed Input/Output size.
Expansion Configuration Failed	Red	Failed to initialize expansion module - Detected invalid expansion module ID. - Overflowed Input / Output Size - Oversized expansion module - Initial protocol failure - Mismatch vendor code between adapter and I/O module.

4.2.5. Field Power Status LED

State	LED state	To indicate
Not Supplied Field Power	Off	Not supplied 24V dc field power.
Supplied Field Power	Green	Supplied 24V dc field power.

4.2.6. Exception indication

To indicate	LED			
	MOD	NET	RUN	IOS
Booting [1]	Green Blinking	Green Blinking		
IAP Mode [2]	Green/Red Toggle	-	Off	Off
Program reset	-	-	Red/Green (Toggle every 0.25s)	-
Factory reset	Red/Green (Toggle every 0.25s)			
IO Watchdog error	Red	-	Red	-
CODESYS Task Watchdog error	-	-	Red	-
CODESYS License error	-	-	Red /Green (Toggle every 2s)	-
Heap memory over flow	Red	Red	-	-
Stack memory over flow or Assertion error	Red	-	-	-
Hard Fault	Red			

'-' : Current LED status
1) BOOTP/DHCP requests a new IP address. (You can change the IP setting mode. Refer to Modbus Register 0x160B.)
2) The IP Address to access IAP web-server during IAP Mode: 192.168.100.10 (Recommended to use FireFox)

4.3. RJ-45 Socket , RS232/485 Port

RJ-45	Signal Name	Description
1	TD+	Transmit +
2	TD-	Transmit -
3	RD+	Receive +
4	-	
5	-	
6	RD-	Receive -
7	-	
8	-	
Case	Shield	

RS 232/485*	Signal Name	Description
1	-	
2	TXD	RS-232 TXD
3	RXD	RS-232 RXD
4	-	
5	GND	RS-232 GND
6	D+	RS-485 D+
7	-	
8	D-	RS-485 D-
9	-	

***COM Port 1: RS-232** **COM Port 2: RS-485**

4.4. Toggle Switch and Push Button

Toggle Switch Status	Module in	Description
UP	RUN	PLC Run
DOWN	STOP	PLC Stop

Push Button	Module is	Description
Press and detach.	Reset	Reset the PLC and then stop.*
Push for 5sec and power Reset	PLC Reset	Erase PLC user program and Retain memory.*
Push for 20sec and power reset	Factory default	Erase PLC user program and PLC parameter reset.*
Hold down and reset the power.	IAP mode	Available for firmware download using FireFox

*Device reset requires a loaded internal Battery. If reset is not working keep the device powered for at least 16 hours

4.5. RTB Terminal Block

Pin No.	Signal Description	Signal Description	Pin No.
0	System Power, 24V	System Power, Ground	1
2	System Power, 24V	System Power, Ground	3
4	F.G	F.G	5
6	Field Power, Ground	Field Power, Ground	7
8	Field Power, 24V	Field Power, 24V	9

- System Power: The power for starting up CPU.
- Field Power: The power for input and output line.

DANGER

Do not use an incorrect voltage/frequency!
The use of an incorrect supply voltage or frequency can cause severe damage to the component.

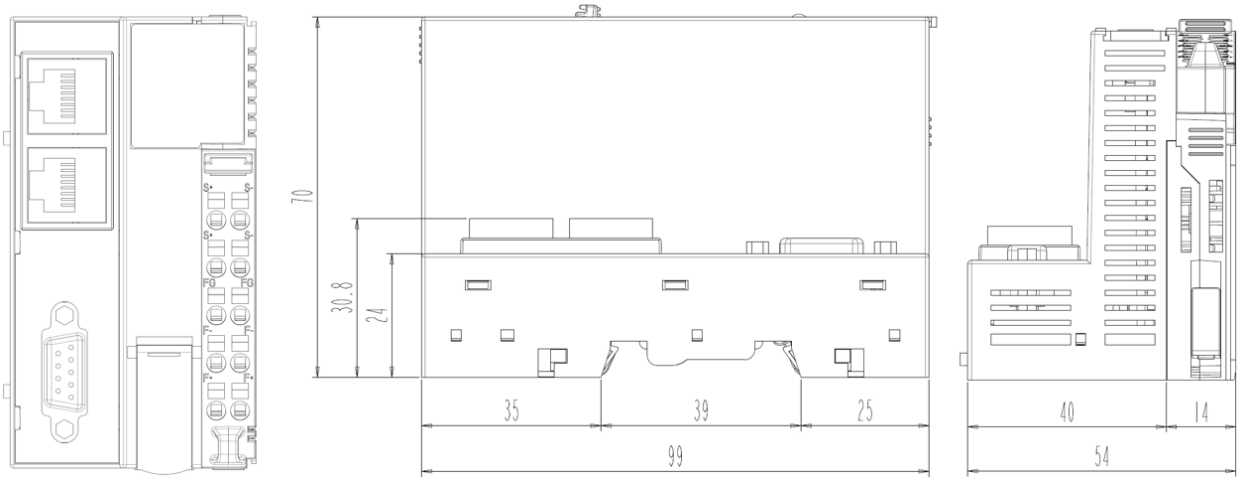
4.6. Pin Description

Communication between the Network adapter and the expansion module as well as system / field power supply of the bus modules is carried out via the internal bus. It is comprised of 6 data pins and 2 field power pins.

No.	Name	Description
1	System Vcc	System supply voltage (5V dc).
2	System GND	System Ground.
3	Token Output	Token output port of Processor module.
4	Serial Output	Transmitter output port of Processor module.
5	Serial Input	Receiver input port of Processor module.
6	Reserved	Reserved for bypass Token.
7	Field GND	Field Ground.
8	Field Vcc	Field supply voltage (24Vdc).

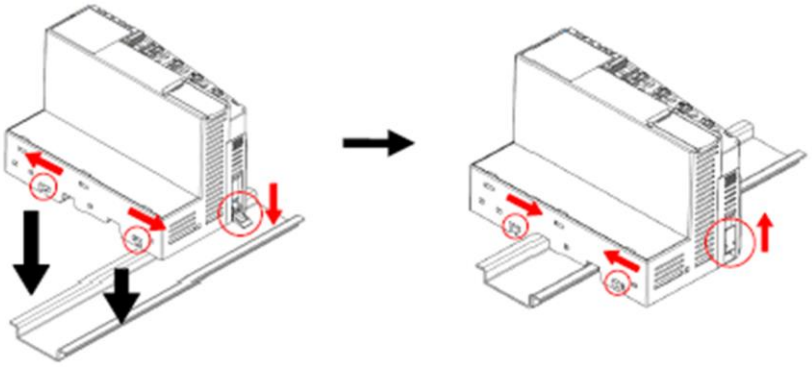
DANGER	
	Do not touch data and field power pins in order to avoid soiling and damage by ESD noise. To prevent ESD noise, it is recommended to use the END module.

4.7. Dimension (mm)

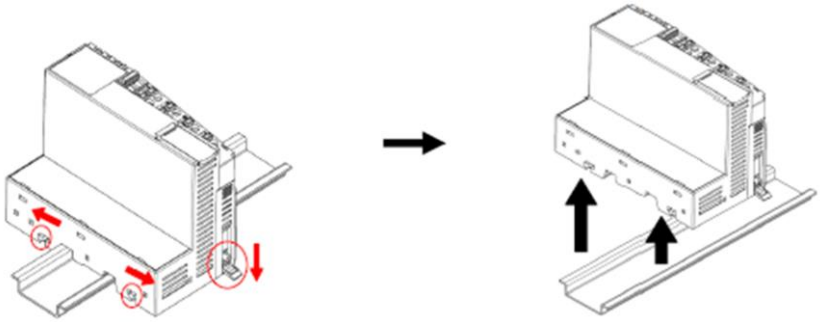


5. Mechanical Setup

5.1. How to mount on Din-Rail



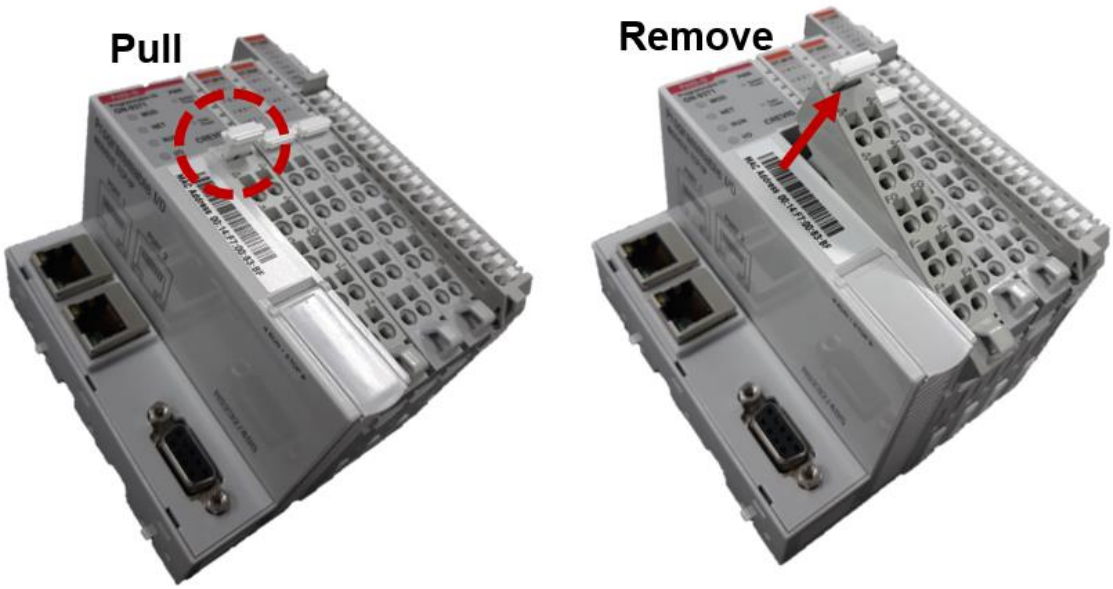
5.2. How to dismount on Din-Rail



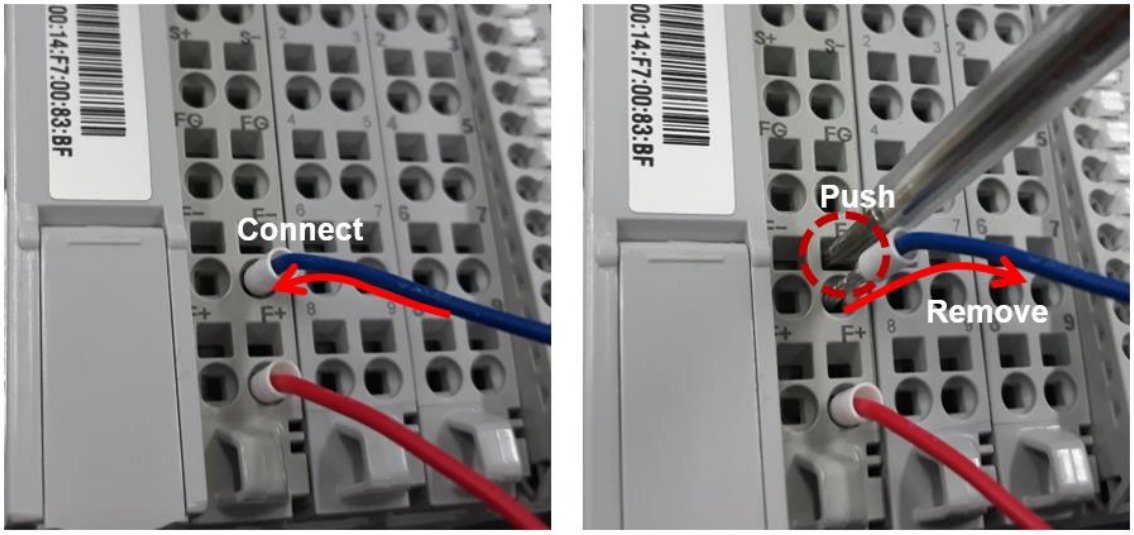
As above figure in order to safeguard the RIO3 module from jamming, it should be fixed onto the DIN rail with locking level. To do so, fold on the upper of the locking lever. To pull out the RIO3 module, unfold the locking lever as above figure.

DANGER	Before starting the work on the components, the voltage supply must be turned off.

5.3. Removable Terminal Block (RTB)



5.4. Method of Wiring



Connecting or removing the cable by pushing the terminal button for the relevant points.

ATTENTION	
	The use of an incorrect supply voltage or frequency can cause severe damage to the component.

6. HX-RIO3 Series PIO (Programmable I/O) Functions

HITACHI IO Guide Pro is compatible with the RIO3-CP3MBT.

The basic parameter set-up and configuration for the PIO is available via the IO Guide Pro.

And user can set up the IP Address, RTC from the Webserver page.

6.1. Connection to IOGuidePro by Modbus RTU (RS-232 or RS-485)

(1) Installation program, 'IO Guide Pro Tool' downloaded by HITACHI Webpage

(2) Open the IO Guide and Click the 'New project' Icon.

(3) Write the 'Project Name', Select the 'Bus Type' & the 'Location' and Click the 'OK'.

(4) After creating a project and Click the ‘Automatic scan’ Icon.

(5) Write the value(Port, Node, Baudrate), and Click the ‘Scan’ button.

(6) After the end to scan the network, Click the ‘OK’.

(7) Now ready to use the IO Guide Pro with RTU.

6.2. Connection to IOGuidePro by Modbus TCP

The process is identical as in the previous chapter. Only Modbus TCP must be chosen in the bus type drop down menu.

(1) Open the IO Guide Pro Tool and Click the ‘New project’ Icon (Same as RTU).
(2) Write the ‘Project Name’, Select the ‘Bus Type’ & the ‘Location’ and Click the ‘OK’.
(3) After creating a project and click the ‘Automatic Scan’ Icon.
(4) Click the ‘Scan’ button.
(5) After the end to scan the network, Click the ‘OK’ button.
(6) Now ready to use the IO Guide Pro with Modbus TCP.

6.3. Confirmation of Network Information

Network Condition about the IP Address, Subnet Mask, Gate Way, Mac Address of RIO3-CP3MBT can be checked, or set up from the following process below by using the IO Guide Pro Tool under Modbus RTU, and Modbus TCP protocol types.

IP Address : Also known as an "IP number" or simply an "IP," this is a code made up of numbers separated by three dots that identifies a particular computer on the Internet. Every computer, whether it be a Web server or the computer you're using right now, requires an IP address to connect to the Internet. IP addresses consist of four sets of numbers from 0 to 255, separated by three dots.

Subnet Mask : A subnet mask is a number that defines a range of IP addresses that can be used in a network. (It is not something you wear on your head to keep subnets out.) Subnet masks are used to designate sub networks, or subnets, which are typically local networks LANs that are connected to the Internet. Systems within the same subnet can communicate directly with each other, while systems on different subnets must communicate through a router.

Gateway : A gateway is either hardware or software that acts as a bridge between two networks so that data can be transferred between a number of computers.

MAC Address : A MAC address is a hardware identification number that uniquely identifies each device on a network. The MAC address is manufactured into every network card, such as an Ethernet card or Wi-Fi card, and therefore can’t change.

(1) Run '[Hitachi] -> [IOGuidePro] -> [Protocol Messenger] -> [Modbus]'

(2) Write the value of each.

