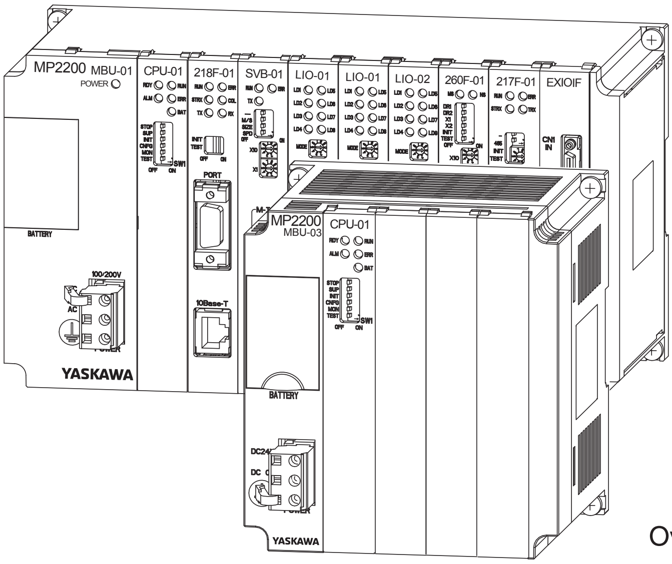
YASKAWA

Machine Controller MP2200
USER'S MANUAL



MANUAL NO. SIEP C880700 14H

Using this Manual

The MP2200 is a Machine Controller that can be greatly expanded by mounting up to nine Optional Modules per rack and connecting up to four racks. The CPU Module for the MP2200 achieves a high-speed motion control cycle of 0.5 ms and can control up to 256 axes.
Please read this manual to ensure correct usage of the MP2200 system and apply to your manufacturing system for control. Keep this manual in a safe place for future reference.

■ Basic Terms

Unless otherwise specified, the following definitions are used:

- MP2200: MP2200 Machine Controller
- MPE720: The Programming Device Software or a Programming Device (i.e., a personal computer) running the Programming Device Software
- PLC: Programmable Logic Controller

■ Manual Configuration

Read the chapters of this manual as required by the purpose.

Chapter	Selecting Models and Peripheral Devices	Studying Specifications and Ratings	Designing the System	Installation and Wiring	Trial Operation	Maintenance and Inspection
Chapter 1 Overview	√	–	–	–	–	–
Chapter 2 Overview of Specifications and Functions	√	√	√	√	–	–
Chapter 3 Mounting and Wiring	–	√	√	√	–	–
Chapter 4 System Startup	√	–	–	–	√	–
Chapter 5 Overview of System Operation	–	–	√	–	√	–
Chapter 6 Built-in Ethernet Communication (Supported by the CPU-03 and CPU-04)	–	–	√	–	√	–
Chapter 7 Easy Programming (Supported by the CPU-03 and CPU-04)	–	–	√	–	√	–
Chapter 8 Maintenance, Inspection, and Troubleshooting	–	–	√	–	√	√

For information on how to use motion parameters and motion commands, refer to *Machine Controller MP2000-series Built-in SVB/SVB-01 Motion Module User's Manual* (manual no.: SIEP C880700 33).

■ Engineering Tool Used in this Manual

The displays for MPE720 version 6 are used for descriptions in this manual.
If you are using MP720 version 5, interpret the displays according to MPE720 version 5.

■ Related Manuals

The following table lists the manuals relating to the MP2200. Refer to these manuals as required.

Manual Name	Manual Number	Contents
Machine Controller MP2000 Series SVA-01 Motion Module User's Manual	SIEP C880700 32	Describes the functions, specifications, and application methods of the MP2000-series SVA-01 Motion Module.
Machine Controller MP2000 Series Built-in SVB/SVB-01 Motion Module User's Manual	SIEP C880700 33	Describes the functions, specifications, and application methods of the MP2000-series Motion Module that is built into the SVA, SVB-01, and SVR Module.
Machine Controller MP2000 Series SVC-01 Motion Module User's Manual	SIEP C880700 41	Describes the functions, specifications, and application methods of the MP2000-series SVC-01 Motion Module.
Machine Controller MP2000 Series MPU-01 Multiple-CPU Module User's Manual	SIEP C880781 05	Describes the functions, specifications, and application methods of the MPU-01 Multiple-CPU Module that can be connected to MP2000 Series Machine Controllers.
Machine Controller MP2000 Series Pulse Output Motion Module PO-01 User's Manual	SIEP C880700 28	Describes the functions, specifications, and application methods of the MP2000-series PO-01 Motion Module.
Machine Controller MP2☐00 Communication Module User's Manual	SIEP C880700 04	Describes the functions, specifications, and application methods of the MP2☐00 Communication Modules (217IF, 218IF, 260IF, 261IF).
Machine Controller MP2300S/MP2310/MP2400 Basic Module Supplement for Ethernet Communications	SIEP C880700 37	Describes how to communicate with devices (PLCs, Windows computers, etc.) connected to the MP2300S/MP2310/MP2400 by Ethernet.
Machine Controller MP2000 Series 262IF-01 FL-net Communication Module User's Manual	SIEP C880700 36	Describes the specifications and communication methods of an FL-net Communication Module that can connect to an MP2000-series Machine Controller.
Machine Controller MP2000 Series 263IF-01 EtherNet/IP Communication Module User's Manual	SIEP C800700 39	Describes the specifications and communication methods of an EtherNet/IP Communication Module that can connect to an MP2000-series Machine Controller.
Machine Controller MP2000 Series EtherCAT Module 264IF-01 User's Manual	SIEP C880700 42	Describes the functions, specifications, and application methods of the 264IF-01, EtherCAT Module for the MP2000 Series.
Machine Controller MP2000 Series 265IF-01 CompoNet Module User's Manual	SIEP C880700 44	Describes the functions, specifications, and application methods of the 265IF-01, CompoNet Module for the MP2000 Series.
Machine Controller MP2000 Series 266IF-01 PROFINET Controller Module User's Manual	SIEP C880700 47	Describes the functions, specifications, and application methods of the 266IF-01, PROFINET Controller Module for the MP2000-series Machine Controllers.
Machine Controller MP2000 Series 266IF-02 PROFINET Device Module User's Manual	SIEP C880700 48	Describes the functions, specifications, and application methods of the 266IF-02, PROFINET Device Module for the MP2000-series Machine Controllers.
Machine Controller MP2000 Series 267IF-01 CC-Link Master Module User's Manual	SIEP C880712 01	Describes the functions, specifications, and application methods of the 267IF-01, CC-Link Master Module for the MP2000-series Machine Controllers.
Machine Controller MP2000 Series I/O Module User's Manual	SIEP C800700 34	Describes functions, specifications, and application methods of the MP2000-series I/O Modules (LIO-01, LIO-02, LIO-04, LIO-05, LIO-06, and DO-01).
Machine Controller MP2000 Series Analog Input/Analog Output Module AI-01/AO-01 User's Manual	SIEP C800700 26	Describes the functions, specifications, and communication methods of the MP2000-series I/O Modules (Al-01 and AO-01).
Machine Controller MP2000 Series Counter Module CNTR-01 User's Manual	SIEP C800700 27	Describes the functions, specifications, and application methods of the MP2000-series CNTR-01 Counter Module.

(cont'd)

Manual Name	Manual Number	Contents
Machine Controller MP900/MP2000 Series User's Manual, Ladder Programming	SIEZ-C887-1.2	Describes the instructions used in MP900/MP2000 ladder programming.
Machine Controller MP2000 Series User's Manual, Motion Programming	SIEP C880700 38	Describes the motion language used with an MP2000-series Machine Controller.
Engineering Tool for MP2000 Series Machine Controller MPE720 Version 6 User's Manual	SIEP C880700 30	Describes the installation and operation of the engineering tool for MP2000-series Machine Controller MPE720 Version 6.
Machine Controller MP2000/MP3000 Series Engineering Tool MPE720 Version 7 User's Manual	SIEP C880761 03	Describes the installation and operation of the engineering tool for MP2000/MP3000-series Machine Controller MPE720 Version 7.
Machine Controller MP900/MP2000 Series MPE720 Software for Programming Device User's Manual	SIEP C880700 05	Describes how to install and operate the MP900/MP2000-series programming system (MPE720).
Machine Controller MP900/MP2000 Series New Ladder Editor User's Manual Programming Instructions	SIEZ-C887-13.1	Describes the programming instructions of the New Ladder Editor, which assists MP900/MP2000-series design and maintenance.
Machine Controller MP900/MP2000 Series New Ladder Editor User's Manual Operation	SIEZ-C887-13.2	Describes the operating methods of the New Ladder Editor, which assists MP900/MP2000-series design and maintenance.
Machine Controller MP900/MP2000 Series User's Manual, MECHATROLINK System	SIEZ-C887-5.1	Describes MECHATROLINK distributed I/O for MP900/MP2000-series Machine Controllers.
Machine Controller MP900/MP2000 Series Linear Servomotor Manual	SIEP C880700 06	Describes the connection methods, setting methods, and other information for Linear Servomotors.

■ Terms Used to Describe "Torque"

Although the term "torque" is commonly used when describing rotary servomotors and "force" or "thrust" are used when describing linear servomotors, this manual uses "torque" when describing both (excluding parameters).

■ Copyrights

Safety Information

The following conventions are used to indicate precautions in this manual. These precautions are provided to ensure the safe operation of the MP2200 and connected devices. Information marked as shown below is important for the safety of the user. Always read this information and heed the precautions that are provided.
The conventions are as follows:

WARNING	Indicates precautions that, if not heeded, could possibly result in loss of life, serious injury, or property damage.
CAUTION	Indicates precautions that, if not heeded, could result in relatively serious or minor injury, or property damage. If not heeded, even precautions classified under CAUTION can lead to serious results depending on circumstances.
PROHIBITED	Indicates prohibited actions. Specific prohibitions are indicated inside . For example, indicates prohibition of open flame.
MANDATORY	Indicates mandatory actions. Specific actions are indicated inside . For example, indicates mandatory grounding.

Safety Precautions

The following precautions are for checking products on delivery, storage, transportation, installation, wiring, operation, application, inspection, and disposal. These precautions are important and must be observed.

■ General Precautions

⚠ WARNING
• Before connecting the machine and starting operation, ensure that an emergency stop procedure has been provided and is working correctly. There is a risk of injury. • Do not touch anything inside the MP2200. There is a risk of electrical shock. • Always keep the front cover attached when power is being supplied. There is a risk of electrical shock. • Observe all procedures and precautions given in this manual for trial operation. Operating mistakes while the servomotor and machine are connected may damage the machine or even cause accidents resulting in injury or death. There is a risk of electrical shock. • Do not remove the front cover, cables, connector, or options while power is being supplied. There is a risk of electrical shock. • Do not damage, pull on, apply excessive force to, place heavy objects on, or pinch cables. There is a risk of electrical shock, operational failure or burning of the MP2200. • Do not attempt to modify the MP2200 in any way. There is a risk of injury or device damage. • Do not approach the machine when there is a momentary interruption to the power supply. When power is restored, the MP2200 and the device connected to it may start operation suddenly. Provide safety measures in advance to ensure human safety in the event that operation restarts suddenly. There is a risk of injury. • Do not allow installation, disassembly, or repairs to be performed by anyone other than specified personnel. There is a risk of electrical shock or injury.

■ Storage and Transportation

⚠ CAUTION

- Do not store or install the MP2200 in the following locations.
  There is a risk of fire, electrical shock, or device damage.
  - Direct sunlight
  - Ambient temperature exceeds the storage or operating conditions
  - Ambient humidity exceeds the storage or operating conditions
  - Rapid changes in temperature or locations subject to condensation
  - Corrosive or flammable gas
  - Excessive dust, dirt, salt, or metallic powder
  - Water, oil, or chemicals
  - Vibration or shock
- Do not overload the MP2200 during transportation.
  There is a risk of injury or an accident.
- Do not under any means subject the MP2200 to an atmosphere that contains halogen gas (fluorine, chloride, bromine, iodine, etc.) during storage, transportation, or installation.
  There is a risk of damage or malfunction.
- If disinfectants or insecticides must be used to treat packing materials such as wooden frames, pallets, or plywood, the packing materials must be treated before the product is packaged, and methods other than fumigation must be used.
  Example: Heat treatment, where materials are kiln-dried to a core temperature of 56°C for 30 minutes or more.
  If the electronic products, which include stand-alone products and products installed in machines, are packed with fumigated wooden materials, the electrical components may be greatly damaged by the gases or fumes resulting from the fumigation process. In particular, disinfectants containing halogen, which includes chlorine, fluorine, bromine, or iodine can contribute to the erosion of the capacitors.

■ Installation

⚠ CAUTION

- Never use the MP2200 in locations subject to water, corrosive atmospheres, or flammable gas, or near burnable objects.
  There is a risk of electrical shock or fire.
- Do not step on the MP2200 or place heavy objects on the MP2200.
  There is a risk of injury.
- Do not block the air exhaust port or allow foreign objects to enter the MP2200.
  There is a risk of element deterioration inside, an accident, or fire.
- Always mount the MP2200 in the specified orientation.
  There is a risk of an accident.
- Do not subject the MP2200 to strong shock.
  There is a risk of an accident.

■ Wiring

⚠ CAUTION

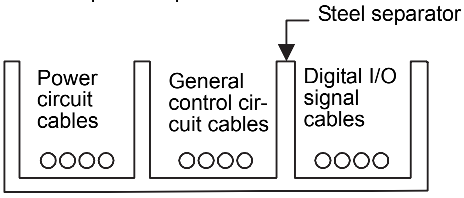
- Check the wiring to be sure it has been performed correctly.
  There is a risk of motor run-away, injury, or an accident.
- Always use a power supply of the specified voltage.
  There is a risk of burning.
- In places with poor power supply conditions, take all steps necessary to ensure that the input power supply is within the specified voltage range.
  There is a risk of device damage.
- Install breakers and other safety measure to provide protection against shorts in external wiring.
  There is a risk of fire.
- Provide sufficient shielding when using the MP2200 in the following locations.
  There is a risk of device damage.
  - Noise, such as from static electricity
  - Strong electromagnetic or magnetic fields
  - Radiation
  - Near to power lines
- When connecting the battery, connect the polarity correctly.
  There is a risk of battery damage or explosion.
- Do not allow the battery to be replaced by anyone other than qualified personnel who have received safety training.
  Improper battery replacement can result in electric shock. It can also cause equipment malfunction, injury to operators, or equipment damage.
- Do not touch the battery terminals when replacing the battery.
  There is a risk of electrostatic discharge failure.

■ Selecting, Separating, and Laying External Cables

⚠ CAUTION

- Consider the following items when selecting the I/O signal lines (external cables) to connect the MP2200 to external devices.
  - Mechanical strength
  - Noise interference
  - Wiring distance
  - Signal voltage, etc.
- Separate the I/O signal lines from the power lines both inside and outside the control box to reduce the influence of noise from the power lines.
  If the I/O signal lines and power lines are not separated properly, malfunctioning may result.

Example of Separated External Cables



Maintenance and Inspection Precautions

⚠ CAUTION
• Do not attempt to disassemble the MP2200. There is a risk of electrical shock or injury. • Do not change wiring while power is being supplied. There is a risk of electrical shock or injury. • When replacing the MP2200, restart operation only after transferring the programs and parameters from the old Module to the new Module. There is a risk of device damage.

Disposal Precautions

⚠ CAUTION
• Dispose of the MP2200 as general industrial waste. • A lithium battery is built into the Controller. After replacing the battery, dispose of the old battery separate from regular waste and in accordance with local regulations.

General Precautions

Observe the following general precautions to ensure safe application.
• The products shown in illustrations in this manual are sometimes shown without covers or protective guards. Always replace the cover or protective guard as specified first, and then operate the products in accordance with the manual. • The drawings presented in this manual are typical examples and may not match the product you received. • If the manual must be ordered due to loss or damage, inform your nearest Yaskawa representative or one of the offices listed on the back of this manual.

Warranty

(1) Details of Warranty

■ Warranty Period

The warranty period for a product that was purchased (hereinafter called “delivered product”) is one year from the time of delivery to the location specified by the customer or 18 months from the time of shipment from the Yaskawa factory, whichever is sooner.

■ Warranty Scope

Yaskawa shall replace or repair a defective product free of charge if a defect attributable to Yaskawa occurs during the warranty period above. This warranty does not cover defects caused by the delivered product reaching the end of its service life and replacement of parts that require replacement or that have a limited service life.
This warranty does not cover failures that result from any of the following causes.

1. Improper handling, abuse, or use in unsuitable conditions or in environments not described in product catalogs or manuals, or in any separately agreed-upon specifications
2. Causes not attributable to the delivered product itself
3. Modifications or repairs not performed by Yaskawa
4. Abuse of the delivered product in a manner in which it was not originally intended
5. Causes that were not foreseeable with the scientific and technological understanding at the time of shipment from Yaskawa
6. Events for which Yaskawa is not responsible, such as natural or human-made disasters

(2) Limitations of Liability

1. Yaskawa shall in no event be responsible for any damage or loss of opportunity to the customer that arises due to failure of the delivered product.
2. Yaskawa shall not be responsible for any programs (including parameter settings) or the results of program execution of the programs provided by the user or by a third party for use with programmable Yaskawa products.
3. The information described in product catalogs or manuals is provided for the purpose of the customer purchasing the appropriate product for the intended application. The use thereof does not guarantee that there are no infringements of intellectual property rights or other proprietary rights of Yaskawa or third parties, nor does it construe a license.
4. Yaskawa shall not be responsible for any damage arising from infringements of intellectual property rights or other proprietary rights of third parties as a result of using the information described in catalogs or manuals.

(3) Suitability for Use

1. It is the customer's responsibility to confirm conformity with any standards, codes, or regulations that apply if the Yaskawa product is used in combination with any other products.
2. The customer must confirm that the Yaskawa product is suitable for the systems, machines, and equipment used by the customer.
3. Consult with Yaskawa to determine whether use in the following applications is acceptable. If use in the application is acceptable, use the product with extra allowance in ratings and specifications, and provide safety measures to minimize hazards in the event of failure.
   - Outdoor use, use involving potential chemical contamination or electrical interference, or use in conditions or environments not described in product catalogs or manuals
   - Nuclear energy control systems, combustion systems, railroad systems, aviation systems, vehicle systems, medical equipment, amusement machines, and installations subject to separate industry or government regulations
   - Systems, machines, and equipment that may present a risk to life or property
   - Systems that require a high degree of reliability, such as systems that supply gas, water, or electricity, or systems that operate continuously 24 hours a day
   - Other systems that require a similar high degree of safety
4. Never use the product for an application involving serious risk to life or property without first ensuring that the system is designed to secure the required level of safety with risk warnings and redundancy, and that the Yaskawa product is properly rated and installed.
5. The circuit examples and other application examples described in product catalogs and manuals are for reference. Check the functionality and safety of the actual devices and equipment to be used before using the product.
6. Read and understand all use prohibitions and precautions, and operate the Yaskawa product correctly to prevent accidental harm to third parties.

(4) Specifications Change

The names, specifications, appearance, and accessories of products in product catalogs and manuals may be changed at any time based on improvements and other reasons. The next editions of the revised catalogs or manuals will be published with updated code numbers. Consult with your Yaskawa representative to confirm the actual specifications before purchasing a product.

Contents

1

Overview

This chapter explains an overview and features of the MP2200 Machine Controller.

1.1 MP2200 Features

The MP2200 is a high-performance, multi-axis Machine Controller for flexible system construction. In addition to I/O and Communication Modules, it has a wide range of Optional Modules, including various Motion Modules that support a variety of motor drives. It provides ideal motion control for a range of machines, from stand-alone machines to FA systems.

■ Flexibility

- The following CPU Modules are available: CPU-01, CPU-02, CPU-03, and CPU-04. The best CPU Module can be selected to match your application and machine specifications.
- With an option slot configuration that enables expansion to 35 slots, Optional Modules can be selected freely and the optimum system can be built for your machine.

■ High Performance

- Control characteristics have been improved by increasing the CPU and Motion Network (MECHATROLINK-II) speed.
  - Completely synchronous operation can be achieved for up to 256 axes.
  - MECHATROLINK-II baud rate: 2.5 times faster than MP920 (previous model)
  - CPU processing speed: 2.0 to 5.0 times faster than MP920 (previous model)
  - Larger user memory area
- High-speed (0.5 ms) motion control is now possible.
- MECHATROLINK-II enables position control, speed control, torque control, and phase control, and makes precise synchronous control possible. The control mode can also be changed online, facilitating complicated machine operations. The range of possible motion control applications is increased even further with the Virtual Motion Module (SVR).
- The following open networks are supported when optional Communication Modules are used.
  - Ethernet
  - DeviceNet
  - PROFIBUS
  - FL-net
  - EtherNet/IP

■ Easy to Use

- Machine startup times can be greatly reduced by using the self-configuration function that automatically detects devices connected to MECHATROLINK and sets the required parameters.
- The application program converter can utilize your previous software assets with their accumulated databanks of specific knowledge to improve the system further.

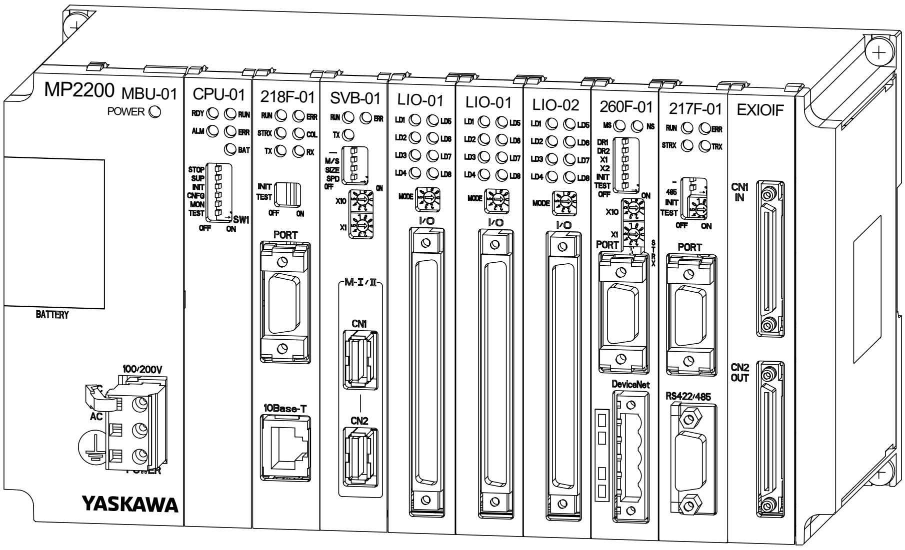
■ Compact

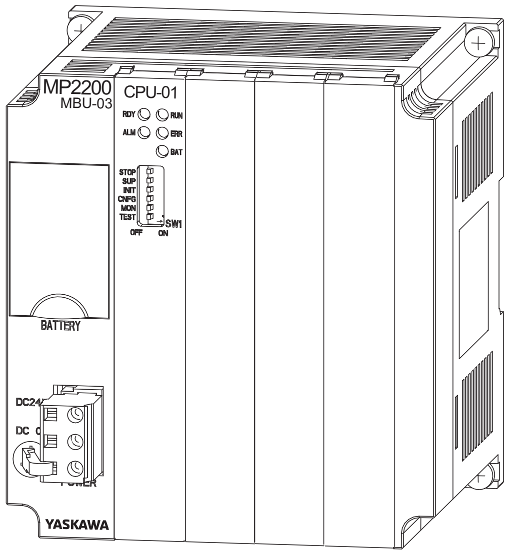
- The mounting area has been reduced to half that of the MP920 (previous model).

1.2 MP2200 Configuration

1.2.1 Basic Unit Appearance

The appearance of the Basic Units is shown below.





1.2.2 MP2200 Modules

The following table shows the names and specifications of the Basic Unit and Optional Modules.

Group		Name	Description	Model	Specifications
Basic Unit	Base Unit	Base Unit for AC power supply	MBU-01	JEPMC-BU2200-E JEPMC-BU2200	85 to 276 VAC 9 slots
		Base Unit for DC power supply	MBU-02	JEPMC-BU2210-E JEPMC-BU2210	24 VDC ±20% 9 slots
		Base Unit for DC power supply	MBU-03	JEPMC-BU2220-E	24 VDC ±20% 4 slots
	CPU Modules	CPU Modules	CPU-01	JAPMC-CP2200-E JAPMC-CP2200	CPU Module for MP2200 system
			CPU-02	JAPMC-CP2210-E JAPMC-CP2210	CPU Module for MP2200 system with 1 Compact Flash slot and 1 USB port
			CPU-03	JAPMC-CP2220-E	CPU Module for MP2200 system with 1 Compact Flash slot and 1 Ethernet port
			CPU-04	JAPMC-CP2230-E	Host CPU Module for MP2200 system with 1 Ethernet port
		Multiple-CPU Module	MPU-01	JAPMC-CP2700-E	Optional Module with CPU + SVC-01 functions

(cont'd)

Group		Name	Description	Model	Specifications
Optional Modules	Motion Modules	MECHATROLINK Motion Module	SVB-01	JAPMC-MC2310-E JAPMC-MC2310	MECHATROLINK-I and -II Interface, 16 axes maximum
			SVC-01	JAPMC-MC2320-E	MECHATROLINK-III Interface, 16 axes maximum
		Analog Output Motion Module	SVA-01	JAPMC-MC2300-E JAPMC-MC2300	Analog output, 2 axes maximum
		Pulse Output Motion Module	PO-01	JAPMC-PL2310-E	Pulse output, 4 axes maximum
	I/O Modules	I/O Module	LIO-01	JAPMC-IO2300-E JAPMC-IO2300	16 inputs, 16 outputs (sink mode output) 1 pulse input
		I/O Module	LIO-02	JAPMC-IO2301-E JAPMC-IO2301	16 inputs, 16 outputs (source mode output) 1 pulse input
		I/O Module	LIO-04	JAPMC-IO2303-E JAPMC-IO2303	32 inputs, 32 outputs (sink mode output)
		I/O Module	LIO-05	JAPMC-IO2304-E JAPMC-IO2304	32 inputs, 32 outputs (source mode output)
		I/O Module	LIO-06	JAPMC-IO2305-E	8 inputs, 8 outputs (sink mode outputs), 1 analog input channel, 1 analog output channel, 1 pulse counter channel
		Output Module	DO-01	JAPMC-DO2300-E JAPMC-DO2300	64 outputs (sink mode output)
		Analog Input Module	AI-01	JAPMC-AN2300-E JAPMC-AN2300	Analog input, 8 channels
		Analog Output Module	AO-01	JAPMC-AN2310-E	Analog input, 4 channels
		Counter Module	CNTR-01	JAPMC-PL2300-E	Reversible counter, 2 channels
	Communication Modules	Ethernet Communication Module	218IF-01	JAPMC-CM2300-E JAPMC-CM2300	RS-232C and Ethernet communication
		Ethernet Communication Module	218IF-02	JAPMC-CM2302-E	RS-232C and Ethernet communication (100 Mbps)
		General-purpose Serial Communication Module	217IF-01	JAPMC-CM2310-E JAPMC-CM2310	RS-232C and RS422/485 communication
		DeviceNet Communication Module	260IF-01	JAPMC-CM2320-E JAPMC-CM2320	RS-232C and DeviceNet communication
		PROFIBUS Communication Module	261IF-01	JAPMC-CM2330-E JAPMC-CM2330	RS-232C and PROFIBUS communication
		FL-net Communication Module	262IF-01	JAPMC-CM2303-E	FL-net communication
		EtherNet/IP Communication Module	263IF-01	JAPMC-CM2304-E	EtherNet/IP communication
		EtherCAT Communication Module	264IF-01	JAPMC-CM2305-E	EtherCAT communication (slave)
		CompoNet Communication Module	265IF-01	JAPMC-CM2390-E	CompoNet communication
		PROFINET Communication Module	266IF-01	JAPMC-CM2306-E	PROFINET communication (master)
			266IF-02	JAPMC-CM2307-E	PROFINET communication (slave)
		CC-Link Communication Module	267IF-01	JAPMC-CM23A0-E	CC-Link communication (master)
		CC-Link IE Field Communication Module	269IF-01	JAPMC-CM2308-E	CC-Link IE Field communication (slave)
		MPLINK/CP-215 Communication Module	215AIF-01	JAPMC-CM2360-E JAPMC-CM2360	RS-232C, MPLINK, and CP-215 communications
	Expansion Interface Module	Connection Interface	EXIOIF*	JAPMC-EX2200-E JAPMC-EX2200	System bus expansion (configuration: 4 racks max.)

* Use an EXIOIF cable that has a maximum length of 6.0 m.

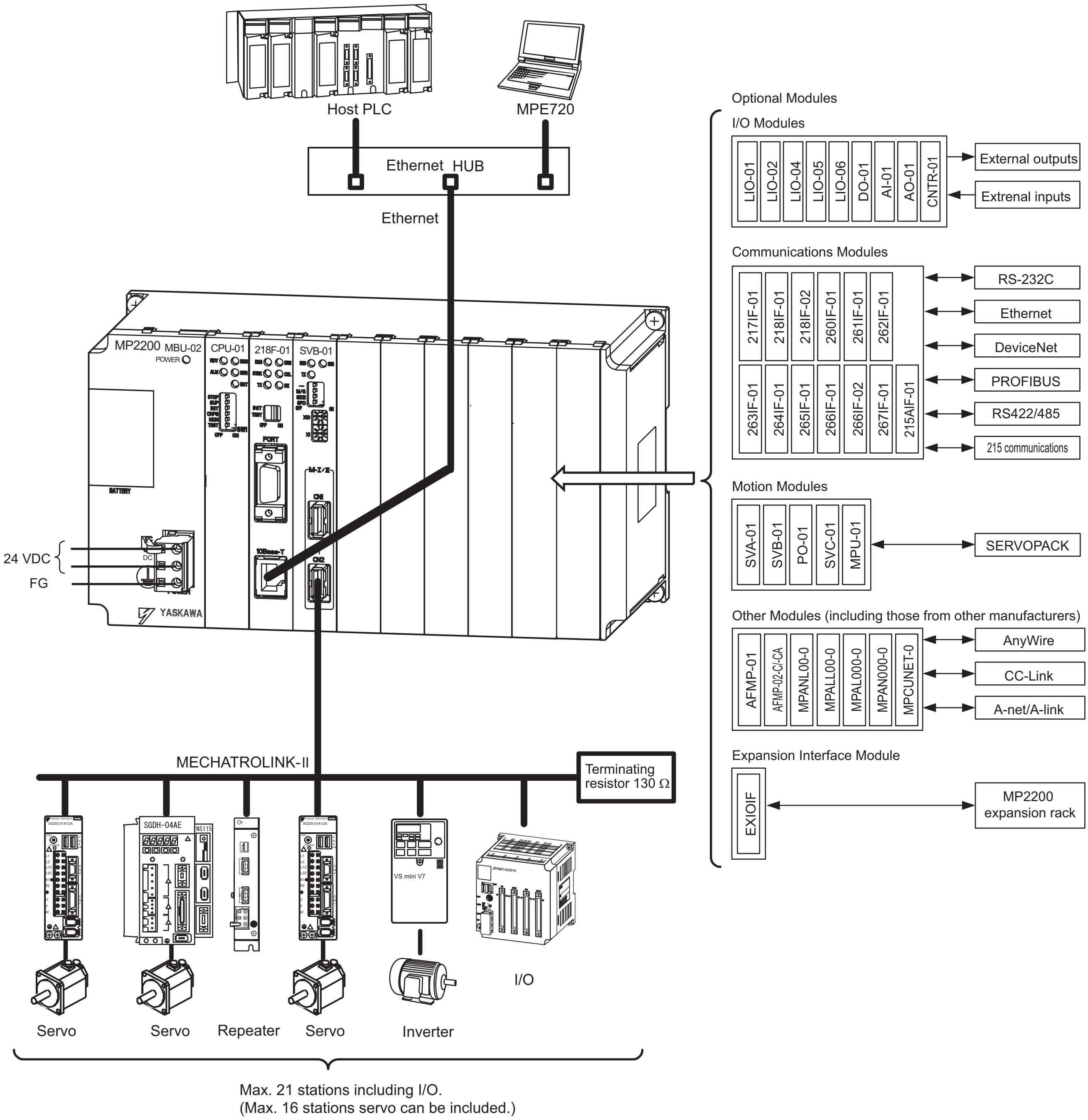
Note: Models that have model numbers ending in "-E" are RoHS compliant.

1.3 System Configuration

1.3.1 Example

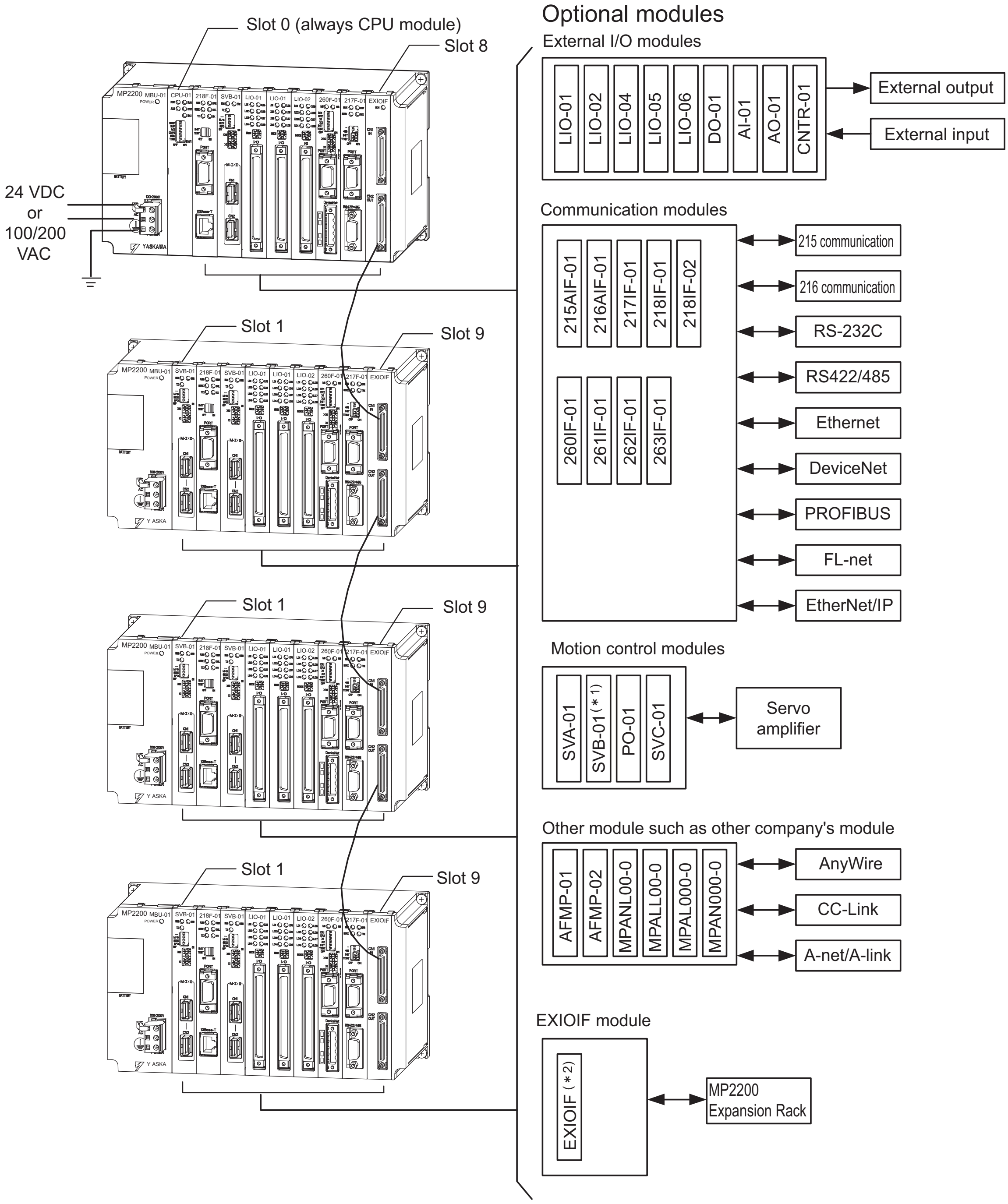
(1) Basic Unit Rack (Rack 1) Configuration

The following diagram shows an example of system configuration.



Note: 1. For the details on the system configuration example, refer to *4.2.1 (1) System Layout Model*.
2. Use the connecting cables and connectors recommended by Yaskawa. Always check the device to be used and select the correct cable for the device.
3. Different SERVOPACKs are connected to MECHATROLINK-I (4 Mbps) and MECHATROLINK-II (10 Mbps). Refer to *1.4.1 Devices Connectable to MECHATROLINK-I/II* and select the appropriate SERVOPACKs.
4. If devices compatible with MECHATROLINK-I and with MECHATROLINK-II are used together, make the settings for MECHATROLINK-I.
5. The user must supply the 24-VDC power supply.
6. When connecting SERVOPACKs via MECHATROLINK, connect the overtravel, zero point return deceleration limit switch, and external latch signals to the SERVOPACKs. For connection, refer to the SERVOPACK's manual.

(2) Maximum Four-Rack Configuration



* 1. A distributed I/O function is provided by the SVB-01 Modules through MECHATROLINK communication.
* 2. Use an EXIOIF cable that has a maximum length of 6.0 m.

1.3.2 System Configuration Precautions

The following precautions must be followed when designing a system using the MP2200.

- Use the connecting cables and connectors recommended by Yaskawa. Yaskawa has a range of cables. Always check the device to be used and select the correct cable for the device.
- Different SERVOPACKs are connected to MECHATROLINK-I and MECHATROLINK-II. Refer to the list and select the appropriate SERVOPACKs.
- The user must supply the 24-VDC power supply.
- The battery backs up M registers, system registers, and trace memory. Always save the program to flash memory whenever it is input or changed.

1.4 Devices Connectable to MECHATROLINK

1.4.1 Devices Connectable to MECHATROLINK-I/II

The devices that are compatible with MECHATROLINK-I/II and can be connected to the MP2200 and the SVB-01 Module are listed below.

(1) Compatible SERVOPACKs

Model Number	Details	MECHATROLINK-I	MECHATROLINK-II
SGD-□□□N SGDB-□□AN	MECHATROLINK-I-compatible AC SERVOPACK	Yes	No
SGDH-□□□E JUSP-NS100	Σ-II Series SGDH SERVOPACK NS100 MECHATROLINK-I Application Module	Yes	No
SGDH-□□□E JUSP-NS115	Σ-II Series SGDH SERVOPACK NS115 MECHATROLINK-II Application Module	Yes	Yes
SGDS-□□□1□□	Σ-III Series SGDS SERVOPACK	Yes	Yes
SGDX-□□□12□	SGDX SERVOPACK	Yes	Yes
SJDE-□□AN□	SJDE SERVOPACK	No	Yes
SGDV-□□□□1□□	SGDV SERVOPACK	Yes	Yes
SGD7S-□□□□10□	SGD7S SERVOPACK	Yes	Yes

(2) Compatible Inverters

Model Number	Details	MECHATROLINK-I	MECHATROLINK-II
CIMR-G7A□ SI-T	Varispeed G7 Inverter with MECHATROLINK interface	Yes	Yes
CIMR-F7A□ SI-T	Varispeed F7 Inverter with MECHATROLINK interface	Yes	Yes
CIMR-V7AA□ SI-T/V7	VSmini V7 Inverter with MECHATROLINK interface	Yes	Yes
CIMR-A□ SI-T3	High Performance Vector Control Drive A1000 MECHATROLINK-II Option Card	Yes	Yes
CIMR-V□ SI-T3/V	Compact Vector Control Drive V1000 MECHATROLINK-II Option Unit	Yes	Yes

(3) Compatible Modules

Model	Details	MECHATROLINK-I	MECHATROLINK-II
JEPMC-IO350	64-point I/O Module 24 VDC, 64 inputs, 64 outputs (sink mode outputs)	Yes	No
JAMSC-120DDI34330	DC Input Module 12/24 VDC, 16 inputs	Yes	No
JAMSC-120DDO34340	DC Output Module 12/24 VDC, 16 outputs	Yes	No
JAMSC-120DAI53330	AC Input Module 100 VAC, 8 inputs	Yes	No
JAMSC-120DAI73330	AC Input Module 200 VAC, 8 inputs	Yes	No
JAMSC-120DAO83330	AC Output Module 100/200 VAC, 8 outputs	Yes	No
JAMSC-120DRA83030	Relay Module Wide voltage range relay contacts, 8 contact outputs	Yes	No
JAMSC-120AVI02030	A/D Module Analog inputs, −10 to 10 V, 4 channels	Yes	No
JAMSC-120AVO01030	D/A Module Analog outputs, −10 to 10 V, 2 channels	Yes	No
JAMSC-120EHC21140	Counter Module Reversible counter, 2 channels	Yes	No
JAMSC-120MMB20230	Pulse Output Module Pulse output, 2 channels	Yes	No
JAMSC-IO2900-E	DC Input Module 12/24 VDC, 16 inputs	Yes	Yes
JAMSC-IO2910-E	DC Output Module 12/24 VDC, 16 outputs	Yes	Yes
JAMSC-IO2920-E	8-point I/O Module 24 VDC, 8 inputs, 8 outputs	Yes	Yes
JAPMC-IO2950-E	Relay Module Wide voltage range relay contacts, 8 contact outputs	Yes	Yes
AB023-M1	MECHATROLINK Bit decentralization I/O terminal (by Anywire Corporation)	Yes	Yes
JEPMC-IO2310 JEPMC-IO2310-E	64-point I/O Module 24 VDC, 64 inputs, 64 outputs (sink mode outputs)	Yes	Yes
JEPMC-IO2330 JEPMC-IO2330-E	64-point I/O Module 24 VDC, 64 inputs, 64 outputs (source mode outputs)	Yes	Yes
JEPMC-PL2900 JEPMC-PL2900-E	Counter Module Reversible counters, 2 channels	Yes	Yes
JEPMC-PL2910 JEPMC-PL2910-E	Pulse Output Module Pulse output, 2 channels	Yes	Yes
JEPMC-AN2900 JEPMC-AN2900-E	A/D Module Analog inputs, −10 to 10 V, 4 channels	Yes	Yes
JEPMC-AN2910 JEPMC-AN2910-E	D/A Module Analog outputs, −10 to 10 V, 2 channels	Yes	Yes
JAPMC-MC2310 JAPMC-MC2310-E	SVB-01 Motion Module	Yes	Yes
JEPMC-REP2000 JEPMC-REP2000-E	MECHATROLINK-II Repeater	No	Yes
JEVSA-YV250 JEVSA-YV250-E	MYVIS (image processing device)	Yes	Yes
JEVSA-YV260□□-E	MYVIS YV260 Machine Vision System	Yes	Yes
JEPMC-MC400	MP940 Motion Controller	Yes	No

1.4.2 Devices Connectable to MECHATROLINK-III

The devices that are compatible with MECHATROLINK-III and can be connected to the SVC-01 Module are listed below.

(1) Compatible SERVOPACKs

Model Number	Details
SGDV-□□□□2□□	SGDV SERVOPACKs with MECHATROLINK-III Communications References
SGD7S-□□□□20□	SGD7S SERVOPACKs with MECHATROLINK-III Communications References
SGD7W-□□□□20□	SGD7W SERVOPACKs with MECHATROLINK-III Communications References

(2) Compatible Inverters

Model Number	Details
CIMR-A□ SI-ET3	High Performance Vector Control Drive A1000 MECHATROLINK-III Option Card
CIMR-V□ SI-ET3/V	Compact Vector Control Drive V1000 MECHATROLINK-III Option Unit

(3) Compatible Modules

Model Number	Details
JAPMC-MC2320-E	SVC-01 Motion Module
JEPMC-MTD2310-E	64-point I/O Module 24 VDC, 64 inputs, 64 outputs
JEPMC-MTA2900-E	Analog Input Module Analog input, 8 channels
JEPMC-MTA2910-E	Analog Output Module Analog output, 4 channels
JEPMC-MTP2900-E	Pulse Input Module Pulse input, 2 channels
JEPMC-MTP2910-E	Pulse Output Module Pulse output, 4 channels

1.5 Cables, Accessories and Options, and Software

1.5.1 Cables

The following table shows the cables that can be connected to the MP2200 Basic Module and Optional Modules.

Module	Connector Name	Cable Name	Model	Specifications
CPU-02	USB	USB cable	Use a commercially available cable.	Used between CPU-02 and external device with mini-B connector
CPU-03 CPU-04	Ethernet	Ethernet communication cable		Cross cable used between CPU-03/CPU-04 and external device (category 5)
SVB-01	M-I/II	MECHATROLINK-I Cables	JEPMC-W6010-□□ with a MECHATROLINK connector and loose wires	Used between the devices listed below SVB-01 and SGD-□□□N SVB-01 and SGDB-□□AN
		MECHATROLINK-I Terminator (terminating resistance)	JEPMC-W6020	–
		MECHATROLINK-II Cables	JEPMC-W6002-□□ with MECHATROLINK connectors on both ends	Used between the devices listed below SVB-01 and MECHATROLINK-II compatible device
			JEPMC-W6003-□□ with MECHATROLINK connectors on both ends with ferrite core	
		MECHATROLINK-II Terminator (terminating resistance)	JEPMC-W6022	–
SVC-01	M-III	MECHATROLINK-III Cables	JEPMC-W6012□□-E with MECHATROLINK-III connectors on both ends with ferrite core	Used between the following devices: SVC-01 and MECHATROLINK-III compatible device
			JEPMC-W6013□□-E with MECHATROLINK-III connectors on both ends with ferrite core	
			JEPMC-W6014□□-E with loose wires on one end with ferrite core	
SVA-01	CH1 CH2	Cable for Analog Reference Input SERVOPACK	JEPMC-W2040-□□	Used between the devices listed below SVA-01 and Analog Reference Input SERVOPACK
MPU-01	M-III	MECHATROLINK-III cable	JEPMC-W6012-□□-E • with MECHATROLINK-III connectors on both ends	Used between the devices listed below MPU-01 and MECHATROLINK-III compatible device
			JEPMC-W6013-□□-E • with MECHATROLINK-III connectors on both ends • with ferrite core	
			JEPMC-W6014-□□-E • with a MECHATROLINK-III connector and loose wires	–
LIO-01 LIO-02	I/O	External I/O Cable	JEPMC-W2061-□□ with loose wires on one end	Used between LIO-01/02 and external I/O device
LIO-04 LIO-05	CN1, CN2	External I/O Cable	JEPMC-W6060-□□ with loose wires on one end	Used between LIO-04/05 and external I/O device
LIO-06	CN1	External I/O Cable	JEPMC-W2064-□□ with loose wires on one end	Used between LIO-06 and external I/O device
DO-01	CN1, CN2	External Output Cable	JEPMC-W6060-□□ with loose wires on one end	Used between DO-01 and external input device
AI-01	CN1, CN2	Analog Input Cable	JEPMC-W6080-□□ with loose wires on one end	Used between AI-01 and analog external input device

(cont'd)

Module	Connector Name	Cable Name	Model	Specifications
AO-01	CN1	Analog Output Cable	JEPMC-W6090-□□ with loose wires on one end	Used between AO-01 and analog external output device
CNTR-01	CN1	Cable for CNTR-01 Module	JEPMC-W2063E-□□ with loose wires on one end	Used between CNTR-01 and external I/O device
PO-01	CN1, CN2	Cable for PO-01 Module	JEPMC-W6060-□□ with loose wires on one end	PO-01 and external I/O device
218IF-01	10Base-T	Ethernet communication cable	Use a commercially available cable.	Cross cable (category 3 min.)
	PORT	RS-232C communication cable	JEPMC-W5310-□□	RS-232C port and 25-pin male D-sub connector
			JEPMC-W5311-□□	RS-232C port and DOS/V
218IF-02	Ethernet	Ethernet communication cable	Use a commercially available cable.	Cross cable or straight cable (category 5)
	PORT	RS-232C communication cable	JEPMC-W5310-□□	RS-232C port and 25-pin male D-sub connector
			JEPMC-W5311-□□	RS-232C port and DOS/V
217IF-01	RS422/485	RS-422/RS-485 communication cable	Use a commercially available cable.	• Module-side connector: 1010214-52A2JL (manufactured by 3M Japan Limited) • Cable-end connector: 10114-3000PE (manufactured by 3M Japan Limited) • Shell: 10314-52A0-008 (manufactured by 3M Japan Limited)
	PORT	RS-232C communication cable	JEPMC-W5310-□□	RS-232C port and 25-pin male D-sub connector
			JEPMC-W5311-□□	RS-232C port and DOS/V
260IF-01	DeviceNet	DeviceNet communication cable	Use a commercially available cable.	Module-side connector: MSTB2.5/5-GF-5.08AU (manufactured by Phoenix Contact)
	PORT	RS-232C communication cable	JEPMC-W5310-□□	RS-232C port and 25-pin male D-sub connector
			JEPMC-W5311-□□	RS-232C port and DOS/V
261IF-01	PROFIBUS	PROFIBUS communication cable	Use a commercially available cable.	Module-side connector: 17LE-13090-27(D33C) (manufactured by DDK Ltd.)
	PORT	RS-232C communication cable	JEPMC-W5310-□□	RS-232C port and 25-pin male D-sub connector
			JEPMC-W5311-□□	RS-232C port and DOS/V

(cont'd)

Module	Connector Name	Cable Name	Model	Specifications
262IF-01	FL-net	Ethernet communication cable	Use a commercially available cable.	Cross cable or straight cable (category 5)
263IF-01	EtherNet/IP			
264IF-01	EtherCAT			
265IF-01	CompoNet	CompoNet communication cable		For details, refer to the "Construction Manual" issued by ODVA.
266IF-01	PROFINET	Ethernet communication cable		Cross cable or straight cable (category 5)
266IF-02				
267IF-01	CC-Link	CC-Link communication cable		CC-Link dedicated cables (three-core shielded twisted-pair cables)
269IF-01	CC-Link IE Field	Ethernet communication cable		Cross cable or straight cable (category 5)
215AIF-01	MPLINK	MPLINK communication cable (MECHATROLINK cable)	JEPMC-W6002-☐☐ • with MECHATROLINK connectors on both ends	Used between 215AI-01 MPLINK and MPLINK compatible device
			JEPMC-W6003-☐☐ • with MECHATROLINK connectors on both ends • with ferrite core	
			JEPMC-W6022	Terminator
	CP-215	CP-215 communication cable	Provided by the customers. Refer to the communication module manual (Manual No.: SIEPC88070004).	
EXIOIF	CH1 IN, CH2 OUT	EXIOIF Cable	JEPMC-W2091-☐☐	Between EXIOIF Modules

1.5.2 Accessories and Options

Name	Accessory/Optional	Model	Remarks
Battery	Accessory	JZSP-BA01	ER3VC + exclusive use connector (BA000517)
Power Supply Connector	Accessory	721-203/026-304	MBU-01 Unit Cable side (manufactured by WAGO Company of Japan, Ltd, black)
		721-203/026-000	MBU-02/MBU-03 Unit Cable side (manufactured by WAGO Company of Japan, Ltd, white)
DIN Rail Mounting Parts	Optional	JEPMC-OP300	1 pair
Cover for Option Slot	Optional	JEPMC-OP2300	Front cover for the unused slot.

1.5.3 Software (Programming Tool)

The MPE720, programming tool for MP2200, is available.

Name	Model	Supported versions				Remarks
		CPU-01	CPU-02	CPU-03	CPU-04	
MPE720	CPMC-MPE720	Version 5.10 or later	Version 5.31 or later	Version 5.50 or later	Version 5.52 or later	CD-ROM
MPE720 Version 6	CPMC-MPE770D	All versions	All versions	Version 6.20 or later	Version 6.22 or later	DVD-ROM
MPE720 Version 7	CPMC-MPE780D	All versions	All versions	All versions	All versions	DVD-ROM

2

Overview of Specifications and Functions

This chapter explains the specifications and functions for the Basic Unit and Optional Modules of the MP2200.

2.1 Specifications

2.1.1 General Specifications

Item		Specifications
Environmental Conditions	Ambient Operating Temperature	0°C to 55°C, but 0 to 50°C for the CPU-03
	Ambient Storage Temperature	–25°C to 85°C
	Ambient Operating Humidity	30% to 95% (with no condensation)
	Ambient Storage Humidity	5% to 95% (with no condensation)
	Pollution Level	Pollution level 2 (conforming to JIS B 3502)
	Corrosive Gas	There must be no combustible or corrosive gas.
	Operating Altitude	2,000 m above sea level or lower
Mechanical Operating Conditions	Vibration Resistance	Conforming to JIS B 3502: • 10 to 57 Hz with single-amplitude of 0.075 mm • 57 to 150 Hz with fixed acceleration of 9.8 m/s^2 • 10 sweeps each in X, Y, and Z directions (sweep time: 1 octave/min.)
	Shock Resistance	Conforming to JIS B 3502: Peak acceleration of 147 m/s^2 (15 G) twice for 11 ms each in the X, Y, and Z directions
Electrical Operating Conditions	Noise Resistance	Conforming to EN 61000-6-2, EN 61000-6-4, EN 55011 (Group 1, Class A)
Installation Requirements	Ground	Ground to 100 Ω max.
	Cooling Method	Natural cooling

2.1.2 Product Specifications

The following table shows the product specifications of the MP2200.

Items		Specifications			
		CPU-01	CPU-02	CPU-03	CPU-04
External Dimensions	Long rack	240 mm × 130 mm × 108 mm			
	Short rack	120 mm × 130 mm × 108 mm			
Maximum Number of Option Slots		35 slots			
Number of Control Axes	Number of Basic Control Axes	–			
	Maximum Number of Control Axes	256 axes (when sixteen SVB-01 Modules are added.)			
	Number of Virtual Axis Controlling Axes	16 axes			
CPU Processing Speed Comparison		1.0	1.0	1.6	2.5
Scan Interval Setting	High-speed Scan	0.5 ms to 32 ms (per 0.5 ms)			
	Low-speed Scan	2.0 ms to 300 ms (per 0.5 ms)			
Communication I/F		Not provided.	USB: USB 2.0 (functions), 12 Mbps, 1 port	Ethernet: 100Base-TX, 1 port Maximum connections: 4 + 2 (I/O message communication)	Ethernet: 100Base-TX, 1 port Maximum connections: 20 + 2 (I/O message communication)
External Memory I/F		Not provided.	One Compact Flash slot		Not provided.
Memory Capacity	SDRAM/DDR	32 MB	32 MB	64 MB	
	SRAM (battery backup)	512 KB	2 MB*1	4 MB*2	
	FLASH	12 MB	16 MB	16 MB	
	Program Capacity	7.5 MB	11.5 MB	11.5 MB	
Programming Language	Ladder Language	√	√	√	
	Motion Language	√	√	√	
	Sequence Program	–	–	√	
	C Language	√	√	√	

* 1. 1 MB max. with battery backup for table data.
* 2. 3 MB max. with battery backup for table data.

Note: Symbols in the table mean as follows.
√: Available, –: Not available

2.1.3 Function Lists

(1) PLC Function Specifications

The following table shows the PLC functions of the MP2200.

Item	Specifications
Control Method	Sequence: High-speed and low-speed scan methods
Programming Language	Ladder diagram: Relay circuit Text-type language: Numeric operations, logic operations, etc.
Scan	Two scan levels: High-speed scan and low-speed scan High-speed scan time setting: 0.5 to 32 ms Low-speed scan time setting: 2 to 300 ms
User Drawings, Functions and Motion Programs	Startup drawings (DWG.A): 64 drawings max. Up to three hierarchical drawing levels Interrupt processing drawings (DWG.I): 64 drawings max. Up to three hierarchical drawing levels High-speed scan process drawings (DWG.H): 200 drawings max. Up to three hierarchical drawing levels Low-speed scan process drawings (DWG.L): 500 drawings max. Up to three hierarchical drawing levels Number of steps: Up to 1,000 steps per drawing User functions: Up to 500 functions Motion programs and sequence programs*1: A total of up to 256
	Revision history of drawings and motion programs Security function for drawings and motion programs
Data Memory	Common data (M) registers: 64 kwords System (S) registers: 8 kwords Drawing local (D) registers: Up to 16 kwords per drawing Drawing constant (#) registers: Up to 16 kwords per drawing Input (I) registers: 32 kwords Output (O) registers: 32 kwords Constant (C) registers: 16 kwords
Trace Memory	Data trace: 128 kwords (32 kwords × 4 groups), 16 points defined
Memory Backup	Program memory: Flash memory: Data other than battery backup data such as definition files, ladder programs, motion programs. Data memory: Battery backup: M registers, S registers, alarm history, trace data, table data*2
Data Types	Bit (relay): ON/OFF Integer: –32768 to +32767 Double-length integer: –2147483648 to +2147483647 Real number: ± (1.175E-38 to 3.402E+38)
Register Designation Method	Register number: Direct designation of register number Symbolic designation: Up to 8 alphanumeric characters (up to 200 symbols per drawing) With automatic numbering or symbol assignment
Instructions	Program control instructions: 14 instructions Direct I/O instructions: 2 instructions Relay circuit instructions: 14 instructions (including set and reset coils) Logic operation instructions: 3 instructions Numeric operation instructions: 16 instructions Numeric conversion instructions: 9 instructions Numeric comparison instructions: 7 instructions Data manipulation instructions: 14 instructions Basic function instructions: 10 instructions Table data manipulation instructions: 11 instructions DDC instructions: 13 instructions System functions: 9 instructions
Optional Functions*3	Clearing D registers at startup Security Battery backup for table data*2

* 1. Sequence programming cannot be used with the CPU-01 or CPU-02.
* 2. Table data cannot be backed up with a battery with the CPU-01.
* 3. For details on optional functions, refer to *Appendix E Optional Functions*.

2.2 Base Unit

2.2.1 Overview of Functions

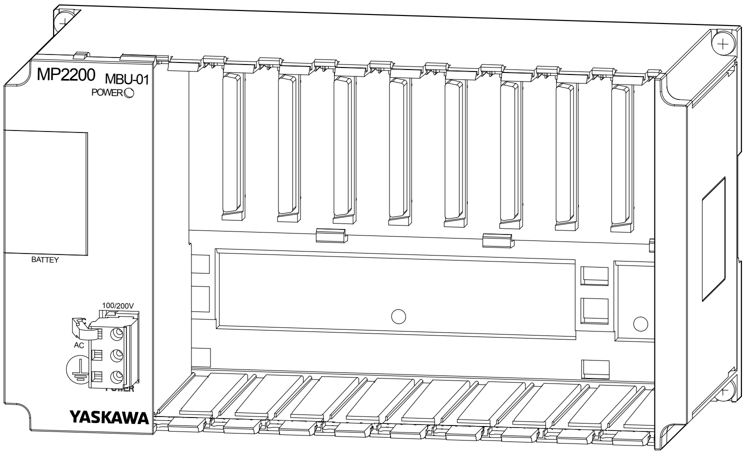
The Base Unit combines the power supply, mounting base board, and frame in one unit. Both AC-input and DC-input power supply Base Units are available. The Base Unit provides either four option slots (short rack) or nine option slots. Any Optional Modules can be mounted in the option slots to create the perfect system for the machinery.

2.2.2 LED Indicators

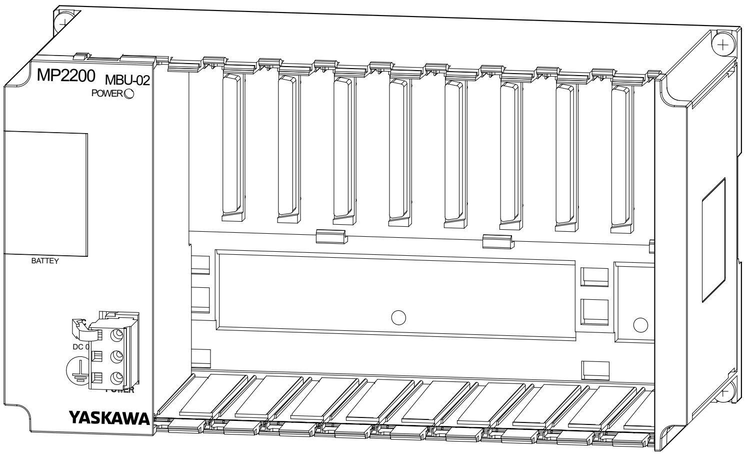
(1) External Appearance

The following figure shows the external appearance of the Base Unit.

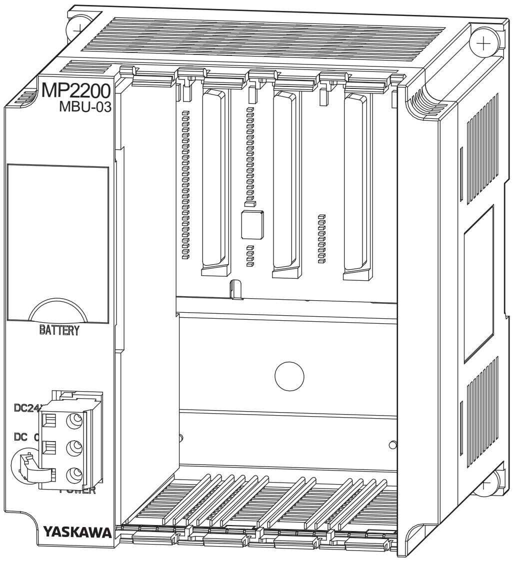
[a] Base Unit with AC-input Power Supply and Nine Slots



[b] Base Unit with DC-input Power Supply and Nine Slots



[c] Base Unit with DC-input Power Supply Short Rack



(2) Indicator

The LED indicator that displays the status of the power supply is detailed in the following table.

The short-rack Base Unit does not have a POWER indicator.

Indicator	Indicator Name	Color	Significance when Lit
POWER ◯	POWER	Green	The power supply is operating normally.

2.2.3 Hardware Specifications

The following table shows the hardware specifications of the Base Units.

<table>
<tr><th colspan="2">Item</th><th colspan="3">Specifications</th></tr>
<tr><td colspan="2">Name</td><td>Base Unit with AC-input power supply</td><td>Base Unit with DC-input power supply</td><td>Base Unit with DC-input power supply for short racks</td></tr>
<tr><td colspan="2">Model Number</td><td>JEPMC-BU2200</td><td>JEPMC-BU2210</td><td>JEPMC-BU2220-E</td></tr>
<tr><td colspan="2">Abbreviation</td><td>MBU-01</td><td>MBU-02</td><td>MBU-03</td></tr>
<tr><td colspan="2">Slot Configuration</td><td colspan="2">Basic Unit Rack (Rack1) Configuration
1 slot for CPU Module
8 slots for Optional Modules (including slots for Expansion Modules)
Optional Expansion Rack (Rack 2, 3, 4) Configuration
9 slots for Optional Modules (because a CPU Module is not mounted to Racks 2, 3, and 4)</td><td>Basic Unit Rack (Rack 1) Configuration
1 slot for CPU Module
3 slots for Optional Modules (including slots for Expansion Modules)
Optional Expansion Rack (Rack 2, 3, 4) Configuration
4 slots for Optional Modules</td></tr>
<tr><td rowspan="10">Power Unit</td><td>Input Voltage</td><td>85 to 132 VAC / 198 to 276 VAC</td><td colspan="2">24 VDC (± 20%)*</td></tr>
<tr><td>Allowable Frequeucy Range</td><td>47 to 63 Hz</td><td colspan="2">–</td></tr>
<tr><td>Input Current</td><td>1.5 A max. (at rated I/O)</td><td>3.0 A max. (at rated I/O)*</td><td>1.0 A max. (at rated I/O)*</td></tr>
<tr><td>Inrush Current</td><td>40 A max. (when completely discharged, with 275-VAC input, rated output)</td><td colspan="2">30 A max. (when completely discharged, rated output)*</td></tr>
<tr><td>Allowable Power Loss Time</td><td>20 ms</td><td colspan="2">1 ms</td></tr>
<tr><td>Rated Voltage</td><td colspan="3">5.0 V</td></tr>
<tr><td>Rated Current</td><td colspan="2">8.0 A</td><td>4.0 A</td></tr>
<tr><td>Output Current Range</td><td colspan="2">0.0 to 8.0 A</td><td>0.0 to 4.0 A</td></tr>
<tr><td>Constant Voltage Precision</td><td colspan="3">±2% max. (including input voltage and output load fluctuations)</td></tr>
<tr><td>Battery</td><td colspan="3">Battery for memory retention attachable</td></tr>
<tr><td colspan="2">Indicators</td><td colspan="3">POWER (green)</td></tr>
<tr><td colspan="2">Dimensions (mm)</td><td colspan="2">240 × 130 × 108 (W × H × D)</td><td>120 × 130 × 108 (W × H × D)</td></tr>
<tr><td colspan="2">Mass</td><td colspan="2">650 g</td><td>500 g</td></tr>
</table>

* For the external 24-V power supply, select a power supply that satisfies the following specifications and that can provide the rated current:

- Allowable output load capacity: 1200 µF or more
- Overcurrent detection is automatically restored by removing causes

However, except that the primary side (AC side) of the external 24V power supply is turned ON/OFF.

2.3 CPU-01 Module

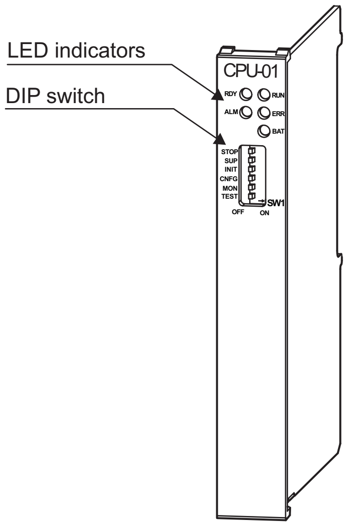
2.3.1 Overview of Functions

The CPU-01 Module is the MP2200 Control Module that controls the Motion, Communication, I/O, and other Optional Modules.

2.3.2 LED Indicators and Switch Settings

(1) External Appearance

The following figure shows the external appearance of the CPU-01 Module.



(2) Indicators

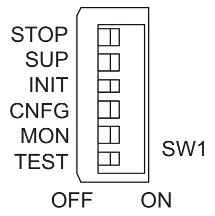
The LED indicators that display the operating status and error details for the Base Unit are detailed in the following table.

Indicators	Indicator Name	Color	Significance when Lit
RDY ○ ○ RUN ALM ○ ○ ERR ○ BAT	RDY	Green	Unit operating normally.
	RUN	Green	User program running.
	ALM	Red	Lights/blinks for warning.
	ERR	Red	Lights/blinks for errors.
	BAT	Red	Battery alarm activated.

Note: Refer to *8.2.3 (2) LED Indicator Meanings* for details on the meaning of indicators.

(3) Switch Settings

The SW1 of the DIP switch sets the operating conditions for the CPU-01 Module when the power is turned ON.



Pin Number	Switch Name	Status	Operating Mode	Default Setting	Details
6	STOP	ON	User program stopped	OFF	Stops the user program execution. Enabled only when debugging.
		OFF	User program running		
5	SUP	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
4	INIT	ON	Memory clear	OFF	Set to ON to clear the memory. If this switch is set to OFF, the program stored in flash memory will be executed.
		OFF	Normal operation		
3	CNFG	ON	Configuration mode	OFF	Set to ON to execute self-configuration for connected devices.
		OFF	Normal operation		
2	MON	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
1	TEST	ON	System use (adjusted before shipment)	OFF	Always set to OFF.
		OFF	Normal operation		

2.3.3 Hardware Specifications

The following table shows the hardware specifications of the CPU-01 Module.

Item	Specifications
Name	CPU-01 Module
Model	JAPMC-CP2200
Abbreviation	CPU-01
Flash Memory	12 MB
SDRAM	32 MB
SRAM	512 KB, M registers, S registers, trace memory, alarm history (battery backup)
Calendar	Seconds to year timer (battery backup)
Indicators	RDY (green) RUN (green) ALM (red) ERR (red) BAT (red)
Switches	STOP SUP INIT CNFG MON TEST
Dimensions (mm)	125 × 95 (H × D)
Mass	66 g

2.3.4 SVR Module (Virtual Motion Module)

(1) Overview

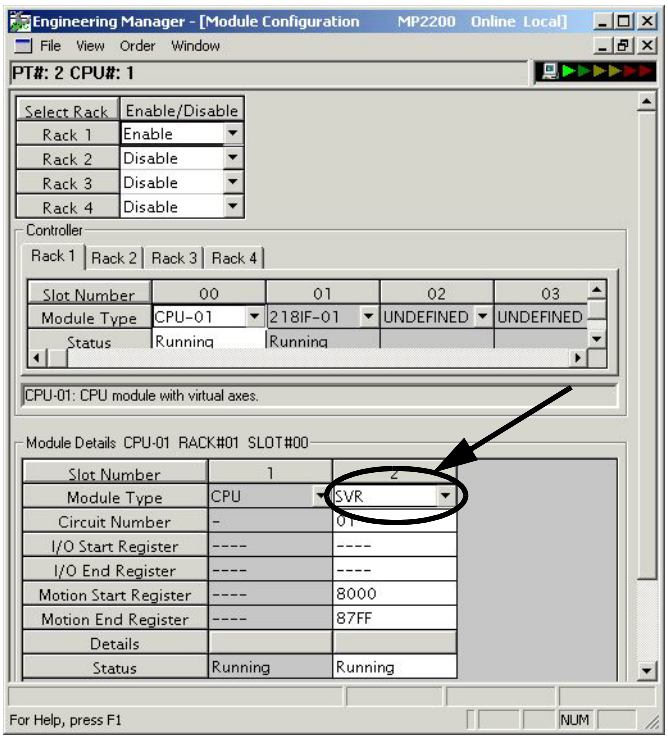
The Virtual Motion Module is a software module provided as a standard feature with the CPU-01 Module. It is not connected to a motor, but provides a virtual axis interface.

The SVR Module has fixed parameters, setting parameters, and monitoring parameters (see *Appendix B SVR Motion Parameters*), and can be accessed from application programs using I/O registers.

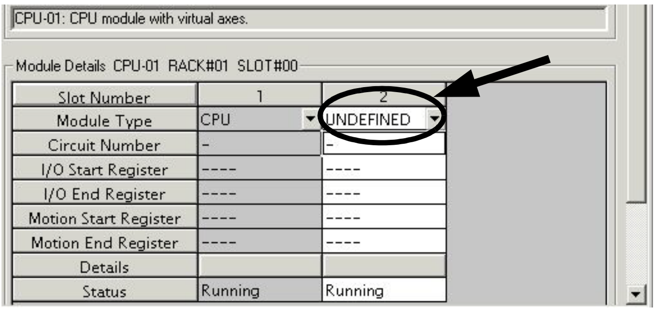
The SVR can be used to control up to 16 virtual axes in the high-speed scan control cycle.

Note: For information on how to use SVR motion parameters and motion commands, refer to *Machine Controller MP2000-series Built-in SVB/SVB-01 Motion Module User's Manual* (manual no.: SIEP C880700 33).

In an MP2200 CPU Module, slot 2 in the default **Module Configuration** Window is for the SVR Module.



Note: If the SVR is not used, MP2200 processing time can be reduced by setting the *Module Type* for SVR to **UNDEFINED** in the **Module Configuration** Window.



(2) Example SVR Usage

The SVR is used in the following two applications.

- **Program testing**: Results are easily obtained without connecting a motor.
- **Generating commands**: If the SVR is used in applications where motion modules are required only for generating commands, such as master axis for phase control or multi-axis synchronous control, then Motion Modules on real axes are no longer required.

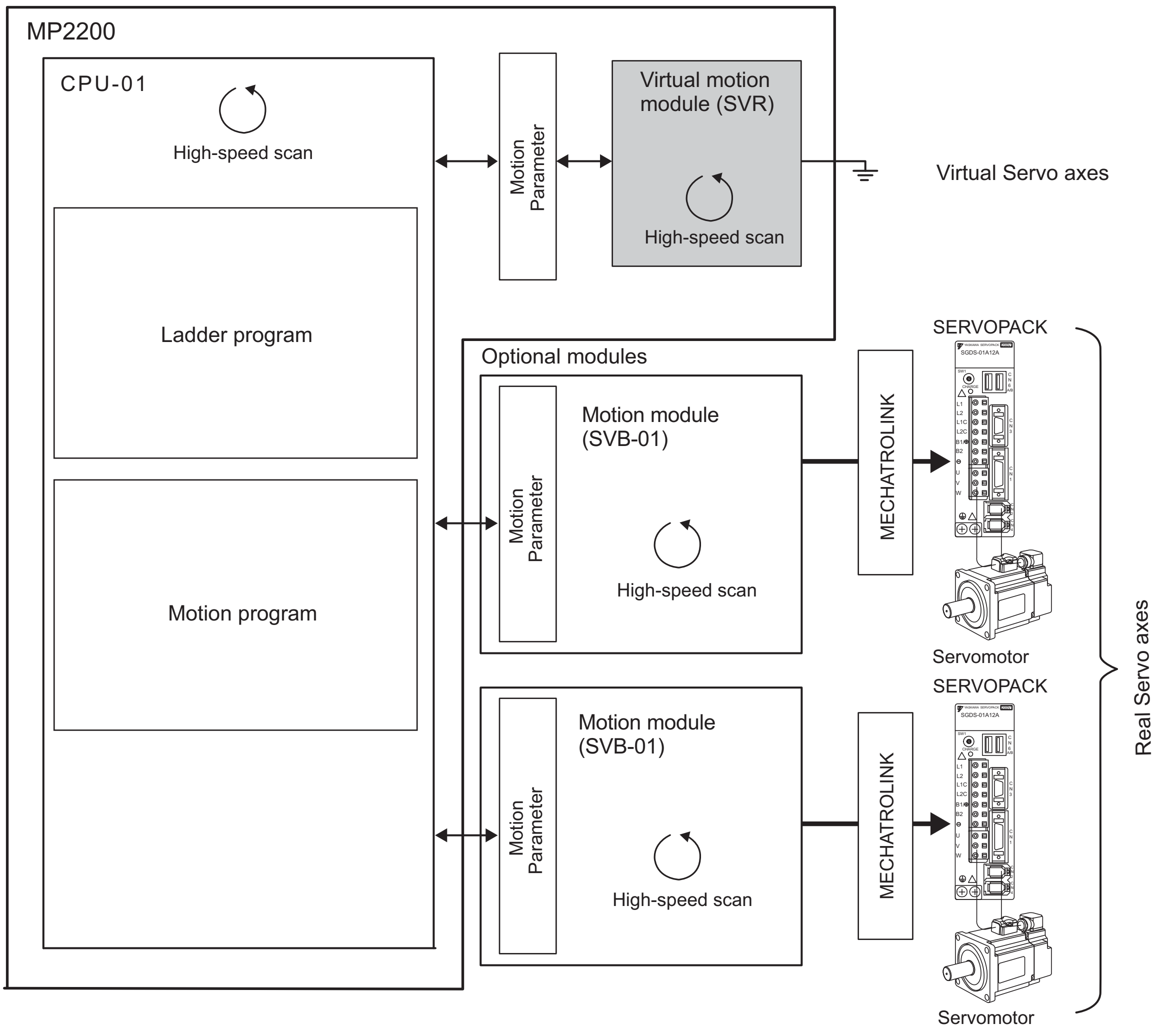
The following table lists application examples of the SVR.

Slot Number	Application Example	Application Method
1	Master axis for phase control	Electronic cam or shaft operation can be achieved by using the SVR for the virtual master axis.
2	Multi-axis synchronous control	Multi-axis synchronous control can be achieved by controlling the SVR from a motion program and then using the ladder program to copy position commands of the SVR to other axes.
3	Sine curve commands	If the motion program is used to perform circular interpolation with the SVR, the axis will operate with a sine curve command.

Note: The software limit function and machine lock function cannot be used with the SVR. The position error will always be 0.