*Protocol : Modbus TCP, Modbus RTU

*ComPort : User Port / Baudrate : 115200(default)

*Address(Hex) : 1600 (IP Address Register)

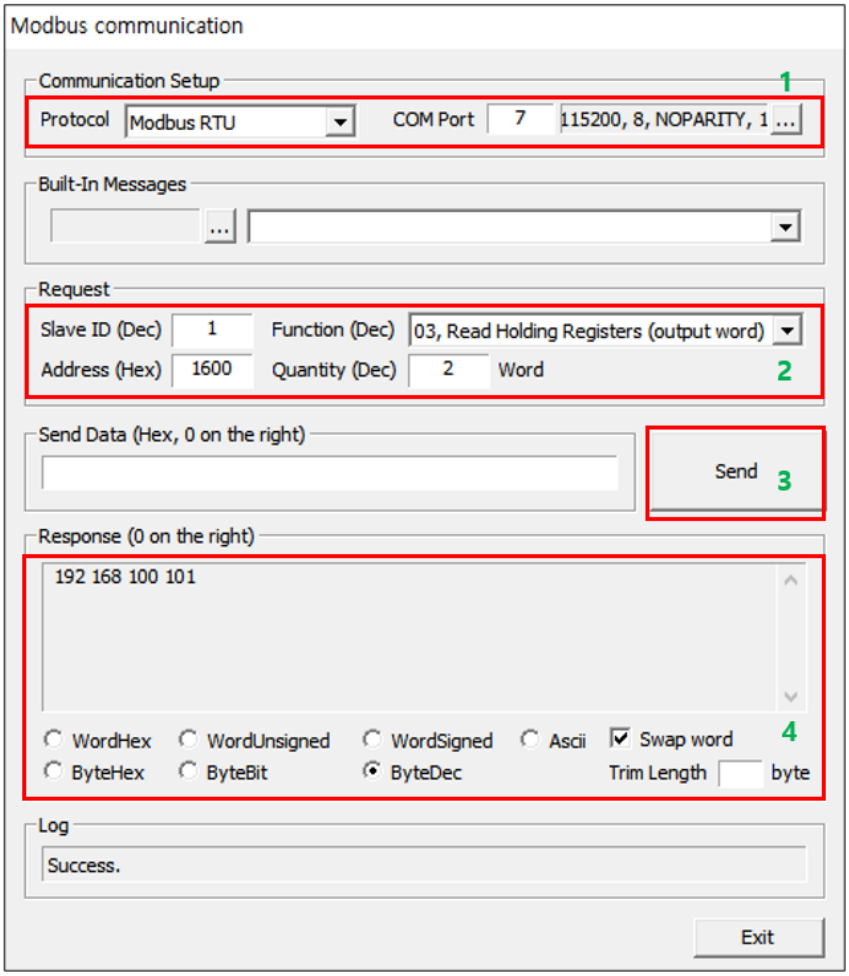
: 1602 (IP Subnet Mask Register)

: 1604 (Gate way Register)

: 1610 (Mac Address Register)

*Function(Dec) : 03, Read Holding Registers

(3) After clicking the ‘send’ button and confirm the necessary information.
If you choose 'ByteDec', easier to see.



6.4. BOOTP / DHCP

'IP Address' setting is available for BOOTP / DHCP.

BOOTP: short for Bootstrap Protocol, is a UDP network protocol used by a network client to obtain its IP address automatically. This is usually done in the bootstrap process of computers or operating systems running on them. The BOOTP server can assign the IP address from a pool of addresses to each client.

DHCP: set of rules used by communications devices such as a computer, router or network adapter to allow the device to request and obtain an IP address from a server which has a list of addresses available for assignment.

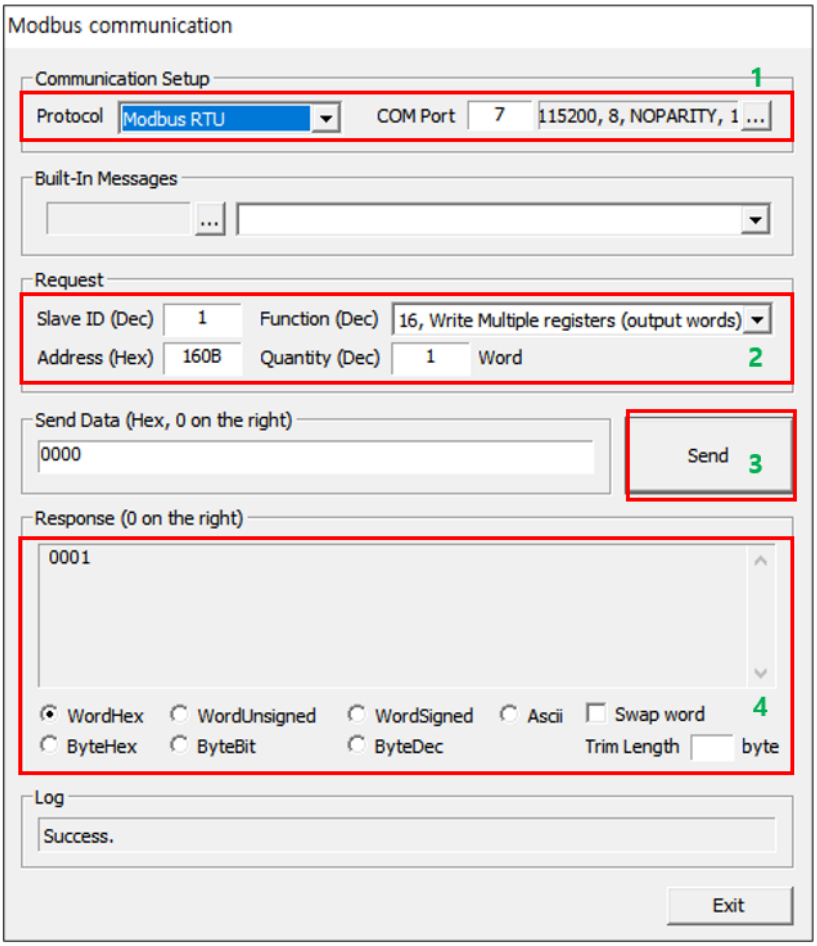
(1) Run '[Hitachi] -> [IOGuidePro] -> [Protocol Messenger] -> [Modbus]'

(1) Write the value of each.

- Protocol : Modbus TCP, Modbus RTU
- ComPort : User Port / Baudrate : 115200(default)
- Address(Hex) : 160B (IP Setting Method Register)
- Function(Dec) : 16, Write Multiple registers

(2) Write the register value and click the ‘Send’ button.

*Not Use : 0000 / *BOOTP Setting : 8000 / *DHCP Setting : 8001



6.5. Setup IP Address

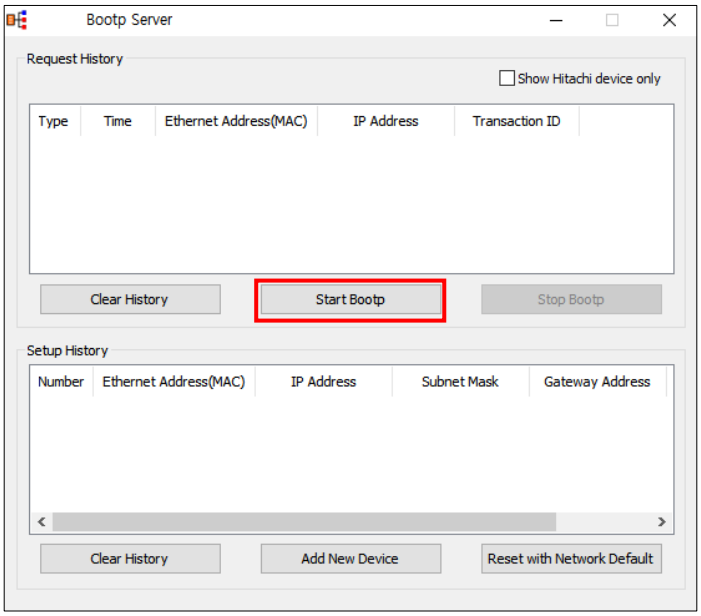
User can assign the IP Address manually via the HITACHI BOOTP Server in IO Guide Pro.

- **Default IP Address is 192.168.100.100**

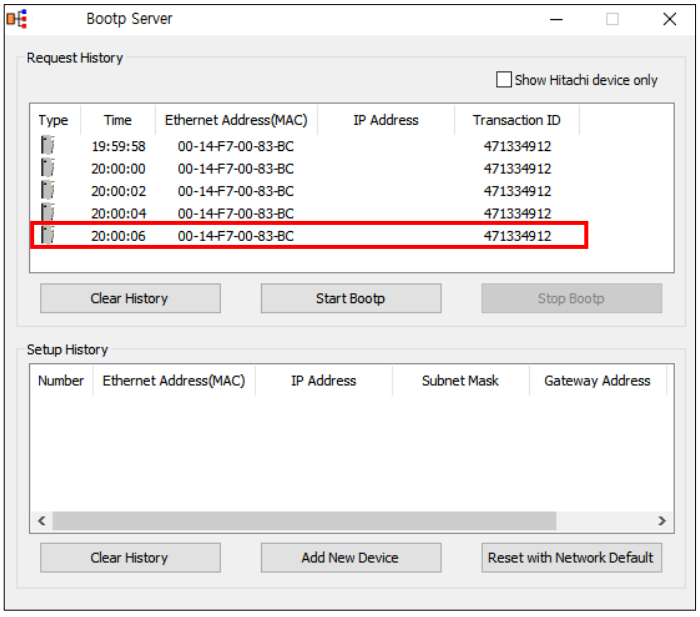
(1) Run '[Hitachi] → [IOGuidePro] → [BOOTP Server]'

IMPORTANT	RIO3-CP3MBT can support DHCP or BOOTP. If DHCP server is working on the same network, BOOTP is not available. In this case, IP set-up is only available from the DHCP server.

(2) Power on the RIO3-CP3MBT, and Click the ‘Start BOOTP’ button.

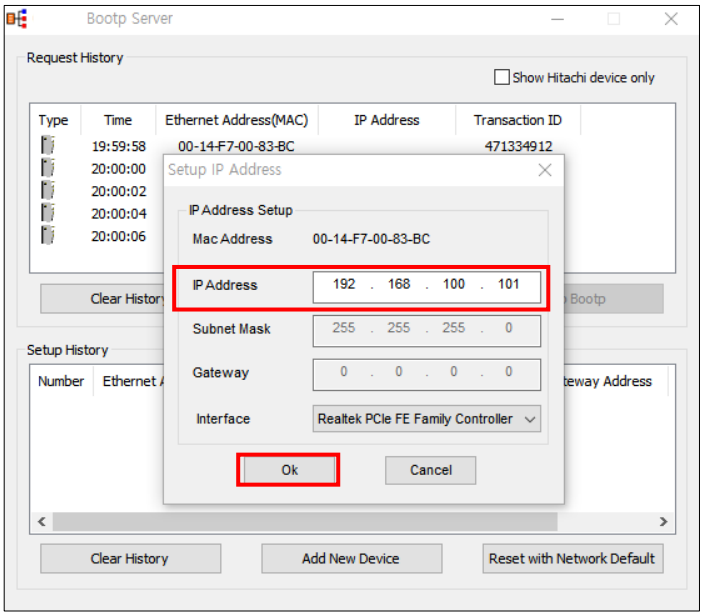


(3) Double Click ‘MAC address of RIO3-CP3MBT.

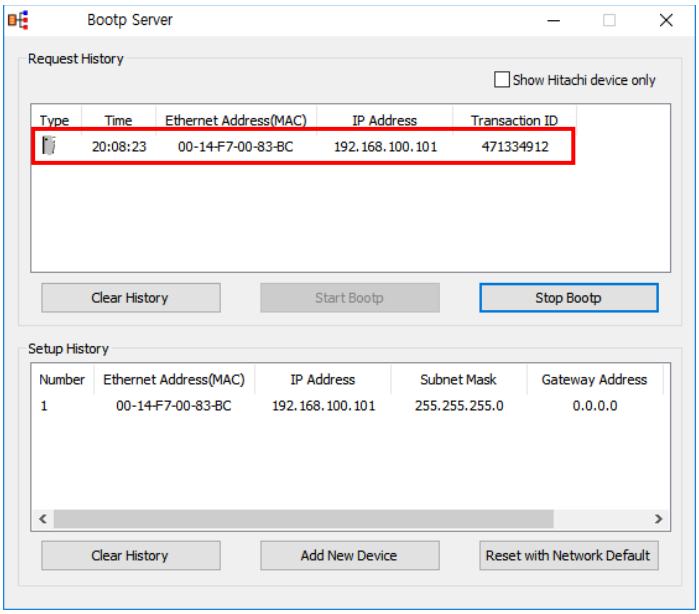


IMPORTANT	Turn on the power of RIO3-CP3MBT, and BOOTP server will retry to get the MAC address or IP address 2 times for 4 seconds. IP set-up can be available within the duration above from the BOOTP server.

(4) Set the IP, and click ‘OK’.



(5) Finish



IMPORTANT	Subnet Mask and Gateway are assigned automatically by the value that is set in the computer.

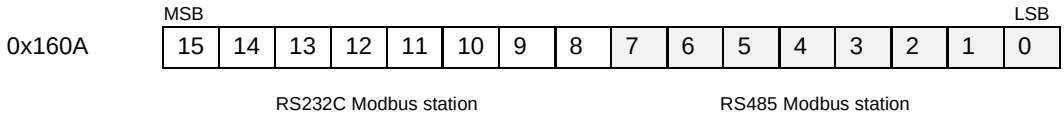
6.6. Serial Communication Settings

Setting according to the each communication state is possible because RIO3-CP3MBT is available for RS232 and RS485 serial communication.

Station Setting

The following illustration is an area of Register 0x160A address that can be used to set the code of Serial communication. High 1byte is the area of rs232, and Low 1byte is the area of rs485.

It is possible to set a maximum of 247 for each area. (default : 1)



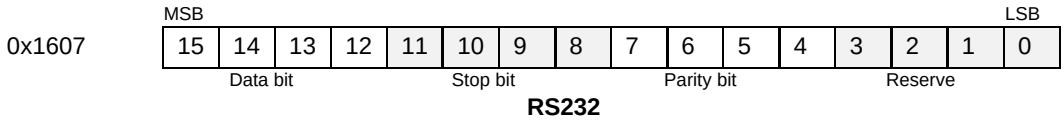
RS232/ RS485 Communication setting

The options for the communication can be selected.

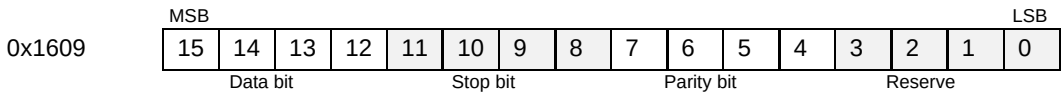
RS232 can be selected from the register address, “0x1606”.

RS485 can be selected from the register address, “0x1608”.

- 1 nibble : Data bit(0 : 8bit(default), 1 : 9bit)
- 2 nibble : Stop bit(0 : 1bit(default), 1 : 2bit)
- 3 nibble : Parity bit(0 : none(default), 1: even, 2 : odd)
- 4 nibble : Reserve



RS232



RS485

Baud rate setting

The baud rate from 2400bps to 115200bps is supported.

RS232 can be selected from the register address, “0x1606”.

RS485 can be selected from the register address, “0x1608”.

0 : 115200(Default) 1 : 2400 2 : 4800 3 : 9600

4 : 19200 5 : 38400 6 : 57600 7 : 115200

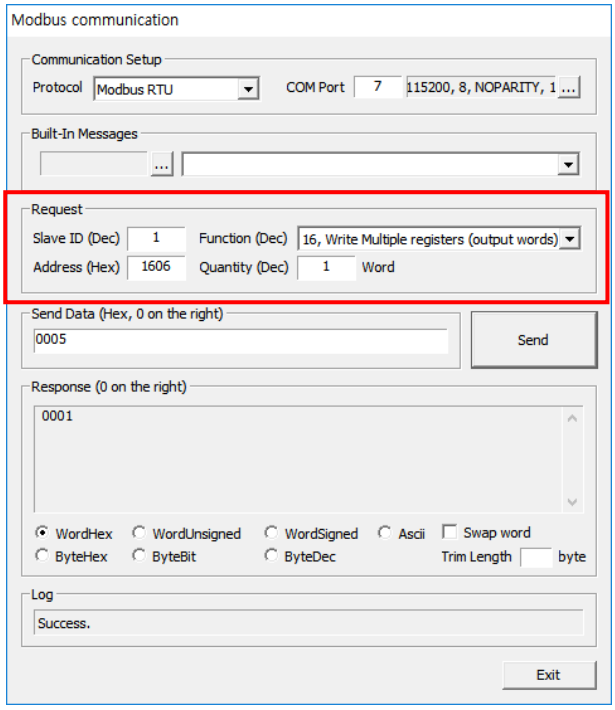
(1) Run '[Hitachi] -> [IOGuidePro] -> [Protocol Messenger] -> [Modbus]'

(2) Write the value of each.