(3) System Configuration Example

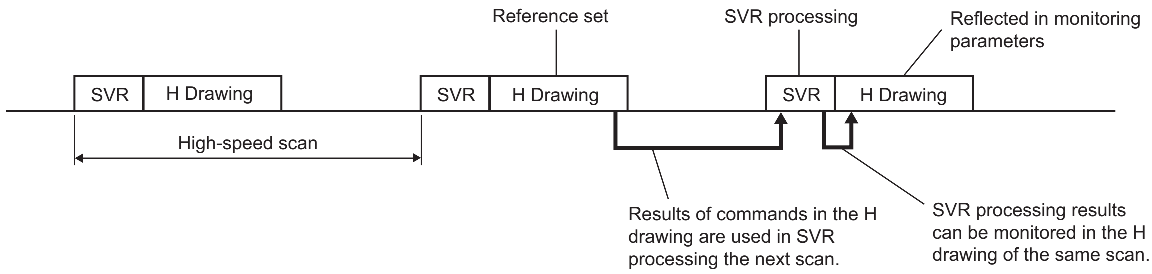
The following figure shows an example system configuration using SVR.



(4) SVR Operation

[a] SVR Execution Timing

The SVR is processed at the beginning of the high-speed scan. SVR processing is performed in the next scan after specifying and the processing results are reflected in the monitoring parameters.



[b] Processing Time

When fixed parameter 0 (Selection of Operation Modes) is set to 0 (Normal Operation Mode), services are started for each of the 16 SVR Module virtual axes.

Note: The default for the Selection of Operation Modes parameter is 1 (Axis Unused).

The following table gives guidelines for the processing time required for each SVR axis.

Command	MP2200
NOP	35 +14 × Number of axes (μs)
POSING	35 +36 × Number of axes (μs)

Note: Number of axes: The number of axes (1 to 16) when Selection of Operation Modes (fixed parameter 0) is set to Normal Operation Mode (0). The formula listed above do not apply when the number of axes is 0.

■ Differences from SVB Simulation Mode

Simulation mode does not have a positioning function, so the position data is refreshed in one scan to the final target position. The SVR has its own positioning function that performs distribution, so like a real module, position data is refreshed each scan for the final target position.

2.4 CPU-02 Module

2.4.1 Overview of Functions

The CPU-02 Module is used exclusively for the MP2200, and has been developed as a higher level of CPU-01 Module.

In addition to the expansion of the user memory, the CPU-02 Module is highly generalized, and has one Compact Flash port and one USB port.

The Compact Flash already in the Compact Flash slot can be used to back up applications directly without going through the MPE720 to save them as a batch. Also, applications can be loaded from the Compact Flash directly to the CPU in batch loads.

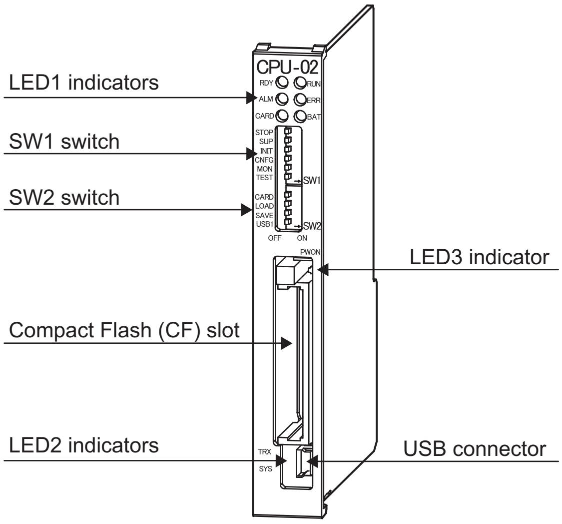
The USB connector has only one channel, which works as a port only for the MPE720. With a USB hub, several CPU-02 Modules can be controlled with one personal computer (MPE720).

Note: MPE720 version 5.31B or later.

2.4.2 External Appearance, Indicators, and Switch Settings

(1) External Appearance

The following figure shows the external appearance of the CPU-02 Module.



Note: A dummy card is inserted in the Compact Flash slot to prevent dust from entering before shipment.

(2) Indicators

The following table shows the indicators that show the operating status of the CPU-02 Module and error information.

Indicators	Indicator Name	Color	Significance when Lit
LED1 RDY ◯ ◯ RUN ALM ◯ ◯ ERR CARD ◯ ◯ BAT	RDY	Green	Unit operating normally. *1
	RUN	Green	User program running. *1
	ALM	Red	Lights/blinks for warning. *1
	ERR	Red	Lights/blinks for errors.*1
	CARD	Green	Accessing the Compact Flash.
	BAT	Red	Battery alarm activated.*1
LED2 TRX ◯ STS ◯	TRX	Green	Communicating with USB. *2
	STS	Red	Lights or blinks for errors. *2
LED3 PWON (On the side of the Compact Flash connector lever)	PWON	Green	Supplying power to the Compact Flash.

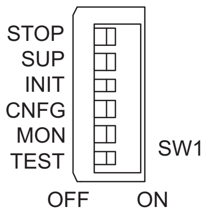
* 1. Refer to *8.2.3 (2) LED Indicator Meanings* on the meaning of indicators.
* 2. Refer to *2.4.6 (3) Details of LED Display (LED2) for USBs* for details of the LED displays.

(3) Switch Settings

[a] SW1

The SW1 of the DIP switch is used to set the operating conditions for the CPU-02 Module when the power supply is turned ON.

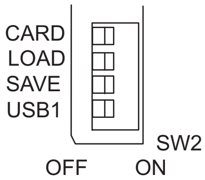
Before turning ON the power supply, set this switch. Any settings made after the power supply is turned ON are invalid.



Pin Number	Switch Name	Status	Operating Mode	Default Setting	Details
6	STOP	ON	User program stopped	OFF	Stops the user program execution. Enabled only when debugging.
		OFF	User program running		
5	SUP	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
4	INIT	ON	Memory clear	OFF	Set to ON to clear the memory. Also set to ON not to save the data in the Compact Flash. If this switch is set to OFF, the program stored in flash memory will be executed.
		OFF	Normal operation		
3	CNFG	ON	Configuration mode	OFF	Set to ON to execute self-configuration for connected devices.
		OFF	Normal operation		
2	MON	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
1	TEST	ON	System use (adjusted before shipment)	OFF	Always set to OFF.
		OFF	Normal operation		

[b] SW2

The SW2 of the DIP switch is used to set the operating conditions for the Compact Flash slot and the USB connector.



Pin Number	Switch Name	Status	Operating Mode	Default Setting	Details
4	CARD	ON	Compact Flash enabled.	OFF	Turns the power supply to the Compact Flash ON or OFF.
		OFF	Compact Flash disabled.		
3	LOAD	ON	Executes batch load from the Compact Flash (Compact Flash → CPU).	OFF	Enabled when switches SW2-4 is ON (Compact Flash enabled). With the LOAD switch set to ON, a batch load from the Compact Flash is executed when the power supply is turned ON.
		OFF	Batch load is not executed.		
2	SAVE	OFF ↓ ON	Save (CPU → Compact Flash).	OFF	Enabled when switches SW2-4 is ON (Compact Flash enabled). By turning it from OFF to ON, a batch save to the Compact Flash is executed.
1	USB1	ON	USB local address1.	OFF	Setting this switch to ON initializes the USB local address and sets it to 1 by force.
		OFF	Start up according to the Module Configuration Definitions of USB.		

2.4.3 Hardware Specifications

The following table shows the hardware specification of the CPU-02 Module.

<table>
<tr><th>Item</th><th colspan="2">Specifications</th></tr>
<tr><td>Name</td><td colspan="2">CPU-02 Module</td></tr>
<tr><td>Model</td><td colspan="2">JAPMC-CP2210</td></tr>
<tr><td>Abbreviation</td><td colspan="2">CPU-02</td></tr>
<tr><td>Flash Memory</td><td colspan="2">16 MB (User’s area: 11.5 MB)</td></tr>
<tr><td>SDRAM</td><td colspan="2">32 MB</td></tr>
<tr><td>SRAM</td><td colspan="2">2 MB, M registers, S registers, trace memory, alarm history (battery backup)</td></tr>
<tr><td>External Memory Interface</td><td colspan="2">Compact Flash interface × 1 slot</td></tr>
<tr><td>Engineering Port</td><td colspan="2">USB interface × 1 channel</td></tr>
<tr><td>Calendar</td><td colspan="2">Seconds to year timer (battery backup)</td></tr>
<tr><td rowspan="2">Indicators</td><td rowspan="2">LED1
RDY (green)
RUN (green)
ALM (red)
ERR (red)
BAT (red)</td><td>LED2
TRX (green)
SYS (red)</td></tr>
<tr><td>LED3
PWON (green)</td></tr>
<tr><td>Switches</td><td>SW1
STOP
SUP
INIT
CNFG
MON
TEST</td><td>SW2
CARD
LOAD
SAVE
USB1</td></tr>
<tr><td>Dimensions (mm)</td><td colspan="2">125 × 95 (H × D)</td></tr>
<tr><td>Mass</td><td colspan="2">75 g</td></tr>
</table>

Note: For information on general specifications, refer to *2.1 Specifications*.

2.4.4 SVR Module (Virtual Motion Module)

Refer to *2.3.4 SVR Module (Virtual Motion Module)*.

2.4.5 CARD Module

The specifications of the Compact Flash interface are described here.

(1) Specifications

[a] General Specifications

The following table shows the general specifications of the Compact Flash interface.

Item	Specifications	Remarks
Slot Standard	For Compact Flash (TYPE-I)	50 pins
Number of Slots	1	–
Interface	PC card ATA	True IDE is not supported.
Applicable Media	Compact Flash (TYPE-1)	The Compact Flash described in the following table is recommended.*
Media Memory Capacity	32 MB, 64 MB, 128 MB, 256MB, 512MB	–
For Partition	Not available	–
Applicable FAT	FAT12/16/32, VFAT (long name)	Up to 256 characters of a file name are available when VFAT is used, but only 246 characters are actually allowed because of full path designation.
Max. No. of Nested Directory	10	–
File Information	Report of updating date available	Calendar function of the controller is used.
Max. Length of File Name and Directory Name	256 characters	Total number of characters in file names and directory names.
Current Directory Function	Not available	–
Max. No. of Simultaneous Open Files	16	–

* Micro drive, I/O device, or HDD is not supported.

- Recommended Compact Flash / Adapter (Available from YASKAWA)

Model	Specifications	Manufacturer
CFI-128MDG	128 MB	Hagiwara Sys-Com Co., Ltd.
CFI-256MDG	256 MB	
CFI-512MDG	512 MB	
CFC-ADP03	PC-card adapter* (for the PCMCIA connector)	

* Use a PC-card type adapter when the Compact Flash is used with a personal computer.
By mounting the Compact Flash on a PC-card adapter, it can be used as a PC-Card Type-II Flash Card.

- Operation Confirmed Compact Flash (Available from an electric appliance shop)

Model	Specifications	Manufacturer
SDCFB-64-J60	64 MB	SanDisk Corporation
SDCFB-128-J60	128 MB	
HPC-CF64V	64 MB	Hagiwara Sys-Com Co., Ltd. (V Series)
HPC-CF128V	128 MB	
HPC-SD128T	128-MB SD memory card	Hagiwara Sys-Com Co., Ltd.
HPC-CDA01	Compact Flash-type SD memory card adapter	

[b] Functions

The following table shows the functional specifications for the Compact Flash interface.

Item	Specifications	Descriptions
Operation Switch	3 pins of SW2	CARD: Enables or disables use of media. LOAD: Executes batch load if set to ON when power supply is turned ON. SAVE: Executes a batch save when set from OFF to ON.
Batch Load	Compact Flash → CPU (Flash memory)	• Executes and controls operation in accordance with settings of SW2. • The data to be transferred is: User application + register.
Batch Save	CPU (SDRAM) → Compact Flash	

[c] Compact Flash-Related System Registers

The following table shows the specifications of the system register related to the Compact Flash interface.

Specifications	Register Number		Description	
Whole capacity of Compact Flash Card	SL00652		Unit: Byte	
Card Status	SW00654	SB006540	0: Compact Flash card not mounted	1: Compact Flash card mounted
		SB006541	0: Power not supplied	1: Power being supplied
		SB006542	0: Compact Flash card not identified	1: Compact Flash card being identified
		SB006543	0: No Compact Flash card access	1: Compact Flash card being accessed
		SB006544	0: –	1: FAT file system being checked
		SB006545 to SB00654F	Reserved for system	
FAT Type	SW00655	0001H	FAT12	
		0002H	FAT16	
		0003H	FAT32	
Reserved for system	SW00656		–	
Reserved for system	SW00657		–	
Batch Load/Save	SW00658	SB006580	During batch load	
		SB006581	Compact Flash card read-out error	
		SB006582	Load file model mismatched	
		SB006583	Load file write-in error	
		SB006584	Flash-storage error	
		SB006585	No batch load folder exists.	
		SB006586	Load error due to prohibition of load (program write protection)	
		SB006587	Reserved for system	
		SB006588	During a batch save	
		SB006589	Compact Flash card write-in error	
		SB00658A	Save file read-out error	
		SB00658B	Security error	
		SB00658C to SB00658F	Reserved for system	
Reserved for system	SW00659		–	

(2) Precautions when Using Compact Flash

Pay attention to the following points when using the Compact Flash.

- **Removing and Reattaching the Compact Flash**

Before removing or inserting the Compact Flash, set the switch SW2-4 (CARD) to OFF, and confirm that the CARD and PWON LED indicator lamps are unlit.

If the Compact Flash is removed or inserted while these LED indicator lamps are lit, the data stored in the Compact Flash may be damaged.

- **Formatting the Compact Flash**

Format the Compact Flash in a computer running on Windows 2000 or Windows XP operating system. The CPU-02 Module cannot be used to format the Compact Flash.

Note: The Compact Flash purchased from YASKAWA has already been formatted.

- **FAT Check and Restoration of Compact Flash**

If the Compact Flash is removed or if the power supply is turned OFF while accessing the Compact Flash with the CARD LED indicator lamp lit, the FAT in the Compact Flash may be damaged.

The CPU-02 Module checks the FAT when the switch SW2-4 (CARD) is set to ON and attempts to restore the data if the FAT is damaged. During data restoration, the CARD LED indicator lamp will flicker ON and OFF, because the Compact Flash is accessed automatically. This is not a failure.

It may take 10 minutes or more to restore the FAT. If in a hurry, forcibly remove the Compact Flash and run an error check on the personal computer to restore the data.

- **INIT Switch Setting and Transfer of Registers M, S, I, and O**

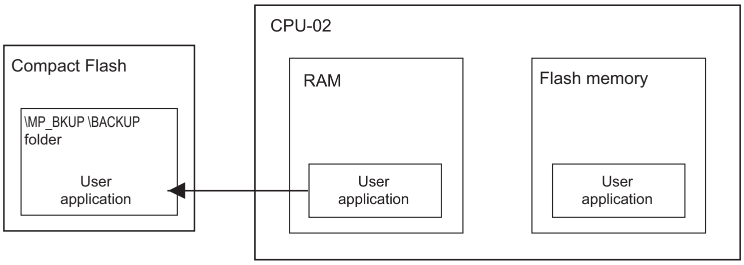
The following table shows registers M, S, I, and O to be transferred between the Compact Flash and the controller according to the status of the INIT the SW1 switch.

Setting of INIT (SW1) Switch	OFF	ON
At batch save	M, S, I, O, C, #, D	C, #, D
At batch load	M, C, #, D	C, #, D

Note:1. S, I, and O registers are not transferred in a batch load regardless of the INIT setting.
2. Registers other than M, S, I, and O registers are transferred both in a batch save and a batch load regardless of the INIT setting.

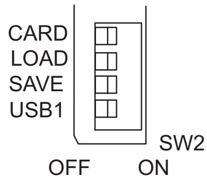
(3) Batch Save to Compact Flash

By using the DIP switches, the batch save function of the CPU-02 Module can save all the user application data from the RAM to the specified folder and file, \MP_BKUP\BACKUP, of the Compact Flash without going through the MPE720.



Note: 1. If data in the Compact Flash has already been saved in a batch or another form written in from the MPE720, it will be cleared when a new batch is saved.
2. A batch load is not possible if a motion register is used for trace definition, or if the relevant motion module is not defined or not mounted.
3. The security function is not provided for program transfers with the Compact Flash in the CPU software (version 2.41 or earlier).

The procedure for batch saving to the Compact Flash is as follows:



1. Confirm that the CARD, LOAD and SAVE of the CPU-02 Module’s SW2 are set to OFF.

2. Insert the formatted Compact Flash into the Compact Flash slot on the CPU-02 Module with the power supply ON.

3. Set the CARD switch to ON.

The PWON LED indicator lamp will light up.

4. Set the SAVE switch to ON.

The batch save will start, and the CARD LED indicator lamp will light up. When a batch save is completed, the CARD LED indicator lamp will go out.

Note: If the operation fails, the ALM LED indicator lamp will light up, and the error will be reported to the system register SW00658. Refer to *3.2.2 [a] Compact Flash (CF) Slot Specifications*.
The ALM LED indicator lamp will turn OFF by re-executing batch save or by restarting the MP2200.

• How to Input Data from the Compact Flash to the MPE720

The MPE720 can read application data that was batch-saved to Compact Flash. For information on how to read the data, refer to the MPE720 user's manual.

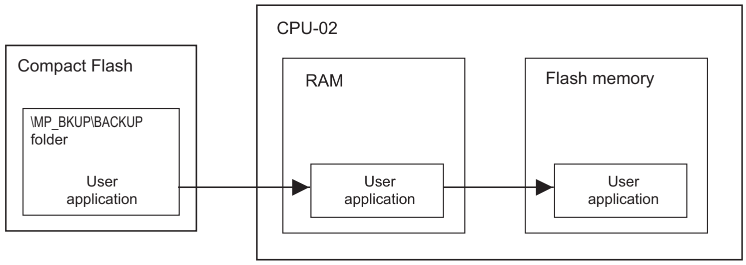
• Writing in from the MPE720 to the Compact Flash

The MPE720 can write application data to the Compact Flash if the application data was batch-dumped from the Controller. If data that is created or edited off-line is written in to the Compact Flash, an error may occur in a batch load.

Refer to the MPE720 user's manual for information on how to write application data to the Compact Flash using the MPE720.

(4) Batch Load from Compact Flash

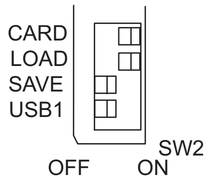
By using only the DIP switches, you can use the batch load function of the CPU-02 Module to read in user application data in the Compact-Flash-specified folder and file, \MP_BKUP\BACKUP, and also flash-store the data without going through the MPE720.



To load data in a batch from the Compact Flash, use the following procedure:

1. Insert the Compact Flash with the backup application data into the Compact Flash slot of the CPU-02 Module with the power supply OFF.

2. Set the CARD and LOAD switches of SW2 of the CPU-02 Module to ON.



3. Turn ON the MP2200 power supply.

The PWON LED indicator lamp will light up, and the batch load will start. The CARD LED indicator lamp will light up, and the RUN LED indicator lamp will blink. When all of the data has been successfully read out from the Compact Flash, the CARD LED indicator lamp will turn OFF.
After the data has been successfully saved, the RUN and PWON LED indicator lamps will turn OFF.

Note: If a load failure occurs, the ALM indicator will light, and the error details will be reported in system register SW00658. (Refer to *2.4.5 (1) [c] Compact Flash-Related System Registers*.) If you restart the MP2200, the ALM LED indicator lamp will turn OFF.

4. Turn OFF the MP2200 power supply.

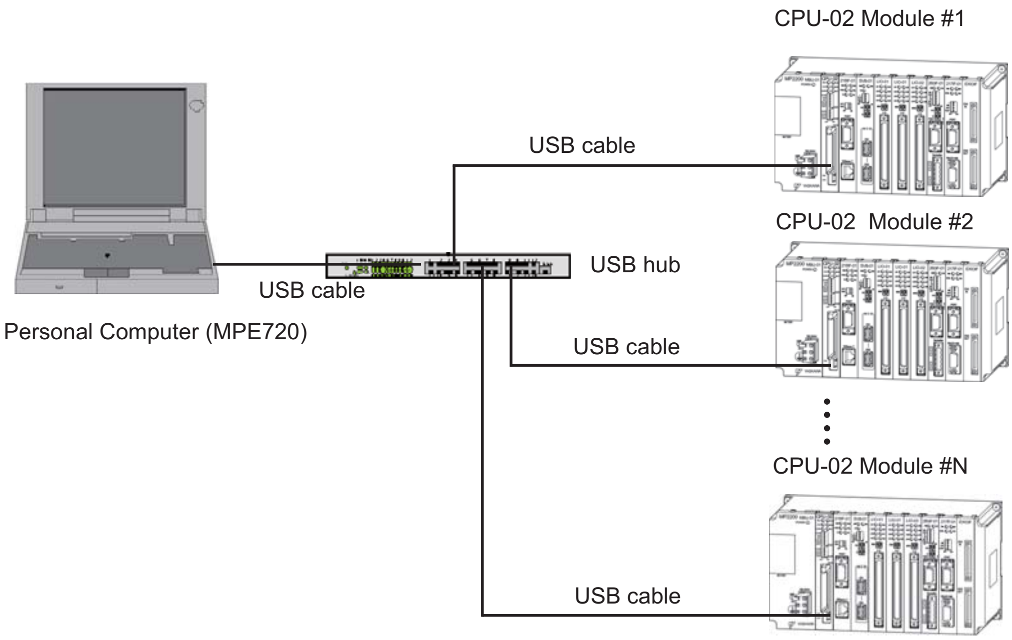
5. Set the CARD, LOAD, and INIT switches to OFF. Then, turn ON the MP2200 power supply to start from the flash memory.

2.4.6 USB Module

(1) Overview

By connecting a computer with MPE720 installed, either directly or via a USB hub with a CPU-02 Module, MP2200 controller engineering can be done through the MPE720-equipped computer.

- Example of Configuration Using Several CPU-02 Modules



Note: 1. USB corresponding communications process and exclusive-use device driver must be installed for this function. For information on the installation method, refer to *2.4.6 (4) Installing the Hardware Driver*.
2. Up to 5 USB hubs can be connected.

(2) General Specifications

The following table describes the general specifications of the USB interface.

Item	Specifications	Remarks
Standard	In accordance with USB2.0 Full-speed function	–
Bus speed	Full speed (12 Mbps)	–
Format	Asynchronous serial	–
Communication Method	Half duplex	–
Connected Devices	One USB host, 127 slaves	Including hub
Cable	USB cable available at most electric appliance stores	Connector: PC side: Series A plug Module side: Series mini-B plug
Cable Length	Max. length in full-speed mode: 5 m (30 m at full speed with Hub 5-layer connection)	–
Connector Type	Series mini-B plug	–
Power Supply	Self-powered type	Shares power supply with the MP2200.

(3) Details of LED Display (LED2) for USBs

Classification	Indicator Name		Indicator Details	Description
	TRX	STS		
Normal Operation	Not lit	Not lit	During normal operation	LED is OFF when USB transmission is not carried out even after the power supply is turned ON.
Transmission	Blinking	Not lit	During USB transmission	TRX flashes during USB transmission.
Error	Undefined	Blinking	Hardware error (Meaning differs depending on the number of blinks.) Number of blinks and error 1: ROM diagnostic error 2: RAM diagnostic error 3: Shared-memory diagnostic error 15: Watchdog timeout error	Hardware error in the USB-control section. The module must be replaced.

(4) Installing the Hardware Driver

Before connecting the MP2200 (CPU-02 Module) to the personal computer via a USB for the first time, the USB communications driver for the MP series must be installed. Prepare the setup disc (Ver 5.31B or later) of the MPE720, and install the USB driver as described in this section.

1. Connect the USB port on the CPU-02 Module to the USB port on the personal computer with a USB cable (series mini-B), and turn ON the MP2200 power supply.

2. A message **Find New Hardware** will be displayed. Then, **Found New Hardware Wizard** will be displayed. Click the **Next** button.

Overview of Specifications and Functions

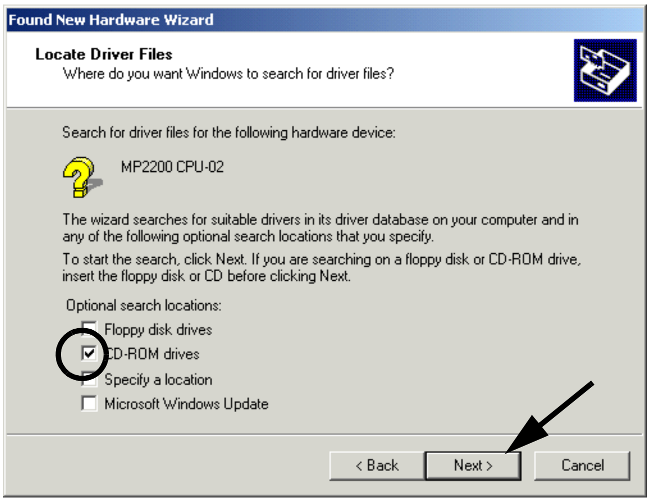
2

3. Select **Display a list of the known drivers for this device so that I can choose a specific driver** and click the **Next** button.

A message will appear, asking where to find the driver.

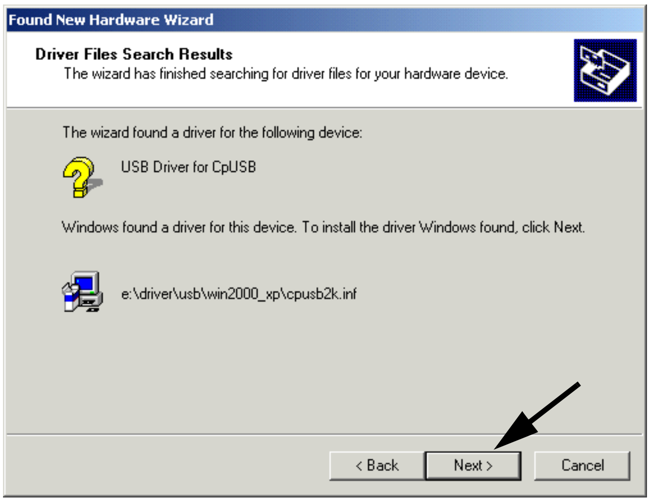
4. Insert the MPE720 setup disc into the CD-ROM drive.

5. Select **CD-ROM drives** and click the **Next** button.



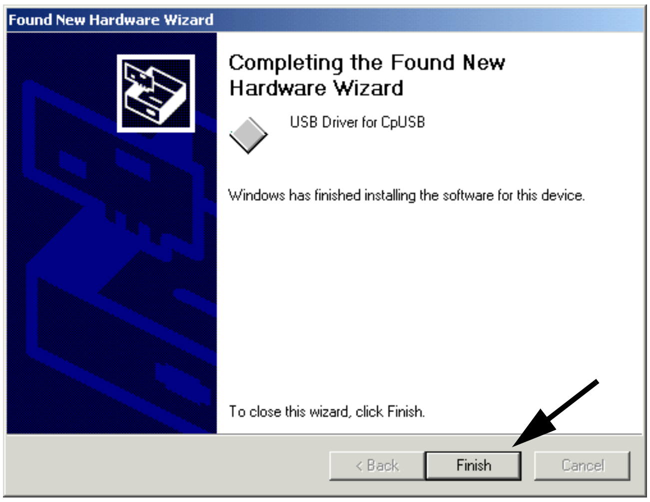
If the driver is found, the search results will appear.

6. Click the **Next** button.



The installation will start.

7. After the driver has been successfully installed, click the **Finish** button.



(5) Removing the Cable from the USB Connector

When removing the cable from the USB connector on the personal computer, the USB hub, or the CPU-02 Module, or when turning OFF the MP2200 power supply connected to a USB, perform the following operations to safely undo the connections.

1. In the task tray at the lower right of the window, click the icon to remove the device.

A message **Stop USB Mass Storage Device-Drive (F:)** will appear.

2. Click inside the window.

The connection will be cut and a window, **The 'USB Mass Storage Device' device can now be safety removed from the system**, will appear. After this message appears, the cable can be safely removed and the MP2200 power supply can be turned OFF.

Click the inside the window or click the **OK** button in the dialog box to close the window.

(6) Creating Project Files for a USB Connection

This section describes how to create a project file that specifies USB for the communication port.

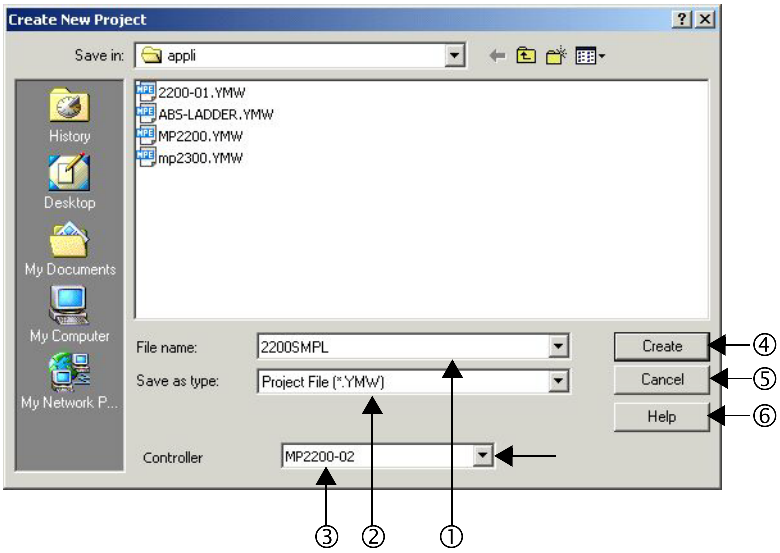
1. Use either of the following methods to start creating a new project file.

- Select ***Project - New*** from the start window.
- Select ***File - New Project*** from the main menu.

Note: A message will be displayed to confirm whether to compile and save any project file that is being edited. The following processing will be performed when a button is pressed.

- Yes Button: The project file will be compiled, saved, and closed.
- No Button: The project file will be closed without compiling or saving it.
- Cancel Button: Creating the new file will be canceled, and editing will continue.

2. Open the folder at the save location, input the file name, and select the file type and machine controller to be used.



The settings and setting procedure are described below.

① File name
Inputs a new name for a project file.

② Save as type
Displays the extensions that can be selected for the project file.
Example: Project File (*.YMW)

③ Controller
Selects the type of Controller to be used.
In this case, select MP2200-02.

④ Create Button
Creates a file with the name that was input in ①.

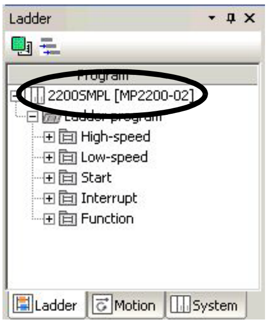
⑤ Cancel Button
Cancels the operation to name the file and save it, and closes the dialog box.

⑥ Help Button
Displays the Select Help Dialog Box.

Note: 1. The following characters cannot be used in file names: / , : * ? " < > |
2. You can select either *Project File (*.YMW)* or *all files (*.*)* for the *Save as type*. A project file (*.YMW) will be created either way.

3. Click the **Create** Button.

A project file create/open message will be displayed and the project file will be created in the specified folder and displayed in the Ladder Subwindow.

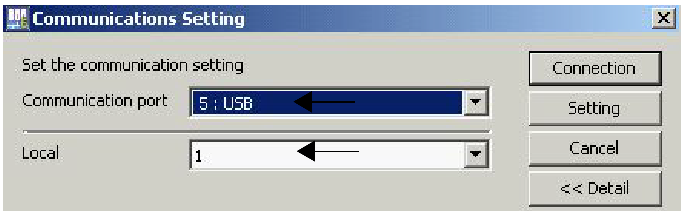


4. Select ***Online - Communication Settings*** from the main menu. Or, click ***Controller - Communication Settings*** in the start window (or ***Project - Communication Settings*** in the start window).

The Communications Settings Dialog Box will appear.

5. Select USB for the communication port.

Click the **Detail** Button, and then set the local address.



6. Click the **Setting** Button.

The communication settings will be completed, and the dialog box will close.

Note: While the machine controller is connected, the Connection Button and Setting Button will be disabled and it will not be possible to set communications. The operation of MPE720 version 6 when connected to the Machine Controller is different depending on whether the connection information (i.e., the settings in the Communications Settings Dialog Box) have been saved in the project files.

7. Select ***File - Overwrite Save*** Project from the main menu.

A message saying to wait while the project file is being saved will be displayed, and then the project file being edited will be saved.

Note: The local address to be set in this paragraph is used to specify the corresponding CPU-02 Modules from the MPE720 (personal computer) side. If one CPU-02 Module is connected to the personal computer, set it to 1. If more than one CPU-02 Module is connected to the personal computer, use the appropriate setting. For details, refer to *2.4.6 (7) Setting the Local Station when Connecting Several CPU-02 Modules via USB*.

(7) Setting the Local Station when Connecting Several CPU-02 Modules via USB

To connect several CPU-02 Modules to the personal computer via the USB hub, perform the following connections and settings. Incorrect settings may overlap local addresses, resulting in an error.

1. Execute self-configuration in advance before the USB connection of a CPU-02 Module that is being used for the first time.

Note: For information on the procedure for executing self-configuration, refer to *5.4 Self-configuration*.

2. If some CPU-02 Modules are already connected, remove all the CPU-02 cables from the USB hub.

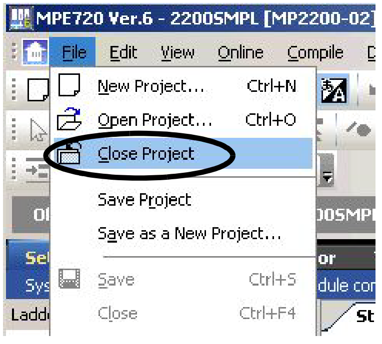
Note: Refer to *2.4.6 (5) Removing the Cable from the USB Connector*.

3. After turning ON the USB1 switch (SW2) of the CPU-02 Modules where the local station is to be set, use the USB cable to connect the CPU-02 Modules to the USB hub.

If the USB1 switch (SW2) is set to ON, the CPU-02 module will operate as local address 1 regardless of the Module Configuration Definitions.

4. Start MPE720 version 6.

Note: If any project file is in MPE720 version 6, close the file.

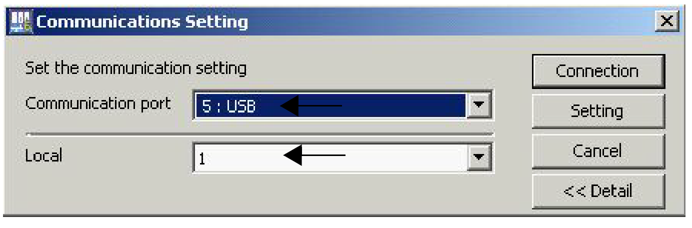


5. Select ***Online - Communications Settings*** from the main menu. Or, click ***Controller - Communication Settings*** in the start window (or ***Project - Communication Settings*** in the start window).

The Communications Settings Dialog Box will be displayed.

6. Select USB for the communication port.

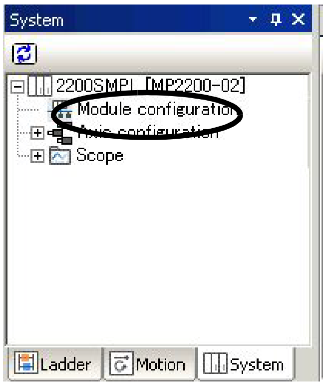
Click the **Detail** Button, and then set the local address to 1.



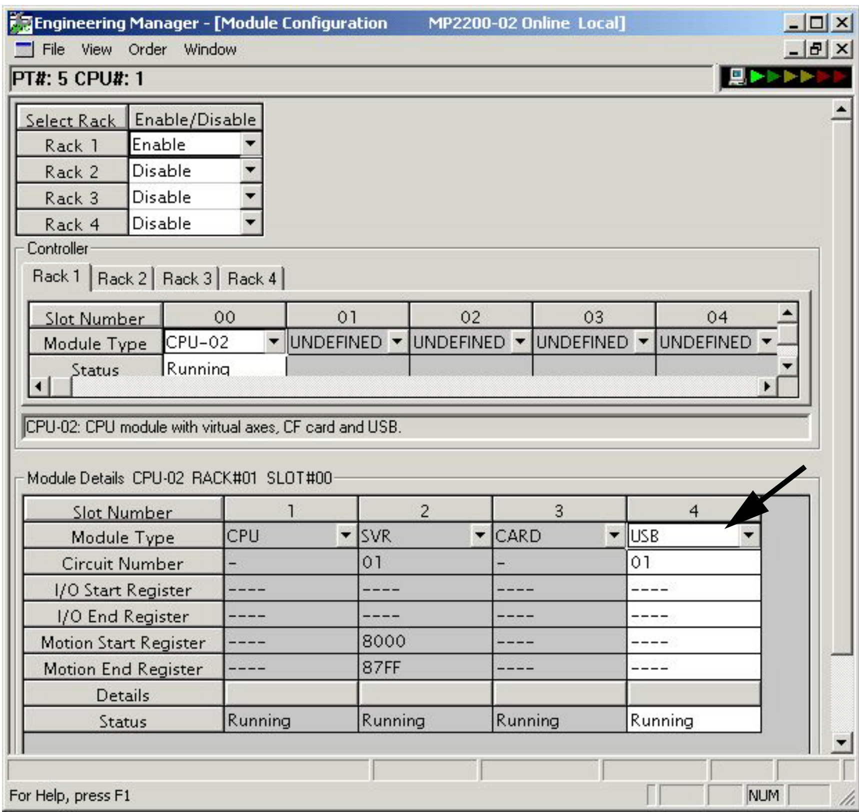
7. Click the **Connection** Button.

The communication settings will be completed, and the system will be connected online.

8. Double-click the **Module configuration** Icon in the System Subwindow to display the window for module configuration definitions.