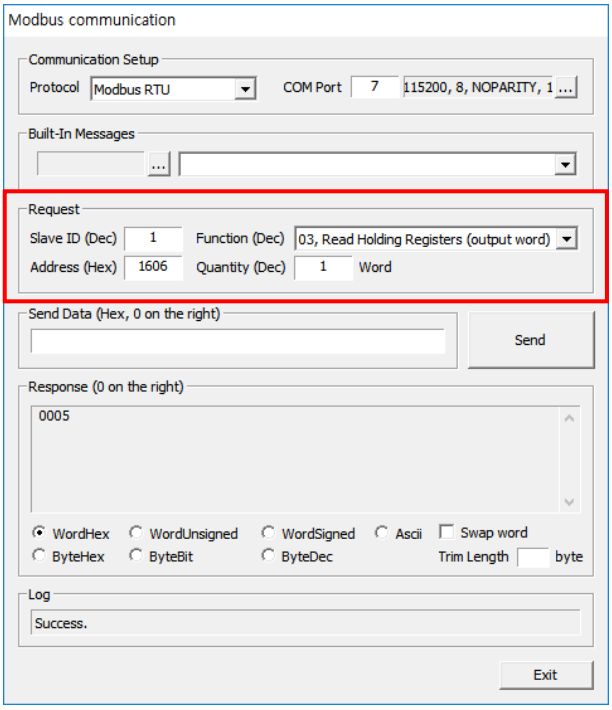
*Protocol : Modbus RTU
*ComPort : User Port / Baud rate : 115200(default)
*Address(Hex) : 1606 (RS232 Baud rate Register)
: 1607 (RS232 Use bit Setting Register)
: 1608 (RS485 Baud rate Register)
: 1609 (RS485 Use bit Setting Register)
*Function(Dec) : When the value is write - 16, Write Multiple registers
When the value is read - 03, Read Holding Registers

(3) confirm the necessary information.

***When the value is write**
Write the desired value and click the Send button.



***When the value is read**
Click the Send button and confirm the value.



6.7. Memory Reset

Data field 0x55AA makes the remote device to restart with factory default setup of EEPROM.

*All expansion slot configuration parameters are cleared.

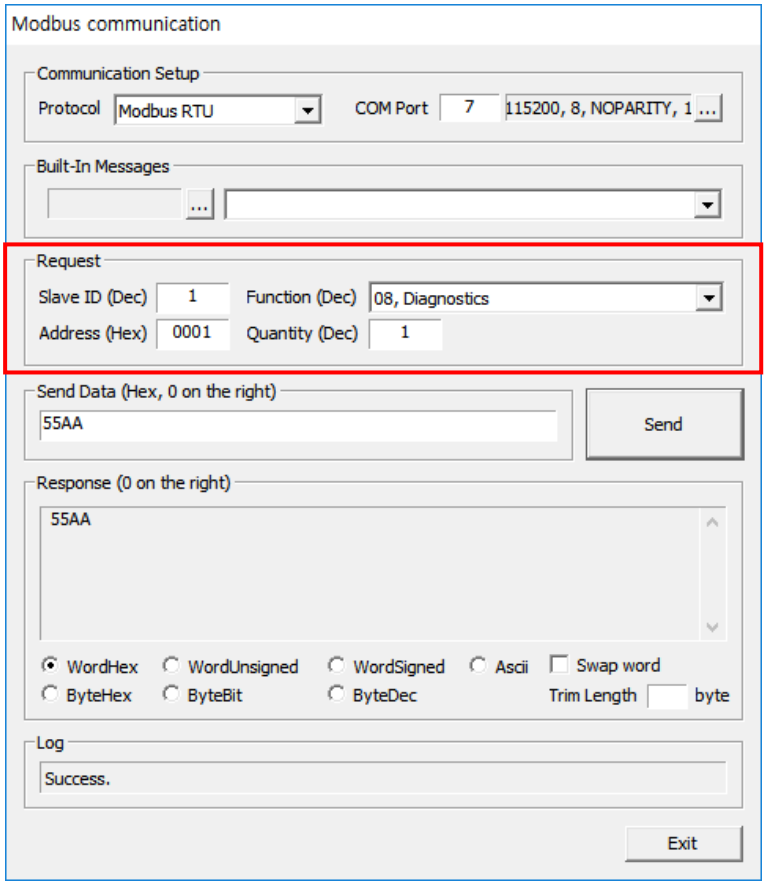
(1) Run '[Hitachi] -> [IOGuidePro] -> [Protocol Messenger] -> [Modbus]'

(2) Write the value of each.
*Protocol : Modbus RTU
*ComPort : User Port / Baudrate : 115200(default)
*Address(Hex) : 0001 (Factory default setup)
*Length(Hex) : 0002
*Function(Dec) : When the value is write – 08, Diagnostics

(3) Write the register value and click the 'Send' button.
*Value : 0x55AA + "Sumcheck" – Program Reset

*Value : 0x55AA + 0xAB7B + "Sumcheck" – Factory Reset

(Please refer to A.4.2 for Sumcheck (0x1006).



6.8. RTC(Real Time Clock) Function

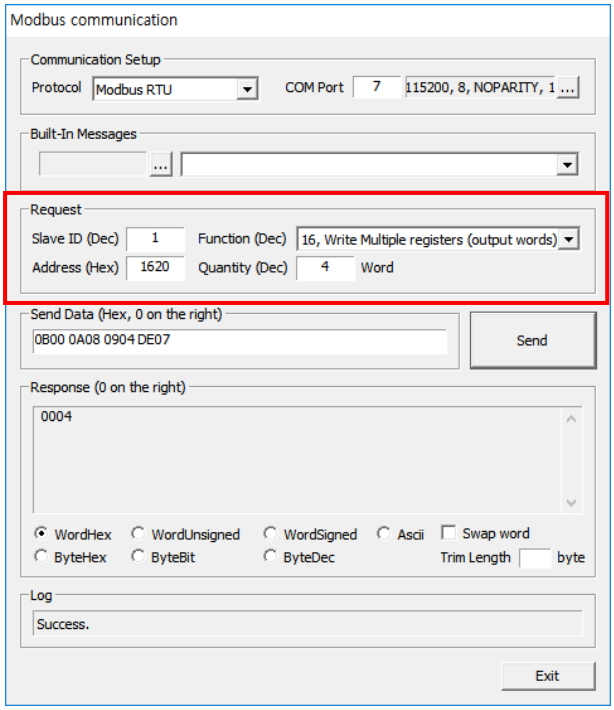
A real-time clock (RTC) is a computer clock (most often in the form of an integrated circuit) that keeps track of the current time. RTC information of RIO3-CP3MBT is stored in address 0x1620 in the Register, also can be read.

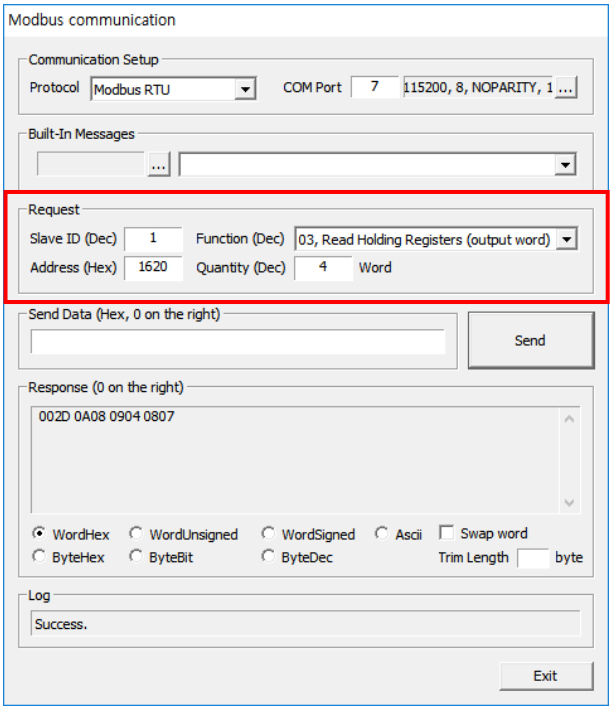
(1) Run '[Hitachi] -> [IOGuidePro] -> [Protocol Messenger] -> [Modbus]'

(2) Write the value of each.
*Protocol : Modbus TCP, Modbus RTU
*ComPort : User Port / Baudrate : 115200(default)
*Address(Hex) : 1620 (RTC Register)
*Function(Dec) : When the value is write - 16, Write Multiple registers
When the value is read - 03, Read Holding Registers

(3) confirm the necessary information.
*When the value is write
Write the desired value and click the Send button.

*When the value is read
Click the Send button and confirm the value.



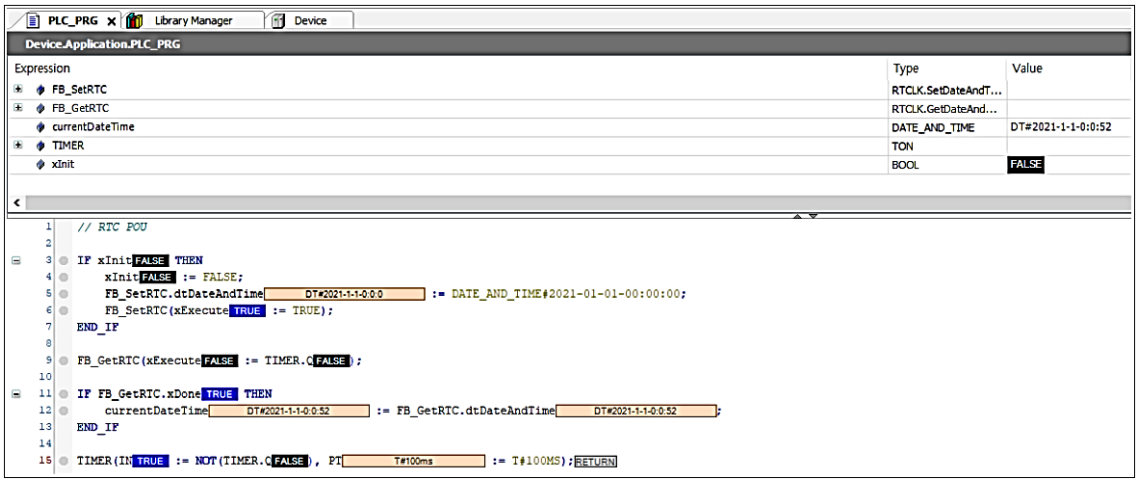


RTC can be checked & set using the following methods;

(4) CODESYS

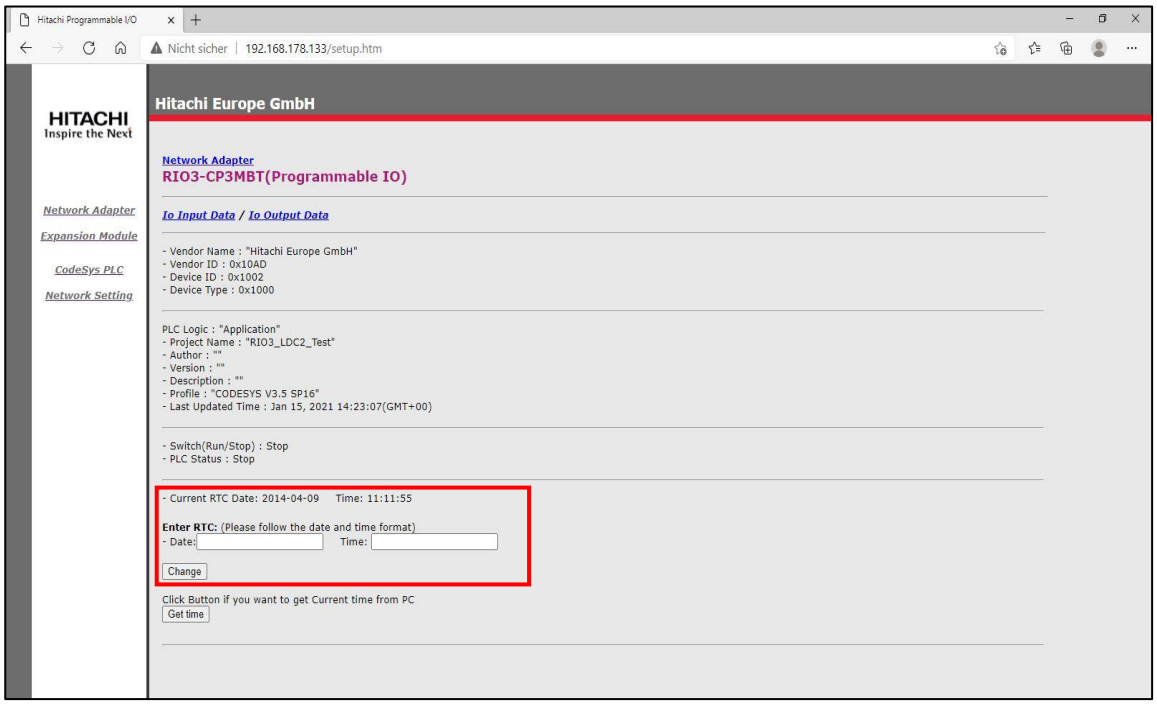
Verify & change the current time.

- 2021Y 01M 01D 00h 00m 00s
- RTV value every 100ms



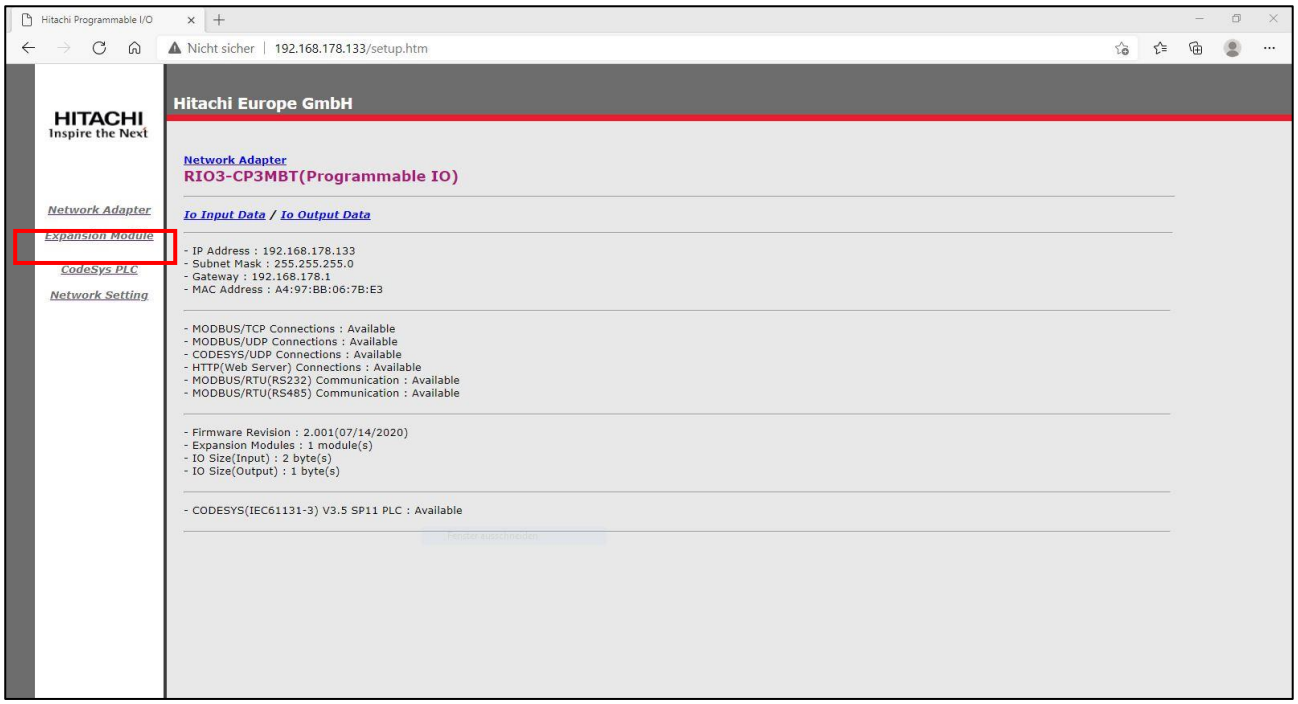
(5) Web-Server

Verify & change the current time.