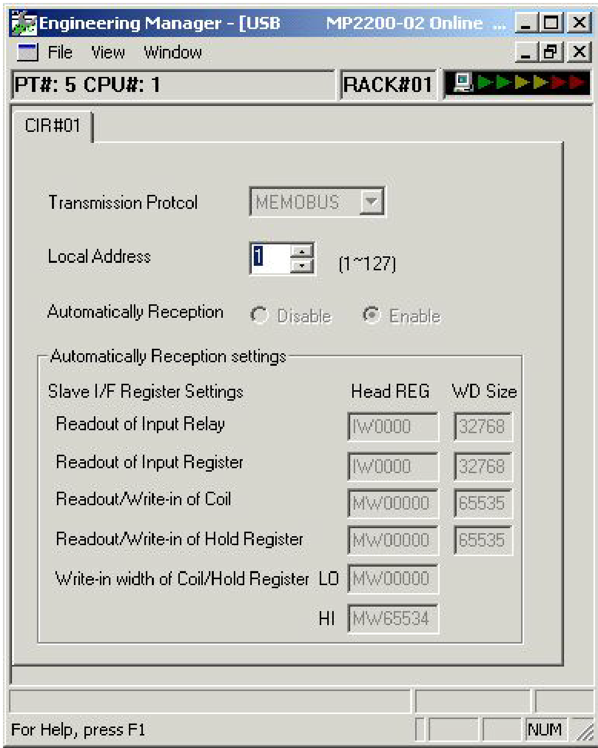
9. Double-click **USB** in the Module Details Area.



The dialog box for setting the local address will be displayed.

10. Set the local address, and then click the **Close** Button.

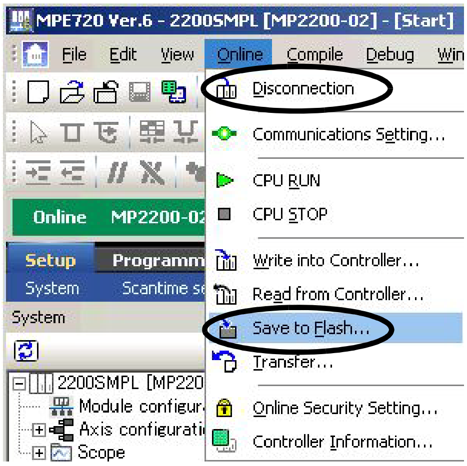
For the local address, set an address that is not already being used by other CPU-02 Modules.



Click the **OK** button in the dialog box to set the local address.

Note: The local address that is set here is used to write-in to the CPU-02 Module.

11. Flash-store the data and log off after returning to the **MPE720 Ver.6** Window.



12. Turn OFF the USB1 switch (SW2) of the CPU-02 Module and then turn the power supply of the MP2200 OFF and back ON again.

The local address of the USB in the CPU-02 Module will be changed to the new address after restart of the power.
To add another CPU-02 Module, repeat the above steps 1 to 9.
Refer to *2.4.6 (6) Creating Project Files for a USB Connection* for the procedure to create a project file to connect to a CPU-02 Module.

2.5 CPU-03 Module

2.5.1 Overview of Functions

The CPU-03 Module is a special CPU Module for the MP2200. It was designed as a upper-end model to provide greater speed and easier operation than the CPU-01 Module and CPU-02 Module.
The CPU-03 Module has one Compact Flash interface and one Ethernet interface with the following characteristics.

• One Standard-feature Slot for Compact Flash (CF) Interface

- Applications can be directly backed up (i.e., batch-saved) to the Compact Flash mounted in the CF slot without going through the MPE720.
- Applications can be loaded directly to the CPU Module from the Compact Flash (i.e., batch-loaded).

• Standard-feature Ethernet (100 Mbps)

- High-speed communication can be performed with the MPE720 Engineering Tool.
- Communication can be performed with a touch panel without using ladder programming (automatic reception).
- Communication can be performed with the host PLC without using ladder programming (I/O message communication).

• Simple Programming

- The operating procedures have been greatly streamlined through execution of motion operations.
- The motion program can be started from the host PLC without programming simply by creating a motion program and registering the execution sequence.

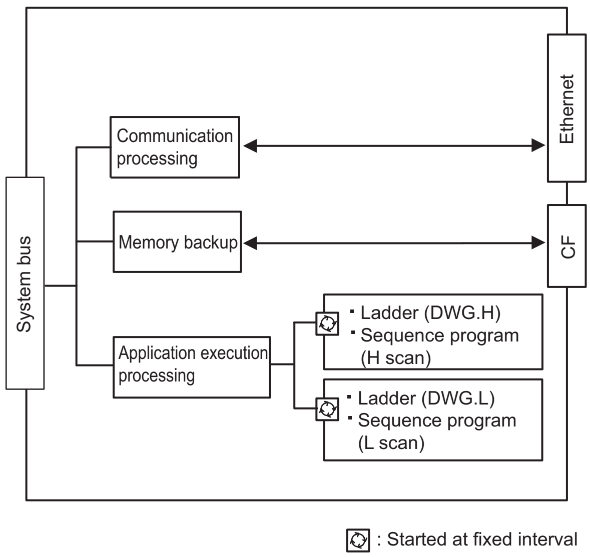
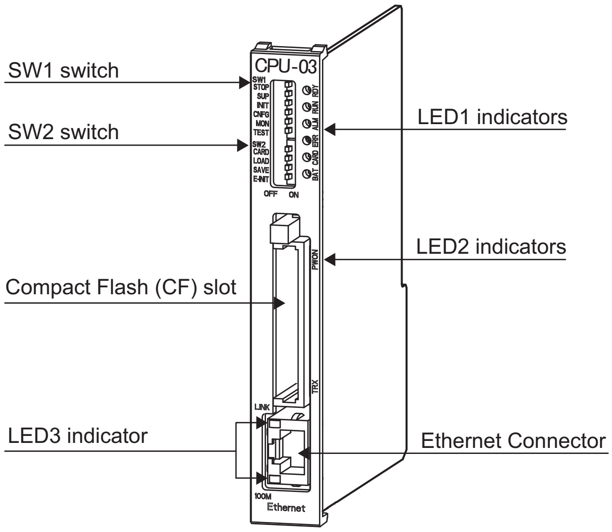


Fig. 2.1 Overview of CPU-03 Module Functions

2.5.2 External Appearance, Indicators, and Switch Settings

(1) External Appearance

The following figure shows the external appearance of the CPU-03 Module.



Note: A dummy card is inserted in the Compact Flash slot to prevent dust from entering before shipment.

(2) Indicators

The following table shows the indicators that show the operating status of the CPU-03 Module and error information.

Indicators	Indicator Name	Color	Significance when Lit
LED1 RDY RUN ALM ERR CARD BAT	RDY	Green	Unit operating normally. *
	RUN	Green	User program running. *
	ALM	Red	Lights/blinks for warning. *
	ERR	Red	Lights/blinks for errors.*
	CARD	Green	Accessing the Compact Flash.
	BAT	Red	Battery alarm activated.*
LED2 PWON TRX (Mounted on the back of the CF connector.)	PWON	Green	Supplying power to the Compact Flash.
	TRX	Green	Transmitting or receiving Ethernet data.
LED3 LINK 100M (Built into the Ethernet connector.)	LINK	Yellow	When there is an Ethernet link.
	100M	Green	Lit: Connected at 100 Mbps or performing auto-negotiation. Not lit: 10 Mbps

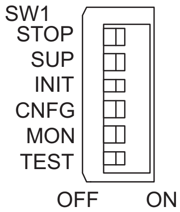
* Refer to *8.2.3 (2) LED Indicator Meanings* on the meaning of indicators.

(3) Switch Settings

[a] SW1

The SW1 of the DIP switch is used to set the operating conditions for the CPU-03 Module when the power supply is turned ON.

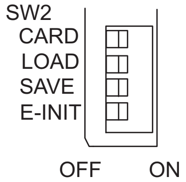
Before turning ON the power supply, set this switch. Any settings made after the power supply is turned ON are invalid.



Pin Number	Switch Name	Status	Operating Mode	Default Setting	Details
6	STOP	ON	User program stopped	OFF	Stops the user program execution. Enabled only when debugging.
		OFF	User program running		
5	SUP	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
4	INIT	ON	Memory clear	OFF	Set to ON to clear the memory. Also set to ON not to save the data in the Compact Flash. If this switch is set to OFF, the program stored in flash memory will be executed.
		OFF	Normal operation		
3	CNFG	ON	Configuration mode	OFF	Set to ON to execute self-configuration for connected devices.
		OFF	Normal operation		
2	MON	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
1	TEST	ON	System use (adjusted before shipment)	OFF	Always set to OFF.
		OFF	Normal operation		

[b] SW2

SW2 is used to set the CF slot and Ethernet operating conditions.



Pin Number	Switch Name	Status	Operating Mode	Default Setting	Details
4	CARD	ON	Compact Flash enabled.	OFF	Turns the power supply to the Compact Flash ON or OFF.
		OFF	Compact Flash disabled.		
3	LOAD	ON	Executes batch load from the Compact Flash (Compact Flash → CPU).	OFF	Enabled when switches SW2-4 is ON (Compact Flash enabled). With the LOAD switch set to ON, a batch load from the Compact Flash is executed when the power supply is turned ON.
		OFF	Batch load is not executed.		
2	SAVE	OFF ↓ ON	Save (CPU → Compact Flash).	OFF	Enabled when switches SW2-4 is ON (Compact Flash enabled). By turning it from OFF to ON, a batch save to the Compact Flash is executed.
1	E-INIT	ON	Transmission parameter defaults for Ethernet	OFF	If E-INIT is ON, the IP address and other transmission parameters will be set to default values at startup.
		OFF	Start up according to the Module Configuration Definitions for Ethernet.		

2.5.3 Hardware Specifications

The following table shows the hardware specification of the CPU-03 Module.

<table>
<tr><th>Item</th><th colspan="2">Specifications</th></tr>
<tr><td>Name</td><td colspan="2">CPU-03 Module</td></tr>
<tr><td>Model</td><td colspan="2">JAPMC-CP2220-E</td></tr>
<tr><td>Abbreviation</td><td colspan="2">CPU-03</td></tr>
<tr><td>Flash Memory</td><td colspan="2">16 MB (User’s area: 11.5 MB)</td></tr>
<tr><td>DDR</td><td colspan="2">64 MB</td></tr>
<tr><td>SRAM</td><td colspan="2">4 MB (3 MB max. with battery backup can be used for table data.)</td></tr>
<tr><td>External Memory Interface</td><td colspan="2">Compact Flash interface × 1 slot</td></tr>
<tr><td>Communication Interface</td><td colspan="2">One Ethernet interface channel</td></tr>
<tr><td>Calendar</td><td colspan="2">Seconds to year timer (battery backup)</td></tr>
<tr><td rowspan="2">Indicators</td><td rowspan="2">LED1
RDY (green)
RUN (green)
ALM (red)
ERR (red)
CARD (green)
BAT (red)</td><td>LED2
PWON (green)
TRX (green)</td></tr>
<tr><td>LED3
LINK (yellow)
100M (green)</td></tr>
<tr><td>Switches</td><td>SW1
STOP
SUP
INIT
CNFG
MON
TEST</td><td>SW2
CARD
LOAD
SAVE
USB1
E-INIT</td></tr>
<tr><td>Dimensions (mm)</td><td colspan="2">125 × 95 (H × D)</td></tr>
<tr><td>Mass</td><td colspan="2">86 g</td></tr>
</table>

Note: For information on general specifications, refer to *2.1 Specifications*.

2.5.4 SVR Module (Virtual Motion Module)

Refer to *2.3.4 SVR Module (Virtual Motion Module)*.

2.5.5 CARD Module

Refer to *2.4.5 CARD Module*.

2.5.6 218IFA Module (Ethernet)

(1) Overview of 218IFA Module Functions

The 218IFA Module is a 10Base-T/100Base-TX Ethernet interface. It is a standard-feature communication interface for the CPU-03 Module.

• 100Mbps transmission speed is supported (100Base-TX).

• Supports the following various communication protocols:

  • Support for MEMOBUS protocol, Extended MEMOBUS protocol
  • Support for MELSEC protocol (A-compatible I/E frame)
  • Support for MODBUS/TCP protocol
  • Support for non-procedure communication

• An I/O message communication function enables you the data exchange in the form of I/O image when communicating with the host PLC, eliminating you from creating a ladder program.

• An automatic receive function eliminates you from creating a ladder program when connected to the indicator and the like.

• The 218IFA Module can be used as the standard interface with the MPE720 Engineering Tool. It provides a simple function for connecting to the MPE720, allowing you to connect to MPE720 without knowing the IP address of the 218IFA Module.

(2) Specification of 218IFA Module

The following table shows the specification of the 218IFA Module.

Items		Description
Communication Interface *1		10Base-T/100Base-TX
Communication Protocol *2		TCP/UDP/IP/ARP/ICMP
Maximum Number of Communication Connections		4+2 (I/O Message communication)
Maximum Number of Communication Channels		4+2 (I/O Message communication)
Message Communication (maximum)	MEMOBUS	Write: 100W Read: 125W
	Extended MEMOBUS	Write: 2043W Read: 2044W
	MELSEC	Write: 1017W Read: 1017W
	MODBUS/TCP	Write: 100W Read: 125W
	Non-procedure	Write: 2046W
I/O Message Communication (maximum)	MEMOBUS	Write: 100W Read: 125W
	Extended MEMOBUS	Write: 1024W Read: 1024W
	MELSEC	Write: 256W Read: 256W
	MODBUS/TCP	Write: 100W Read: 125W
Automatic Receive	MEMOBUS	Provided
	Extended MEMOBUS	Provided
	MELSEC	Provided
	MODBUS/TCP	Provided
Non-procedure Receive Buffer Mode Selection *3		Provided
Simple Function for Connecting with Engineering Tool		Provided

* 1. Communication Interface

The discrimination between 10Base-T/100Base-TX and full-duplex/half-duplex is done by 218IFA based on the remote device. When connecting to a device without automatic negotiation function, set the remote device to half-duplex mode.

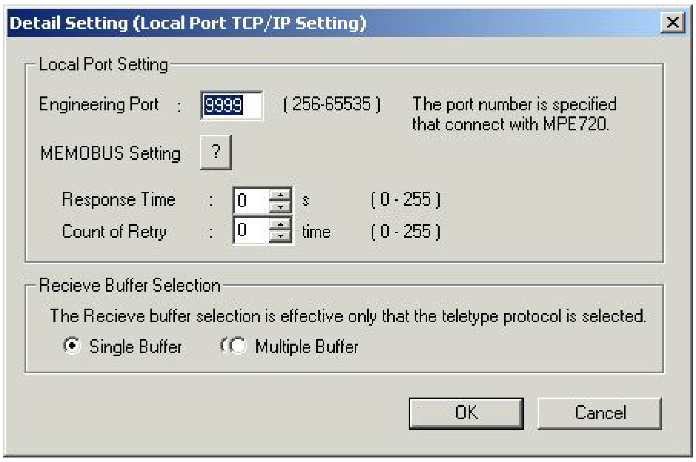
Correspondence of Communication Mode

218IFA Module	Remote Device				
	Automatic Negotiation	10Base-T Half-duplex	10Base-T Full-duplex	100Base-TX Half-duplex	100Base-TX Full-duplex
Automatic Negotiation	Depends on the remote device	Communicates in 10Base-T half-duplex mode	Unable to communicate	Communicates in 100Base-TX half-duplex mode	Unable to communicate

* 2. Communication Protocols

- TCP(Transmission Control Protocol): Connection-oriented transport layer protocol
- UDP(User Datagram Protocol): Connectionless transport layer protocol
- IP(Internet Protocol): Protocol for establishing a communication link between computers
- ICMP(Internet Control Message Protocol): Error control protocol for IP protocol
- ARP(Address Resolution Protocol): Address resolving protocol. Protocol for converting IP address into MAC address

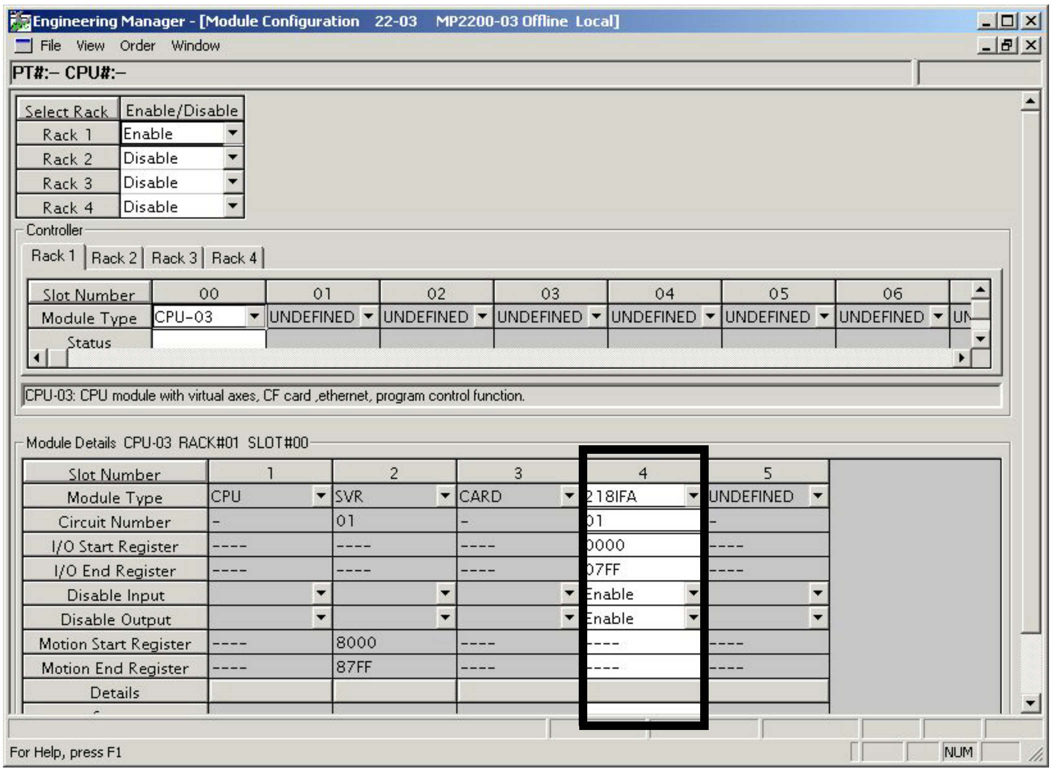
* 3. Non-procedure Receive Buffer Mode Selection
When the non-procedure application protocol of the 218IFA is used, either a single buffer or multiple buffers can be selected for the receive buffers in the 218IFA.
If multiple buffers are selected, 20 data items per connection can be processed at the same time with data continuously received at an interval shorter than the startup interval of the MSG-RCV function.



(3) Module Configuration Definition

[a] Module Configuration Window Details

Click **CPU-03** in the **Controller** area to display the details of the Basic Modules' functions in the **Module Details** area. The cell No.4 provides a detailed definition of 218IFA.



Items displayed in the **Module Details** area show the following meanings:

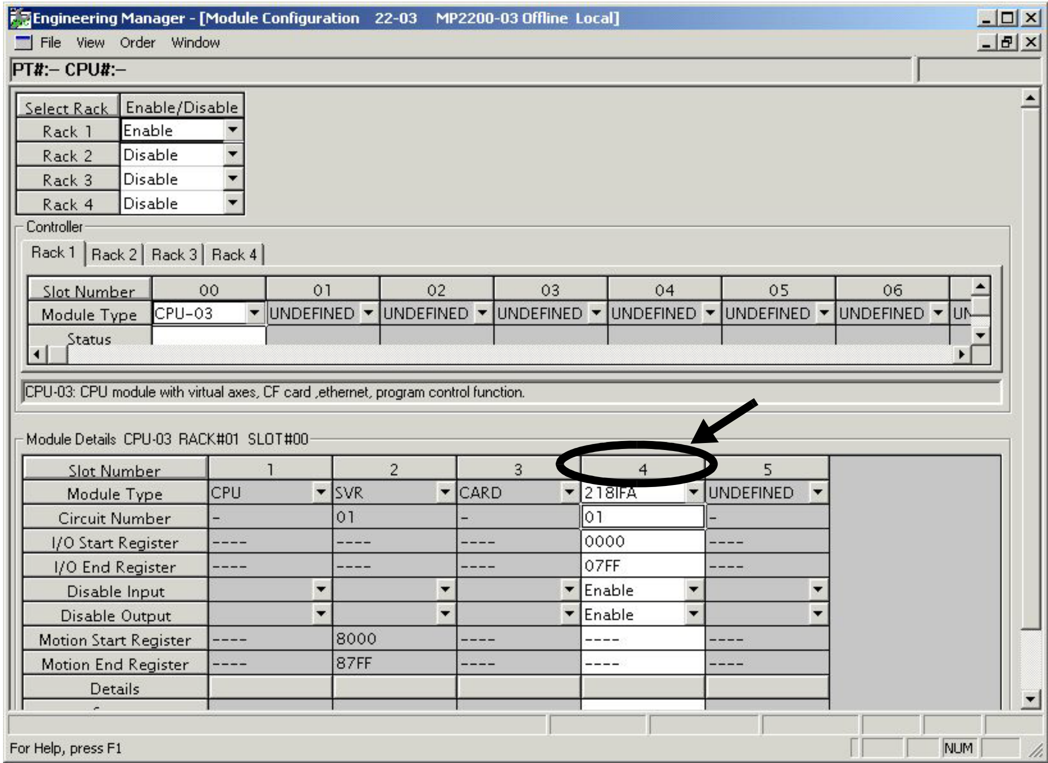
Items	Descriptions	Change
Slot Number	Sub-slot number. Double-click it to open the 218IFA detailed definition window.	–
Module Type	A module name is shown. Changing the name to UNDEFINED enables you to disable 218IFA functions.	√
Circuit Number	Module's line number (valid range: 01-08)	√
I/O Start Register	Start register of the I/O register used in the I/O message communication of 218IFA (valid range: 0000-7FFFh, size: 800h words)	√
I/O End Register	End register of the I/O register used in the I/O message communication of 218IFA (valid range: 0000-7FFFh, size: 800h words)	√
Disable Input	Input Enable/Disable.	√
Disable Output	Output Enable/Disable.	√
Motion Start Register	Not used. Fixed at "– – – –".	–
Motion End Register	Not used. Fixed at "– – – –".	–
Details	Not used.	–
Status	218IFA module status in online mode.	–

√: Available, –: Not available

(4) 218IFA Module Detailed Definition Window

[a] Displaying the 218IFA Module Detailed Definition Window

Select **CPU-03** in the **Controller** Area of the **Module Configuration** Window and then double-click the **Slot Number** cell for the 218IFA Module in the **Module Details** Area to display the detailed definition for the 218IFA Module.

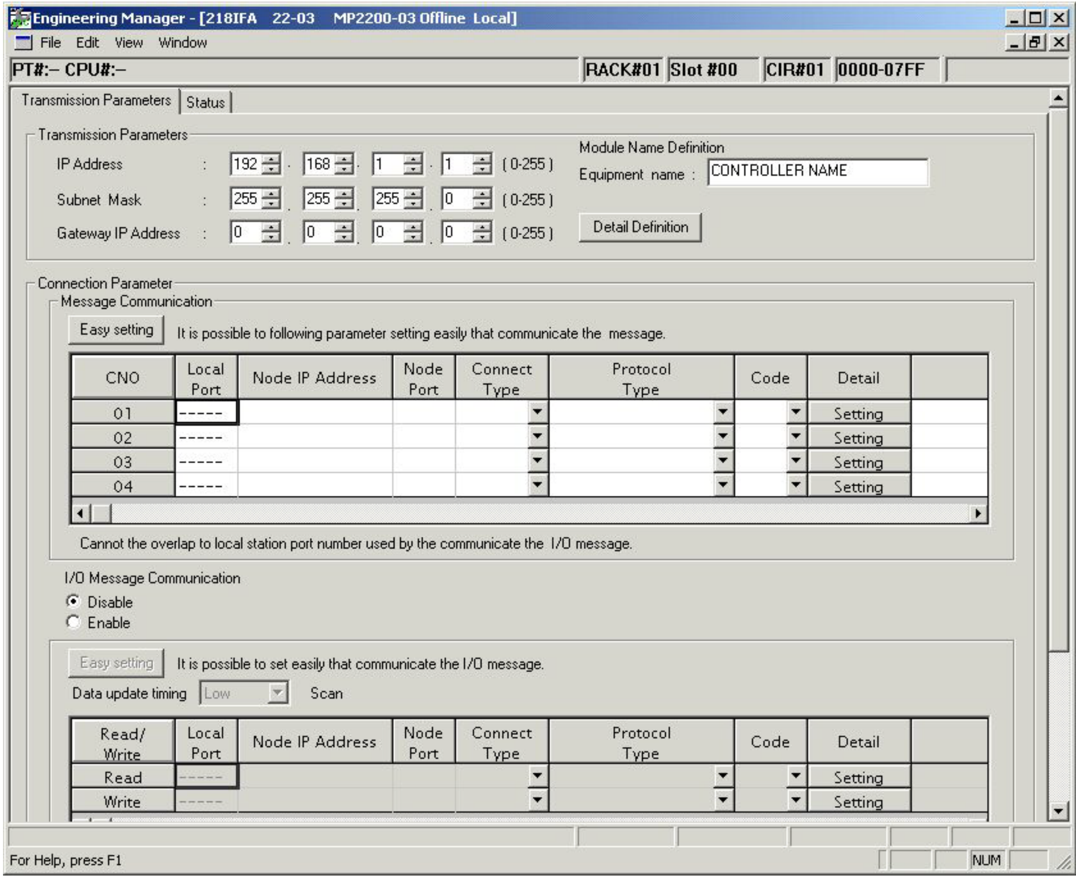


[b] 218IFA Module Detailed Definition Window

The 218IFA Module detailed definition window is composed of **Transmission Parameter** and **Status** Tabs, and each tab is changed with a click.

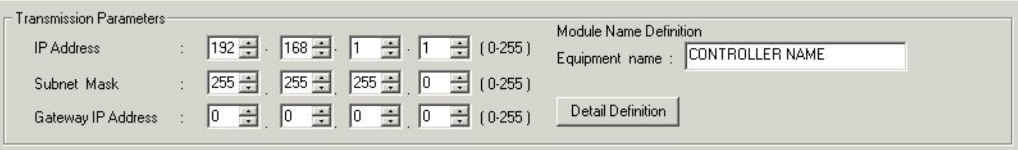
1. Transmission Parameters Tab

The **Transmission Parameters** Tab sets 218IFA transmission parameters. The setting details are as follows:

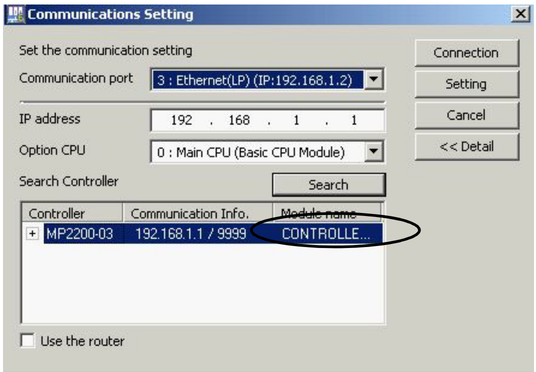


■ Transmission Parameters Setting Items

Sets local transmission parameters for 218IFA.

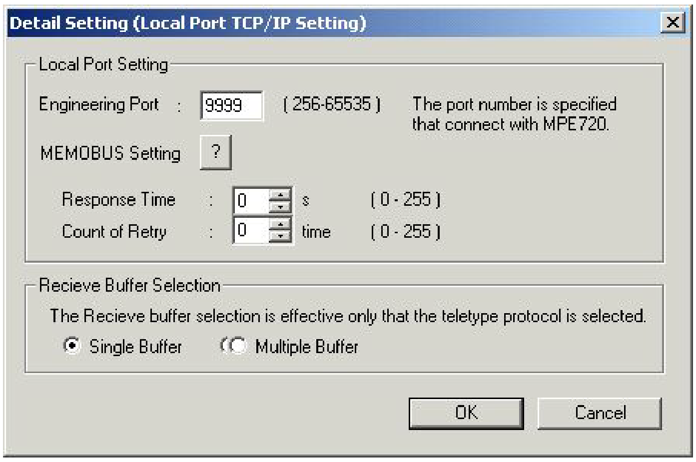


The following table shows each setting item.

Item	Setting Range	Details	Default
IP Address	0.0.0.1 to 255.255.255.254	Sets 218IFA IP address. However, the following addresses are excluded: 127.xxx.xxx.xxx xxx.xxx.xxx.000 xxx.xxx.xxx.255	192.168.001.001
Subnet Mask	0.0.0.0 to 255.255.255.254	Sets the 218IFA subnet mask.	255.255.255.000
Gateway IP Address (router IP address)	0.0.0.0 to 255.255.255.254	Sets the 218IFA default gateway IP address. However, the following addresses are excluded: 127.xxx.xxx.xxx xxx.xxx.xxx.000 (except 000.000.000.000) xxx.xxx.xxx.255 When you do not use it, set it to 000.000.000.000.	000.000.000.000
Equipment Name	Up to 16 single-byte characters	218IFA can be any name. The name specified here is displayed as a search result in the **Module name** field of **Search Controller** list when running the **Search** in the **Communications Setting** Dialog Box of MPE720 Ver.6. 	CONTROLLER NAME
Detailed Definition	–	Opens the dialog box for setting the engineering communication with MPE720 and the MEMOBUS communication.	–

■ Detail Setting Dialog Box of Transmission Parameters Setting

Sets the engineering communication with MPE720 and the message communication.

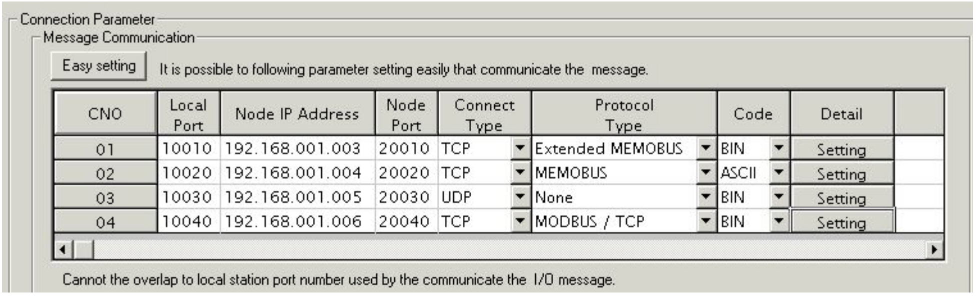


The following table shows each setting item.

Item	Setting Range	Details	Default
Engineering Port	256 to 65535	Specify the 218IFA port number used in the engineering communication with MPE720. Note: When changing this setting, you must also change the engineering port value in the logical port setting detailed window of the MPE720 communication process. The port number cannot be 9998 or 10000.	9999
Response Time	0 to 255 (sec)	Specify the wait time until a remote response is returned after sending a command, when carrying out a message communication using MSG-SND function. (value zero waits infinitely.) If the **Count of Retry** is zero, set **Response Time** to zero. Note: If no response is returned after the setting period expires, a time-out occurs, retry the transmission the number of times specified by **Count of Retry**.	0
Count of Retry	0 to 255 (time)	Specify the command retransmit number of times when a timeout is detected after setting period of **Response Time** expires. Note: If no response is returned after as many retries as the retransmit number of times, an error is returned to the MSG-SND function.	0

■ Message Communication Item of Connection Parameter Setting

Sets the connection parameters for the message communication using MSG-SND/MSG-RCV function and the message communication using automatic receive function.



The following table shows each setting item.

<table>
<tr><th>Item</th><th>Setting Range</th><th>Details</th><th>Default</th></tr>
<tr><td>Easy Setting</td><td>–</td><td>Opens the Message Communication Easy Setting Dialog Box for the connection parameters. The content of the selected connection is displayed.</td><td>–</td></tr>
<tr><td>CNO (Connection Number)</td><td>1 to 4</td><td>In 218IFA Ethernet communication, remote stations are distinguished by their connection numbers.
This connection number is used in remote connection number (PARAM02) of the parameter list (PARAM) of the MSG-SND/ MSG-RCV function.</td><td>–</td></tr>
<tr><td>Local Port</td><td>256 to 65535</td><td>Specify the 218IFA port number for each connection. 218IFA establishes a message communication with the connection with this port number only. Set an unique channel number for the port number of this connections.
To delete the port number, enter zero.
Note: When the connection type = UDP, the port number cannot be 9998 or 10000.</td><td>0</td></tr>
<tr><td>Node IP Address</td><td>0.0.0.0 to 255.255.255.254</td><td>Set the remote IP address for each connection. However, the following addresses are excluded:
127.xxx.xxx.xxx
xxx.xxx.xxx.000 (except 000.000.000.000)
xxx.xxx.xxx.255
Note: When 0.0.0.0 is set, it will enter into “Unpassive open mode.” When 218IFA is within the network specified by the subnet mask, it responds to the connection request from the remote station regardless of the remote IP address setting.</td><td>000.000.000.000</td></tr>
<tr><td>Node Port</td><td>0 and
256 to 65535</td><td>Specify the remote port number for each connection. A pair of remote IP address and remote port number must not be duplicated.
Note: In case of “Unpassive open mode,” set it to zero.</td><td>0</td></tr>
<tr><td>Connect Type</td><td>TCP,
UDP</td><td>Select a transport layer protocol.
TCP: Transmission control protocol
UDP: User datagram protocol</td><td>TCP</td></tr>
<tr><td>Protocol Type</td><td>Extended MEMOBUS,
MEMOBUS,
MELSEC,
None,
MODBUS/TCP</td><td>Select an application layer protocol.
<table>
<tr><th>Protocol Type</th><th>Overview</th></tr>
<tr><td>Extended MEMOBUS</td><td>Yaskawa’s Extended MEMOBUS protocol.</td></tr>
<tr><td>MEMOBUS</td><td>Yaskawa’s MEMOBUS protocol.</td></tr>
<tr><td>MELSEC</td><td>Ethernet I/F protocol for the sequencer (A series) manufactured by Mitsubishi Electric Corporation.</td></tr>
<tr><td>Non-procedure</td><td>General-purpose message communication. Transmits and receives continuous data intact in the specified register.</td></tr>
<tr><td>MODBUS/TCP</td><td>Industrial Ethernet protocol proposed by Modicon, Inc.</td></tr>
</table></td><td>Extended MEMOBUS</td></tr>
</table>

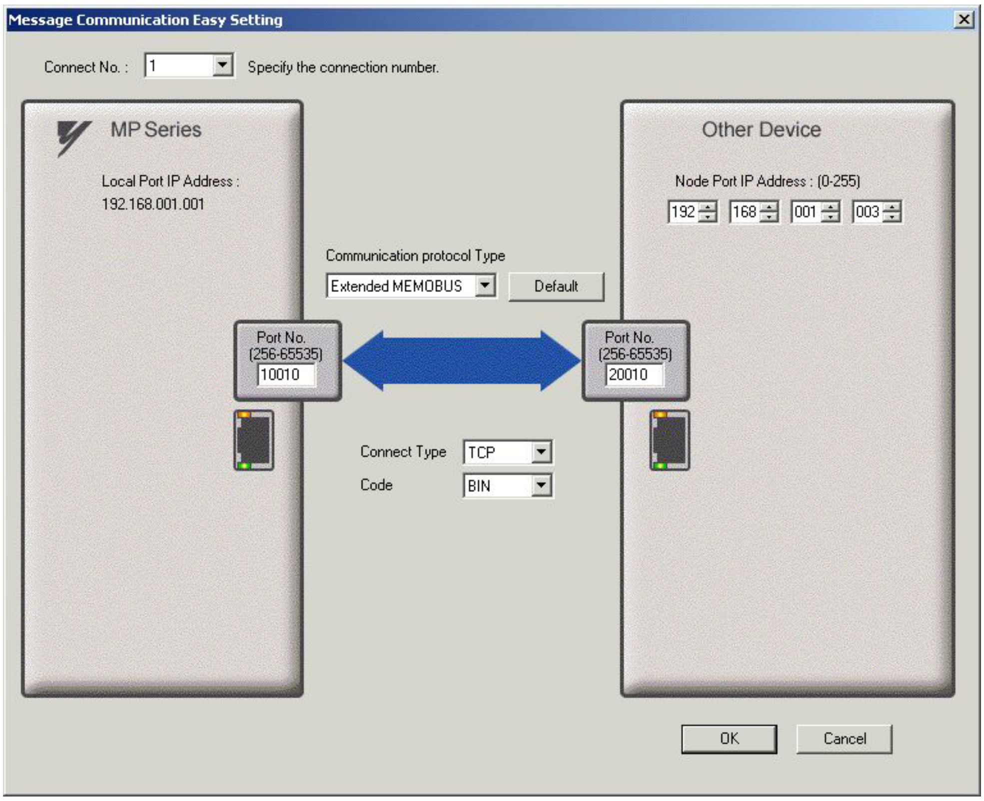
(cont'd)

<table>
<tr><th>Item</th><th>Setting Range</th><th>Details</th><th>Default</th></tr>
<tr><td>Code</td><td>ASCII
BIN
RTU</td><td>Select a code type for the message communication data.
Depending on protocol type, available codes are restricted as follows:
<table>
<tr><th rowspan="2">Protocol Type</th><th colspan="3">Code</th></tr>
<tr><th>ASCII</th><th>BIN</th><th>RTU</th></tr>
<tr><td>Extended MEMOBUS</td><td>√</td><td>√</td><td>–</td></tr>
<tr><td>MEMOBUS</td><td>√</td><td>–</td><td>√</td></tr>
<tr><td>MELSEC</td><td>√</td><td>√</td><td>–</td></tr>
<tr><td>Non-procedure</td><td>√</td><td>√</td><td>–</td></tr>
<tr><td>MODBUS/TCP</td><td>–</td><td>√</td><td>–</td></tr>
</table>
√ : Available, – : Not available</td><td>ASCII</td></tr>
<tr><td>Detail (Automatically Receive)</td><td>–</td><td>Opens the Detail setting Dialog Box. To open the dialog box, double-click this button.
Note: The automatic receive function is valid only for a connection when the connection number = 1.</td><td>–</td></tr>
<tr><td>Remote Station Name</td><td>Up to 32 single-byte characters (16 double-byte characters)</td><td>Any text can be entered as a connection comment.</td><td>Blank</td></tr>
</table>

Note: For the 218IFC of the CPU-04, the setting range for the connection number is larger, i.e., 1 to 20.