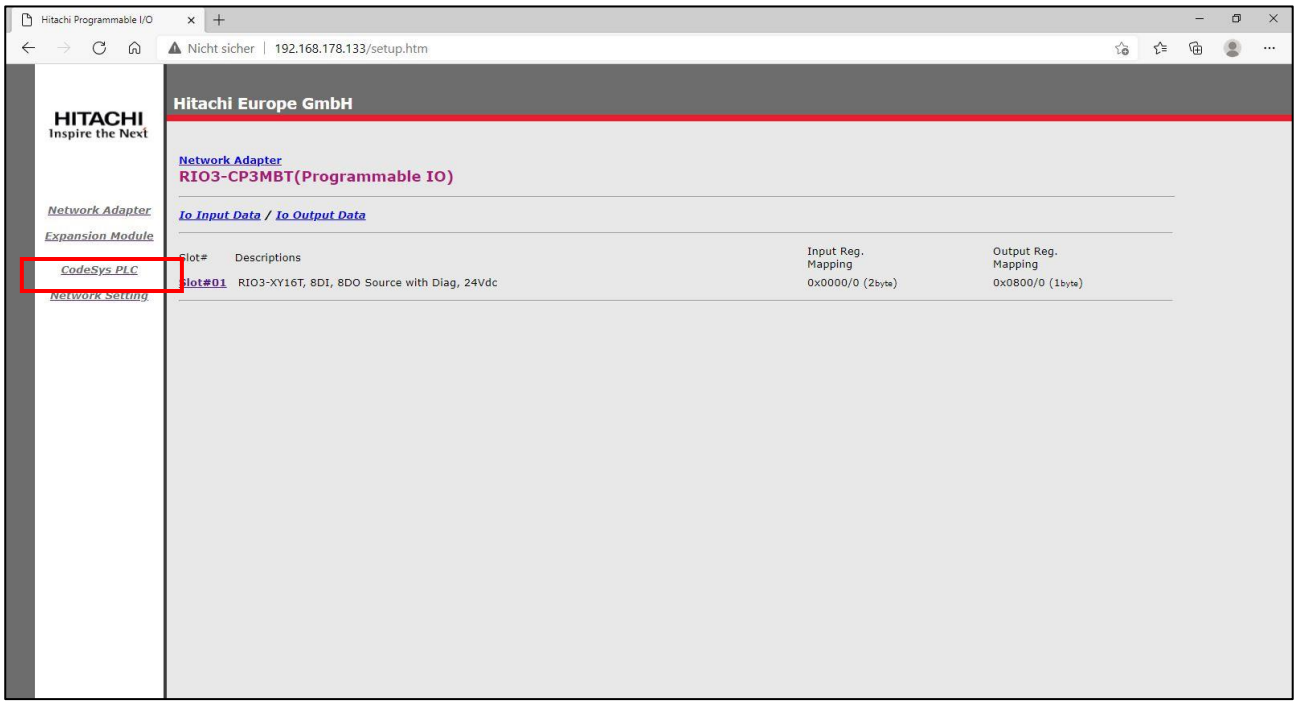


6.9. Webserver

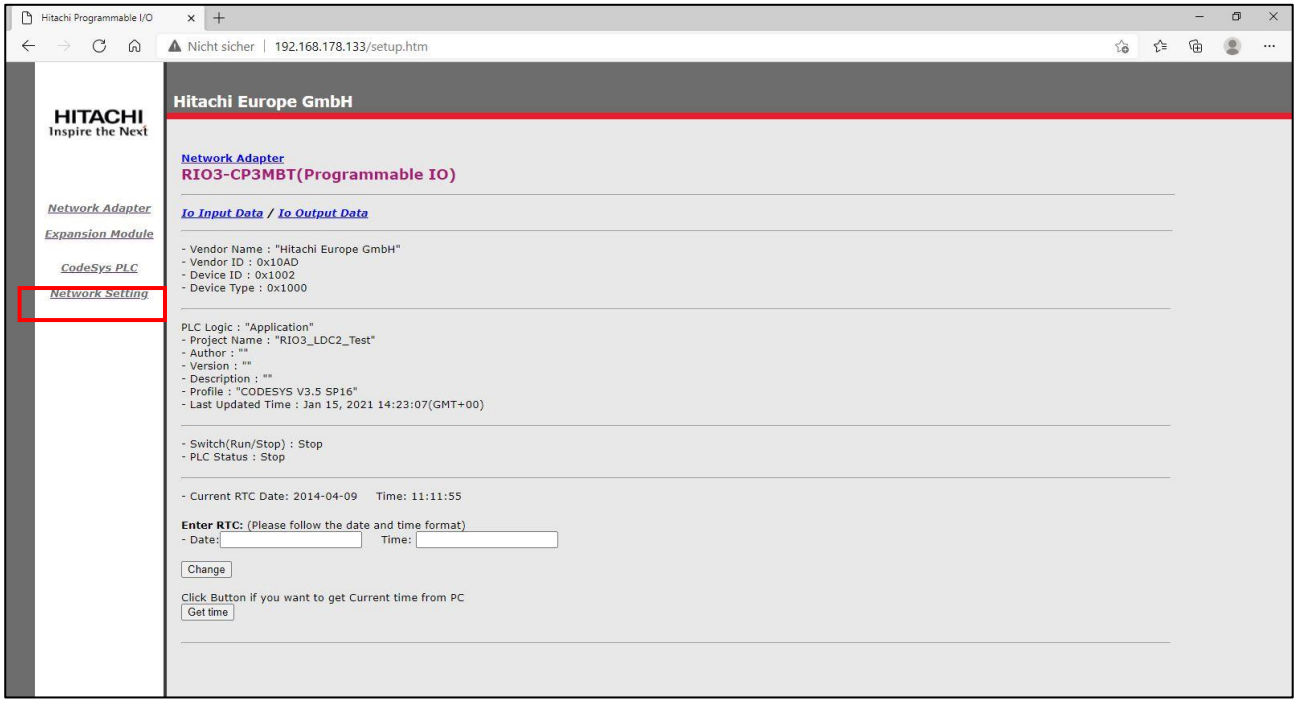
(1) Main page is showing various information for PIO status.
(2) To access the webserver, ***IP Address/setup.htm*** (is required) (e.g. "*192.168.178.133/setup.htm")*



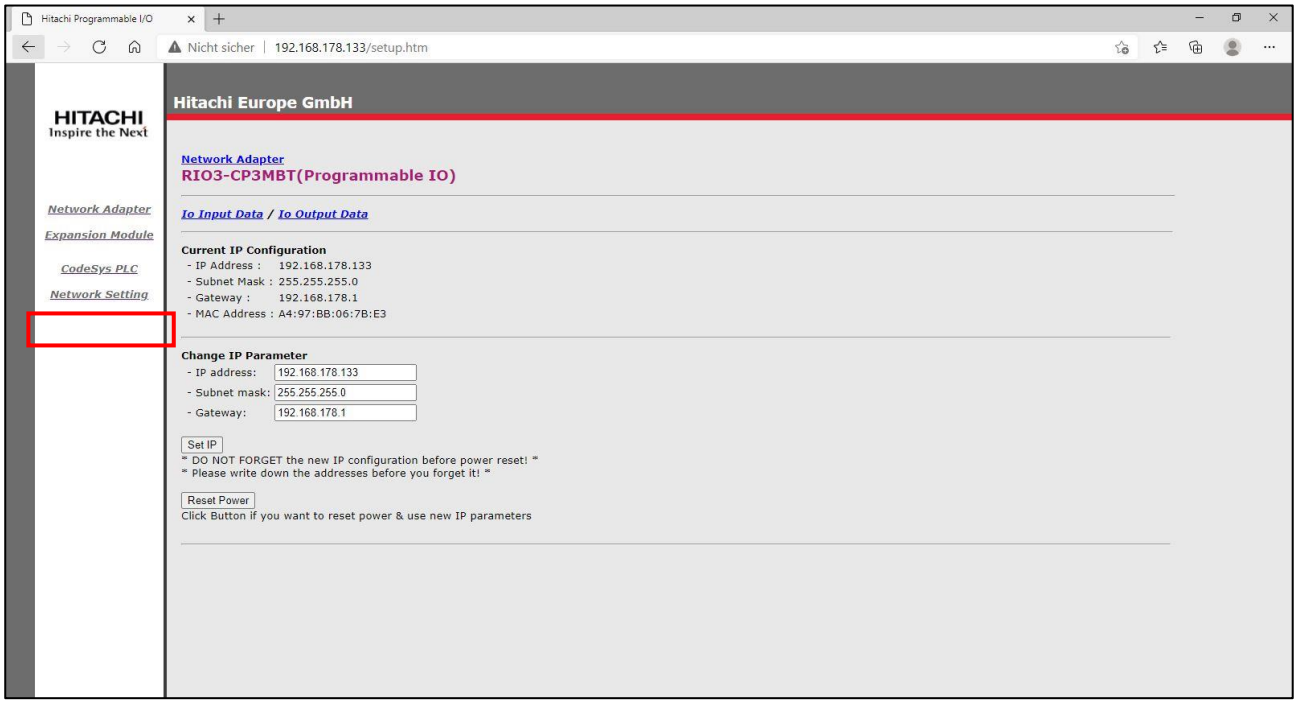
(3) When you click the expansion menu, user can check the extension module status.



(4) User can easily change and set the RTC time. If you click the ‘Get Time’ button, the clock time from PC will be adapted to PIO.



(5) Using the Network setting, user can set the IP/Subnet mask/Gateway.



6.10. IP Default Setting

When user forget the IP address, After power on the PIO and push the Reset switch in front of PIO.

The switch should be pushed for 20 seconds at least. PIO will be fall into the factory default mode.

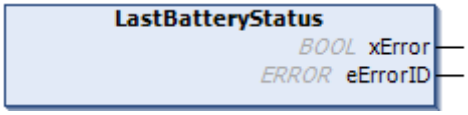
All of PIO LED will blink in Green/Red.
The default IP setting is

IP address	192.168.100.100
Subnet Mask	255.255.255.0
Gateway	192.168.100.254

7. External Feature Library

7.1. Battery State of charge

❖ RTC data or RETAIN data will not be saved if the battery is not charged enough.



Scope	Name	Type	Comment
Output	xError	BOOL	TRUE: Battery charging required FALSE: Battery charged
Output	eErrorID	Error	(1) No_Err (2) Battery_Reset_RTC (3) Battery_Reset_Retain (4) Battery_Reset_RTC_And_Retain

(1) Code about Battery State

```
PROGRAM Battery
VAR
    Battery_FLAG : BOOL;
    Battery : ExtFeatures.LastBatteryStatus;
    Error : BOOL;
    ErrorCode : ExtFeatures.ERROR;
END_VAR
```

```
IF Battery_FLAG = TRUE THEN
    Battery();
    Error := Battery.xError;
    ErrorCode := Battery.eErrorID;
    Battery_FLAG := FALSE;
END_IF
```

(2) Result

❖ IF xError : It means “True”, so the battery should be charged through supplying the system power.

Battery	ExtFeatures.LastBat...	
xError	BOOL	TRUE
eErrorID	ERROR	BATTERY_RESET_RTC_AND_RETAIN
Error	BOOL	TRUE
ErrorCode	ERROR	BATTERY_RESET_RTC_AND_RETAIN

7.2. Read / Write Special Register (PIO Parameter Access)

- The MODBUS Special Register Map listed in the specification can do “read / write” by programming logic through the library as the followings ;

7.2.1. Read Special Register

Scope	Name	Type	Comment
Input	wAddress	WORD	Special register address
	wSize	WORD	Word size to read
Output	wBuffer	ARRAY [0..127] OF WORD	read Buffer
	wRegSize	WORD	
	eErrorID	ERROR	(1) No_Err : No Error (2) Invalid_Address : Incorrect address – unsupported address (3) Invalid_Size : Incorrect access size

7.2.2. Write Special Register

Scope	Name	Type	Comment
Input	wAddress	WORD	Special register address
	wSize	WORD	Word size to write
	wBuffer	ARRAY [0..127] OF WORD	Write buffer
Output	wRegSize	WORD	
	eErrorID	ERROR	(1) No_Err : No Error (2) Invalid_Address : Incorrect address – unsupported address (3) Invalid_Size : Incorrect access size

7.2.3. Example to change Special Register

(1) Test to change the RS-232 Baud rate – 7 (115200)

Address	Access	Type, Size	Description
0x1600(5632)	Read	2words	IP Address. (ex : A8C0 6464 = 192.168.100.100)
0x1602(5634)	Read	2words	Subnet Mask. (ex : FFFF 0000 = 255.255.0.0)
0x1604(5636)	Read	2words	Gate way. (ex : A8C0 0100 = 192.168.0.1)
0x1606(5638)*	Read /Write	1word	RS-232C Baud rate. (2400bps~115200bps) - 0 : 115200 (default) - 1 : 2400 - 2 : 4800 - 3 : 9600 - 4 : 19200 - 5 : 38400 - 6 : 57600 - 7 : 115200

(2) Write Special Register sample logic

❖ After setting the value, the login & download is required.

- wAddress : Special Register Address = 16#1606
- wSize : Special Register size = 1
- wBuffer : buffer that perform write = 16#0007(115200)

```
(*================== Write Special Register ==================*)
IF Write_Flag THEN
    Write.wAddress      := 16#1606;// special register address
    Write.wSize         := 1;      // word to write
    Write.wBuffer[0]    := 16#0007;
    Write();
    Write_Flag := FALSE;
END_IF
```

(3) Read Slot Parameter sample logic

❖ Log-in is required in order to verify that the Write is successful

- wAddress : Special Register Address = 16#1606
- wSize : Special Register size = 1
- wBuffer : buffer that perform Read

```
IF Read_Flag THEN
    Read.wAddress    := 16#1606; // speci
    Read.wSize       := 1;       // word
    Read();
    Read_Flag        := FALSE;
END_IF
```

wBuffer[0]	WORD	16#0007
wBuffer[1]	WORD	16#0000
wBuffer[2]	WORD	16#0000

7.3. Read / Write Slot Parameter

- **Caution** : Continuous writing of the I/O parameter can cause problems in the I/O slot. So it needs to be used only when required.

7.3.1. Read Slot Parameter

Scope	Name	Type	Comment
Input	bSlotAddress	BYTE	Slot address
	bSize	BYTE	Byte size to read
Output	bParamSize	BYTE	parameter size
	bBuffer	ARRAY [0..254] OF BYTE	read Buffer
	eErrorID	ERROR	(1) No_Err : No Error (2) Invalid_Address : Incorrect slot address – 0 or exceeded number of expansion slots (3) Invalid_Size : Incorrect parameter size – 0 or exceeded parameter size

7.3.2. Write Slot Parameter

Scope	Name	Type	Comment
Input	bSlotAddress	BYTE	Slot address
	bOffset	BYTE	Offset
	bSize	BYTE	Byte size to write
	bBuffer	ARRAY [0..254] OF BYTE	write buffer
Output	bParamSize	BYTE	parameter size
	eErrorID	ERROR	(1) No_Err : No Error (2) Invalid_Address : Incorrect slot address – 0 or exceeded number of expansion slots (3) Invalid_Size : Incorrect parameter size – (bSize or bSize+bOffset) > parameter size of slot

7.3.3. Example to access the extended I/O module parameter

RIO3-AXH4V
Valid Parameter length: 6 Bytes

Parameter Data

Bit No	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Byte 0	Voltage Range for Channel 0 (H00: 0~10Vdc, H01: 0~5Vdc, H02: 1~5Vdc)							
Byte 1	Voltage Range for Channel 1 (H00: 0~10Vdc, H01: 0~5Vdc, H02: 1~5Vdc)							
Byte 2	Voltage Range for Channel 2 (H00: 0~10Vdc, H01: 0~5Vdc, H02: 1~5Vdc)							
Byte 3	Voltage Range for Channel 3 (H00: 0~10Vdc, H01: 0~5Vdc, H02: 1~5Vdc)							
Byte 4	Filter Time (H00: Default Filter (20) / H01: Fastest ~ / H3E : Slowest)							
Byte 5	Not used (=00)							

(1) Create a WriteslotParam sample logic

- ❖ After setting the value, the login & download is required.