■ Message Communication Easy Setting Dialog Box

Graphically sets connection parameters for each connection.
Basically, the same settings as those used for **Message Communication** in the **Connection Parameter** are used.
If this dialog box is opened before the items in **Communication Parameter** have been set, the default value for each connection will be automatically stored.



The following table provides the default values for each connection stored when the items in **Communication Parameter** are not yet set and this dialog box is opened.

Item	Default			
	Connection Number 01	Connection Number 02	Connection Number 03	Connection Number 04
Local Port Number	10001	10002	10003	10004
Node IP Address	192.168.1.2	192.168.1.3	192.168.1.4	192.168.1.5
Node Port Number	10001	10002	10003	10004
Communication Protocol Type	Extended MEMOBUS			
Connect Type	TCP			
Code	BIN			

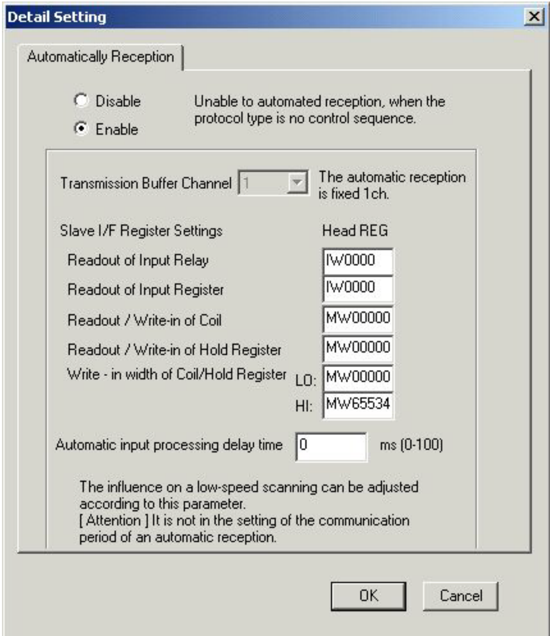
Note: For the 218IFC of the CPU-04, the setting range for the connection number is larger, i.e., 1 to 20.

By clicking the **Default** Button, default values are set for each data code type according to the selected communication protocol type.
The following table shows the default values for **Code** (each data code type).

Communication Protocol Type	Default for Data Code Type
Extended MEMOBUS	BIN
MEMOBUS	RTU
MELSEC	BIN
Non-procedure	BIN
MODBUS/TCP	BIN

■ Automatically Reception Dialog Box for Message Communication

The automatic receive function can be enabled only for connections where the connection number = 1.
The automatic receive function enables the system to automatically run a function equivalent to the MSG-RCV function.



The following table explains each setting item.

Item	Setting Range	Details	Default
Automatically Reception Enable/Disable	Enable/Disable	Select whether to enable automatic reception. Note: When the local port number is not yet set, it becomes invalid regardless of the enable/disable selection.	Enable
The setting items below can only be set when the Automatically Reception is set to “Enable.”			
Transmission Buffer Channel	Cannot be set (fixed to 1)	The communication buffer channel is usually used for data exchanged between the MSG-SND/MSG-RCV function and 218IFA. The communication buffer channel is associated with the connection according to the input item “CH-NO” for the MSG-SND/MSG-RCV function and node connection number (PARAM02) setting for the parameter list (PARAM). When automatic reception is running, the function equivalent to the MSG-RCV function is realized by using the communication buffer channel number “1.”	1
Readout of Input Relay	IW0000 to IWFFFF	Set a start register of the input relay used for the automatic reception.	IW0000
Readout of Input Register	IW0000 to IWFFFF	Set a start register of the input register used for the automatic reception.	IW0000
Readout/Write-in of Coil	MW00000 to MW65534	Set a start read/write register of the coil used for the automatic reception.	MW00000
Readout/Write-in of Hold Register	MW00000 to MW65534	Set a start read/write register of the holding register used for automatic reception.	MW00000
Write-in Width of Coil/Hold Register (LO)	MW00000 to MW65534	Set a write range (LO) of the coil/holding registers used for automatic reception.	MW00000
Write-in Width of Coil/Hold Register (HI)	MW00000 to MW65534	Set a write range (HI) of the coil/holding registers used for the automatic reception.	MW65534

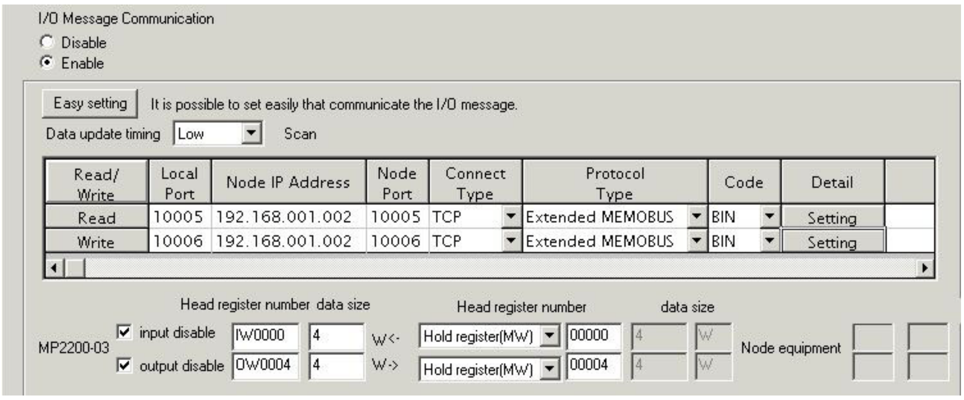
The following table provides the valid setting items for each communication protocol type.

Setting Item	Communication Protocol Type				
	Extended MEMOBUS	MEMOBUS	MELSEC	Non-proce-dure	MODBUS/TCP
Readout of Input Relay	√	√	–	–	√
Readout of Input Register	√	√	–	–	√
Readout/Write-in of Coil	√	√	–	–	√
Readout/Write-in of Hold Register	√	√	√	–	√
Write-in Width of Coil/Hold Register (LO)	√	√	√	–	√
Write-in Width of Coil/Hold Register (HI)	√	√	√	–	√

Note: √ : Enable
– : Disable

■ I/O Message Communication Item Connection Parameter Setting

Sets connection parameters for I/O message communication.
I/O message communication exchanges the data using I/O images with the remote device.



The following table explains each setting item.

Item	Setting Range	Details	Default
I/O Message Communication Enable/Disable	Enable/Disable	Select whether to enable I/O message communications.	Disable
The setting items below can only be set when the I/O Message Communication is set to “Enable.”			
Easy Setting	–	Opens the **Message Communication Easy Setting** Dialog Box for the read/write connection parameters.	–
Data Update Timing	High Scan/ Low Scan	Set when to update the I/O data for the controller side when the I/O message communication is established.	L Scan
Read/Write	–	In 218IFA Ethernet communications, remote stations are distinguished by their connection numbers. I/O message communications have a connection for each read/write.	
Local Port	256 to 65535	Specify the 218IFA port number for each read/write connection. To delete the port number setting, enter zero. To use only a read or a write connection, set the other port number to zero to delete the connection. Note: When the connection type = UDP, the port number cannot be 9998 or 10000.	0
Node IP Address	0.0.0.1 to 255.255.255.254	Set a remote IP address for both read and write connections. Set a common value for both read and write. However, the following addresses cannot be used: 127.xxx.xxx.xxx xxx.xxx.xxx.000 xxx.xxx.xxx.255	000.000.000.000
Node Port	256 to 65535	Specify the remote port number for each read/write connection. A pair of a remote IP address and remote port number must not be duplicated.	0
Connect Type	TCP UDP	Select a transport layer protocol. TCP: Transmission control protocol UP: User datagram protocol	TCP

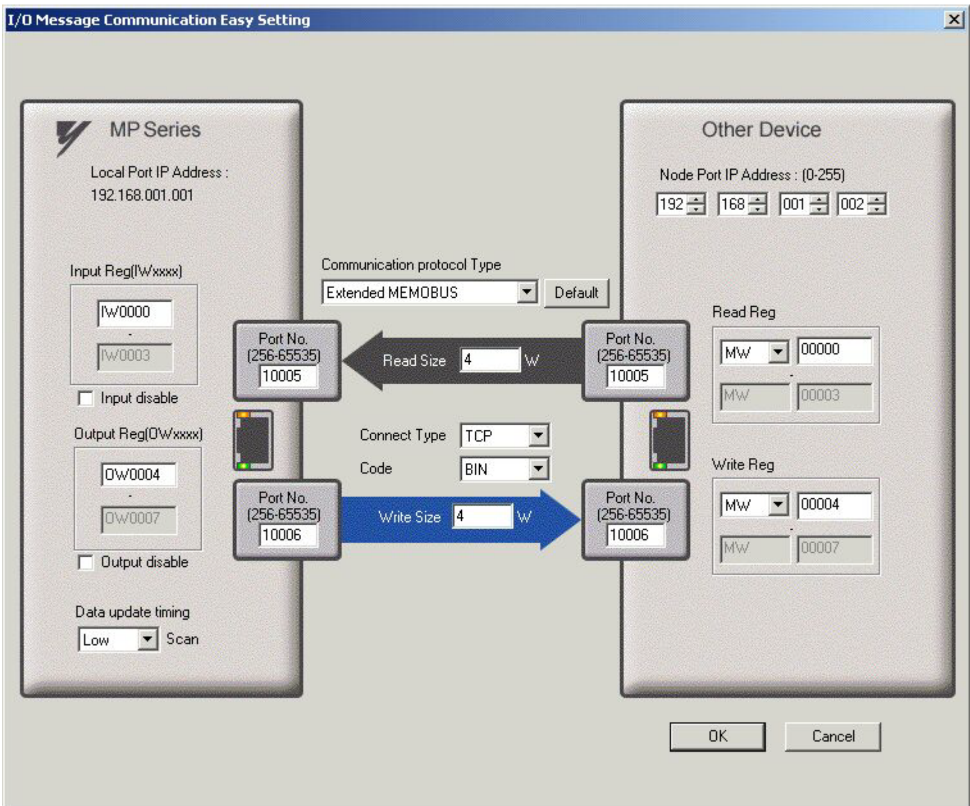
(cont'd)

<table>
<tr><th>Item</th><th>Setting Range</th><th>Details</th><th>Default</th></tr>
<tr><td>Protocol Type</td><td>Extended MEMOBUS
MEMOBUS
MELSEC
MEMOBUS/TCP</td><td>Select an application layer protocol.
<table><tr><th>Protocol Type</th><th>Overview</th></tr><tr><td>Extended MEMOBUS</td><td>Yaskawa's Extended MEMOBUS protocol.</td></tr><tr><td>MEMOBUS</td><td>Yaskawa's MEMOBUS protocol.</td></tr><tr><td>MELSEC</td><td>Ethernet I/F protocol for the sequencer (A series) manufactured by Mitsubishi Electric Corporation.</td></tr><tr><td>MODBUS/TCP</td><td>Industrial Ethernet protocol proposed by Modieon, Inc.</td></tr></table></td><td>Extended MEMOBUS</td></tr>
<tr><td>Code</td><td>ASCII
BIN
RTU</td><td>Select a code type for the message communication data.
Depending on protocol type, available codes are restricted as follows:
<table><tr><th rowspan="2">Protocol Type</th><th colspan="3">Code</th></tr><tr><th>ASCII</th><th>BIN</th><th>RTU</th></tr><tr><td>Extended MEMOBUS</td><td>√</td><td>√</td><td>–</td></tr><tr><td>MEMOBUS</td><td>√</td><td>–</td><td>√</td></tr><tr><td>MELSEC</td><td>√</td><td>√</td><td>–</td></tr><tr><td>MODBUS/TCP</td><td>–</td><td>√</td><td>–</td></tr></table>√ : Available
–: Not available</td><td>ASCII</td></tr>
<tr><td>Detail (Remote Station Name)</td><td>Up to 32 single-byte characters (16 double-byte characters)</td><td>Any text can be entered as a connection comment.</td><td>Blank</td></tr>
<tr><td>Input Disable</td><td>Enable/disable</td><td>Select whether to update the input data in the I/O message communication.</td><td>enable</td></tr>
<tr><td>Output Disable</td><td>Enable/disable</td><td>Select whether to update the output data in the I/O message communication.</td><td>enable</td></tr>
<tr><td rowspan="2">Head Register Number Data Size</td><td>IW0000 to IW7FFF</td><td>Set a start address of the input register of the MP2200 side for storing the data read from the remote device.

Note1: "□□□□" represents a start I/O register number specified by the 218IFA cell in the Module Details field of the Module Configuration Window.</td><td>IW□□□□
(Note1)</td></tr>
<tr><td>OW0000 to OW7FFF</td><td>Set a start address of the MP2200 side output register for referencing the data written in the remote device.
Note2: "□□□□" represents a start I/O register number specified by the 218IFA cell in the Module Details field of the Module Configuration Window.</td><td>OW□□□□
+ 4
(Note2)</td></tr>
<tr><td rowspan="2">Data Size</td><td rowspan="2">Varies according to protocol type</td><td>Specify the data size (word) read from the remote device.</td><td>4</td></tr>
<tr><td>Specify the data size (word) written in the remote device.</td><td>4</td></tr>
<tr><td rowspan="2">Head Register Number for the Node Equipment</td><td rowspan="2">Varies according to protocol type</td><td>Specify the register type and the start register address for the remote device to read.</td><td rowspan="2">Varies according to protocol type.</td></tr>
<tr><td>Specify the register type and the start register address for the remote device to write.</td></tr>
<tr><td rowspan="2">Data Size of the Node Equipment</td><td rowspan="2">Display only</td><td>Generally, the same value specified in MP2200 data size is shown.
By way of exception, when MELSEC is selected for Protocol Type and a bit device such as input relay (X)/ output relay (Y)/ internal relay (M)/ link relay (B) is selected for read register, the data size is displayed in bits.</td><td>4</td></tr>
<tr><td>Generally, the same value specified in MP2200 data size is shown.
By way of exception, when MELSEC is selected for Protocol Type and a bit device such as input relay (X)/ output relay (Y)/ internal relay (M)/ link relay (B) is selected for read register, the data size is displayed in bits.</td><td>4</td></tr>
</table>

■ I/O Message Communication Easy Setting Dialog Box

Graphically adjusts the setting for the read/write connection parameters.
Basically, the same settings as those used for **I/O Message Communication** in the **Connection Parameter** are used. If this dialog box is opened before the items in **Communication Parameter** have been set, the default values for read/write connection will be automatically stored.



The following table provides the default values for each connection stored when the **Connection Parameter** are not yet set and this dialog box is opened.

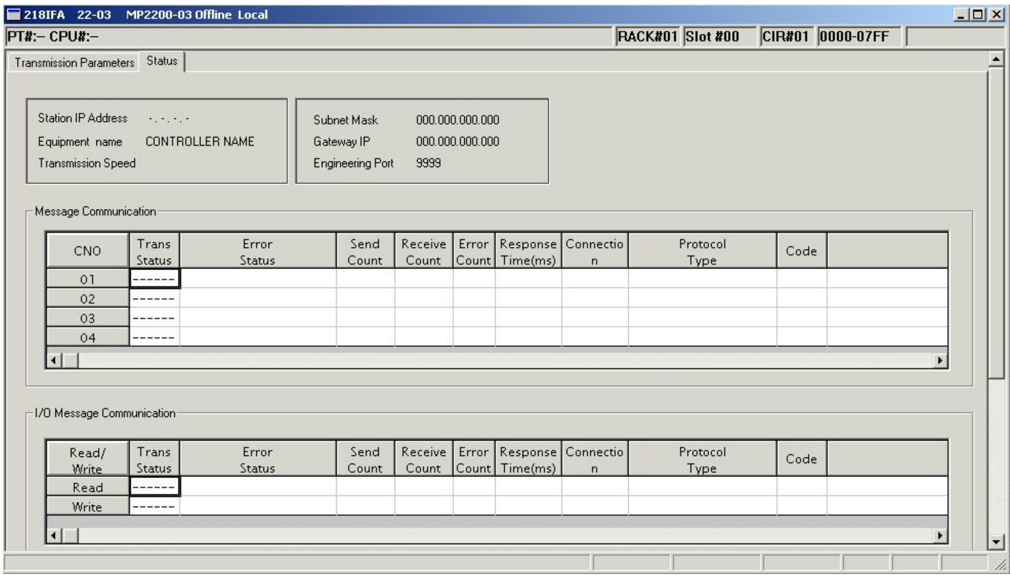
Item			Default
MP Series	Local Port IP Address		Values set in **Transmission Parameter** setting items are shown.
	Local Port Number	Read	10005
		Write	10006
	Input Reg (Register) (IW□□□□)		Start I/O register number specified in the 218IFA cell in the **Module Details** field of the **Module Configuration** Window.
	Input Disable		Not checked (enable)
	Output Reg(Register) (OW□□□□)		Start I/O register number specified in the 218IFA cell in the **Module Details** field of the **Module Configuration** Window + 4.
	Data Update Timing		Low
Other Device	Node Port IP Address		192.168.1.2
	Node Port Number	Read	10005
		Write	10006
	Read Reg (Register)		MW00000
	Write Reg (Register)		MW00004
Communication Protocol Type			Extended MEMOBUS
Read Size			4
Write Size			4
Connect Type			TCP
Code			BIN

In addition, click the **Default** Button to set the default values for data code type, local I/O register setting, read/write size, and node read/write register setting according to the selected communication protocol type.
The following table provides these default values.

Communication Protocol Type	Default			
	Data Code Type	Local Input/Output Register Setting	Read/Write Size	Node Read/Write Register Setting
Extended MEMOBUS	BIN	IW□□□□ to IW□□□□+ 3 (input) OW□□□□+ 4 to OW□□□□+7 (output)	4 (read) 4 (write)	MW0000 to MW0003 (read) MW0004 to MW0007 (write)
MEMOBUS	RTU	Same as above	Same as above	Same as above
MELSEC	BIN	Same as above	Same as above	D0000 to D0003 (read) D0004 to D0007 (write)
MODBUS/TCP	BIN	Same as above	Same as above	4X00001 to 4X0004 (read) 4X00005 to 4X0008 (write)

2. Status Tab

In the **Status** Tab, each setting for 218IFA transmission definition and transmission status is shown.
The displayed contents are as follows:



■ Transmission Parameter Item

Item	Displayed Content	Default
Station IP Address	Displays local IP address specified in the **Transmission Parameters** Tab.	000.000.000.000
Equipment Name	Displays equipment name specified in the **Transmission Parameters** Tab. When the equipment name is not yet set, nothing is shown.	NULL
Transmission Speed	Displays transmission rate retrieved from the status information. (Fixed at Automatic)	Automatic
Subnet Mask	Displays a subnet mask set in the **Transmission Parameters** Tab.	000.000.000.000
Gateway IP	Displays a default gateway IP address set in the **Transmission Parameters** Tab.	000.000.000.000
Engineering Port	Displays a port number set in the detailed definition of the **Transmission Parameters** Tab.	9999

■ Message Communication and I/O Message Communication Items

Item	Displayed Content	Default
Trans (Transmission) Status	Displays the transmission status for each connection.	–
Error Status	If an error is indicated in the transmission status, the error details are shown.	–
Send Count	Displays the number of packets transmitted to the remote station.	–
Receive Count	Displays the number of packets received from the remote station.	–
Error Count	Displays the number of errors that occurred in each connection.	–
Response Time (ms)	Displays the time taken to receive a response after issuing a command in the MSG-SND function of the message communication and the I/O message communication.	–
Connection	Displays the connection type set in the **Transmission Parameters** Tab.	–
Protocol Type	Displays the protocol of the connection parameter set in the **Transmission Parameters** Tab.	–
Code	Displays the code type of the data set in the **Transmission Parameters** Tab.	–
Detail (Node Station Name)	Displays the remote station name set in the **Transmission Parameters** Tab.	–

Note: 1. Trans (Transmission) Status
In online mode, displays the transmission status for each connection.

Transmission Status	State
IDLE	IDLE
WAIT	WAIT (waiting for connection)
CONNECT	CONNECT (capable of transmitting and receiving data)
–	Unused connection

2. Error Status
If an error is indicated in the transmission status, the error details are shown.

Error Status	State	Remarks
No Error	Normal	–
Socket Generation Error	System error	Socket generation failed
Local Port Number Error	Error in setting the local port number (the same address is bound while disconnecting the TCP connection)	Bind error (duplicated port number)
		A bind error occurred while aborting using the MSG function and ending the connection. The error occurs if Execute is turned ON within one minute after an Abort is completed.
		Before the connection was completed, another function issued a command to the same remote station.
Socket Attribute Change Error	System error (in TCP)	An error occurred while setting a socket attribute.
Connection Error (M-SND)	Connection error (when actively open in TCP, a connection is rejected by the node station)	Tried to connect using the MSG-SND function, but the connection was rejected by the remote station, and the command was reset.
		When disconnecting the cable, retried connecting for one minute (default value) without a response.
Connection Error (M-RCV)	Connection error (when passively open in TCP)	An error occurred while receiving the connection from the MSG-RCV function.
System Error	System error	A socket polling (select specification) error occurred while receiving data.
Data Transmit Error (TCP)	Data transmit error (in TCP, either there is no node station or a node station did not startup.)	A response transmit error occurred in the MSG-RCV function. An error also occurred in the MSG-SND function. An error occurred only in TCP when there was no node station to transmit or a node station was rebooted.
Data Transmit Error (UDP)	Data transmit error (in UDP)	A transmit request was issued to a nonexistent socket.
Data Receive Error (TCP)	Data receive error (in TCP, a request to disconnect the connection is received from the node station)	An error occurred when disconnecting the connection from the node station. It also may occur even when close is processed properly.
Data Receive Error (UDP)	Data receive error (in UDP)	A data receive command was issued to a nonexistent socket.
Socket Option Change Error	System error	Error when changing a socket option
Data Change Error	Data change error	Protocol change error

2.5.7 M-EXECUTOR Module (Motion Program Executor)

This section explains the M-EXECUTOR Module (motion program executor) function and its detail window.

(1) M-EXECUTOR Module Function Overview

The M-EXECUTOR Module is a software module that executes a motion or sequence program.
The M-EXECUTOR Module enables the following features:

• Executing a motion program without using a ladder program

Conventionally, in order to execute a motion program, you need to incorporate an MSEE command into a ladder program. The M-EXECUTOR Module allows you to execute the motion program without incorporating the MSEE command into the ladder program.

Note: You can incorporate a MSEE command into the ladder program as ever.

• Controlling a motion program without using a ladder program

You can map any register to the control signal of the motion program registered in the M-EXECUTOR Module.
So, without a ladder program, this allows you to directly control a motion program from a host PLC or other device.

• Describing sequence control in motion language

As a new programming method called sequence programming has been added to the CPU-03 Module in the MP2200 Series.
A sequence program is a scan execution type program where a process is completed with one scan. It employs a text language similar to a motion program.
You can use the sequence program as an alternative to the ladder program.
For information about commands available for sequence programming, refer to the *Machine Controller MP2000 Series Users Manual Motion Programming* (manual no.: SIEP C880700 38).

(2) M-EXECUTOR Module Specification

[a] Programs Capable of Registration in M-EXECUTOR

The following table shows programs capable of registration in M-EXECUTOR.

Program Type		Number of Registrations	Remarks
Motion Program		16*	* Up to 16 programs in total
Sequence Program	Startup	1	
	Interrupt	Disable	
	H Scan	16*	
	L Scan	16*	

[b] Program Control Method

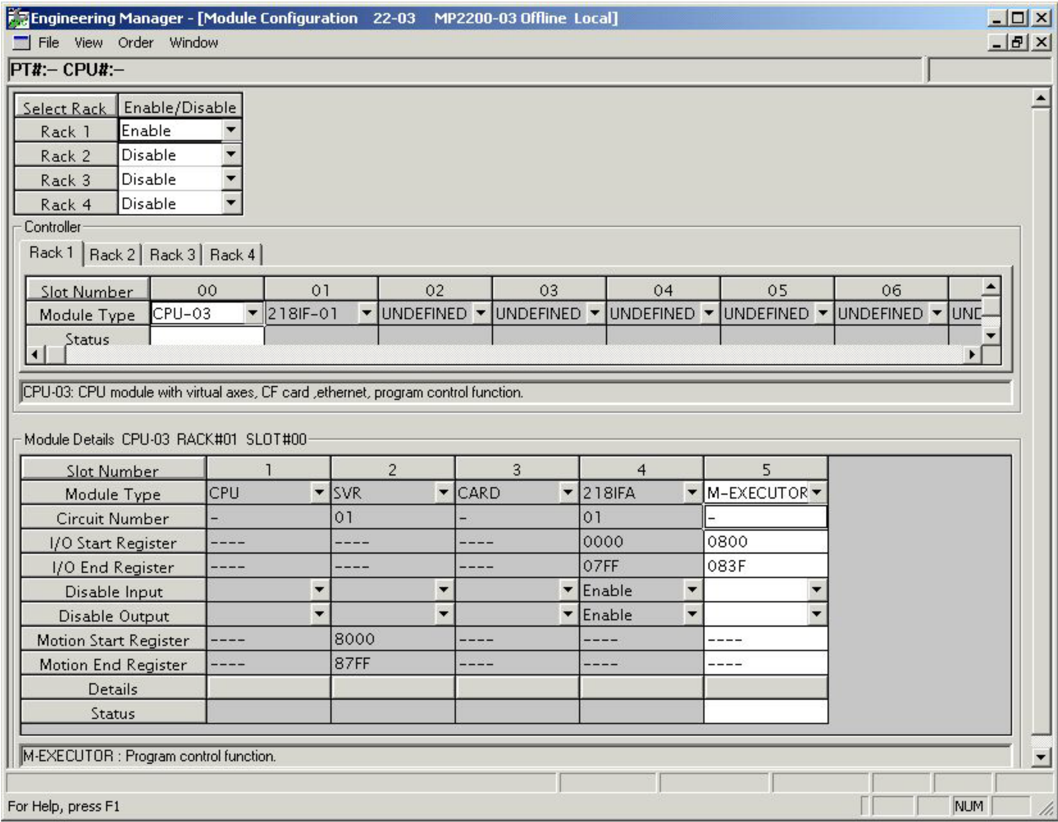
The following table shows the program control methods registered in M-EXECUTOR.

<table>
<tr><th>Item</th><th>Motion Program</th><th>Sequence Program</th></tr>
<tr><td>Execution Method</td><td>Sequential Execution</td><td>Startup: Event driven
H Scan: Scan execution
L Scan: Scan execution</td></tr>
<tr><td>System Work</td><td colspan="2">1:1 correspondence between the definition number and system work
(The number of program definitions is set in the MPE720 window.)
<table>
<tr><th>Definition No.</th><th>System Work Number</th></tr>
<tr><td>No. 1</td><td>1</td></tr>
<tr><td>No. 2</td><td>2</td></tr>
<tr><td>·</td><td>·</td></tr>
<tr><td>No. 16</td><td>16</td></tr>
</table></td></tr>
<tr><td>Program Designation Method</td><td>Direct or indirect designation</td><td>Direct designation</td></tr>
<tr><td>Program Startup Method</td><td>Registers the program in the definition and turns ON the start signal</td><td>Starts up when registered in the definition</td></tr>
<tr><td>Override Setting for Interpolation</td><td>Yes</td><td>No</td></tr>
<tr><td>I/O Link Definition</td><td>Yes</td><td>No</td></tr>
<tr><td>S Register Report Function of Motion Program Status</td><td colspan="2">Yes</td></tr>
<tr><td>Number of Parallels</td><td>1 to 8 (4 main parallels × 2 sub parallels)</td><td>1</td></tr>
<tr><td>Execute an Error Drawing when Operation Error Occurred</td><td colspan="2">Yes</td></tr>
</table>

(3) Module Configuration Definition

[a] Details of Module Configuration Window

Click **CPU-03** in the **Controller** area to display the details of the basic module functions in the **Module Details** area.
The cell No.5 provides a detailed definition of M-EXECUTOR.
For CPU-03, "UNDEFINED" is the default for No.5.



Items displayed in the **Module Details** area show the following:

Item	Description	Change
Slot Number	Sub-slot number. Double-click to open the M-EXECUTOR detailed definition window.	–
Module Type	A module name appears. Changing the name to UNDEFINED enables you to disable M-EXCUTOR functions.	√
Circuit Number	Not used. Fixed to “–”.	–
I/O Start Register	Start register of the M-EXECUTOR I/O register (valid range: 0000-7FFFh, size: 40h words)	√
I/O End Register	End register of the M-EXECUTOR I/O register (valid range: 0000-7FFFh, size: 40h words)	√
Disable Input	Not used. Fixed at “blank”.	–
Disable Output	Not used. Fixed at “blank”.	–
Motion Start Register	Not used. Fixed at “– – – –”.	–
Motion End Register	Not used. Fixed at “– – – –”.	–
Details	Not used.	–
Status	M-EXECUTOR Module status in online mode.	–

√ : Available, – : Not available

■ I/O Register Details

An I/O register assigned to M-EXECUTOR is used to run a motion program and sequence program, and to monitor a sequence program.
M-EXECUTOR I/O register details are as follows:

M-EXECUTOR Input Register

M-EXECUTOR Input Register	Item	
IW□□□□ + 0	Definition No.1	Status
IW□□□□ + 1		Spare
IW□□□□ + 2		Spare
IW□□□□ + 3		Spare
IW□□□□ + 4	Definition No.2	Status
IW□□□□ + 5		Spare
IW□□□□ + 6		Spare
IW□□□□ + 7		Spare
	.	
	.	
	.	
IW□□□□ + 3C	Definition No.16	Status
IW□□□□ + 3D		Spare
IW□□□□ + 3E		Spare
IW□□□□ + 3F		Spare

M-EXECUTOR Output Register

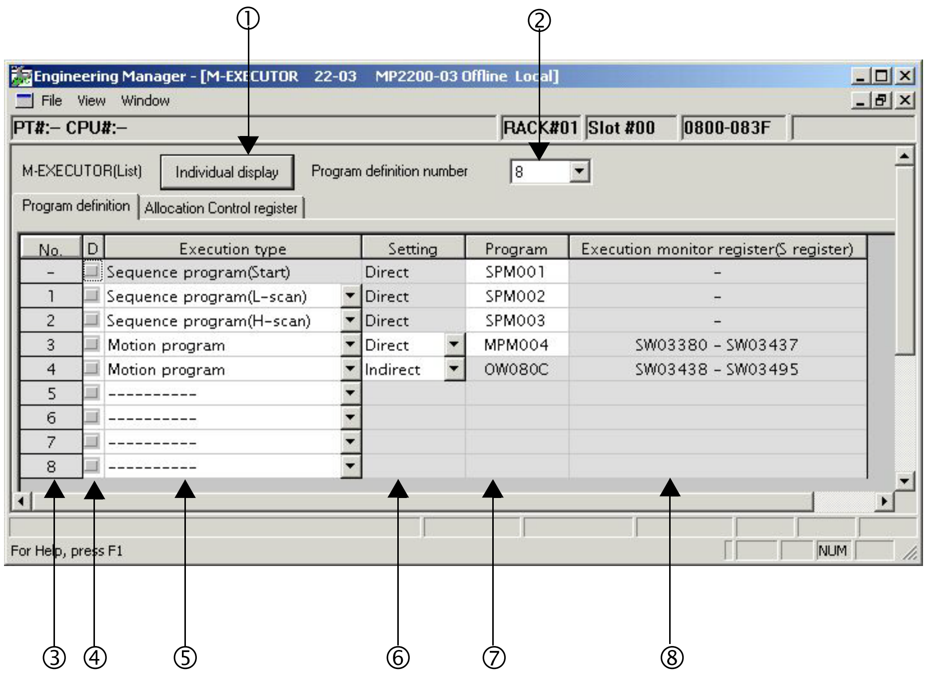
M-EXECUTOR Output Register	Item	
OW□□□□ + 0	Definition No.1	Program number
OW□□□□ + 1		Control signal
OW□□□□ + 2		Override
OW□□□□ + 3		Spare
OW□□□□ + 4	Definition No.2	Program number
OW□□□□ + 5		Control signal
OW□□□□ + 6		Override
OW□□□□ + 7		Spare
	.	
	.	
	.	
OW□□□□ + 3C	Definition No.16	Program number
OW□□□□ + 3D		Control signal
OW□□□□ + 3E		Override
Ow□□□□ + 3F		Spare

(4) Detailed Definition Window

This section describes the M-EXECUTOR detailed definition window.

■ Program Definition Tab (M-EXECUTOR (list) Window)

The **Program definition** Tab allows you to register a motion or sequence program to run.
Programs are executed according to the scan, in ascending numeric order.
A white cell can be set by the user, and a grey cell cannot be set by the user.



① Individual display
Display the **Program execution registry screen** dialog box (M-EXECUTOR (individual display)).

② Program definition number
Sets the number of program definitions to register in the M-EXECUTOR Module.
The valid range is 0-16 (8 by default).

③ No.
Shows the program execution order. Processed according to the scan in ascending numeric order.

④ D
Enables/disables the definition. Uncheck to enable the definition.

⑤ Execution type

Sets the program execution type.

Execution Type	Program to Execute	Execution Conditions
----------	None	None (select this to delete the definition)
Sequence Program (startup)	Sequence program	Power ON (Executed once only, when power turned ON)
Sequence Program (L scan)		Periodical startup (Executed each low-speed scan)
Sequence Program (H scan)		Periodical startup (Executed each high-speed scan)
Motion Program	Motion program	Turns ON the program operation start request of the control signal (runs when the program operation start request is ON).

⑥ Setting

Sets the a program designation.

The way to designate a program may differ according to the program.

Designa-tion Method	Motion Program	Sequence Program	Remarks
Direct	Enable	Enable	The way to designate the program number Example: MPM001, SPM002, and so on
Indirect	Enable	Disable	The way to designate the register for storing the program number Example: OW0C0C, and so on (refers to MPM001 by storing 1 in OW0C0C)

⑦ Program

Sets a program number.

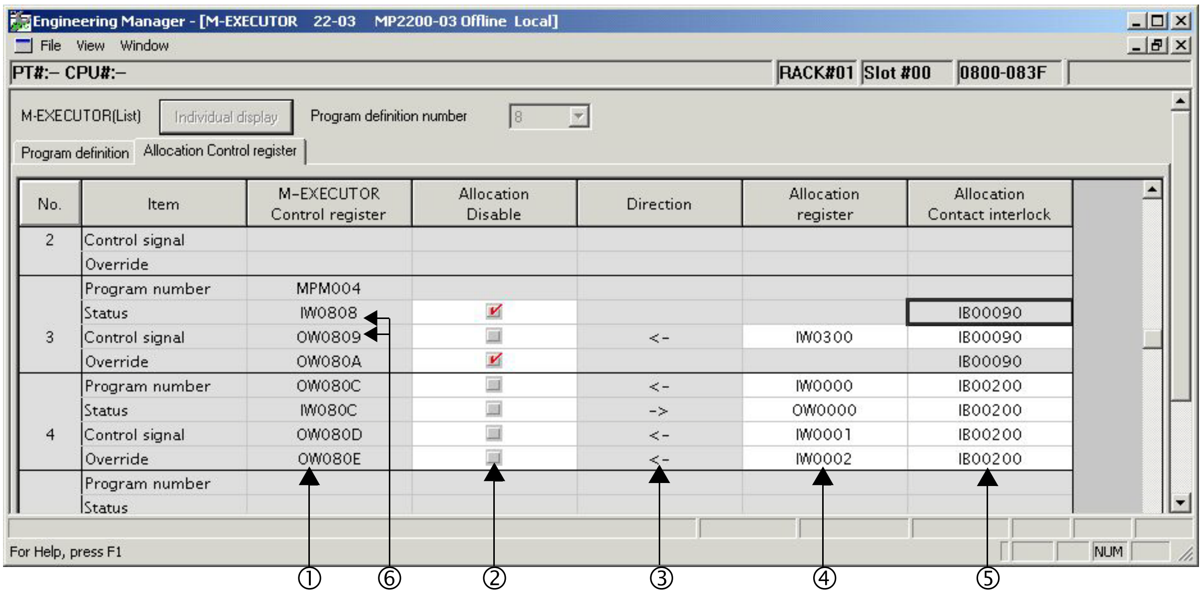
Execution Type	Remarks
Sequence Program (startup, L scan, H scan)	Enter “1” and press ENT to automatically input “SPM001.” You can save an unregistered program or exit this tab page without setting (blank), but in these cases, the program will not be executed.
Motion Program	Direct designation: Enter “1” and press ENT to automatically input “MPM001.” You can save an unregistered program or exit this tab page without setting (blank), but in these cases, the program will not be executed. Indirect designation: O register of M-EXECUTOR Module is automatically set. It cannot be set by the user.

⑧ Execution monitor register (S Register)

When the execution type is set to motion program, the range of the execution monitor registers (S registers) is shown. For more information on the execution monitor register, refer to *5.2.3 (6) Monitor the Motion Program Execution Information using S register*.

■ Allocation Control Register Tab

The **Allocation Control register** Tab Page sets a allocation register.
A white cell can be set by the user, and a shaded cell cannot be set by the user.



① M-EXECUTOR Control register

Displays an I/O register mapped to the M-EXECUTOR Module.
Controls the motion program and monitors the state, using the M-EXECUTOR control register.

M-EXECUTOR Control Register	Usage
Program Number	Sets a program number. This register is used only when set to an indirect designation.
Status	Monitors the program execution status.
Control Signal	Controls the program.
Override	Sets an override value when running a move command for the interpolation system.

Note: For details on the M-EXECUTOR control register, refer to items (4) and (5) under *5.2.3 Motion Programs*.

② Allocation Disable

Enables/disables the allocation register. Uncheck to enable the definition.

③ Direction

Displays the data I/O direction.

④ Allocation register

Data is exchanged between mapping and M-EXECUTOR control registers in real-time.
Any register can be mapped to the allocation register.

Registers that can be set as a Allocation Register
Word type I, O, M (except the motion register)

⑤ Allocation Contact interlock

An allocation contact interlock is used to control the data exchange between the allocation register and M-EXECUTOR control registers. When the allocation contact interlock is ON, data can be exchanged between the allocation register and M-EXECUTOR control registers.
Any register bit can be mapped to the allocation contact interlock.

Registers that can be set as an Allocation Contact Interlock
Bit type I, O, S, M, C (except the motion register)

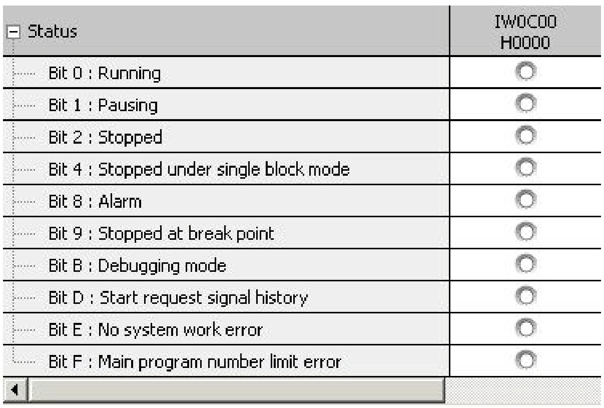
■ Caution

An allocation contact interlock is used to interlock the operation of a motion program.
When setting an allocation register, be sure to set the allocation contact interlock.

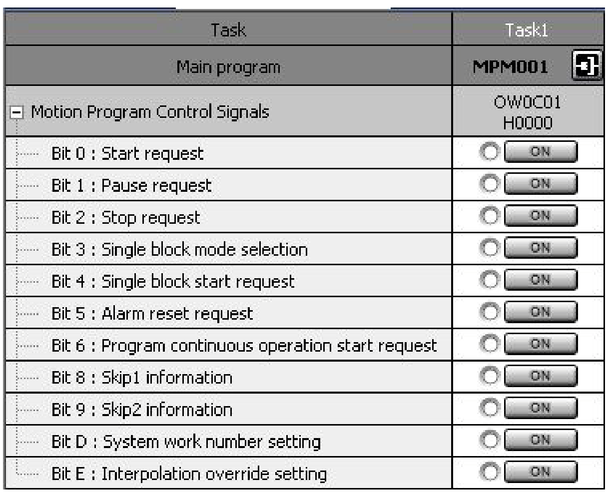
⑥ Status, Control Signal Details

Double-click the status and control register to display the bit detail.
You can check the signal sequence and status here.

- Status

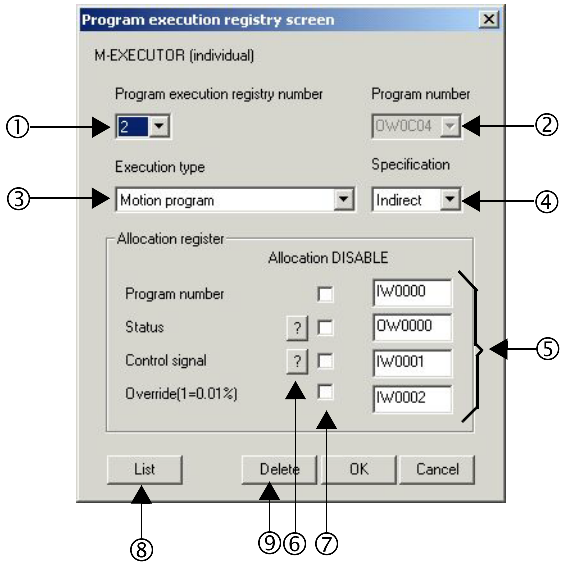


- Control Signal



■ Program execution registry screen Dialog Box (M-EXECUTOR (individual) Dialog Box)

Click the **Individual Display** Button in the M-EXECUTOR (list) window to display this dialog box.
The items that can be set are similar to those in the **Program Definition** Tab Page and the **Allocation Control Register** Tab Page.



① Program execution registry number
Selects a program execution registration No.

② Program number
Sets a program number.

③ Execution type
Sets the program execution type.

④ Specification
Sets the method of designating a program.

⑤ Allocation register
Sets an allocation register.

⑥ Status, Control signal
Displays the status and the signal sequence of the control register.

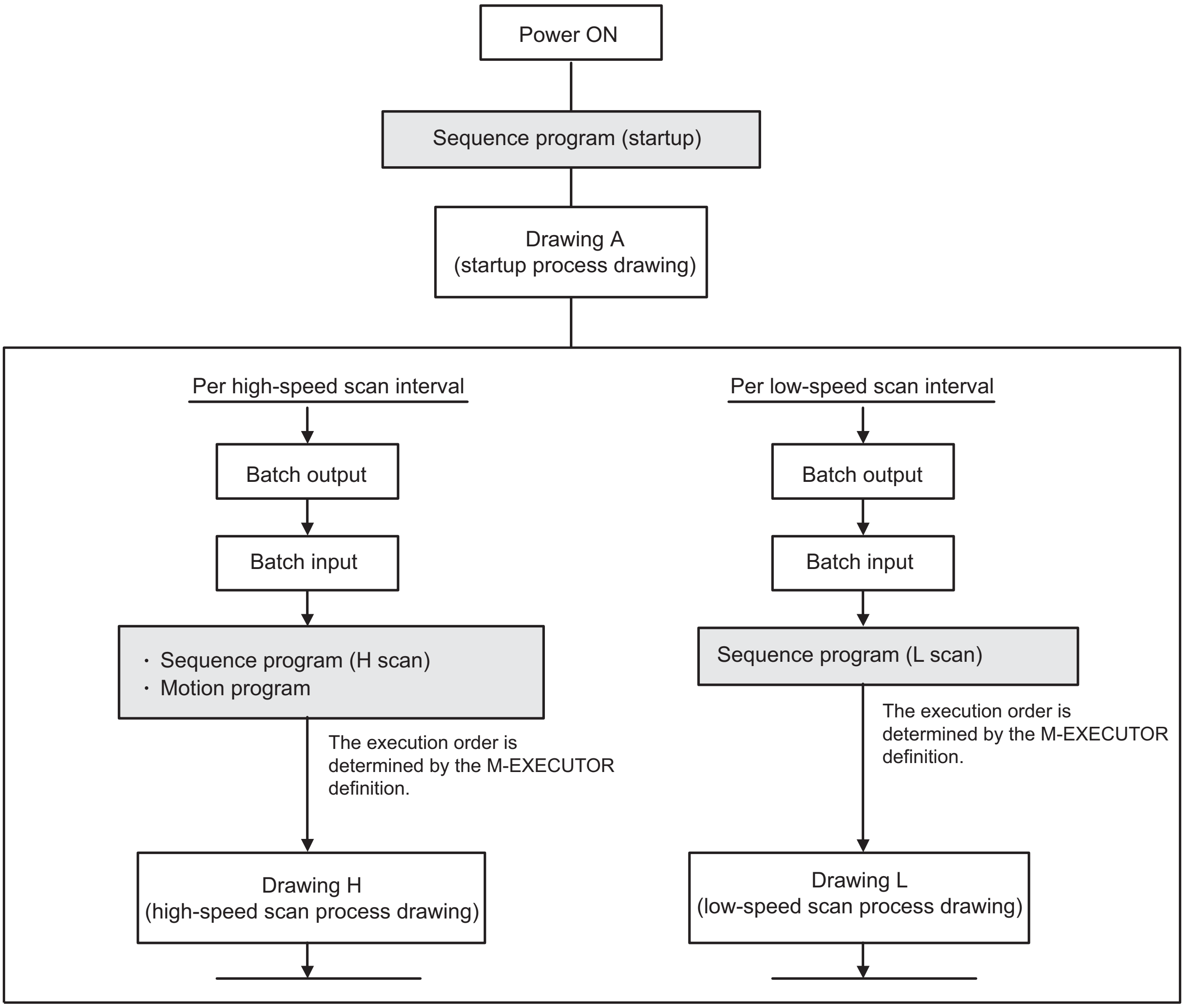
⑦ Allocation DISABLE
Enables/disables the allocation register. Uncheck to enable the definition.

⑧ List
Displays the M-EXECUTOR (list) window.

⑨ Delete
Deletes a definition.

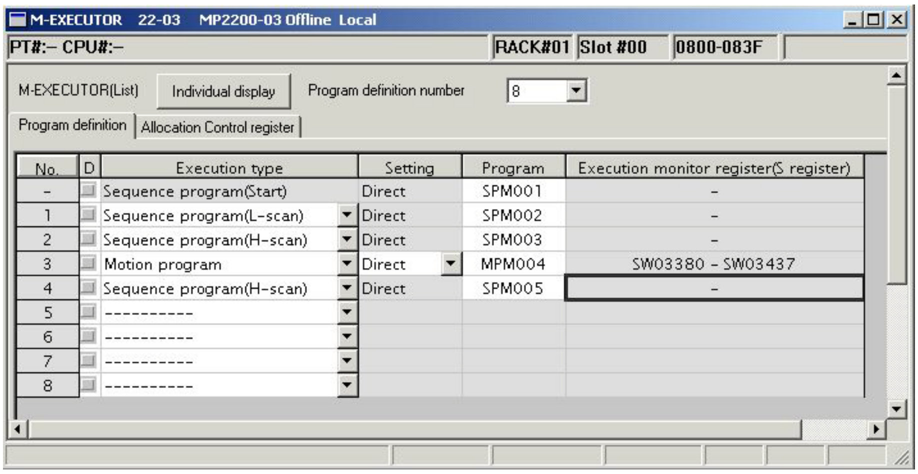
(5) Execution Scheduling

Programs registered in M-EXECUTOR are executed on the basis of their priorities (execution type).
Programs registered in M-EXECUTOR are executed just before the ladder process.



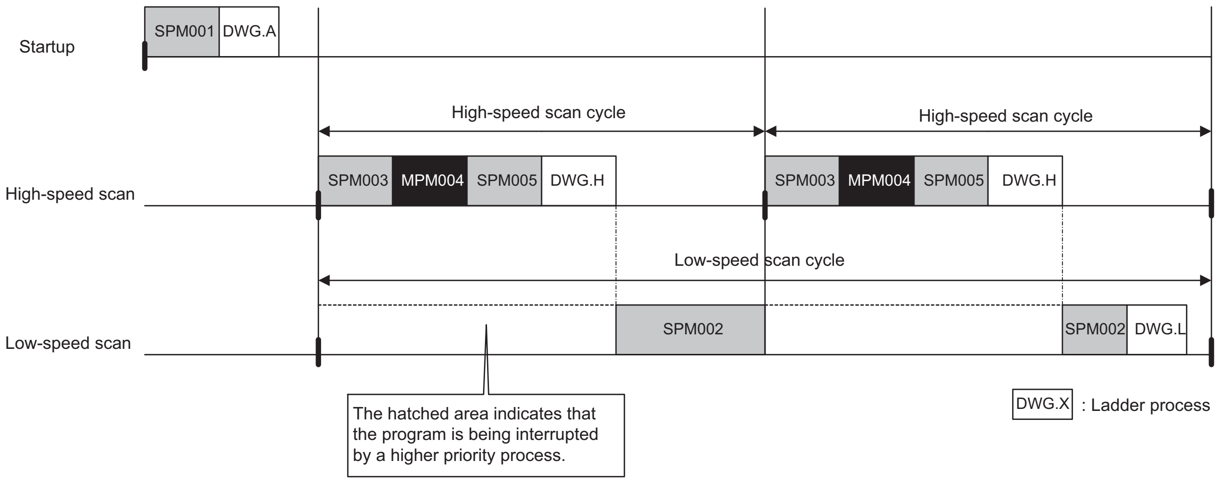
An execution example is as follows:

- M-EXECUTOR program definition



- Execution scheduling

  The following diagram shows the execution scheduling when set in the tab page above.



2.6 CPU-04 Module

2.6.1 Overview of Functions

The CPU-04 Module is a special CPU Module for the MP2200. It was designed as a upper-end model to provide greater speed and easier operation than the CPU-03 Module.
The CPU-04 Module has one Ethernet interface with the following characteristics.

• Standard-feature Ethernet (100 Mbps)

- High-speed communication can be performed with the MPE720 Engineering Tool.
- Communication can be performed with a touch panel without using ladder programming (automatic reception).
- Communication can be performed with the host PLC without using ladder programming (I/O message communication).

• Simple Programming

- The operating procedures have been greatly streamlined through execution of motion operations.
- The motion program can be started from the host PLC without programming simply by creating a motion program and registering the execution sequence.

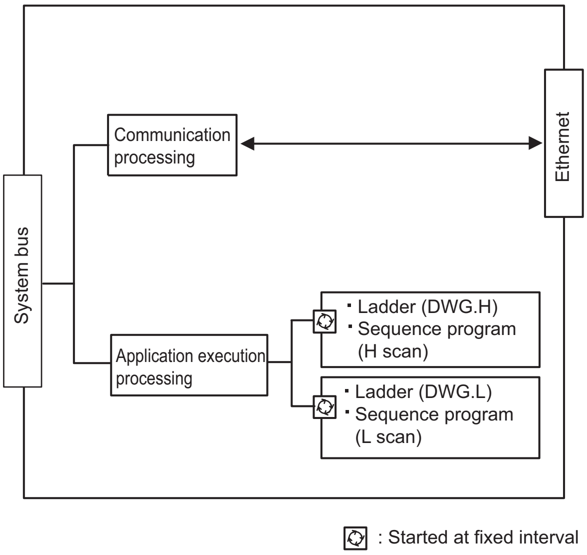
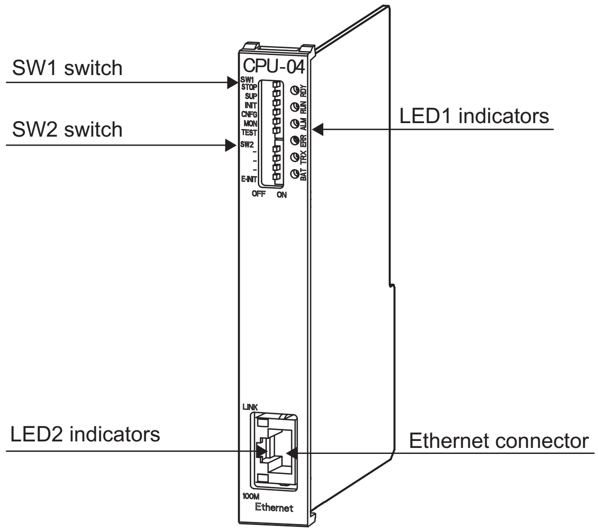


Fig. 2.2 Overview of CPU-04 Module Functions

2.6.2 External Appearance, Indicators, and Switch Settings

(1) External Appearance

The following figure shows the external appearance of the CPU-04 Module.



(2) Indicators

The following table shows the indicators that show the operating status of the CPU-04 Module and error information.

Indicators	Indicator Name	Color	Significance when Lit
LED1 RDY RUN ALM ERR TRX BAT	RDY	Green	Unit operating normally. *
	RUN	Green	User program running. *
	ALM	Red	Lights/blinks for warning. *
	ERR	Red	Lights/blinks for errors.*
	TRX	Green	Transmitting or sending Ethernet data.
	BAT	Red	Battery alarm activated.*
LED2 LINK 100M (Built into the Ethernet connector.)	LINK	Yellow	When there is an Ethernet link.
	100M	Green	Lit: Connected at 100 Mbps or performing auto-negotiation. Not lit: 10 Mbps

* Refer to *8.2.3 (2) LED Indicator Meanings* on the meaning of indicators.

Overview of Specifications and Functions

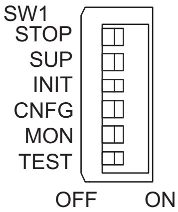
2

(3) Switch Settings

[a] SW1

The SW1 of the DIP switch is used to set the operating conditions for the CPU-04 Module when the power supply is turned ON.

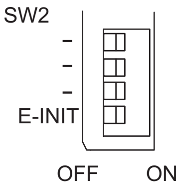
Before turning ON the power supply, set this switch. Any settings made after the power supply is turned ON are invalid.



Pin Number	Switch Name	Status	Operating Mode	Default Setting	Details
6	STOP	ON	User program stopped	OFF	Stops the user program execution. Enabled only when the power is turned ON.
		OFF	User program running		
5	SUP	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
4	INIT	ON	Memory clear	OFF	Set to ON to clear the memory. Also set to ON not to save the data in the Compact Flash. If this switch is set to OFF, the program stored in flash memory will be executed.
		OFF	Normal operation		
3	CNFG	ON	Configuration mode	OFF	Set to ON to execute self-configuration for connected devices.
		OFF	Normal operation		
2	MON	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
1	TEST	ON	System use (adjusted before shipment)	OFF	Always set to OFF.
		OFF	Normal operation		

[b] SW2

The SW2 of the DIP switch is used to set the Ethernet operating conditions.



Pin Number	Switch Name	Status	Operating Mode	Default Setting	Details
4	---	ON	System use	OFF	Always set to OFF.
		OFF			
3	---	ON	System use	OFF	Always set to OFF.
		OFF			
2	---	ON	System use	OFF	Always set to OFF.
		OFF			
1	E-INIT	ON	Transmission parameter defaults for Ethernet	OFF	If E-INIT is ON, the IP address and other transmission parameters will be set to the default values at startup.
		OFF	Startup according to the Module Configuration Definitions for Ethernet.		

2.6.3 Hardware Specifications

The following table shows the hardware specification of the CPU-04 Module.

Item	Specifications	
Name	CPU-04 Module	
Model	JAPMC-CP2230-E	
Abbreviation	CPU-04	
Flash Memory	16 MB (User's area: 11.5 MB)	
DDR	64 MB	
SRAM	4 MB (3 MB max. with battery backup can be used for table data.)	
Communication Interface	One Ethernet interface channel	
Calendar	Seconds to year timer (battery backup)	
Indicators	LED1 RDY (green) RUN (green) ALM (red) ERR (red) TRX (green) BAT (red)	LED2 LINK (yellow) 100M (green)
Switches	SW1 STOP SUP INIT CNFG MON TEST	SW2 E-INIT
Dimensions (mm)	125 × 95 (H × D)	
Mass	105 g	

Note: For information on general specifications, refer to *2.1 Specifications*.

2.6.4 SVR Module (Virtual Motion Module)

Refer to *2.3.4 SVR Module (Virtual Motion Module)*.

2.6.5 218IFC Module (Ethernet)

(1) Overview of 218IFC Module Functions

The 218IFC Module is a 10Base-T/100Base-TX Ethernet interface. It is a standard-feature communication interface for the CPU-04 Module.

- 100Mbps transmission speed is supported (100Base-TX).
- Supports the following various communication protocols:
  - Support for MEMOBUS protocol, Extended MEMOBUS protocol
  - Support for MELSEC protocol (A-compatible 1E frames, QnA-compatible 3E frames)
  - Support for MODBUS/TCP protocol
  - Support for OMRON protocol
  - Support for non-procedure communication
- An I/O message communication function enables you the data exchange in the form of I/O image when communicating with the host PLC, eliminating you from creating a ladder program.
- An automatic receive function eliminates you from creating a ladder program when connected to the indicator and the like.
- The 218IFC Module can be used as the standard interface with the MPE720 engineering tool. It provides a simple function for connecting to the MPE720, allowing you to connect to MPE720 without knowing the IP address of the 218IFC Module.

(2) Specification of 218IFC Module

The following table shows the specification of the 218IFC Module.

Items		Description
Communication Interface *1		10Base-T/100Base-TX
Communication Protocol *2		TCP/UDP/IP/ARP/ICMP
Maximum Number of Communication Connections		20+2 (I/O Message communication)
Maximum Number of Communication Channels		10+2 (I/O Message communication)
Message Communication (maximum)	MEMOBUS	Write: 100W Read: 125W
	Extended MEMOBUS	Write: 2043W Read: 2044W
	MELSEC (A-compatible 1E)	Write: 1017W Read: 1017W
	MELSEC (QnA-compatible 3E)	Write: 960W Read: 960W
	MODBUS/TCP	Write: 100W Read: 125W
	OMRON	Write: 996W Read: 999W
	Non-procedure	Write: 2046W
I/O Message Communication (maximum)	MEMOBUS	Write: 100W Read: 125W
	Extended MEMOBUS	Write: 1024W Read: 1024W
	MELSEC (A-compatible 1E, QnA-compatible 3E)	Write: 256W Read: 256W
	MODBUS/TCP	Write: 100W Read: 125W
	OMRON	Write: 996W Read: 999W
Automatic Receive	MEMOBUS	Provided
	Extended MEMOBUS	Provided
	MELSEC (A-compatible 1E, QnA-compatible 3E)	Provided
	MODBUS/TCP	Provided
Non-procedure receive buffer mode selection *3		Provided
Simple Function for Connecting with Engineering Tool		Provided

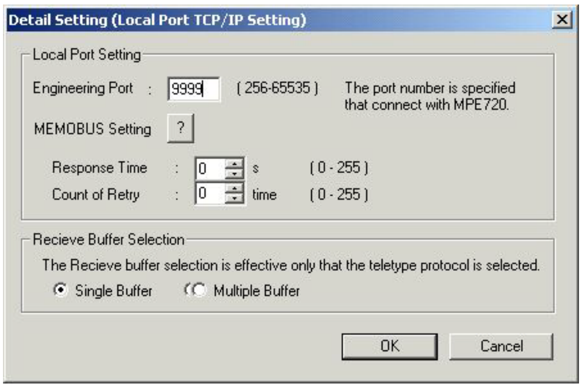
* 1. Communication Interface
The discrimination between 10Base-T/100Base-TX and full-duplex/half-duplex is done by 218IFA based on the remote device. When connecting to a device without automatic negotiation function, set the remote device to half-duplex mode.

Correspondence of Communication Mode

218IFC Module	Device to be connected				
	Automatic Negotiation	10Base-T Half-duplex	10Base-T Full-duplex	100Base-TX Half-duplex	100Base-TX Full-duplex
Automatic Negotiation	Depends on the remote device	Communicates in 10Base-T half-duplex mode	Unable to communicate	Communicates in 100Base-TX half-duplex mode	Unable to communicate

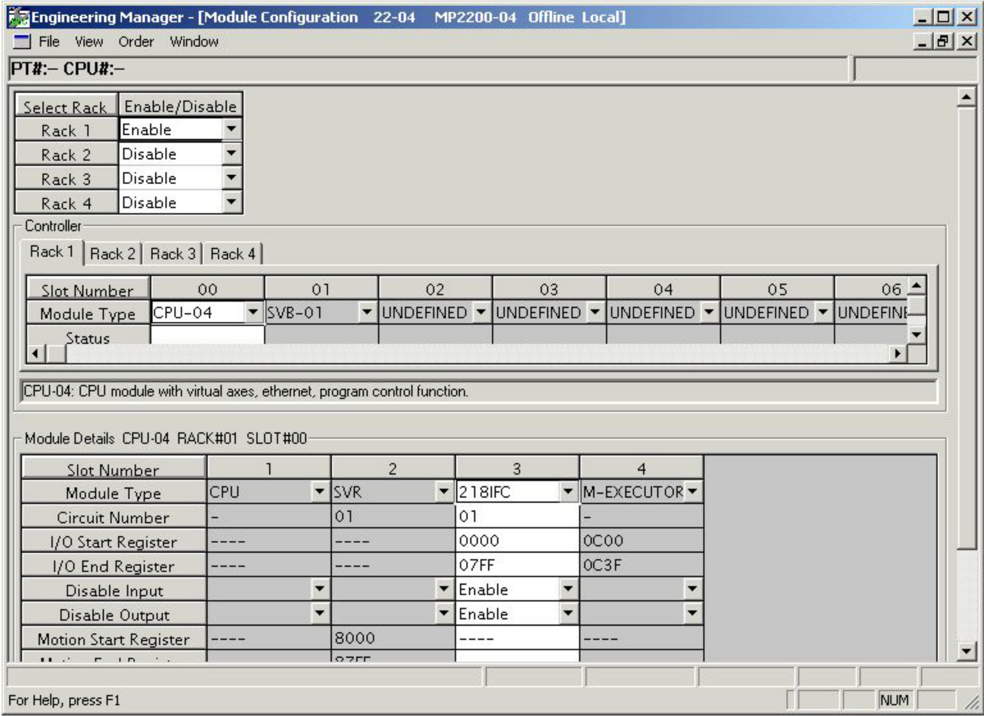
* 2. Communication protocols
- TCP(Transmission Control Protocol): Connection-oriented transport layer protocol
- UDP(User Datagram Protocol): Connectionless transport layer protocol
- IP(Internet Protocol): Protocol for establishing a communication link between computers
- ICMP(Internet Control Message Protocol): Error control protocol for IP protocol
- ARP(Address Resolution Protocol): Address resolving protocol. Protocol for converting IP address into MAC address

* 3. When the non-procedure application protocol of the 218IFC is used, either a single buffer or multiple buffers can be selected for the receive buffers in the 218IFC.
If multiple buffers are selected, 20 data items per connection can be processed at the same time with data continuously received at an interval shorter than the startup interval of the MSG-RCV function.



(3) Module Configuration Definition

Refer to *2.5.6 (3) Module Configuration Definition*.
The following window shows the module configuration for the CPU-04.



(4) 218IFC Module Detailed Definition Window

Refer to *2.5.6 (2) Specification of 218IFA Module*.
For the 218IFC, the setting range for the connection number is larger, i.e., 1 to 20.

2.6.6 M-EXECUTOR Module (Motion Program Executor)

Refer to *2.5.7 M-EXECUTOR Module (Motion Program Executor)*.

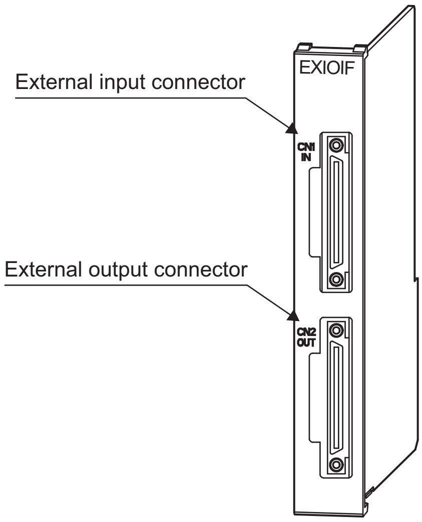
2.7 EXIOIF Module

2.7.1 Overview of Functions

The EXIOIF Module is a special expansion rack interface for the MP2200. This Module can be used to configure an MP2200 system with up to four racks.

2.7.2 External Appearance

The following figure shows the external appearance of the EXIOIF Module.



2.7.3 Hardware Specifications

The following table shows the hardware specifications of the EXIOIF Module.

Item	Specifications
Abbreviation	EXIOIF
Model	JAPMC-EX2200
Function	Expansion rack interface (maximum 4-Rack configuration)
Expansion Bus Connector	HDRA-EC68LFDT-SL (HONDA)
Expansion Bus Interface	IEEE-488 (GPIB): Equivalent to SN75160 (TI).
Rack No. Recognition	The CPU Module automatically recognizes rack 1 from the expansion cable connection. When nothing is connected to the IN connector, a basic unit rack (rack 1) configuration is used. Racks 2 to 4 are in the order that racks are connected to rack 1.
Module Type	MP2200 Optional Module The EXIOIF Module is recognized as an Optional Module. It can be mounted in any slot.
Dimensions (mm)	125×95 (H $\times$ D)
Mass	80 g

2.8 Optional Module

This section provides an option module overview. For more information on its specifications, functions, connections, settings, etc., refer to the following documents separately.

2.8.1 Optional Module Overview List

<table>
<tr><th>Classification</th><th>Optional Module Name</th><th>Module Overview</th><th>Reference Manual</th></tr>
<tr><td rowspan="4">Motion Module</td><td>SVB-01 Module</td><td>The SVB-01 Module is a motion module equipped with a MECHATROLINK supporting interface.
The adoption of MECHATROLINK enables reduced wiring and multi-axis control. In addition, the support for MECHATROLINK-II standard allows you to control position, speed, torque, and phase, realizing precise synchronous control. Also, complex mechanical operation can be achieved by changing the control mode during axis operation.
■ Features
• Up to 21 slave stations per module are connectable (up to 16 servo axes are controllable)
• Because synchronization between modules is enabled, adaptable to interpolation and synchronous control between modules
• An SVB-01 Module that is used as a slave can be connected to a host controller equipped with MECHATROLINK communication.
• Self-configuration function allows you to automatically map slave devices connected to MECHATROLINK.
• SERVOPACK parameters can be managed over the networks.</td><td>Machine Controller MP2000 Series Built-in SVB/SVB-01 Motion Module User's Manual (manual no.: SIEP C880700 33)</td></tr>
<tr><td>SVC-01 Module</td><td>The SVC-01 Module is a motion module equipped with a MECHATROLINK-III interface.
The adoption of MECHATROLINK enables reduced wiring and multi-axis control. In addition, support for MECHATROLINK-III allows you to control positions, speeds, torque, and phases, realizing precise synchronous control. Also, complex mechanical operation can be achieved by changing the control mode during axis operation.
■ Features
• Control performed in a shorter cycle but for the same maximum number of axes as with MECHATROLINK-II.
• Up to 21 slave stations per module are connectable (up to 16 servo axes are controllable).
• Because synchronization between modules is enabled, adaptable to interpolation and synchronous control between modules.
• An SVC-01 Module that is used as a slave can be connected to a host controller equipped with MECHATROLINK-III communication.
• Self-configuration allows you to automatically map slave devices connected to MECHATROLINK-III.
• SERVOPACK parameters can be managed over networks.</td><td>Machine Controller MP2000 Series SVC-01 Motion Module User's Manual (manual no.: SIEP C880700 41)</td></tr>
<tr><td>SVA-01 Module</td><td>The SVA-01 Module is a motion control module with analog output. Capable of controlling a two-axes servo per module or an inverter.
The module has two connectors (CN1, CN2) for connection to a SERVOPACK and an external I/O. Each connector is equipped with an analog output to command speed and torque, an analog input to monitor feedback speed and torque, a pulse input phase-A, B, and C (5V differential), and a general-purpose digital input/output.
The control cycle is fixed at 500µs, so precise control is enabled regardless of high-speed scan cycles.
■ Features
• Two axes servo module with analog output
• Each axis can independently perform position control, speed command output, torque command output, and phase control functions.
• Self-configuration function allows you to automatically map modules.</td><td>Machine Controller MP2000 Series SVA-01 Motion Module User's Manual (manual no.: SIEP C880700 32)</td></tr>
<tr><td>PO-01 Module</td><td>The PO-01 Module is a motion module with pulse output and a four-axes interface. Applicable to connection to a stepping motor or SERVOPACK.</td><td>Machine Controller MP2000 Series Pulse Output Motion Module PO-01 User's Manual (manual no.: SIEP C880700 28)</td></tr>
</table>

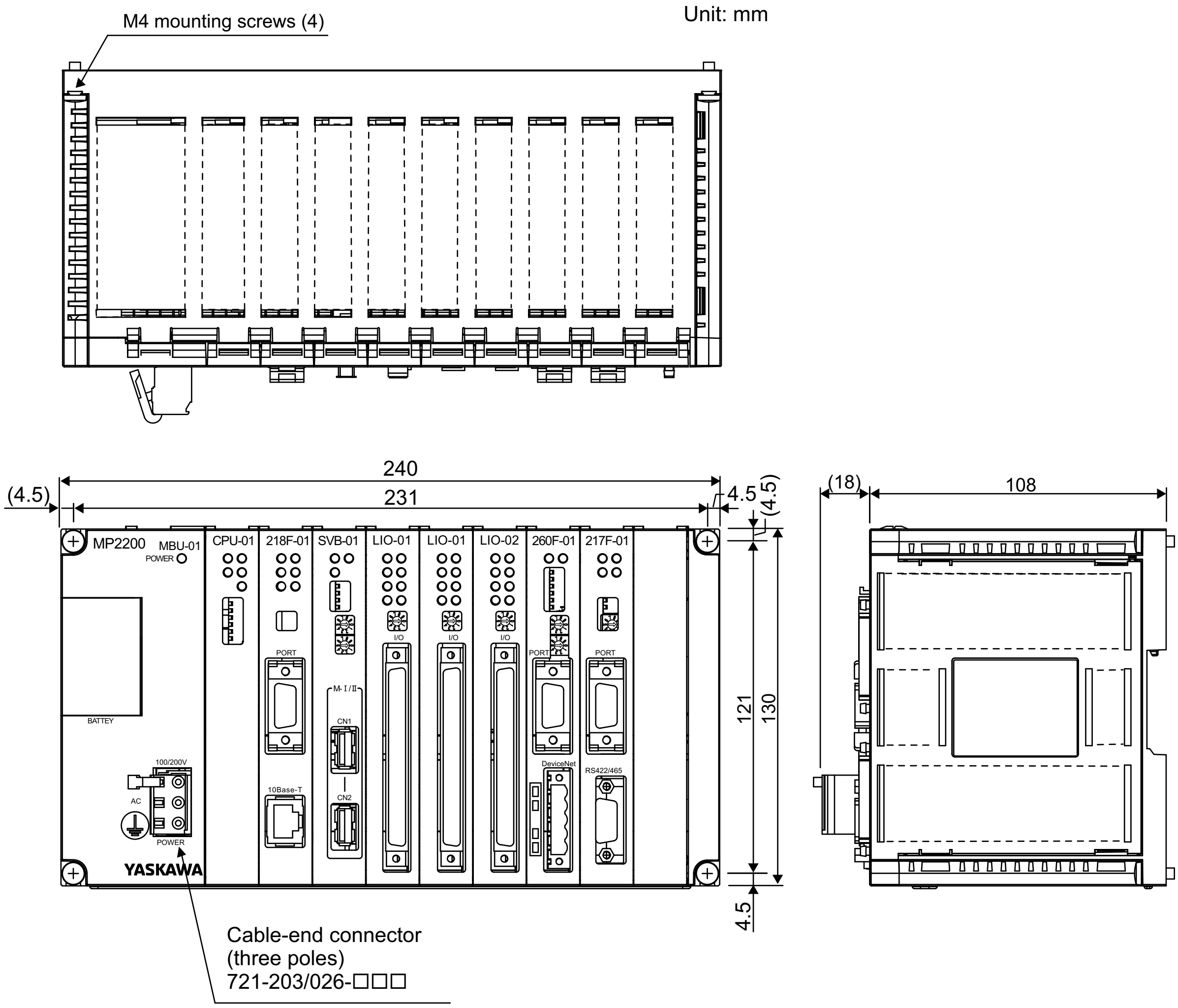
Classification	Optional Module Name	Module Overview	Reference Manual
Input/Output Module	LIO-01/ LIO-02 Module	Digital I/O and pulse counter functions. As a digital I/O function, equipped with 16 digital inputs (DI), 16 digital outputs (DO) (LIO-01: sink output, LIO-02: source output). As a pulse counter function, one pulse input (PI). As for when to input/output for digital I/O and pulse counter functions, input/output for each MP2200 high-speed (High)/ low-speed (Low) scan is carried out at a constant cycle.	*Machine Controller MP2000 Series I/O Module User's Manual* (manual no.: SIEP C880700 34)
	LIO-04/ LIO-05 Module	As a digital I/O function, equipped with 32 digital inputs (DI), and 32 digital outputs (DO) (LIO-04: sink output, LIO-05: source output).	
	LIO-06 Module	As digital I/O function, equipped with 8 digital inputs (DI), 8 digital outputs (DO), one analog input channel, one analog output channel, and one pulse counter input channel.	
	DO-01 Module	As a digital output function, equipped with 64 digital outputs (DO) (sink output).	
	AI-01 Module	8 channel analog input module. For the input, capable of selecting from three options: -10V to +10V, 0V to +10V, or 0 to 20 mA.	*Machine Controller MP2000 Series Analog Input/Analog Output Module AI-01/AO-01 User's Manual* (manual no.: SIEP C880700 26)
	AO-01 Module	4 channel analog output module. For the output, select one from two options: -10V to +10V, or 0V to +10V.	
	CNTR-01 Module	2 channel reversible counter module. 5V differential/ 12V input is optional, and phase-A or -B/ sign/ add-subtract method is optional.	*Machine Controller MP2000 Series Counter Module CNTR-01 User's Manual* (manual no.: SIEP C880700 27)
Communication Module	218IF-01 Module	Equipped with serial interface (RS-232C) and Ethernet interface. Allows you to connect to a personal computer, HMI equipment, or controller by other makers via PORT or 10Base-T connector.	*Machine Controller MP2□00 Communication Module User's Manual* (manual no.:SIEP C880700 04)
	218IF-02 Module	Equipped with serial interface (RS-232C) and Ethernet interface. Allows you to connect to a personal computer, HMI equipment, or controller by other makers via PORT or 100Base-TX connector.	
	217IF-01 Module	Equipped with serial interfaces (RS-232C and RS422/485). Allows you to connect to a personal computer, HMI equipment, or controller by other makers via PORT or RS422/485 connector.	
	260IF-01 Module	Equipped with serial interface (RS-232C) and DeviceNet interface. Allows you to connect to a controller by other makers via DeviceNet connector. Also, allows you to connect to a personal computer or HMI equipment by other makers via the PORT connector.	
	261IF-01 Module	Equipped with serial interface (RS-232C) and PROFIBUS interface. Allows you to connect to a controller by other makers via the PROFIBUS connector. Also, allows you to connect to a personal computer or HMI equipment by other makers via the PORT connector.	
	215AIF-01 Module	MPLINK and CP-215 specifications. MPLINK specification is equipped with one line of our original real-time core network interface MPLINK transmission and a serial interface (RS-232C). CP-215 specification is equipped with one line of our original real-time core network interface CP-215 transmission and a serial interface (RS-232C).	
	262IF-01 Module	Equipped with an FL-net interface. Allows you to connect to a personal computer, HMI, or FA controller via a 100Base-TX or 10Base-T connector.	*Machine Controller MP2000 Series 262IF-01 FL-net Communication Module User's Manual* (manual no: SIEP C880700 36)
	263IF-01 Module	Equipped with an EtherNet/IP interface. Allows you to connect to a personal computer or EtherNet/IP device via the EtherNet/IP connector.	*Machine Controller MP2000 Series 2631F-01 EtherNet/IP Communication Module User's Manual* (manual no.: SIEP C880700 39)

2.9 External Dimensions

2.9.1 Basic Unit

The following figure shows the external appearance of the Basic Unit.