- bSlotaddress : Slot Address : 1 (if first slot : 1 / if 10 th slot : 10)
- bSize : Data size of the parameters of the Module : 6
- boffset : offset : 0
- bBuffer[0] : Channel 0 Parameter Byte : H0(0~10V)
- bBuffer[1] : Channel 1 Parameter Byte : H1(0~5V)
- bBuffer[2] : Channel 2 Parameter Byte : H2(1~5V)

```
// ############ Write slot parameter ##################
IF Write_Flag THEN
    Writeslotparam.bslotAddress := 1; // Slot Address (First slot)
    Writeslotparam.bSize        := 6; // number of bytes to write
    Writeslotparam.bOffset      := 0; // offset
    // HEX 00 : 0~10V / H01 : 0~5V / H02 : 1~5V
    Writeslotparam.bBuffer[0]   := 16#00; //parameter byte 0
    Writeslotparam.bBuffer[1]   := 16#01; //parameter byte 1
    Writeslotparam.bBuffer[2]   := 16#02; //parameter byte 2
    Writeslotparam.bBuffer[3]   := 16#00; //parameter byte 3
    Writeslotparam.bBuffer[4]   := 16#00; //parameter byte 4
    Writeslotparam();
```

(2) Create a ReadslotParam sample logic

- ❖ Log-in is required in order to verify that the Write is successful

- bsize : Set the parameter data size to read
- bSlotaddress : Slot Address(if first slot : 1 / if 10 th slot : 10)

```
IF Read_Flag THEN
    MemUtils.memset(ADR(Read_Data[0]), 0, SIZEOF(Read_Data))

    ReadSlotParam.bSlotAddress := 1;    // Slot Address (Fir
    ReadSlotParam.bSize := 6;           // number of bytes t
    ReadSlotParam();

    MemUtils.SysMemCpy(ADR(Read_Data[0]),ADR(ReadSlotParam.b
    Read_Flag := FALSE;
END_IF
```

Read_Data[0]	BYTE	16#00
Read_Data[1]	BYTE	16#01
Read_Data[2]	BYTE	16#02
Read_Data[3]	BYTE	16#03
Read_Data[4]	BYTE	16#00
Read_Data[5]	BYTE	16#00

8. Upgrade Firmware

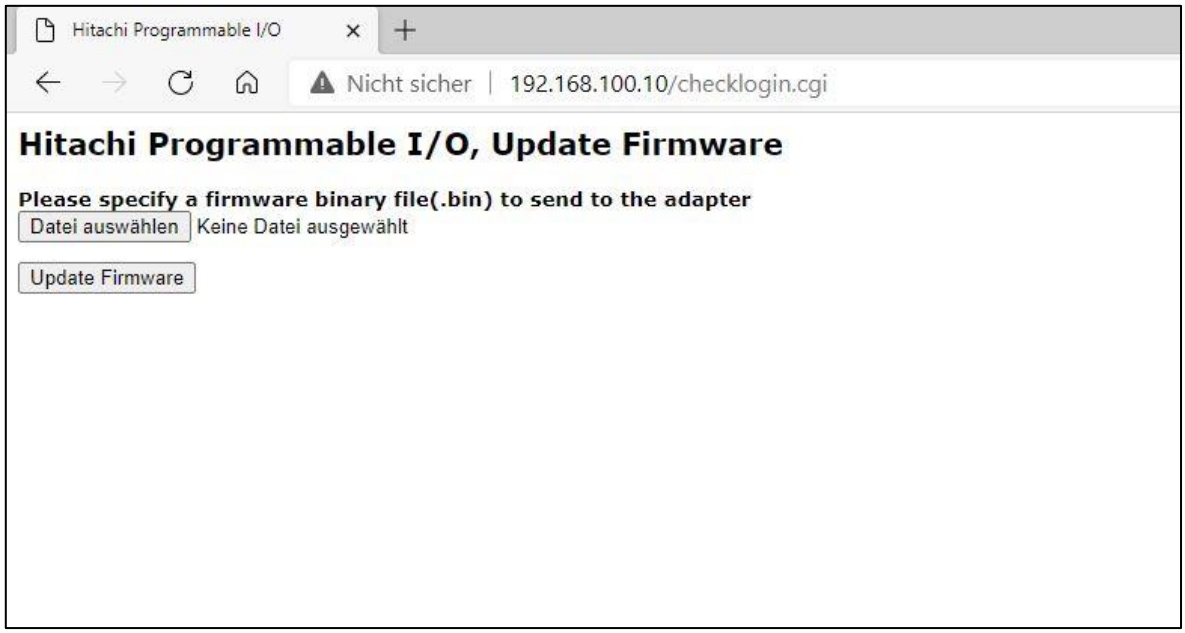
8.1. Using IAP over Ethernet

(1) Apply a power with pushing a reset button(Mod LED will blink Green/Red).

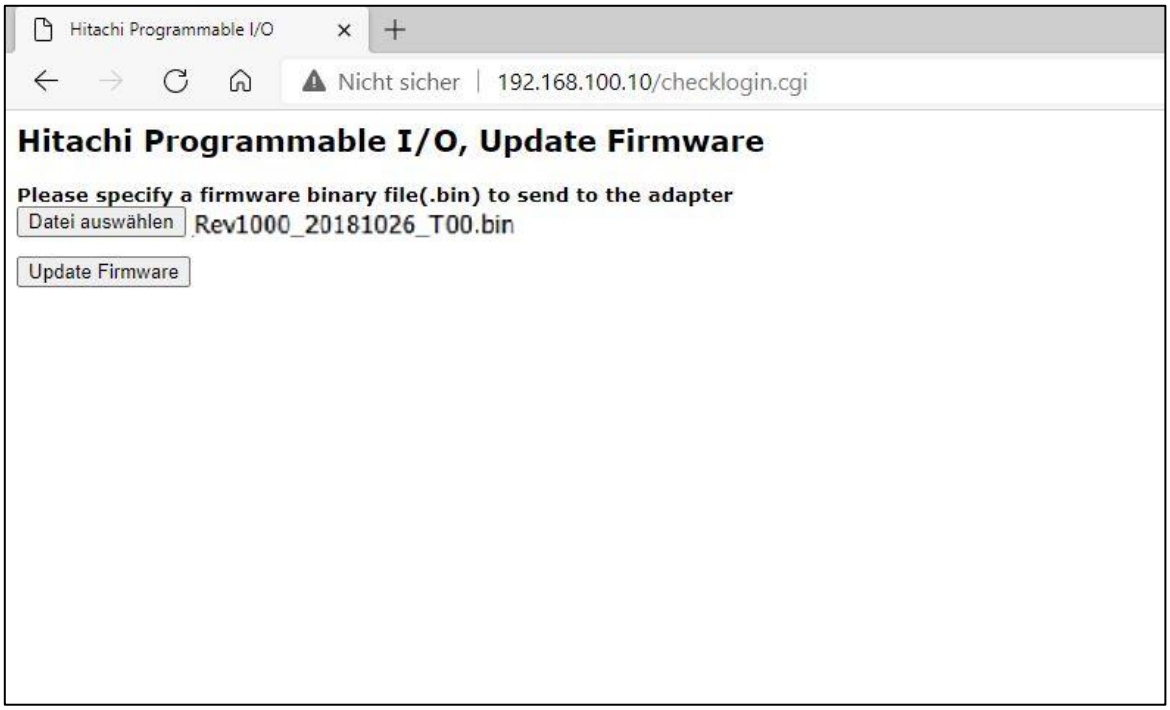
(2) Execute Firefox.(It is recommended to use Firefox)

(3) Connect to 192.168.100.10 and login **(User ID :Hitachi / Password : Hitachi)**

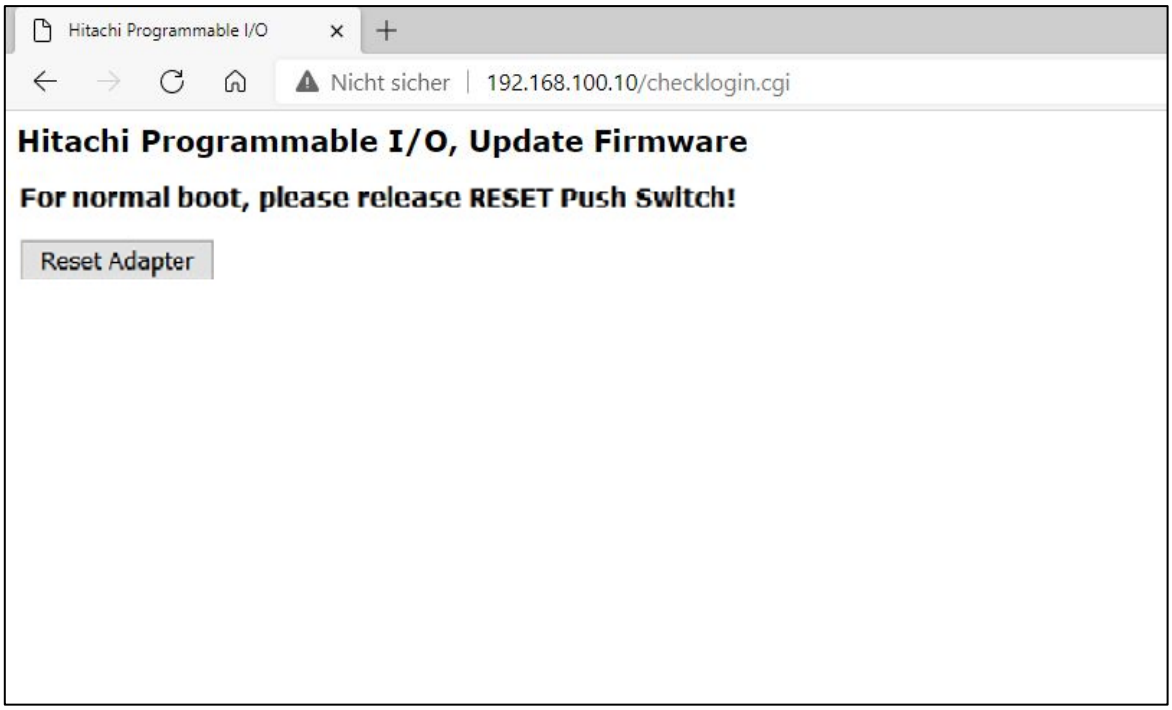
(4) Search the file to download using search button.



(5) Click Update Firmware Button.



(6) If it finish, you can see a below message (File Upload Done!)
And click 'Reset Adapter' button.



9. CPU Usage

CPU load can be checked as the following and below is an example

(1) PIO RTOS Task Scheduling

Main Tasks related to CPU Usage

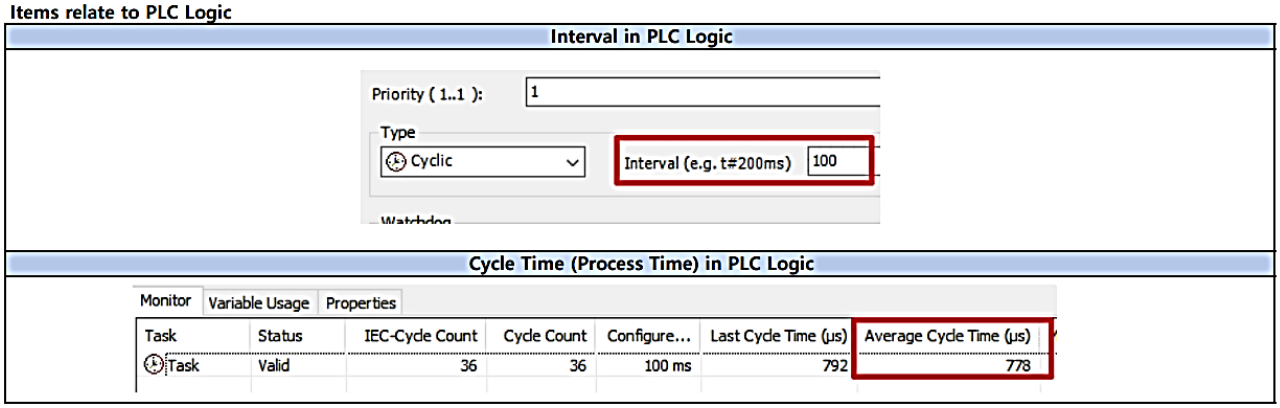
PRIORITY	RTOS TASK	DESCIPTION
HIGH	CODESYS PLC	CODESYS Runtime System + PLC Logic (User PLC Program)
MID	IO SCAN	Communicate & Update with Expansion I/O Modules
LOW	OPC UA SERVER	OPC UA server

* Based on the priority of the task, lower-priority tasks will grant the CPU usage only after higher-priority tasks have utilized their required resources.

* CODESYS PLC > IO SCAN > OPC UA SERVER

(2) CODESYS PLC TASK's CPU Usage

Items relate to PLC Logic



CODESYS_PLC (CPU Usage %)	Default CPU usage + (Remain CPU Usage / Interval * Cycle Time) = 30 + (70 / Interval * Cycle Time)

* The CPU Usage of 'CODESYS PLC TASK' is determined by the following : Default CPU Usage (CODESYS Runtime System) + Interval & Cycle Time in PLC Logic(User PLC Program).

* Default CPU Usage(CODESYS Runtime System) : All processes required for running CODESYS except the User PLC Program. It utilizes a fixed CPU usage of about 30%.

* All tests were conducted on F/W rev 3.003.

10. Troubleshooting

10.1. How to diagnose by LED indicator

LED Status	Cause	Action
All LED turns off	No power	Check main power Cable
	System power is not supplied.	Contact Sales team and send module for repair.
MOD LED flashes green	Failure of initialization EEPROM parameter.	Contact Sales team and send module for repair.
MOD LED flashes red	- Excess of expansion slot - Excess of IO size - Wrong IO composition - Occurrence of EEPROM checksum error	- Use expansion slot up to 63. - Compose that IO total size is not excess. - Check composition I/O Module
MOD LED is red	- Wrong address ID - Occurrence critical error in firmware	Contact Sales team and send module for repair.
I/O LED turns off	- Failure of realization expansion Module - None expansion Module	Check connector status both NA series and expansion module.
I/O LED flashes red	Failure of configuration baud rate	- Check communication cable with Master - Check power for master.
	Failure of initialization I/O	- Use expansion slot up to 63. - Compose that IO total size is not excess.
		NA series notice unidentified expansion module ID. Check status of expansion module.
I/O LED is red	Failure of exchanging I/O data	Check status of expansion IO connection.
RUN LED flashed Green	PLC program stop	Check the toggle switch is up.
RUN LED flashed red	Failure of Module Configuration	Check the module hardware and software configurations are the same.

10.2. How to diagnose when device couldn’t communicate

Inspection of wrong or omission cable connection

- Check status of cable connection for each node.
- Check that all color matches between connector and cable.
- Check wire omission.

Configuration of Node address

- Check duplication node address.

Configuration of Master

- Check configuration of master
- Check whether to do download or don’t
- Check composition is right
  - Configuration of communication baud rate
  - I/O size
  - Configuration of each node

Ground and environment

- Check ground is contacted
- Check environment factor (temperature, humidity, etc.) is in less than regular limit

11. MODBUS INTERFACE

11.1. MODBUS Interface Register / Bit Map

11.1.1. Register Map

Start Address	Read/Write	Description	Func. Code
0x0000 ~	Read	Process input image registers (Real Input Register)	4, 23
0x0800 ~	Read/Write	Process output image registers (Real Output Register)	3, 16, 23
0x1000 ~	Read	Adapter Identification special registers.	3, 4, 23
0x1020 ~	Read/Write	Adapter Watchdog, other time special register.	3, 4, 6, 16, 23
0x1100 ~	Read/Write	Adapter Information special registers.	3, 4, 6, 16, 23
0x2000 ~	Read/Write	Expansion Slot Information special registers.	3, 4, 6, 16, 23

* The special register map must be accessed by read/write of every each address (one address).

11.1.2. Bit Map

Start Address	Read/Write	Description	Func. Code
0x0000 ~	Read	Process input image bits All input registers area is addressable by bit address. Size of input image bit is size of input image register * 16.	2
0x0800 ~	Read/Write	Process output image bits All output registers area is addressable by bit address. Size of output image bit is size of output image register * 16.	1, 5, 15

11.2. MODBUS Transmission Mode

Two different serial transmission modes are defined: The RTU mode and the ASCII mode. It defines the bit contents of message fields transmitted serially on the line. It determines how information is packed into the message fields and decoded.

11.2.1. RTU Transmission Mode

When devices communicate on a MODBUS serial line using the RTU (Remote Terminal Unit) mode, each 8–bit byte in a message contains two 4–bit hexadecimal characters. The main advantage of this mode is that its greater character density allows better data throughput than ASCII mode for the same baudrate. Each message must be transmitted in a continuous stream of characters.

Start	Address	Function	Data	CRC Check	End
≥ 3.5 chars	1 char	1 char	Up to 252 chars	2 chars	≥ 3.5 chars

11.2.2. ASCII Transmission Mode

When devices are setup to communicate on a MODBUS serial line using ASCII (American Standard Code for Information Interchange) mode, each 8–bit byte in a message is sent as two ASCII characters. This mode is used when the physical communication link or the capabilities of the device does not allow the conformance with RTU mode requirement regarding timers management.

Start	Address	Function	Data	CRC Check	End
1 char	2 chars	2 chars	Up to 252 chars	2 chars	2 chars CR,LF

11.3. Supported MODBUS Function Codes

Function Code	Function	Description	Unicast / Broadcast
1 (0x01)	Read Coils	Read output bit	Unicast
2 (0x02)	Read Discrete Inputs	Read input bit	Unicast
3 (0x03)	Read Holding Registers	Read output word	Unicast
4 (0x04)	Read Input Registers	Read input word	Unicast
5 (0x05)	Write Single Coil	Write one bit output	Unicast / Broadcast
6 (0x06)	Write Single Register	Write one word output	Unicast / Broadcast
8 (0x08)	Diagnostics (Serial Line only)	Read diagnostic register	Unicast
15 (0x0F)	Write Multiple Coils	Write a number of output bits	Unicast / Broadcast
16 (0x10)	Write Multiple registers	Write a number of output words	Unicast / Broadcast
23 (0x17)	Read / Write Multiple register	Read a number of input words / Write a number of output words	Unicast

- Refer to MODBUS APPLICATION PROTOCOL SPECIFICATION V1.1a

11.3.1. 1 (0x01) Read Coils

This function code is used to read from 1 to 2000 contiguous status of coils in a remote device. The Request PDU specifies the starting address, i.e. the address of the first coil specified, and the number of coils. In the PDU Coils are addressed starting at zero. Therefore coils numbered 1-16 are addressed as 0-15. The coils in the response message are packed as one coil per bit of the data field. Status is indicated as 1= ON and 0= OFF.

- **Request**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x01	0x01
Starting Address Hi	0x10	0x10
Starting Address Lo	0x00	0x00
Quantity of Outputs Hi	0x00	0x00
Quantity of Outputs Lo	0x0A	0x0A
Error Check (CRC/LRC)	-	0xB8, 0xAB
End of Frame	-	t1-t2-t3

- **Response**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x01	0x01
Byte Count	0x02	0x02
Output Status	0x55	0x55
Output Status	0x02	0x02
Error Check (CRC/LRC)	-	0x8F, 0x6D
End of Frame	-	t1-t2-t3

* In case of address 0x1015~0x1000 output bit value: 00000010_01010101.

11.3.2. 2 (0x02) Read Discrete Inputs

This function code is used to read from 1 to 2000 contiguous status of discrete inputs in a remote device. The Request PDU specifies the starting address, i.e. the address of the first input specified, and the number of inputs. In the PDU Discrete Inputs are addressed starting at zero. Therefore Discrete inputs numbered 1-16 are addressed as 0-15. The discrete inputs in the response message are packed as one input per bit of the data field.
Status is indicated as 1= ON; 0= OFF.

- **Request**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x02	0x02
Starting Address Hi	0x00	0x00
Starting Address Lo	0x00	0x00
Quantity of Inputs Hi	0x00	0x00
Quantity of Inputs Lo	0x0A	0x0A
Error Check (CRC/LRC)	-	0xF8, 0x6B
End of Frame	-	t1-t2-t3

- **Response**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x02	0x02
Byte Count	0x02	0x02
Input Status	0x80	0x80
Input Status	0x00	0x00
Error Check (CRC/LRC)	-	0x50, 0x78
End of Frame	-	t1-t2-t3

- In case of address 0x0015~0x0000 output bit value: 00000000_10000000.

11.3.3. 3 (0x03) Read Holding Registers

This function code is used to read the contents of a contiguous block of holding registers in a remote device. The Request PDU specifies the starting register address and the number of registers.
The register data in the response message are packed as two bytes per register, with the binary contents right justified within each byte. For each register, the first byte contains the high order bits and the second contains the low order bits.

- **Request**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x03	0x03
Starting Address Hi	0x08	0x08
Starting Address Lo	0x00	0x00
Quantity of Register Hi	0x00	0x00
Quantity of Register Lo	0x02	0x02
Error Check (CRC/LRC)	-	0xC6, 0x0D
End of Frame	-	t1-t2-t3

- **Response**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x03	0x03
Byte Count	0x04	0x04
Output Register #0 Hi	0x11	0x11
Output Register #0 Lo	0x22	0x22
Output Register #1 Hi	0x33	0x33
Output Register #1 Lo	0x44	0x44
Error Check (CRC/LRC)	-	0x2D, 0xC6
End of Frame	-	t1-t2-t3

- In case of address 0x0800, 0x0801 output register value: 0x1122, 0x3344.

11.3.4. 4 (0x04) Read Input Registers

This function code is used to read from 1 to approx. 125 contiguous input registers in a remote device. The Request PDU specifies the starting register address and the number of registers. The register data in the response message are packed as two bytes per register, with the binary contents right justified within each byte. For each register, the first byte contains the high order bits and the second contains the low order bits.

- **Request**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x04	0x04
Starting Address Hi	0x00	0x00
Starting Address Lo	0x00	0x00
Quantity of Register Hi	0x00	0x00
Quantity of Register Lo	0x02	0x02
Error Check (CRC/LRC)	-	0x71, 0xAD
End of Frame	-	t1-t2-t3

- **Response**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x04	0x04
Byte Count	0x04	0x04
Input Register #0 Hi	0x00	0x00
Input Register #0 Lo	0x80	0x80
Input Register #1 Hi	0x00	0x00
Input Register #1 Lo	0x00	0x00
Error Check (CRC/LRC)	-	0x9C, 0x6C
End of Frame	-	t1-t2-t3

- In case of address 0x0000, 0x0001 input register value: 0x0080, 0x0000.

11.3.5. 5 (0x05) Write Single Coil

This function code is used to write a single output to either ON or OFF in a remote device. The requested ON/OFF state is specified by a constant in the request data field. A value of FF 00 hex requests the output to be ON. A value of 00 00 requests it to be OFF. All other values are illegal and will not affect the output.

- **Request**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x05	0x05
Starting Address Hi	0x10	0x10
Starting Address Lo	0x01	0x01
Quantity of Outputs Hi	0xFF	0xFF
Quantity of Outputs Lo	0x00	0x00
Error Check (CRC/LRC)	-	0xD9, 0x5C
End of Frame	-	t1-t2-t3

- **Response**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x05	0x05
Output Address Hi	0x10	0x10
Output Address Lo	0x01	0x01
Output Value Hi	0xFF	0xFF
Output Value Lo	0x00	0x00
Error Check (CRC/LRC)	-	0xD9, 0x5C
End of Frame	-	t1-t2-t3

- Output bit of address 0x1001 turns ON.

11.3.6. 6 (0x06) Write Single Register

This function code is used to write a single holding register in a remote device. Therefore register numbered 1 is addressed as 0. The normal response is an echo of the request, returned after the register contents have been written.

- **Request**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x06	0x06
Starting Address Hi	0x08	0x08
Starting Address Lo	0x00	0x00
Quantity of Outputs Hi	0x11	0x11
Quantity of Outputs Lo	0x22	0x22
Error Check (CRC/LRC)	-	0x07, 0x85
End of Frame	-	t1-t2-t3

- **Response**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x06	0x06
Output Address Hi	0x08	0x08
Output Address Lo	0x00	0x00
Output Value Hi	0x11	0x11
Output Value Lo	0x22	0x22
Error Check (CRC/LRC)	-	0x07, 0x85
End of Frame	-	t1-t2-t3

- In case of address 0x0800 outputs register value: 0x0000 changes to 0x1122.

11.3.7. 8 (0x08) Diagnostics

MODBUS function code 08 provides a series of tests for checking the communication system between a client (Master) device and a server (Slave), or for checking various internal error conditions within a server.
The function uses a two–byte sub-function code field in the query to define the type of test to be performed. The server echoes both the function code and sub-function code in a normal response. Some of the diagnostics cause data to be returned from the remote device in the data field of a normal response.

- **Request**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x08	0x08
Sub-Function Hi	0x00	0x00
Sub-Function Lo	0x00	0x00
Data Hi	0x11	0x11
Data Lo	0x22	0x22
Error Check (CRC/LRC)	-	0x6C, 0x24
End of Frame	-	t1-t2-t3

- **Response**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x08	0x08
Sub-Function Hi	0x00	0x00
Sub-Function Lo	0x00	0x00
Data Hi	0x11	0x11
Data Lo	0x22	0x22
Error Check (CRC/LRC)	-	0x6C, 0x24
End of Frame	-	t1-t2-t3

✓ **Sub-function 0x0000(0) Return Query Data**
The data passed in the request data field is to be returned (looped back) in the response.
The entire response message should be identical to the request.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x0000(0)	Any	Echo Request Data	

✓ **Sub-function 0x0001(1) Restart Communications Option**

The remote device could be initialized and restarted, and all of its communications event counters are cleared. Especially, data field 0x55AA makes the remote device to restart with factory default setup of EEPROM.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x0001(1)	0x0000, 0xFF00	Echo Request Data	Reset
0x0001(1)	0x55AA	Echo Request Data	Reset with Default setting[1)]
0x0001(1)	0x55AA+0xAB7B+sumcheck[4)]	Echo Request Data	Reset with Factory Default[2)]
0x0001(1)	0x55AA+0xAA55+sumcheck[4)]	Echo Request Data	Reset with Factory Default[3)]

1),2),3)All expansion slot configuration parameters are cleared.

2),3) IP address, Subnet Mask Address, Gateway Address, RS232/485 setting, and BOOTP/DHCP mode will be the factory default value

3)Mac address will be the factory default value

4)Refer the A.4.2 sum check(0x1006)

✓ **Sub-function 0x000A(10) Clear Counters and Diagnostic Register**

The goal is to clear all counters and the diagnostic register. Counters are also cleared upon power–up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000A(10)	0x0000	Echo Request Data	

✓ **Sub-function 0x000B(11) Return Bus Message Count**

The response data field returns the quantity of messages that the remote device has detected on the communications system since its last restart, clear counters operation, or power–up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000B(11)	0x0000	Total Message Count	

✓ **Sub-function 0x000D(13) Return Bus Exception Error Count**

The response data field returns the quantity of MODBUS exception responses returned by the remote device since its last restart, clear counters operation, or power–up.
Exception responses are described and listed in section 6.2.11.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000D(13)	0x0000	Exception Error Count	

✓ **Sub-function 0x000E(14) Return Slave Message Count**

The response data field returns the quantity of messages addressed to the remote device, or broadcast, that the remote device has processed since its last restart, clear counters operation, or power–up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000E(14)	0x0000	Slave Message Count	

✓ **Sub-function 0x000F(15) Return Slave No Response Count**

The response data field returns the quantity of messages addressed to the remote device for which it has returned no response (neither a normal response nor an exception response), since its last restart, clear counters operation, or power–up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000F(15)	0x0000	Slave No Response Count	

✓

✓ **Sub-function 0x0064(100) Return Slave MODBUS, Extension module Status**

The response data field returns the status of MODBUS and Extension module addressed to the remote device.

This status values are identical with status 1word of input process image. Refer to 5.3.1.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x0064(100)	0x0000	MODBUS, Extension module Status	Same as status 1word

11.3.8. 15 (0x0F) Write Multiple Coils

This function code is used to force each coil in a sequence of coils to either ON or OFF in a remote device. The Request PDU specifies the coil references to be forced. Coils are addressed starting at zero. A logical '1' in a bit position of the field requests the corresponding output to be ON. A logical '0' requests it to be OFF. The normal response returns the function code, starting address, and quantity of coils forced.

- **Request**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x0F	0x0F
Starting Address Hi	0x10	0x10
Starting Address Lo	0x00	0x00
Quantity of Outputs Hi	0x00	0x00
Quantity of Outputs Lo	0x0A	0x0A
Byte Count	0x02	0x02
Output Value #0	0x55	0x55
Output Value #1	0x01	0x01
Error Check (CRC/LRC)	-	0x21, 0XC9
End of Frame	-	t1-t2-t3

- **Response**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x0F	0x0F
Starting Address Hi	0x10	0x10
Starting Address Lo	0x00	0x00
Quantity of Outputs Hi	0x00	0x00
Quantity of Outputs Lo	0x0A	0x0A
Error Check (CRC/LRC)	-	0xD1, 0x6A
End of Frame	-	t1-t2-t3

- In case of address 0x1015~0x1000 output bit value: 00000000_00000000 changes to 00000001_01010101.

11.3.9. 16 (0x10) Write Multiple Registers

This function code is used to write a block of contiguous registers (1 to approx. 120 registers) in a remote device.
The requested written values are specified in the request data field. Data is packed as two bytes per register.
The normal response returns the function code, starting address, and quantity of registers written.

- **Request**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x0F	0x0F
Starting Address Hi	0x08	0x08
Starting Address Lo	0x00	0x00
Quantity of Registers Hi	0x00	0x00
Quantity of Registers Lo	0x02	0x02
Byte Count	0x04	0x04
Register Value #0 Hi	0x11	0x11
Register Value #0 Lo	0x22	0x22
Register Value #1 Hi	0x33	0x33
Register Value #1 Lo	0x44	0x44
Error Check (CRC/LRC)	-	0x3B, 0x12
End of Frame	-	t1-t2-t3

- **Response**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x0F	0x0F
Starting Address Hi	0x08	0x08
Starting Address Lo	0x00	0x00
Quantity of Outputs Hi	0x00	0x00
Quantity of Outputs Lo	0x02	0x02
Error Check (CRC/LRC)	-	0x43, 0xCE
End of Frame	-	t1-t2-t3

- In case of address 0x0800, 0x0801 output register value: 0x0000, 0x0000 changes to 0x1122, 0x3344.

11.3.10. 23 (0x17) Read/Write Multiple Registers

This function code performs a combination of one read operation and one write operation in a single MODBUS transaction. The write operation is performed before the read. The request specifies the starting address and number of holding registers to be read as well as the starting address, number of holding registers, and the data to be written. The byte count specifies the number of bytes to follow in the write data field.
The normal response contains the data from the group of registers that were read. The byte count field specifies the quantity of bytes to follow in the read data field.

- **Request**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x17	0x17
Read Starting Address Hi	0x08	0x08
Read Starting Address Lo	0x00	0x00
Quantity of Read Hi	0x00	0x00
Quantity of Read Lo	0x02	0x02
Write Starting Address Hi	0x08	0x08
Write Starting Address Lo	0x00	0x00
Quantity of Write Hi	0x00	0x00
Quantity of Write Lo	0x02	0x02
Byte Count	0x04	0x04
Write Reg. Value #0 Hi	0x11	0x11
Write Reg. Value #0 Lo	0x22	0x22
Write Reg. Value #1 Hi	0x33	0x33
Write Reg. Value #1 Lo	0x44	0x44
Error Check (CRC/LRC)	-	0x88, 0x3F
End of Frame	-	t1-t2-t3

- **Response**

Field name	Example	RTU
Start of Frame	-	t1-t2-t3
Slave Address	0x07	0x07
Function Code	0x17	0x17
Byte Count	0x04	0x04
Write Reg. Value #0 Hi	0x11	0x11
Write Reg. Value #0 Lo	0x22	0x22
Write Reg. Value #1 Hi	0x33	0x33
Write Reg. Value #1 Lo	0x44	0x44
Error Check (CRC/LRC)	-	0x2E, 0xD2
End of Frame	-	t1-t2-t3

- In case of address 0x0800, 0x0801 output register value: 0x0000, 0x0000 changes to 0x1122, 0x3344.