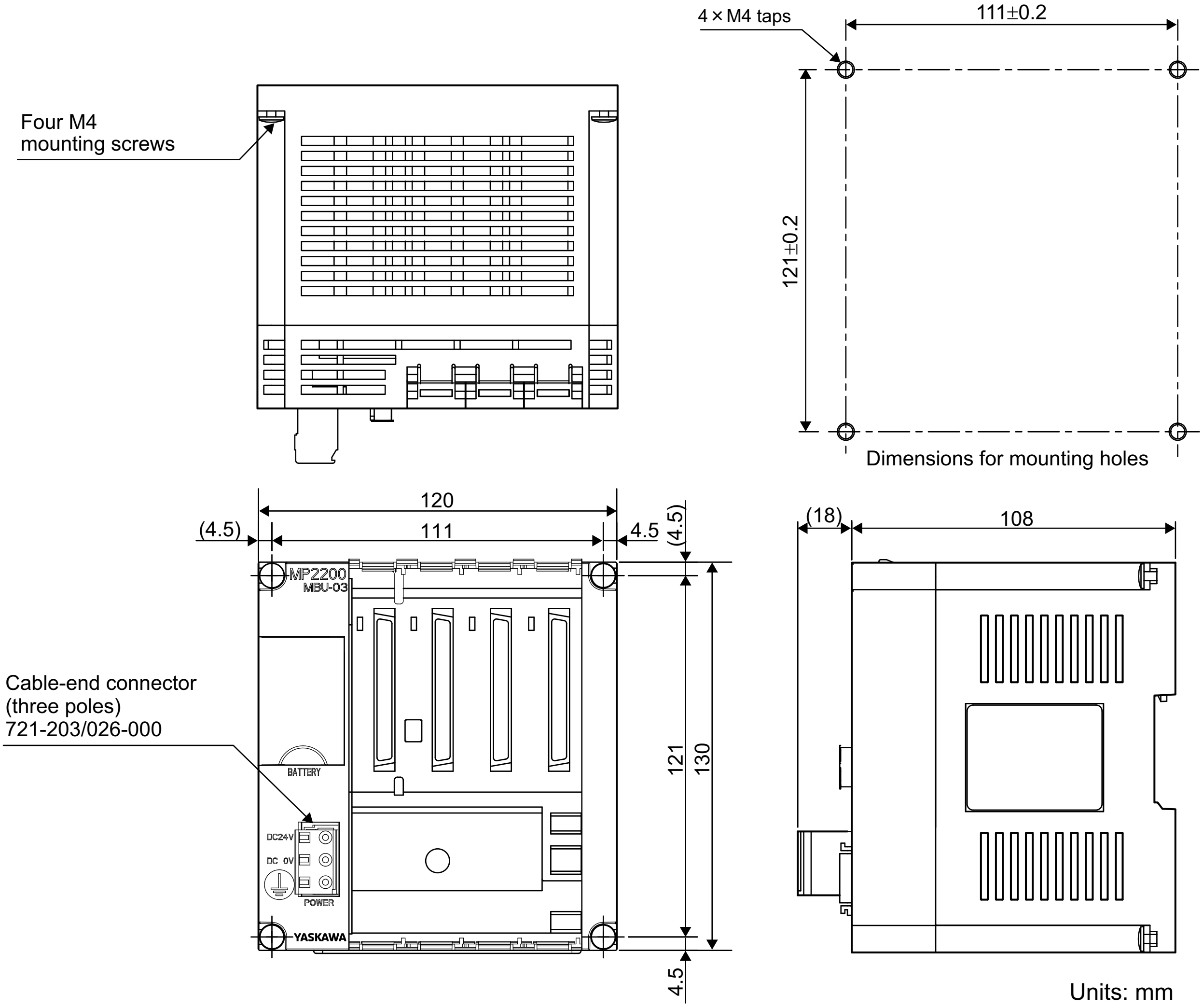
(1) Nine-slot Base Unit



Note: 1. Cable-end connector is attached to the power connector.
Models: MBU-01 721-203/026-304 (Black)
MBU-02 721-203/026-000 (White)

2. Attachment
   - Handle for power connector (model: 231-131)
     These handles are used when connecting a cable to the cable-end connector.

(2) Four-slot Base Unit

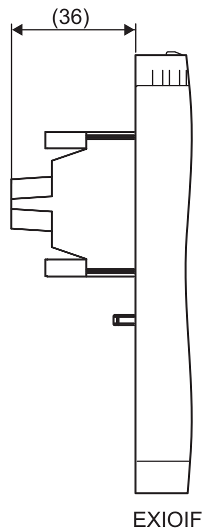


Note: 1. Cable-end connector is attached to the power connector.
Model: 721-203/026-000 (White)
2. Attachment
- Handle for power connector (model: 231-131)
These handles are used when connecting a cable to the cable-end connector.

2.9.2 Mounting Procedure for EXIOIF Connector

The EXIOIF Module has the following dimensions.
Height: 125 mm, Depth: 95 mm

The following figure shows the mounting dimensions for the connector of the EXIOIF Module.



3

Mounting and Wiring

This chapter explains how to handle MP2200 and the connection methods for each module.

3.1 Mounting MP2200

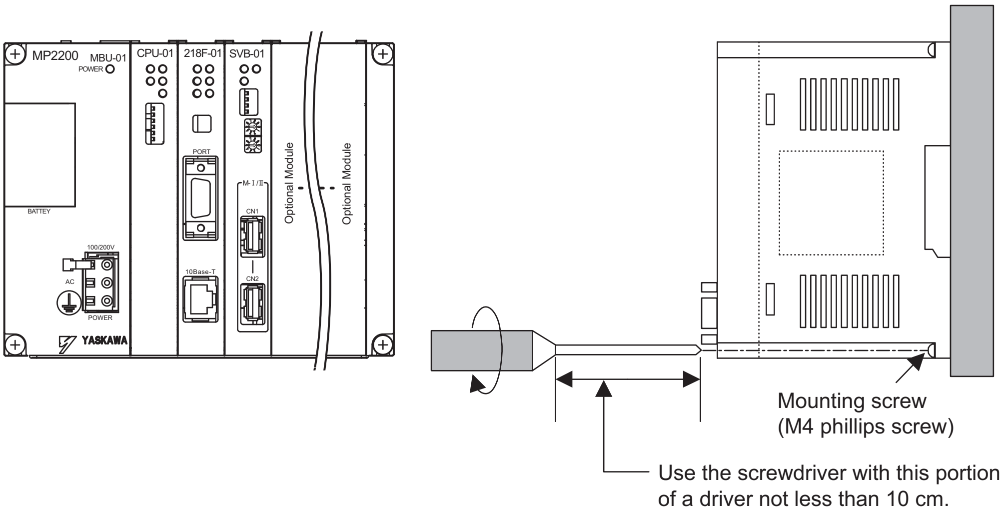
3.1.1 Method

There are two methods for mounting MP2200.

- Using screws
- Using DIN rail

(1) Screwed Method

Push the MP2200 mounted clamp onto the mounting plate as shown in the following figure, and use four mounting screws to firmly secure the clamp.

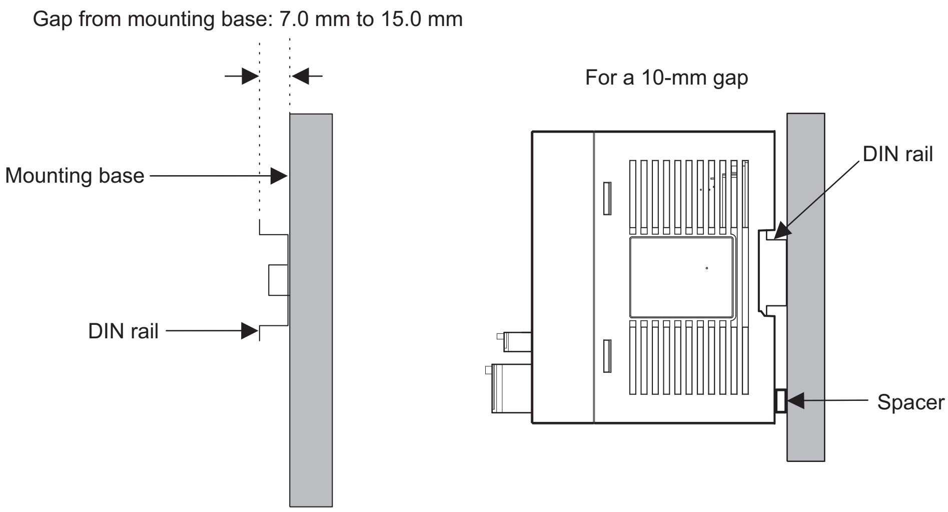


Note: Vertically mount it on the wall as shown in the figure above.

(2) DIN Rail Mounting

[a] DIN Rails and Spacer

Several types of DIN rails are available: with 7-mm to 15-mm gap from the mounting base as shown in the following diagram. If mounting a MP2200 using DIN rail with 10-mm gap, install a spacer on the rear of the MP2200 near the bottom to protect the MP2200 from vibration and shock.

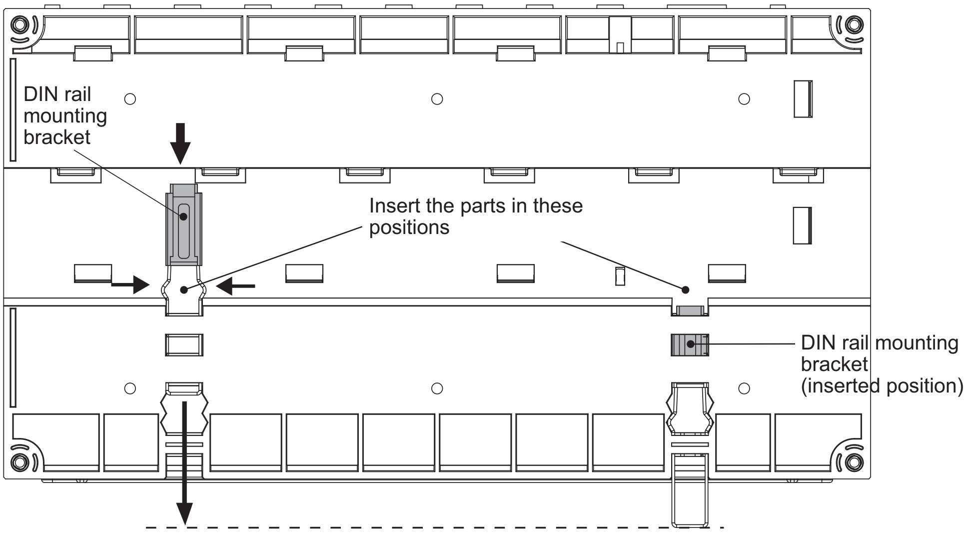


[b] Procedure for Mounting to DIN Rail

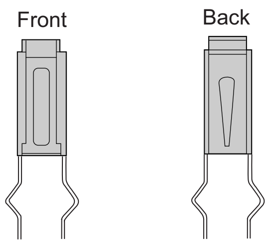
Use the following procedure to attach the DIN rail mounting parts to the MP2200 and then mount the MP2200 to the DIN rail.

1. Insert the DIN rail mounting brackets to the dotted line in the two slots on the rear of the MP2200 as shown in the following figure.

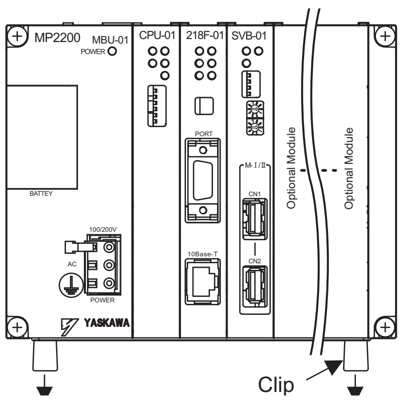
MP2200 Rear Side



Note: The following figure shows the front and back of a mounting clip. Insert each clip so that its front faces outward.



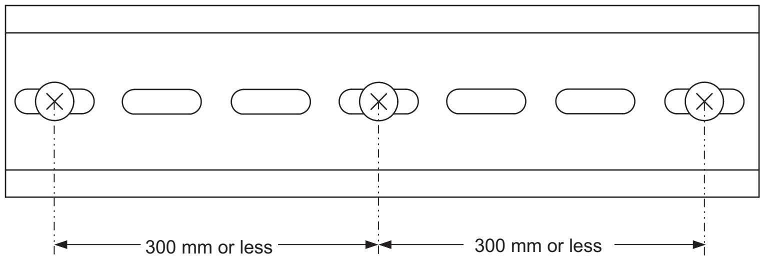
2. Pull the DIN rail mounting clips down to release them.



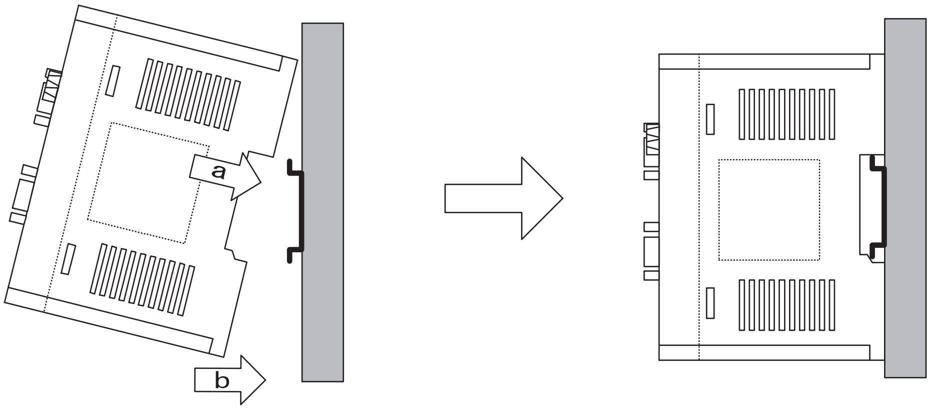
Mounting and Wiring
3

■ Fixing a DIN Rail

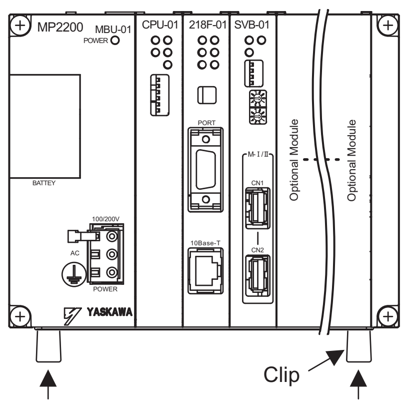
Make sure to fix a DIN rail at 300mm or less pitch as shown in the figure below.



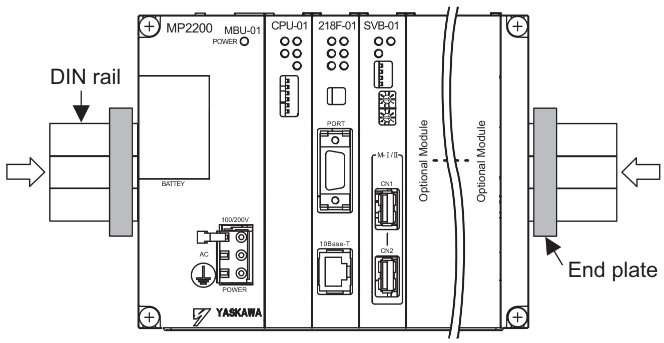
3. Hook the MP2200 to the top of the DIN rail (a), and then push the MP2200 towards the mounting base to secure it in place (b).



4. Push the DIN rail mounting clips to lock them in place..



5. Place end plates on both sides of the MP2200 to secure it to the DIN rail.

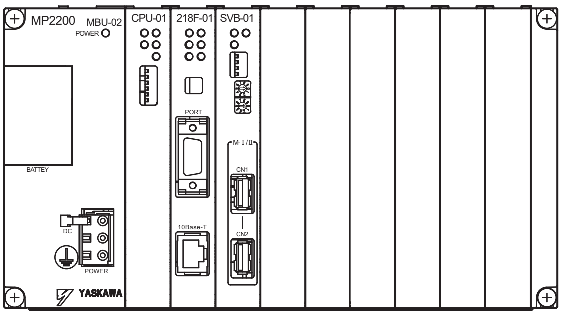


This completes the installation procedure.

3.1.2 MP2200 Mount Direction

Be sure to mount the MP2200 using screwed method or DIN rail.

View from front, when attached



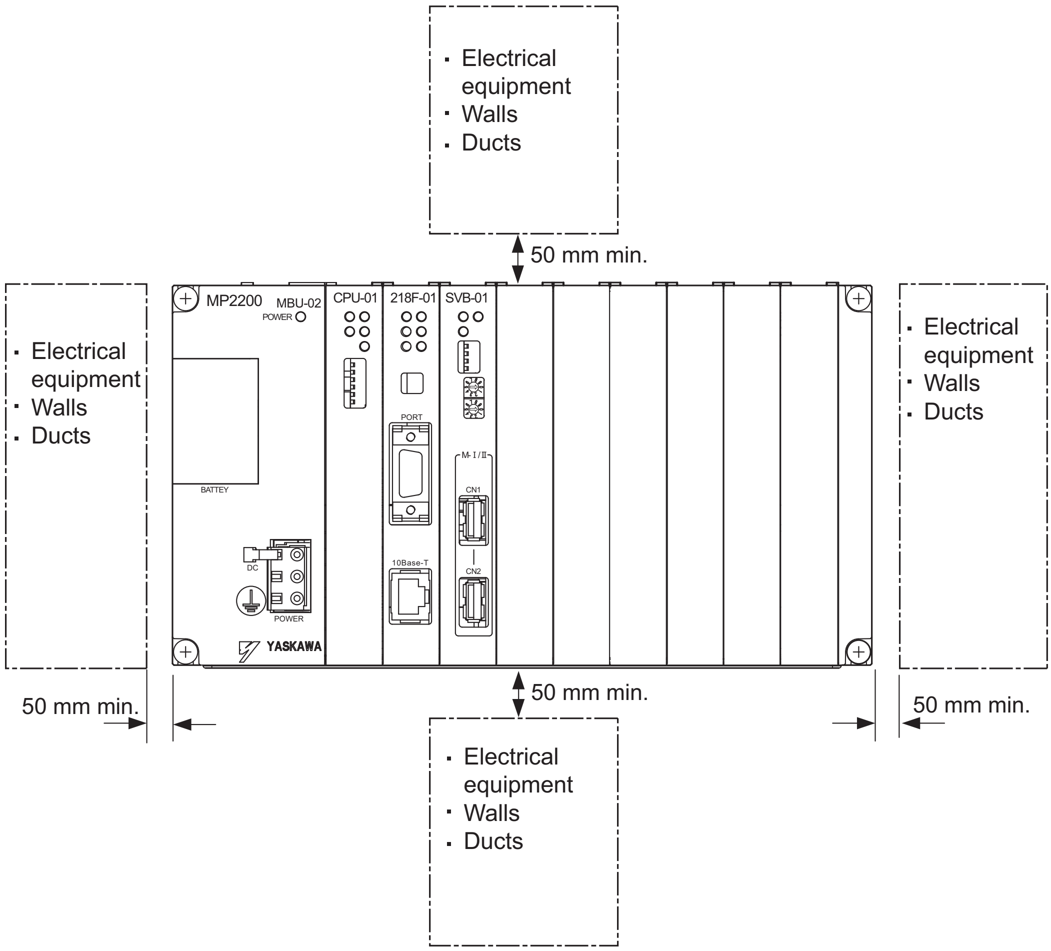
3.1.3 Installation Clearance

To enable proper ventilation, install the MP2200 with the clearance shown in the following figure.

■ Mounting Conditions

- Above and below: 50 mm min.
- Left and right: 50 mm min.

Note: This clearance applies to an ambient temperature of 55°C max.



3.1.4 Replacing and Adding Optional Modules

Use the following procedures to replace and add Optional Modules.

(1) Preparations

1. Create a backup data file.

Use the MPE720 to save the MP2200 program on a computer (right-click the **PLC**, and select ***Transfer - All Files - From Controller to MPE720***.)

2. Remove the MP2200.

Turn OFF the power supply and disconnect all cables from the MP2200. Then remove the MP2200 from the panel or rack and place on a workbench or other area with sufficient space.

(2) Removing Optional Modules

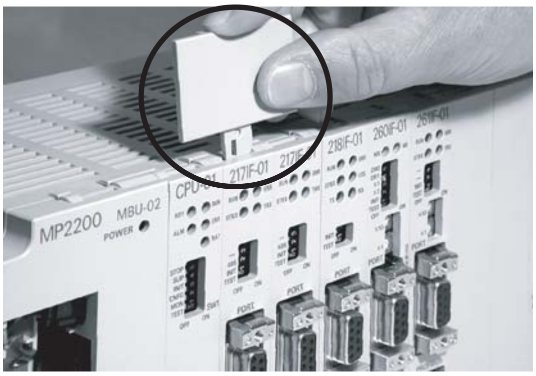
1. Remove the battery cover.

Pull the notch on the side of the MP2200 towards you to remove the battery cover.

2. Remove the panel of Optional Module.

Insert the protruding part of the battery cover into the slot on top of the panel of Optional Module to unhook, as shown in the diagram. Face the front of the battery cover towards you for this operation.

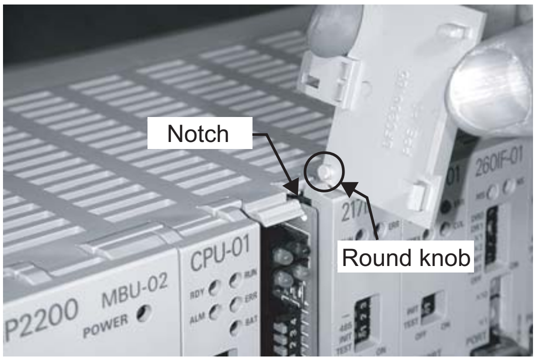
- Remove the front cover (optional) from the empty slot before mounting an Optional Module in an empty slot.



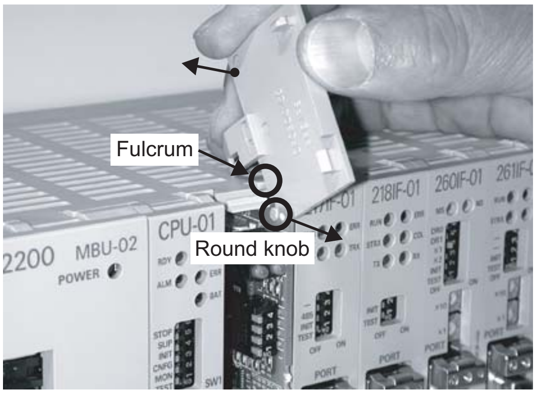
Unhook the bottom in the same way.

3. Remove the Optional Module from the mounting base.

Pull the top of the panel of the Optional Module towards you to remove it. A notch on the Optional Module will be visible from the gap in the cover. Hook the round knob on the battery cover, shown in the diagram, into the notch in the Optional Module.

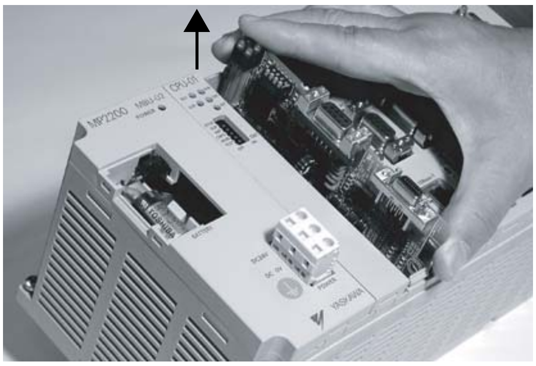


Hold the center of the battery cover as shown in the following diagram. Push the battery cover down and out, rotating from the round knob to disconnect the Module and mounting base connectors, and then pull the Optional Module forward.



4. Pull out the Optional Module.

Hold the Module on the top and bottom and pull it out straight. Hold the edges of the Module and avoid touching the parts on the Module.



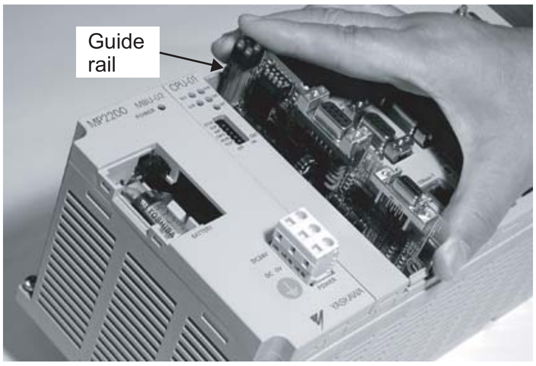
Put the removed Module into the bag that was supplied with and store the Module in this bag.

(3) Installing Optional Modules

1. Insert Optional Modules.

Hold the top and bottom of the Module to be installed, line up the Module on the left-side guide rail inside the Option Slot, and then insert it straight.

Note: The FG bar on the inside bottom of the Unit Case may be damaged if the Module is not inserted straight.



2. Mount on to the mounting base.

Once the Optional Module has been completely inserted, place your hand on the front face of the Optional Module and push hard until the Optional Module has been inserted into the mounting base connectors. The front face of the Optional Module and the hook will be aligned when the Optional Module has been installed properly.

3. Install the panel of the Optional Module.

Place the hole on the bottom of the panel of the Optional Module onto the hook on the bottom of the MP2200. Next, hook the hole at the top of the panel of the Optional Module onto the hook at the top of the MP2200.

This completes the Optional Module mounting procedure.
Be sure to attach the option cover (model: JEPMC-OP2300) on the empty slot.

3.2 Connecting Basic Units

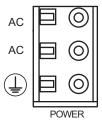
3.2.1 Connecting Base Units

(1) MBU-01 Power Supply Connector

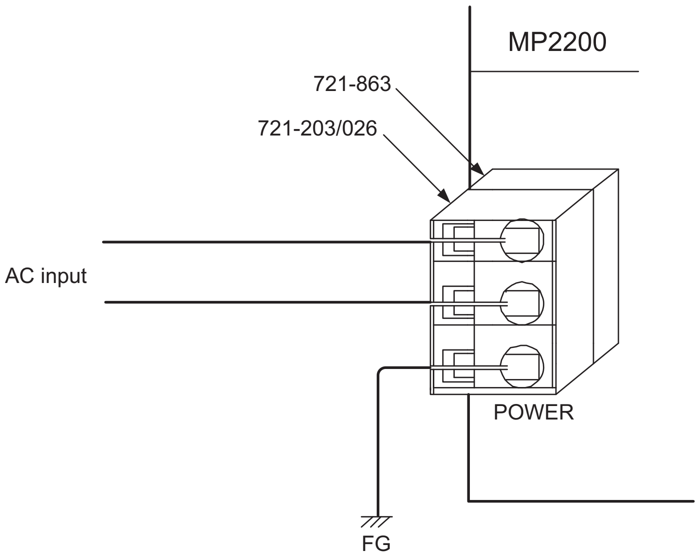
[a] Specifications, Pin Arrangement, and Connection Procedure

Supply 100/200 VAC to the MP2200. Connect the power supply connector as shown in the following diagram.

Name	Connector Name	No. of Pins	Connector Model			
			Module	Cable	Manufacturer	Cable Model
Power Supply Connector	POWER	3	721-863	721-203/026	WAGO Company of Japan, Ltd	–



Pin No.	Signal Name	Description
3	AC	AC input
2	AC	AC input
1	FG	Frame ground (Ground to 100 Ω or less.)

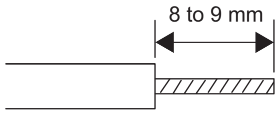


[b] Making a 100/200-VAC Power Supply Cable

The power supply terminal has a removable connector. Use the following procedure to wire the terminal to the power supply connector. Use 1.5 to 2.5 mm^2 (AWG16 to AWG13) twisted-pair cable to connect the 100/200-VAC power supply to the power supply connector on the MP2200.

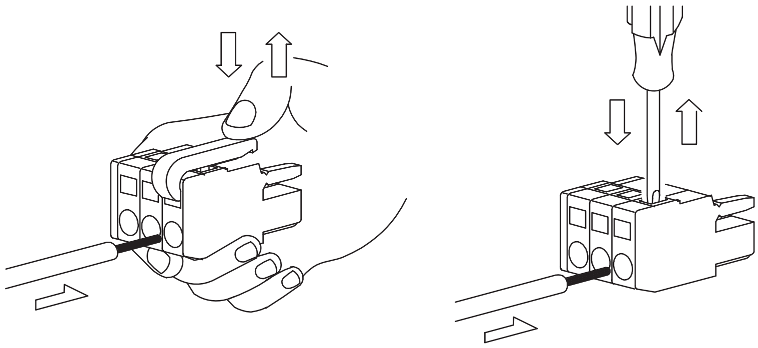
1. Strip the end of the cable.

Strip approx. 8 to 9 mm the end of the wires.



2. Clamp the wires.

Insert the wires all the way to the back of the plug and clamp them securely.

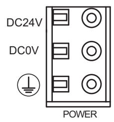


(2) MBU-02/MBU-03 Power Supply Connector

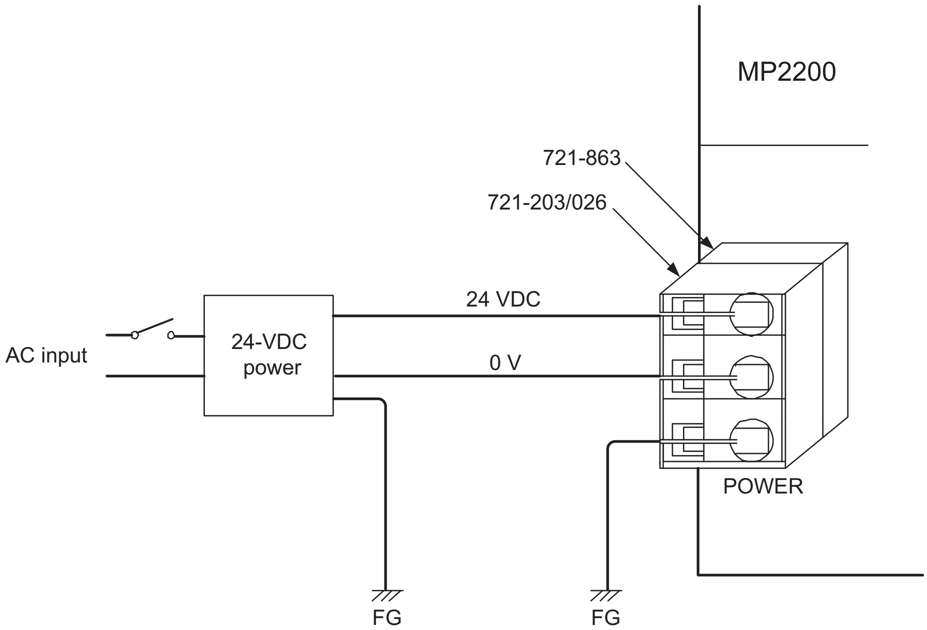
[a] Specifications, Pin Arrangement, and Connection Procedure

Supply 24 VDC to the MP2200. Connect the power supply connector as shown in the following diagram.

Name	Connector Name	No. of Pins	Connector Model			
			Module	Cable	Manufacturer	Cable Model
Power Supply Connector	POWER	3	721-863	721-203/026	WAGO Company of Japan, Ltd	–



Pin No.	Signal Name	Description
3	24V	24-VDC input
2	0V	0-V input
1	FG	Frame ground (Ground to 100 Ω or less.)



Note: Use an insulated 24-VDC power supply. Attach the power supply switch on the AC side. If the switch is attached on the 24-VDC side, there will be an inrush current of approximately 40 A when the power is turned ON.

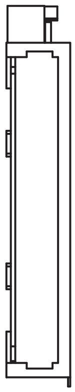
[b] Making a 24-VDC Power Supply Cable

The power supply terminal has a removable connector. Use the following procedure to wire the terminal to the power supply connector. Use 0.8 to 2.6 mm^2 (AWG18 to AWG13) twisted-pair cable to connect the 24-VDC power supply to the power supply connector on the MP2200.
For the procedure to make the cable, refer to *3.2.1 (1) [b] Making a 100/200-VAC Power Supply Cable* .

3.2.2 CPU-02 Module Connections

(1) CARD Module Connector Details

[a] Compact Flash (CF) Slot Specifications



CF slot (Compact Flash not inserted)

• Connector Specifications

Name	No. of Pins	Connector Model		
		Module side	Cable side	Manufacturer
CF slot	50	MI21A-50PD-SF-EJR(31)	–	Hirose

(2) USB Connectors

[a] USB Connector Specifications

Series Mini-B plug

[b] USB Cable

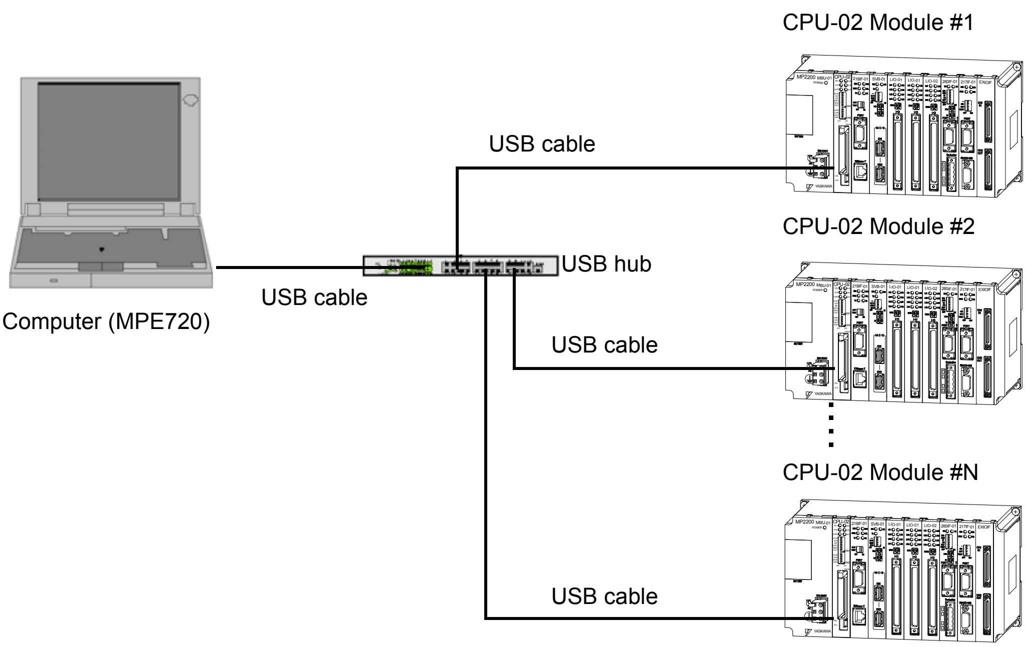
Use a commercially-available USB cable.

Connectors:

- Computer: Series A plug
- Module: Series Mini-B plug

[c] USB Interface Connection Example

The following figure shows a connection example using multiple CPU-02 Modules.



3.2.3 CPU-03/CPU-04 Module Connections

(1) CARD Module Connector Details (except CPU-04)

Refer to *3.2.2 (1) CARD Module Connector Details* .

(2) Ethernet Connector Details

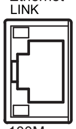
Connects to a personal computer or HMI device by Ethernet (100Base-TX /10Base-T).

[a] Ethernet Connector Specification and Pin Arrangement/ Indicator Lamp

The following table provides the Ethernet connector specifications.

Name	Connector Name	Number of Pins	Connector Model		
			Module Side	Cable Side	Manufacturer
Ethernet	Ethernet	8	RJ-45 CAT5 Socket	RJ-45 CAT5 Plug	Pulse Engineering

The following table provides Ethernet connector pin arrangement/ indicator lamp details.



Pin Number	Signal Name	Description
1	TXD+	Transmitted data + side
2	TXD-	Transmitted data – side
3	RXD+	Received data + side
4	–	–
5	–	–
6	RXD-	Received data – side
7	–	–
8	–	–

Indicator Name	Indicator Color	Description
LINK	Yellow	Lit: Connect Unlit: Unconnected
100M	Green	Lit: Connected at 100Mbps, or automatically negotiating Unlit: Connected at 10Mbps

[b] Ethernet Cable

For the Ethernet cable, use a twisted pair cable with RJ-45 connector.