11.4. MODBUS Special Register Map

The special register map can be accessed by function code 3, 4, 6 and 16. Also the special register map must be accessed by read/write of every each address (one address).

11.4.1. Adapter Register Mapping

Address	IEC Address	Contents
0x0000~0x07FF	%IW0~%IW2047	2048 words Input and Internal memory (Area is write-protected)
0x0800~0x0FFF	%QW0~%QW2047	2048 words Output and Internal memory (Area is write-enabled)
0x1000~0x1FFF	-	Special Function Register (PIO Information)
0x2000~0x2FFF	-	Special Function Register (Slot Information)
0x4000~0x5FFF	%MW0~%MW8191	2048 words Internal memory (Area is write-enabled)

11.4.2. Adapter Identification Special Register (0x1000, 4096)

Address	Access	Type, Size	Description
0x1000(4096)	Read	1 word	Vendor ID = 0x02E5(741), Hitachi. Co., Ltd.
0x1001(4097)	Read	1 word	Device Type = 0x000C, Network Adapter
0x1002(4098)	Read	1 word	Product Code = 0x9120(RIO3-CP3MBT)
0x1003(4099)	Read	1 word	Firmware revision, if 0x0101, revision 1.001
0x1005(4101)	Read	String up to 34bytes	Product name string First 1word is length of valid character string Example) response as following “00 1D 52 4E 2D 39 32 32 32 2C 50 72 6F 66 69 62 75 73 20 41 64 61 70 74 65 72 2C 52 42 55 53 00 00 000” Valid character size = 0x0017 =29 characters
0x1006(4102)	Read	1 word	Sum check of EEPROM
0x1010(4112)	Read	2 words	Firmware release date
0x101E(4126)	Read	15words	Composite Id of following address 0xA8C0(Lo_IPAddr) 0x3264(Hi_IPAddr), 0xFFFF(Lo_NetMask) 0x00FF(Hi_NetMask), 0xA8C0(GateWay) 0xFE64(GateWay), 0x1400(MacAddr) 0x00F7(MacAddr), 0xBA83(MacAddr) 0x02E5(VendorCode), 0x000C(DeviceType) 0x9120(ProductCode), 0x0200(FW_Rev) 0x0420(FW_ReleasData), 0x2018(FW_ReleasYear)

* String Type consists of valid string length (first 1word) and array of characters.

11.4.4. Adapter Information Special Register (0x1100, 4352)

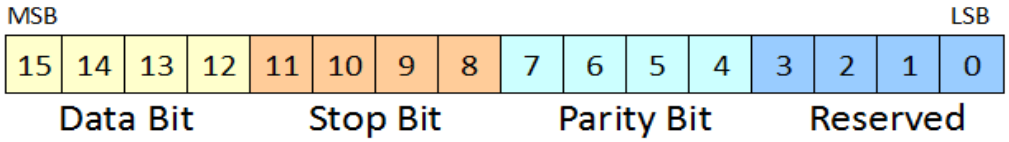
Address	Access	Type, Size	Description
0x1102(4354)	Read	1word	Start address of input image word register. =0x0000
0x1103(4355)	Read	1word	Start address of output image word register. =0x0800
0x1104(4356)	Read	1word	Size of input image word register.
0x1105(4357)	Read	1word	Size of output image word register.
0x1106(4358)	Read	1word	Start address of input image bit. = 0x0000
0x1107(4359)	Read	1word	Start address of output image bit. =0x1000
0x1108(4360)	Read	1word	Size of input image bit.
0x1109(4361)	Read	1word	Size of output image bit.
0x110D(4365)	Read/Write	1word	Field Power On/OFF, Run/Stop Switch, Reset Switch *Field Power On : 0x8010 / Field Power Off : 0x0000 *Stop : 0x0000 / Run : 0x0001 / Reset Switch : 0x0002 ex) 0x8013 : Field Power On + Run + Reset Switch ON
0x110E(4366)	Read	Up to 63 words	Expansion slot's RIO3-number
0x1110(4368)	Read	1word	Number of expansion slot
0x1113(4371)	Read	Up to 63 words	Expansion slot Module Id. Refer to Appendix A.1 Product List. First 1word is adapter's module ID.
0x111E(4382)	Read	1word	Reserved. Adapter IO identification vendor code.

* After the system is reset, the new "Set Value" action is applied.
* If the slot location is changed, set default value automatically (all expansion slots are live).

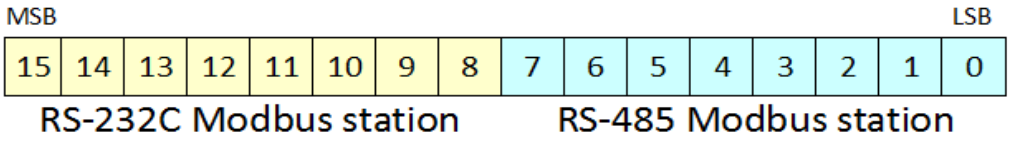
11.4.5. Adapter Setting Special Register (0x1600, 5632)

Address	Access	Type, Size	Description
0x1600(5632)	Read	2words	IP Address. (ex : C0A8 6464 = 192.168.100.100)
0x1602(5634)	Read	2words	Subnet Mask. (ex : FFFF FF00 = 255.255.255.0)
0x1604(5636)	Read	2words	Gate way. (ex : C0A8 0001 = 192.168.0.1)
0x1606(5638)*	Read/Write	1word	RS-232 Baudrate. (2400bps~115200bps) 0 : 115200(Default) 1 : 2400 2 : 4800 3 : 9600 4 : 19200 5 : 38400 6 : 57600 7 : 115200
0x1607(5639)*	Read/Write	1word	RS-232 Setting. - 1 nibble : Data bit(0 : 8bit(default), 1 : 9bit, 2 : 7bit) - 2 nibble : Stop bit(0 : 1bit(default), 1 : 2bit) - 3 nibble : Parity bit(0 : none(default), 1: odd, 2 : even) - 4 nibble : Reserve
0x1608(5640)	Read/Write	1word	RS-485 Baudrate. (2400bps~115200bps) 0 : 115200(Default) 1 : 2400 2 : 4800 3 : 9600 4 : 19200 5 : 38400 6 : 57600 7 : 115200
0x1609(5641)*	Read/Write	1word	RS-485 Setting. - 1 nibble : Data bit(0 : 8bit(default), 1 : 9bit, 2 : 7bit) - 2 nibble : Stop bit(0 : 1bit(default), 1 : 2bit) - 3 nibble : Parity bit(0 : none(default), 1: odd, 2 : even) - 4 nibble : Reserve
0x160A(5642)**	Read/Write	1word	MODBUS Station. - High 1byte : Station No. of RS-232C (default : 1) - Low 1byte : Station No. of RS-485 (default : 1)
0x160B(5643)	Read/Write	1word	IP Setting Method. - Not Use : 0x0000 - BOOTP : 0x8000 (default) - DHCP : 0x8001
0x1610(5648)	Read	3words	Mac Address (ex : 0014 F700 0101 = 00.14.F7.00.01.01)
0x1614(5652)	Read/Write	1word	Serial connection Method - 0x0000 : HITACHI Modbus/RTU(Default) - 0x8000 : RS232 Enable for CODESYS Function block / RTU Master CODESYS Serial Port Config Setting: COM Port 1 - 0x8001 : RS485 Enable for CODESYS Function block / RTU Master CODESYS Serial Port Config Setting: COM Port 2
0x1620(5664)	Read/Write	4words	RTC - 1 word : 00ss (ss : sec) - 2 word : hhmm (hh : hour, mm : min) - 3 word : mmdd (mm : month, dd : day) - 4 word : yyyy (yyyy : year) (ex : 0010 0F28 0317 07E0 = 2016 - 03.23 - 15:40 - 16)

***RS-232C/485 Setting :** This description for 0x1607/0x1609 register with bit.



Modbus Station : This description for 0x160A register with bit.



11.4.6. Expansion Slot Information Special Register (0x2000, 8192)

Each expansion slot has 0x20(32) address offset and same information structure.

Slot#1	0x2000(8192)~0x201F(8223)	Slot#2	0x2020(8224)~0x203F(8255)
Slot#3	0x2040(8256)~0x205F(8287)	Slot#4	0x2060(8288)~0x207F(8319)
Slot#5	0x2080(8320)~0x209F(8351)	Slot#6	0x20A0(8352)~0x20BF(8383)
Slot#7	0x20C0(8384)~0x20DF(8415)	Slot#8	0x20E0(8416)~0x20FF(8447)
Slot#9	0x2100(8448)~0x211F(8479)	Slot#10	0x2120(8480)~0x213F(8511)
Slot#11	0x2140(8512)~0x215F(8543)	Slot#12	0x2160(8544)~0x217F(8575)
Slot#13	0x2180(8576)~0x219F(8607)	Slot#14	0x21A0(8608)~0x21BF(8639)
Slot#15	0x21C0(8640)~0x21DF(8671)	Slot#16	0x21E0(8672)~0x21FF(8703)
Slot#17	0x2200(8704)~0x221F(8735)	Slot#18	0x2220(8736)~0x223F(8767)
Slot#19	0x2240(8768)~0x225F(8799)	Slot#20	0x2260(8800)~0x227F(8831)
Slot#21	0x2280(8832)~0x229F(8863)	Slot#22	0x22A0(8864)~0x22BF(8895)
Slot#23	0x22C0(8896)~0x22DF(8927)	Slot#24	0x22E0(8928)~0x22FF(8959)
Slot#25	0x2300(8960)~0x231F(8991)	Slot#26	0x2320(8992)~0x233F(9023)
Slot#27	0x2340(9024)~0x235F(9055)	Slot#28	0x2360(9056)~0x237F(9087)
Slot#29	0x2380(9088)~0x239F(9119)	Slot#30	0x23A0(9120)~0x23BF(9151)
Slot#31	0x23C0(9152)~0x23DF(9183)	Slot#32	0x23E0(9184)~0x23FF(9215)
Slot#33	0x2400(9216)~0x241F(9247)	Slot#34	0x2420(9248)~0x243F(9279)
Slot#35	0x2440(9280)~0x245F(9311)	Slot#36	0x2460(9312)~0x247F(9343)
Slot#37	0x2480(9344)~0x249F(9375)	Slot#38	0x24A0(9376)~0x24BF(9407)
Slot#39	0x24C0(9408)~0x24DF(9439)	Slot#40	0x24E0(9440)~0x24FF(9471)
Slot#41	0x2500(9472)~0x251F(9503)	Slot#42	0x2520(9504)~0x253F(9535)
Slot#43	0x2540(9536)~0x255F(9567)	Slot#44	0x2560(9568)~0x257F(9599)
Slot#45	0x2580(9600)~0x259F(9631)	Slot#46	0x25A0(9632)~0x25BF(9663)
Slot#47	0x25C0(9664)~0x25DF(9695)	Slot#48	0x25E0(9696)~0x25FF(9727)
Slot#49	0x2600(9728)~0x261F(9759)	Slot#50	0x2620(9760)~0x263F(9791)
Slot#51	0x2640(9792)~0x265F(9823)	Slot#52	0x2660(9824)~0x267F(9855)
Slot#53	0x2680(9856)~0x269F(9887)	Slot#54	0x26A0(9888)~0x26BF(9919)
Slot#55	0x26C0(9920)~0x26DF(9951)	Slot#56	0x26E0(9952)~0x26FF(9983)
Slot#57	0x2700(9984)~0x271F(10015)	Slot#58	0x2720(10016)~0x273F(10047)
Slot#59	0x2740(10048)~0x275F(10079)	Slot#60	0x2760(10080)~0x277F(10111)
Slot#61	0x2780(10112)~0x279F(10143)	Slot#62	0x27A0(10144)~0x27BF(10175)
Slot#63	0x27C0(10176)~0x27DF(10207)		

Address Offset	Expansion Slot#1	Expansion Slot#2	Expansion Slot#3	……..	Expansion Slot#62	Expansion Slot#63
+ 0x00(+0)	0x2000(8192)	0x2020(8224)	0x2040(8256)	…….	0x27A0(9120)	0x27C0(9152)
+ 0x01(+1)	0x2001(8193)	0x2021(8225)	0x2041(8257)	…….	0x27A1(9121)	0x27C1(9153)
+ 0x02(+2)	0x2002(8194)	0x2022(8226)	0x2042(8258)	…….	0x27A2(9122)	0x27C2(9154)
+ 0x03(+3)	0x2003(8195)	0x2023(8227)	0x2043(8259)	…….	0x27A3(9123)	0x27C3(9155)
+ 0x04(+4)	0x2004(8196)	0x2024(8228)	0x2044(8260)	…….	0x27A4(9124)	0x27C4(9156)
+ 0x05(+5)	0x2005(8197)	0x2025(8229)	0x2045(8261)	…….	0x27A5(9125)	0x27C5(9157)
+ 0x06(+6)	0x2006(8198)	0x2026(8230)	0x2046(8262)	…….	0x27A6(9126)	0x27C6(9158)
+ 0x07(+7)	0x2007(8199)	0x2027(8231)	0x2047(8263)	…….	0x27A7(9127)	0x27C7(9159)
+ 0x08(+8)	0x2008(8200)	0x2028(8232)	0x2048(8264)	…….	0x27A8(9128)	0x27C8(9160)
+ 0x09(+9)	0x2009(8201)	0x2029(8233)	0x2049(8265)	…….	0x27A9(9129)	0x27C9(9161)
+ 0x0A(+10)	0x200A(8202)	0x202A(8234)	0x204A(8266)	…….	0x27AA(9130)	0x27CA(9162)
+ 0x0B(+11)	0x200B(8203)	0x202B(8235)	0x204B(8267)	…….	0x27AB(9131)	0x27CB(9163)
+ 0x0C(+12)	0x200C(8204)	0x202C(8236)	0x204C(8268)	…….	0x27AC(9132)	0x27CC(9164)
+ 0x0D(+13)	0x200D(8205)	0x202D(8237)	0x204D(8269)	…….	0x27AD(9133)	0x27CD(9165)
+ 0x0E(+14)	0x200E(8206)	0x202E(8238)	0x204E(8270)	…….	0x27AE(9134)	0x27CE(9166)
+ 0x0F(+15)	0x200F(8207)	0x202F(8239)	0x204F(8271)	…….	0x27AF(9135)	0x27CF(9167)
+ 0x10(+16)	0x2010(8208)	0x2030(8240)	0x2050(8272)	…….	0x27B0(9136)	0x27D0(9168)
+ 0x11(+17)	0x2011(8209)	0x2031(8241)	0x2051(8273)	…….	0x27B1(9137)	0x27D1(9169)
+ 0x12(+18)	0x2012(8210)	0x2032(8242)	0x2052(8274)	…….	0x27B2(9138)	0x27D2(9170)
+ 0x13(+19)	0x2013(8211)	0x2033(8243)	0x2053(8275)	…….	0x27B3(9139)	0x27D3(9171)
+ 0x14(+20)	0x2014(8212)	0x2034(8244)	0x2054(8276)	…….	0x27B4(9140)	0x27D4(9172)
+ 0x15(+21)	0x2015(8213)	0x2035(8245)	0x2055(8277)	…….	0x27B5(9141)	0x27D5(9173)
+ 0x16(+22)	0x2016(8214)	0x2036(8246)	0x2056(8278)	…….	0x27B6(9142)	0x27D6(9174)
+ 0x17(+23)	0x2017(8215)	0x2037(8247)	0x2057(8279)	…….	0x27B7(9143)	0x27D7(9175)
+ 0x18(+24)	0x2018(8216)	0x2038(8248)	0x2058(8280)	…….	0x27B8(9144)	0x27D8(9176)
+ 0x19(+25)	0x2018(8217)	0x2038(8249)	0x2058(8281)	…….	0x27B9(9145)	0x27D9(9177)
+ 0x1A(+26)	0x201A(8218)	0x203A(8250)	0x205A(8282)	…….	0x27BA(9146)	0x27DA(9178)
+ 0x1B(+27)	0x201B(8219)	0x203B(8251)	0x205B(8283)	…….	0x27BB(9147)	0x27DB(9179)
+ 0x1C(+28)	0x201C(8220)	0x203C(8252)	0x205C(8284)	…….	0x27BC(9148)	0x27DC(9180)
+ 0x1D(+29)	0x201D(8221)	0x203D(8253)	0x205D(8285)	…….	0x27BD(9149)	0x27DD(9181)
+ 0x1E(+30)	0x201E(8222)	0x203E(8254)	0x205E(8286)	…….	0x27BE(9150)	0x27DE(9182)
+ 0x1F(+31)	0x201F(8223)	0x203F(8255)	0x205F(8287)	…….	0x23BF(9151)	0x27DF(9183)

Address Offset	Access	Type, Size	Description
+ 0x00(+0)	Read	1word	Slot module id. Refer to Appendix A.1 Product List.
+ 0x02(+2) **	Read	1word	Input start register address of input image word this slot.
+ 0x03(+3) **	Read	1word	Input word's bit offset of input image word this slot.
+ 0x04(+4) **	Read	1word	Output start register address of output image word this slot.
+ 0x05(+5) **	Read	1word	Output word's bit offset of output image word this slot.
+ 0x06(+6) **	Read	1word	Input bit start address of input image bit this slot.
+ 0x07(+7) **	Read	1word	Output bit start address of output image bit this slot.
+ 0x08(+8) **	Read	1word	Size of input bit this slot
+ 0x09(+9) **	Read	1word	Size of output bit this slot
+ 0x0A(+10)**	Read	n words	Read input data this slot
+ 0x0B(+11)**	Read/Write	n words	Read/write output data this slot
+ 0x0E(+14)	Read	1word	RIO3-number
+ 0x0F(+15)	Read	String Upto 72bytes	First 1word is length of valid character string.
+ 0x10(+16)	Read	1word	Size of configuration parameter byte
+ 0x11(+17)**	Read/Write	n words	Read/write Configuration parameter data, up to 8byte. ***
+ 0x17(+23)	Read	1word	Firmware Revision
+ 0x18(+24)	Read	1word	Expansion Module Revision
+ 0x19(+25)	Read	2word	Firmware release data

* After the system is reset, the new "Set Value" action is applied.

** Nothing of output, input, and memory or configuration parameter corresponding slot returns Exception 02.

*** Slot Configuration parameter saved by internal EEPROM during power cycle until slot position changed.

*** All of output modules and special modules have the slot configuration parameter data. Refer to Document. (RIO3_Configuration_Parameter_Memory_Register_Rev1.01)

- **IO Data Code Format (1 word)**

Item	#15	#14	#13	#12	#11	#10	#9	#8	#7	#6	#5	#4	#3	#2	#1	#0	Word
Field	Output IO code								Input IO code								
Field	Date Type		Data Length						Date Type		Data Length						
Example)																	
ST-3214	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0x0084
ST-1224	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0x00C4
ST-1228	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0x0041
ST-4123	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0x8200
ST-221F	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0x4200
ST-2324	1	1	0	0	0	1	0	0	1	1	0	0	0	1	0	0	0xC4C4

Input/output Data Type: 0 0: No I/O Data / 0 1: Byte Data / 1 0: Word Data / 1 1: Bit Data

Input/output Data Length: 0 0 0 0 0 0 0: 0 Bit/Byte/Word / 0 0 0 0 0 0 1: 1 Bit/Byte/Word / 0 0 0 0 0 1 0: 2 Bit/Byte/Word
0 0 0 0 0 1 1: 3 Bit/Byte/Word / 0 0 0 0 0 1 1: 3 Bit/Byte/Word

11.5. I/O Process Image Map

11.5.1. Example of Input Process Image (Input Register) Map

Input image data depends on slot position and expansion slot data type. Input process image data is only ordered by expansion slot position when input image mode is uncompressed (mode 0, 2). But, when input image mode is compressed (mode 1, 3), input process image data is ordered by expansion slot position and slot data type. Input process image mode can be set by special register 0x1114(4372). Refer to 6.3.3.

- **Example slot configuration**

Slot Address	Module Description
#0	MODBUS Adapter
#1	4-discrete input
#2	8-discrete input
#3	4-analog input
#4	16-discrete input
#5	4-discrete input
#6	8-discrete input
#7	4-discrete input
#8	4-analog input

- **Input Process Image Mode#0** (Status(1word) + Uncompressed Input Processing Data)

Status (1word)

Addr.	#15	#14	#13	#12	#11	#10	#9	#8	#7	#6	#5	#4	#3	#2	#1	#0
0x0000	EW	0	0	0	0	0	0	0	FP	Internal protocol Status						
0x0001	Discrete In 8pts (Slot#2)								Empty, Always 0				Discrete In 4pts (Slot#1)			
0x0002	Analog Input Ch0 high byte (Slot#3)								Analog Input Ch0 low byte (Slot#3)							
0x0003	Analog Input Ch1 high byte (Slot#3)								Analog Input Ch1 low byte (Slot#3)							
0x0004	Analog Input Ch2 high byte (Slot#3)								Analog Input Ch2 low byte (Slot#3)							
0x0005	Analog Input Ch3 high byte (Slot#3)								Analog Input Ch3 low byte (Slot#3)							
0x0006	Discrete In high 8pts (Slot#4)								Discrete In low 8pts (Slot#4)							
0x0007	Discrete In 8pts (Slot#6)								Empty, Always 0				Discrete In 4pts (Slot#5)			
0x0008	Analog Input Ch0 low byte (Slot#8)								Empty, Always 0				Discrete In 4pts (Slot#7)			
0x0009	Analog Input Ch1 low byte (Slot#8)								Analog Input Ch0 high byte (Slot#8)							
0x0010	Analog Input Ch2 low byte (Slot#8)								Analog Input Ch1 high byte (Slot#8)							
0x0011	Analog Input Ch3 low byte (Slot#8)								Analog Input Ch2 high byte (Slot#8)							
0x0012									Analog Input Ch3 high byte (Slot#8)							

✓ **Expansion Module Status :**

0: Normal Operation
1: Internal protocol Standby
2: Internal protocol Communication Fault
3: Slot Configuration Failed
4: No Expansion Slot

✓ **FP (Field Power) :**

0: 24Vdc Field Power On.
1: 24Vdc Field Power Off

✓ **EW (MODBUS Error Watchdog) :**

0: No Error Watchdog
1: Error Watchdog once more since its last restart, clear counters operation, or power-up.

- **Input Process Image Mode#1** (Status(1word) + Compressed Input Processing Data)

Status (1word)

Addr.	#15	#14	#13	#12	#11	#10	#9	#8	#7	#6	#5	#4	#3	#2	#1	#0
0x0000	EW	0	0	0	0	0	0	0	FP	Internal protocol Status						
0x0001	Analog Input Ch0 high byte (Slot#3)								Analog Input Ch0 low byte (Slot#3)							
0x0002	Analog Input Ch1 high byte (Slot#3)								Analog Input Ch1 low byte (Slot#3)							
0x0003	Analog Input Ch2 high byte (Slot#3)								Analog Input Ch2 low byte (Slot#3)							
0x0004	Analog Input Ch3 high byte (Slot#3)								Analog Input Ch3 low byte (Slot#3)							
0x0005	Analog Input Ch0 high byte (Slot#8)								Analog Input Ch0 low byte (Slot#8)							
0x0006	Analog Input Ch1 high byte (Slot#8)								Analog Input Ch1 low byte (Slot#8)							
0x0007	Analog Input Ch2 high byte (Slot#8)								Analog Input Ch2 low byte (Slot#8)							
0x0008	Analog Input Ch3 high byte (Slot#8)								Analog Input Ch3 low byte (Slot#8)							
0x0009	Discrete In low 8pts (Slot#4)								Discrete In 8pts (Slot#2)							
0x0010	Discrete In 8pts (Slot#6)								Discrete In high 8pts (Slot#4)							
0x0011					Discrete In 4pts (Slot#7)				Discrete In 4pts (Slot#5)				Discrete In 4pts (Slot#1)			

✓ **Input Assembly Priority :**

1) Analog Input Data (Word type)
2) 8 or 16 points Discrete Input Data (Byte type)
3) 4 points Input Data (Bit type)
4) 2 points Input Data (Bit type)

- **Input Process Image Mode#2** (Uncompressed Input Processing Data without Status), Default Input Image

Addr.	#15	#14	#13	#12	#11	#10	#9	#8	#7	#6	#5	#4	#3	#2	#1	#0
0x0000	Discrete In 8pts (Slot#2)								Empty, Always 0				Discrete In 4pts (Slot#1)			
0x0001	Analog Input Ch0 high byte (Slot#3)								Analog Input Ch0 low byte (Slot#3)							
0x0002	Analog Input Ch1 high byte (Slot#3)								Analog Input Ch1 low byte (Slot#3)							
0x0003	Analog Input Ch2 high byte (Slot#3)								Analog Input Ch2 low byte (Slot#3)							
0x0004	Analog Input Ch3 high byte (Slot#3)								Analog Input Ch3 low byte (Slot#3)							
0x0005	Discrete In high 8pts (Slot#4)								Discrete In low 8pts (Slot#4)							
0x0006	Discrete In 8pts (Slot#6)								Empty, Always 0				Discrete In 4pts (Slot#5)			
0x0007	Analog Input Ch0 low byte (Slot#8)								Empty, Always 0				Discrete In 4pts (Slot#7)			
0x0008	Analog Input Ch1 low byte (Slot#8)								Analog Input Ch0 high byte (Slot#8)							
0x0009	Analog Input Ch2 low byte (Slot#8)								Analog Input Ch1 high byte (Slot#8)							
0x0010	Analog Input Ch3 low byte (Slot#8)								Analog Input Ch2 high byte (Slot#8)							
0x0011									Analog Input Ch3 high byte (Slot#8)							

- **Input Process Image Mode#3** (Compressed Input Processing Data without Status)

Addr.	#15	#14	#13	#12	#11	#10	#9	#8	#7	#6	#5	#4	#3	#2	#1	#0
0x0000	Analog Input Ch0 high byte (Slot#3)								Analog Input Ch0 low byte (Slot#3)							
0x0001	Analog Input Ch1 high byte (Slot#3)								Analog Input Ch1 low byte (Slot#3)							
0x0002	Analog Input Ch2 high byte (Slot#3)								Analog Input Ch2 low byte (Slot#3)							
0x0003	Analog Input Ch3 high byte (Slot#3)								Analog Input Ch3 low byte (Slot#3)							
0x0004	Analog Input Ch0 high byte (Slot#8)								Analog Input Ch0 low byte (Slot#8)							
0x0005	Analog Input Ch1 high byte (Slot#8)								Analog Input Ch1 low byte (Slot#8)							
0x0006	Analog Input Ch2 high byte (Slot#8)								Analog Input Ch2 low byte (Slot#8)							
0x0007	Analog Input Ch3 high byte (Slot#8)								Analog Input Ch3 low byte (Slot#8)							
0x0008	Discrete In low 8pts (Slot#4)								Discrete In 8pts (Slot#2)							
0x0009	Discrete In 8pts (Slot#6)								Discrete In high 8pts (Slot#4)							
0x0010					Discrete In 4pts (Slot#7)				Discrete In 4pts (Slot#5)				Discrete In 4pts (Slot#1)			

* S-Series uses the byte-oriented register mapping.
* Size of input image bit is size of input image register *16.

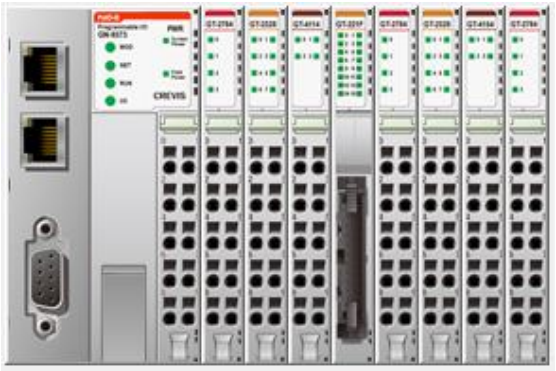
✓ **Input Assembly Priority :**

1) Analog Input Data (Word type)
2) 8 or 16 points Discrete Input Data (Byte type)
3) 4 points Input Data (Bit type)
4) 2 points Input Data (Bit type)

11.5.2. Example of Output Process Image (Output Register) Map

Output image data depends on slot position and expansion slot data type. Output process image data is only ordered by expansion slot position when output image mode is uncompressed (mode 0). But, when output image mode is compressed (mode 1), output process image data is ordered by expansion slot position and slot data type.
Output process image mode can be set by special register 0x1115(4373). Refer to 6.3.3.

- **For example slot configuration**



Slot Address	Module Description
#0	MODBUS Adapter
#1	4-discrete output
#2	8-discrete output
#3	4-analog output
#4	16-discrete output
#5	4-discrete output
#6	8-discrete output
#7	4-analog output
#8	4-discrete output

- **Output Process Image Mode#0** (Uncompressed Output Processing Data), **default output image**

Addr.	#15	#14	#13	#12	#11	#10	#9	#8	#7	#6	#5	#4	#3	#2	#1	#0
0x0800	Discrete out 8pts (Slot#2)								Empty, Don't care				Discrete out 4pts (Slot#1)			
0x0801	Analog out Ch0 high byte (Slot#3)								Analog out Ch0 low byte (Slot#3)							
0x0802	Analog out Ch1 high byte (Slot#3)								Analog out Ch1 low byte (Slot#3)							
0x0803	Analog out Ch2 high byte (Slot#3)								Analog out Ch2 low byte (Slot#3)							
0x0804	Analog out Ch3 high byte (Slot#3)								Analog out Ch3 low byte (Slot#3)							
0x0805	Discrete out high 8pts (Slot#4)								Discrete out low 8pts (Slot#4)							
0x0806	Discrete out 8pts (Slot#6)								Empty, Don't care				Discrete out 4pts (Slot#5)			
0x0807	Analog out Ch0 high byte (Slot#7)								Analog out Ch0 low byte (Slot#7)							
0x0808	Analog out Ch1 high byte (Slot#7)								Analog out Ch1 low byte (Slot#7)							
0x0809	Analog out Ch2 high byte (Slot#7)								Analog out Ch2 low byte (Slot#7)							
0x0810	Analog out Ch3 high byte (Slot#7)								Analog out Ch3 low byte (Slot#7)							
0x0811	Empty, Don't care								Empty, Don't care				Discrete out 4pts (Slot#8)			

- **Output Process Image Mode#1** (Compressed Output Processing Data)

Addr.	#15	#14	#13	#12	#11	#10	#9	#8	#7	#6	#5	#4	#3	#2	#1	#0
0x0800	Analog out Ch0 high byte (Slot#3)								Analog out Ch0 low byte (Slot#3)							
0x0801	Analog out Ch1 high byte (Slot#3)								Analog out Ch1 low byte (Slot#3)							
0x0802	Analog out Ch2 high byte (Slot#3)								Analog out Ch2 low byte (Slot#3)							
0x0803	Analog out Ch3 high byte (Slot#3)								Analog out Ch3 low byte (Slot#3)							
0x0804	Analog out Ch0 high byte (Slot#7)								Analog out Ch0 low byte (Slot#7)							
0x0805	Analog out Ch1 high byte (Slot#7)								Analog out Ch1 low byte (Slot#7)							
0x0806	Analog out Ch2 high byte (Slot#7)								Analog out Ch2 low byte (Slot#7)							
0x0807	Analog out Ch3 high byte (Slot#7)								Analog out Ch3 low byte (Slot#7)							
0x0808	Discrete out low 8pts (Slot#4)								Discrete out 8pts (Slot#2)							
0x0809	Discrete out 8pts (Slot#6)								Discrete out high 8pts (Slot#4)							
0x0810	Empty, Don't care				Discrete out 4pts (Slot#8)				Discrete out 4pts (Slot#5)				Discrete out 4pts (Slot#1)			

* S-Series uses the byte-oriented register mapping.
* Size of input image bit is size of input image register *16.

✓ **Output Assembly Priority :**

1) Analog Output Data (Word type)
2) 8 or 16 points Discrete Output Data (Byte type)
3) 4 points Output Data (Bit type)
4) 2 points Output Data (Bit type)

11.6. MODBUS Reference

MODBUS Reference Documents
http://www.Modbus.org

MODBUS Tools
http://www.Modbustools.com, MODBUS poll
http://www.win-tech.com, MODSCAN32

A. APPENDIX A - Product List

Please refer the separate HX-RIO3 product list document