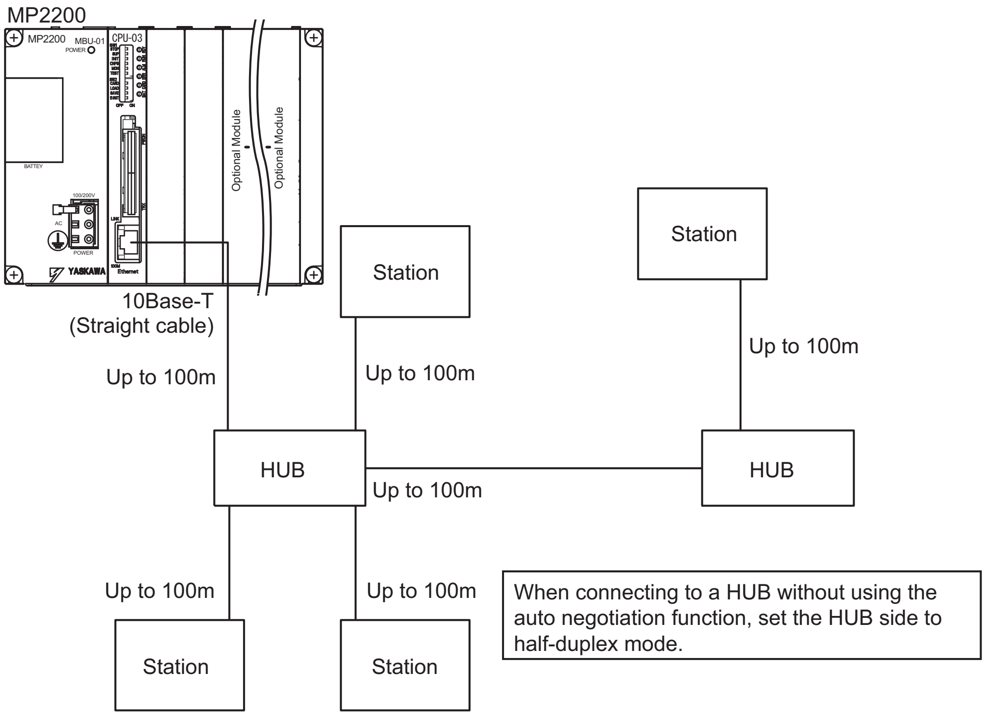
Ethernet Type	Category	Remarks
10Base-T	Category 3 or more	• When connecting to remote device through a hub: Straight cable • When connecting to remote device without using a hub: Cross cable
100Base-TX	Category 5 or more	

[c] Ethernet Connection Example

The following are examples of Ethernet network connections via 10Base-T cable:

■ Connection Example 1

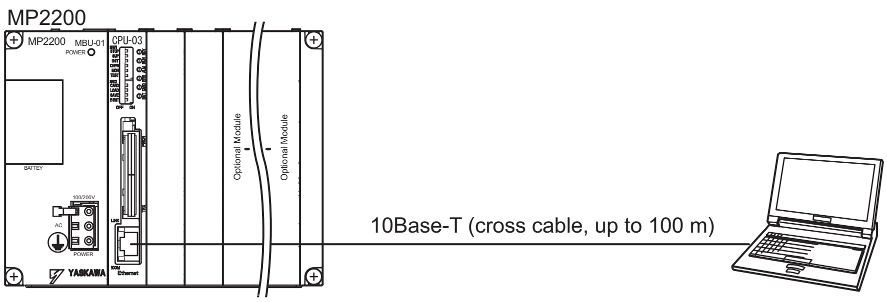
When using a repeater HUB:



Specification

Item	When Connecting to a Repeater HUB	When Connecting to a Switching HUB
Cable Length between Node-HUB	100 m or less	100 m or less
Cable Length between HUBs	100 m or less	100 m or less
Number of HUBs between Nodes	Up to four	Unlimited

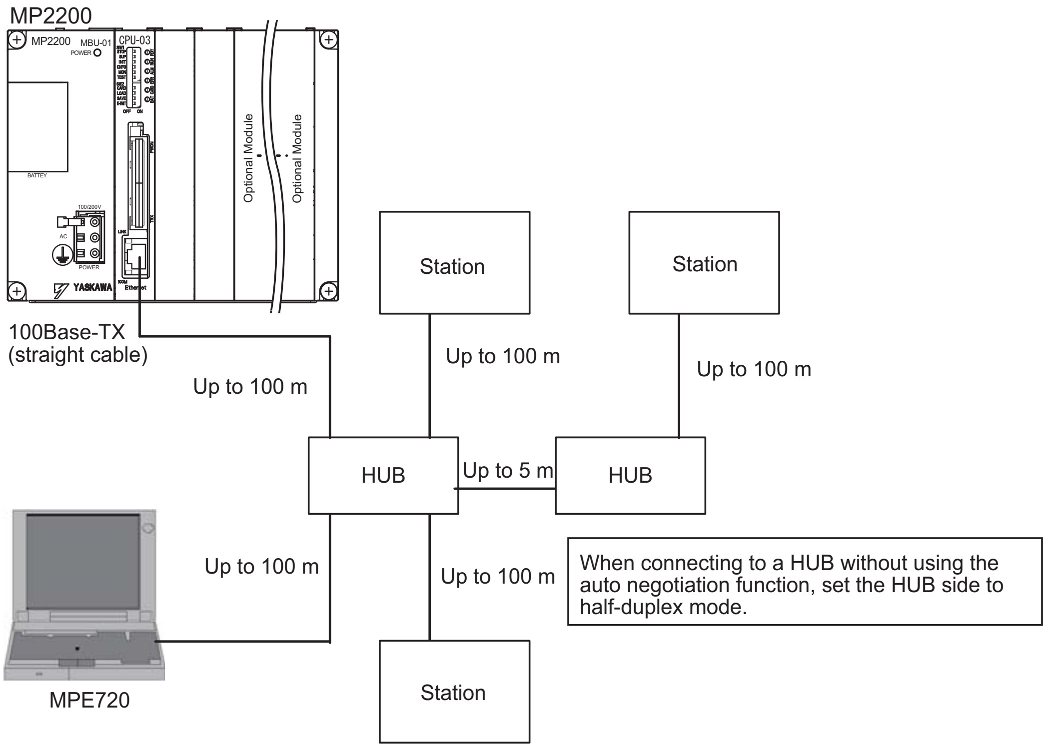
■ Connection Example 2



The following are examples of Ethernet network connections via 100Base-TX cable:

■ Connection Example 3

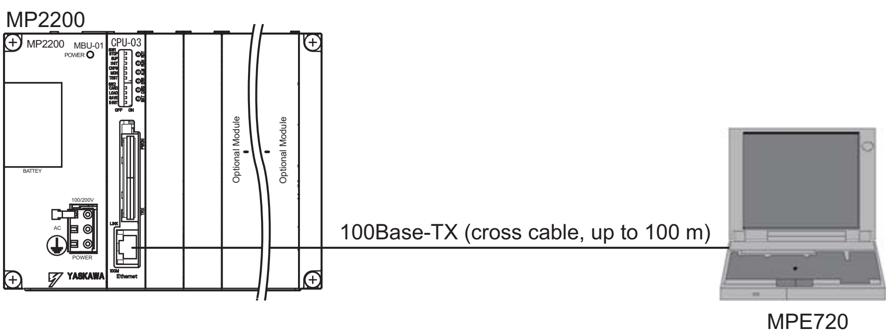
When using a repeater HUB:



Specification

Item	When Connecting to a Repeater HUB	When Connecting to a Switching HUB
Cable Length between Node-HUB	100 m or less	100 m or less
Cable Length between HUBs	5 m or less	100 m or less
Number of HUBs between Nodes	Up to two	Unlimited

■ Connection Example 4



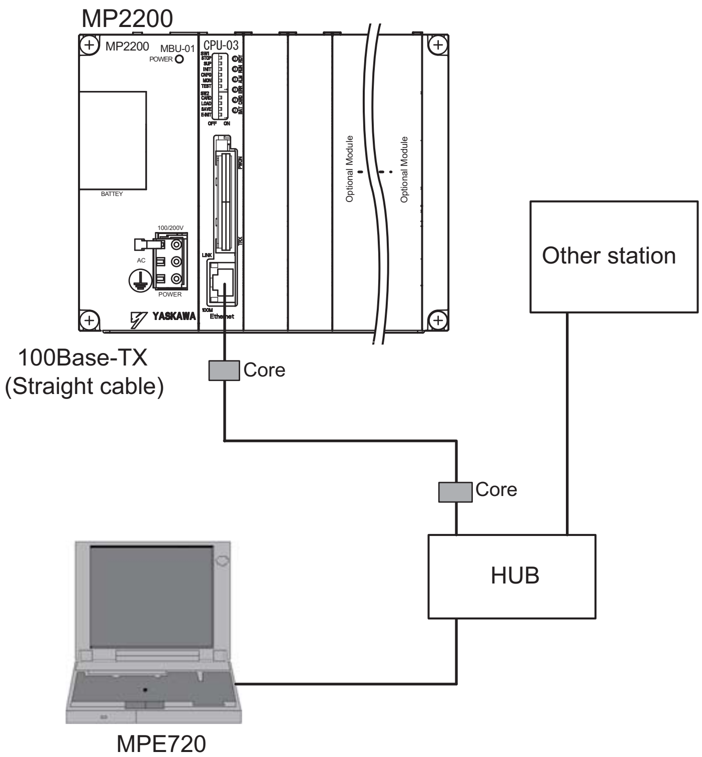
■ Caution

High frequency wave noise from other devices in the installation environment may cause error in communications using 100Base-TX connections. When constructing a system, use MP2200 protective measures to avoid the influence of high frequency wave noise as follows:

1 Wiring
Wire Ethernet cables so that they are well-separated from other cable systems, such as the main circuit or power lines.

2 Communication system (100Base-TX)
- Communicate data to a remote device through TCP/IP communication.
- If necessary, increase the number of communication retries.

3 Attach a ferrite core.
Attach a ferrite core in the manner described below:
Ethernet : Attach it to the communication port side and the external device side of the MP2200 main unit.



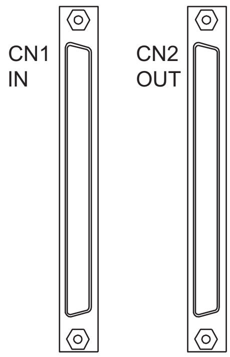
Note: Recommended ferrite core

Model:	Manufacturer
E04SR301334	Seiwa Electric Mfg. Co., Ltd

3.2.4 EXIOIF Module Connections

(1) Connectors

The following diagram shows EXIOIF Module connectors.

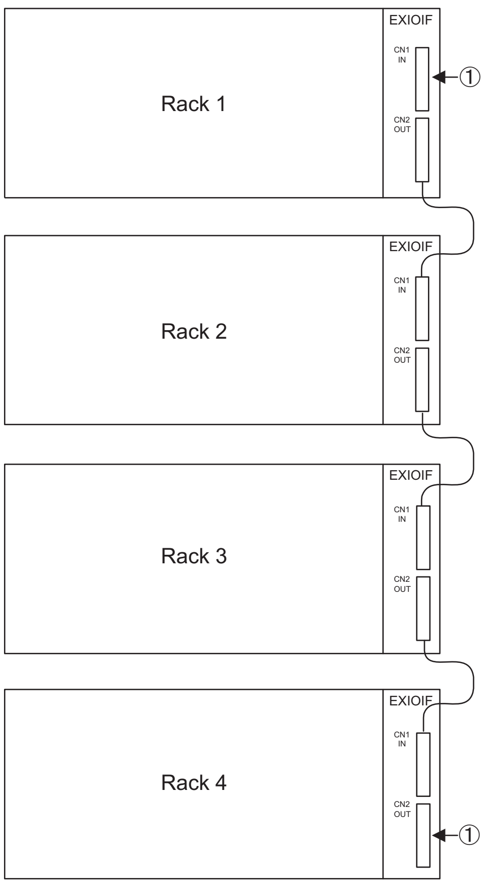


These connectors are used to expand the number of mounting bases to a maximum of four racks.

They are connected using the following cables.

• JEPMC-W2091-□□

The following diagram shows how to connect the external I/O connectors.



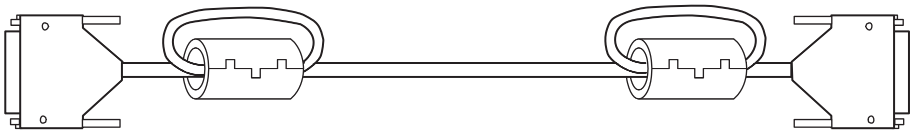
Note: Attach the enclosed dust caps to the unused connectors (1) in the above figure.

(2) Cables

Name	Model	Length
Connection Cables (with connectors on both ends)	JEPMC-W2091-A5	0.5 m
	JEPMC-W2091-01	1.0 m
	JEPMC-W2091-2A5	2.5 m

Note: 1. The total cable length when adding expansion racks is 6.0 m max. Connect the shield to the connector shell.
2. Connection method: 1:1
3. Cable specifications: Shielded cable, equivalent to UL20276, 0.08 mm^2 (AWG28), two ferrite cores attached

(3) Cable Appearance

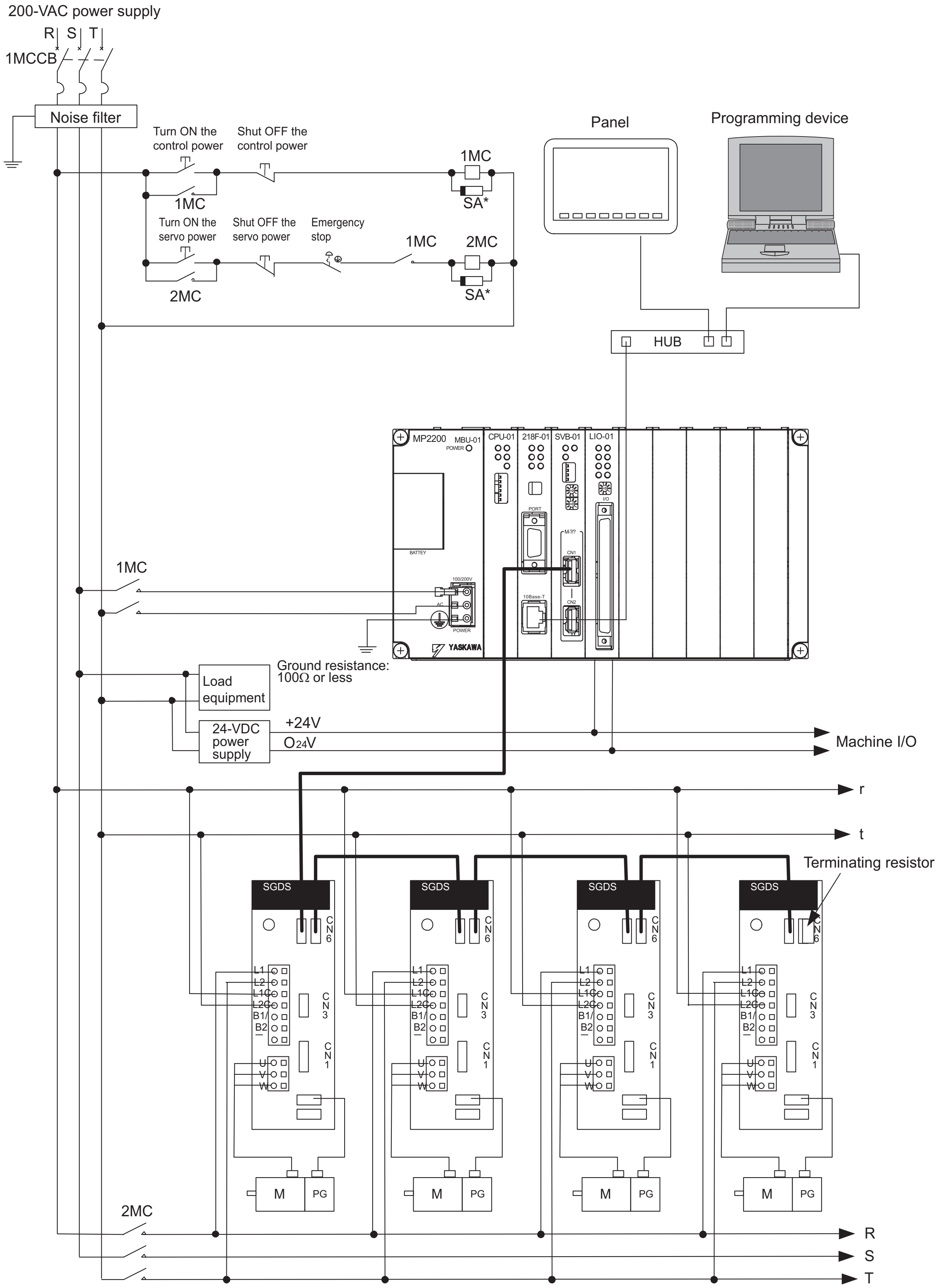


3.2.5 System Connection Example

The following diagram shows a connection example of a system using the MP2200.

The following diagram shows a connection example for the system using a 200-VAC power supply.

Note: Select the SERVOPACK, 24-VDC power supply to use in accordance with the input power supply specification.



* SA: Surge absorber (to absorb switching surge)

4

System Startup

This chapter explains how to start up a model system using the programming tool MPE720 Ver.6. Note that the procedure for designing a mechanical system has been omitted here.

4.1 System Startup Overview

This section describes the procedure to set up a system and to confirm the execution of motion program using the program shown below.
The motion program created here is simple and has only three lines to move two axes 150,000 pulses from the current position and stop them.

```
INC;        Specify an incremental mode
MOV         [A1]150000, [B1]150000;Position two-axes 150,000 pulses
END;
```

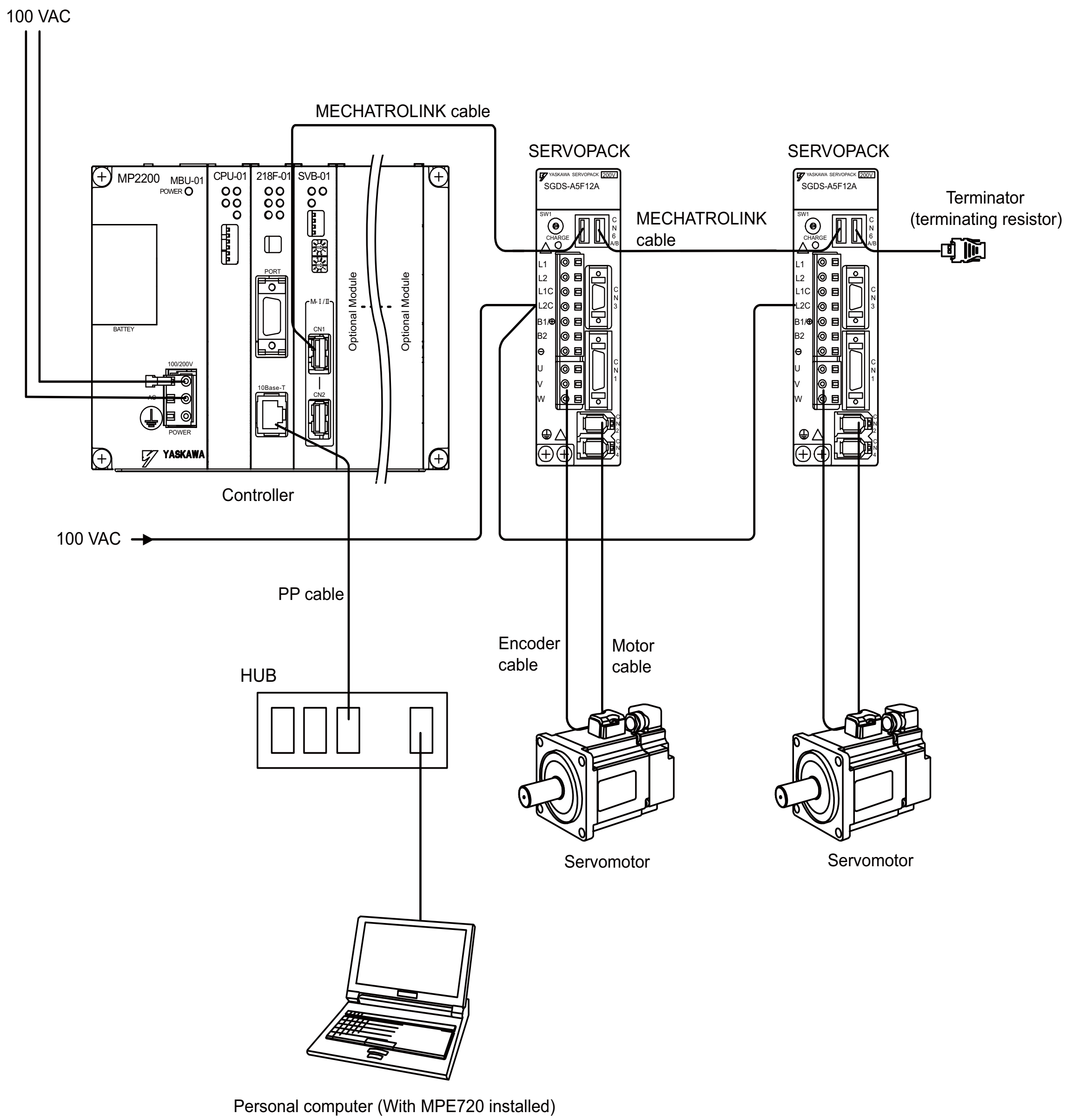
4.2 Preparation

This section explains the steps of "wiring," "self-configuration," and "test operation" for starting up the model system.

4.2.1 Wiring

We use the following layout model to explain the startup of the model system. Prepare each device listed on the next page and connect them as shown in the figure below.

(1) System Layout Model



[a] Required Equipment

Product Name		Model	Q'ty
MP2200	Base Unit with AC power supply	JEPMC-BU2200	1
	CPU-01 Module	JAPMC-CP2200	1
	SVB-01 Module	JAPMC-MC2310	1
	218IF-01 Module	JAPMC-CM2300	1
MECHATROLINK cable (0.5m)		JEPMC-W6002-A5	2
Terminator (terminating resistor)		JEPMC-W6022	2
Σ-III SERVOPACK		SGDS-A5F12A	2
Σ-III servomotor		SGMAS-A5A2A21	2
Motor cable (3m)		JZSP-CSM01-03	2
Encoder cable (3m)		JZSP-CSP05-03	2
HUB (commercialized product)		LSW-TX-8EP	1
MPE720 Ver.6		CPMC-MPE770	1
LAN cable (for Ethernet connection)		Commercialized straight cable	2
Personal computer		Commercialized product	1

■ Caution

- Install MPE720 Ver.6 in the personal computer before starting step 1.
  For information on its installation, refer to the *Engineering Tool for MP2000 Series Machine Controller MPE720 Version 6 Users Manual* (manual no.: SIEP C880700 30).
- Set the PC Ethernet port in advance. For the procedure to set up communication, refer to the *Engineering Tool for MP2000 Series Machine Controller MPE720 Version 6 Users Manual* (manual no.: SIEP C880700 30).
- The SERVOPACK station number (SW1) is set to 1 and 2.
- In a 1:1 connection without HUB, use a cross cable as a LAN cable.

4.2.2 Self Configuration

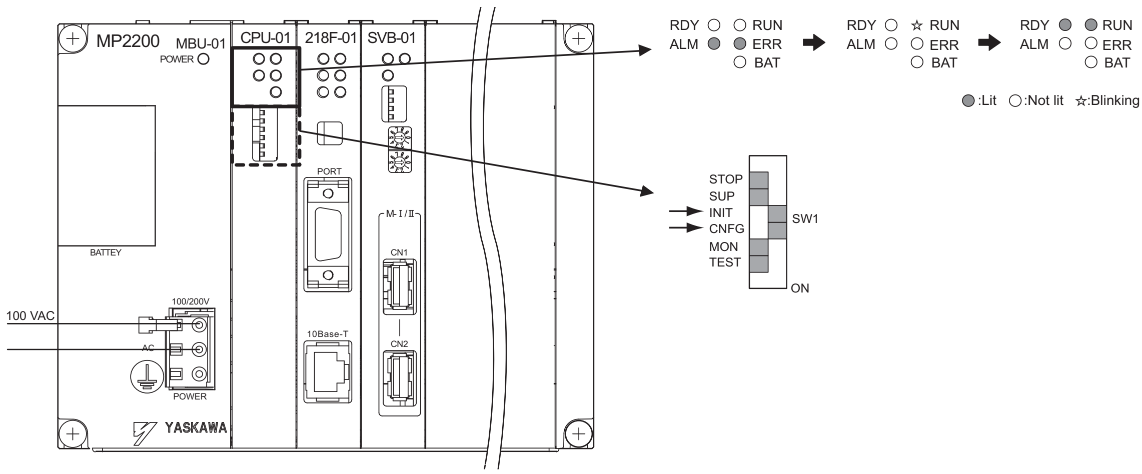
Run the self configuration to automatically recognize devices connected to the MECHATROLINK connector.
Steps for self configuration are as follows.

1. Check that the Σ-III SERVOPACK power supply is ON.

2. Turn OFF the 100-VAC power supply to the MP2200.

3. Set the INIT and CNFG pins on the DIP switch (SW1) to ON on the MP2200.

4. Turn ON the 100-VDC power supply to the MP2200, and confirm that the indicators change as follows:



5. Self configuration is complete, and MECHATROLINK slave device information has been written to a definition information file.

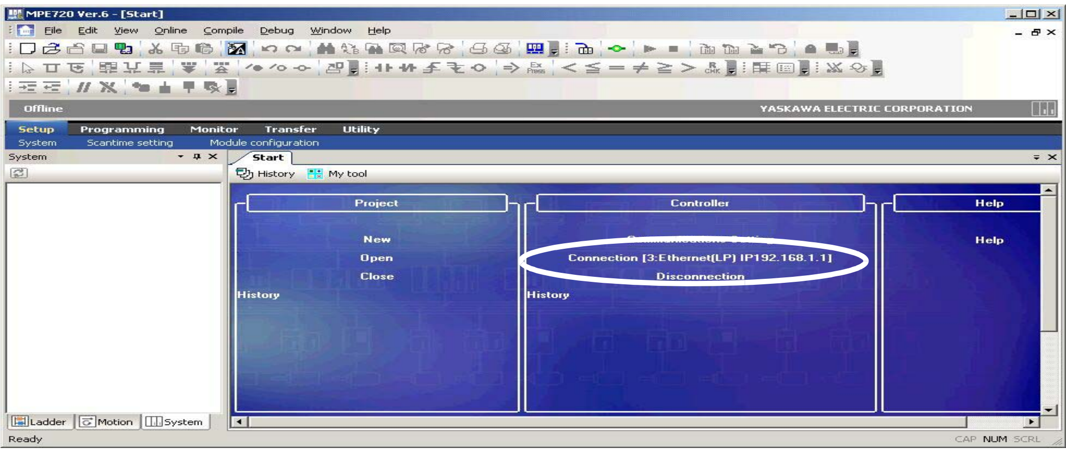
6. Set the INIT and CNFG pins on the DIP switch (SW1) to OFF on the MP2200.

4.2.3 Test Operation

Confirm that the machine controller can command axis servo ON/OFF and jog operation.

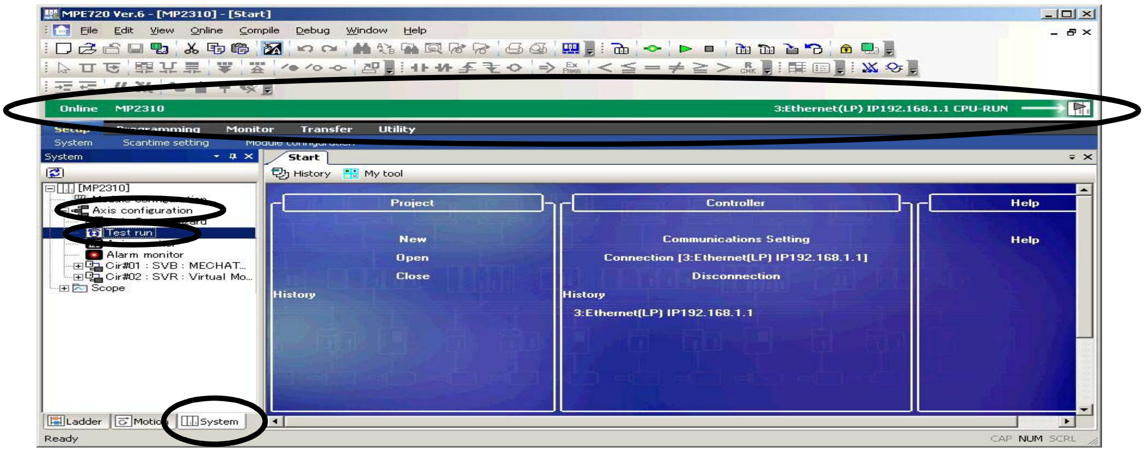
(1) Starting and Connecting MPE720 Ver.6

Launch MPE720Ver.6 and click “3:Ethernet(LP)192.168.1.1” to connect to the controller.

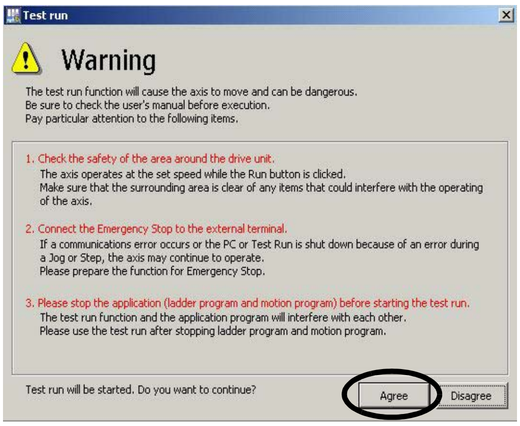


(2) Operating Manually in the Test Run Window

When the connection is complete, the display will change from offline to online.

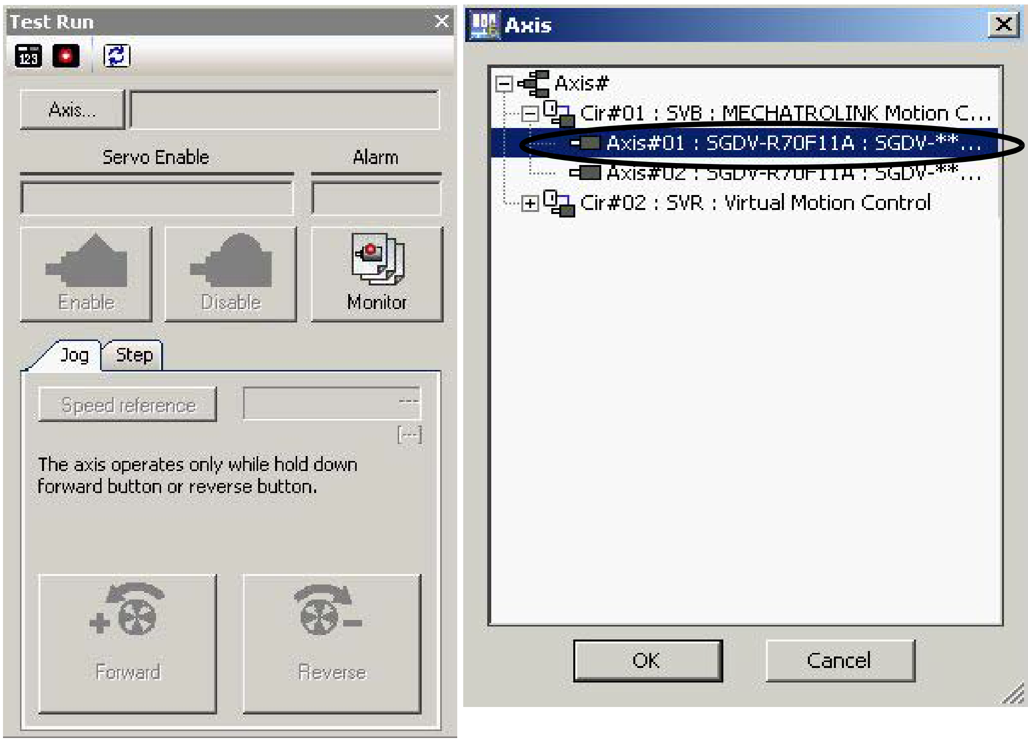


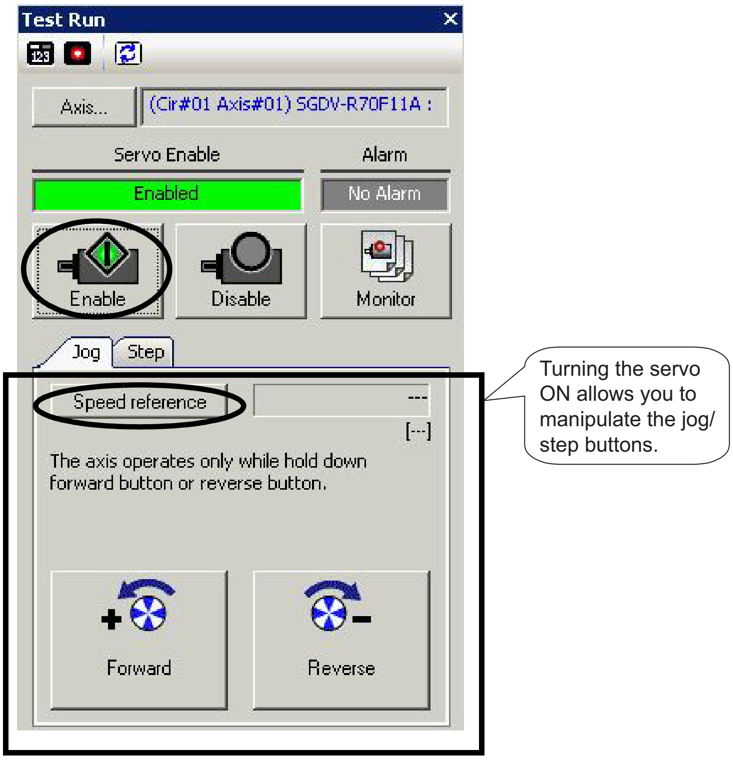
1. Click **System** in the subwindow and double-click ***Axis configuration - Test run*** to display a warning dialog a box for the test run. Click the **Agree** Button.



2. Axis Selection and Servo ON

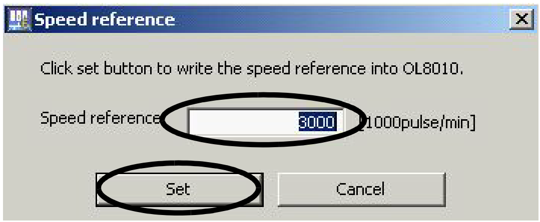
Set an axis number in the **Axis** Window and click the **Enable** (Servo ON) Button in the **Test Run** Window.

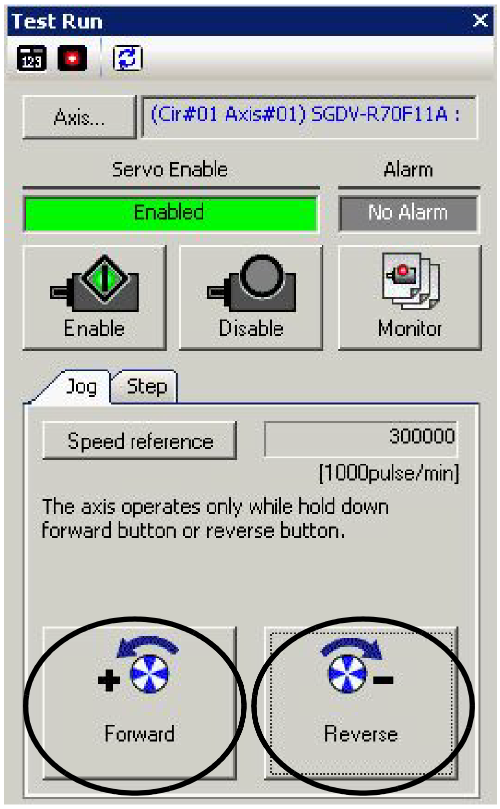




3. Jog Operation

Click the **Speed reference** Icon and set a speed reference value, and check that the axis rotates normally while the **Forward** Button or **Reverse** Button is pressed.





The operation check of the first axis is complete.

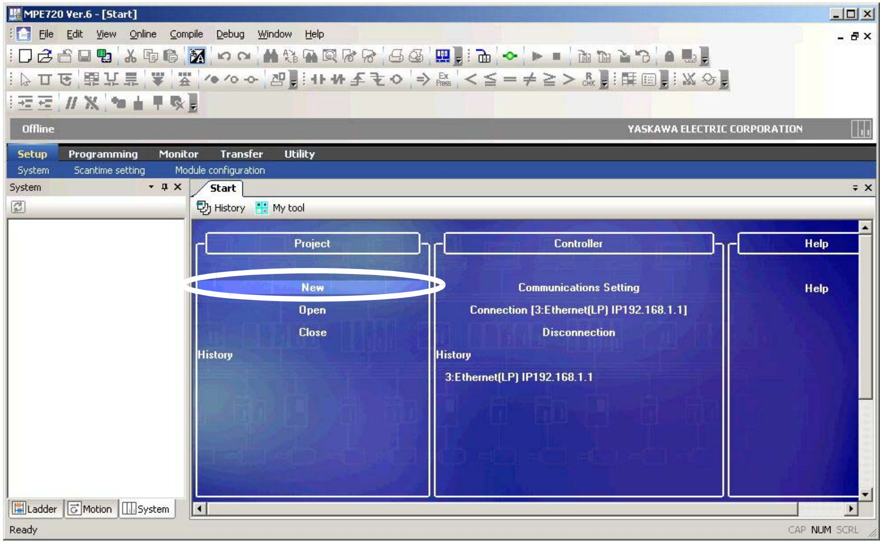
Press the **Axis ...** Button to change to “Axis #02” in the **Axis** Window, and perform the steps 1 to 3 above.

4.3 Programming

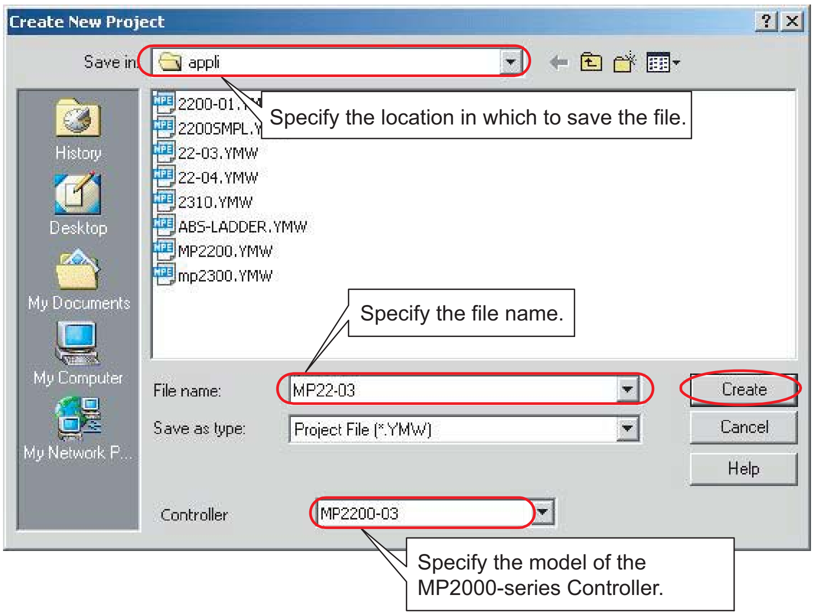
4.3.1 Creating a Project

1. Double-click the icon on the personal computer desktop to start MPE720 version 6.

2. Once MPE720 version 6 has started, click **New**.



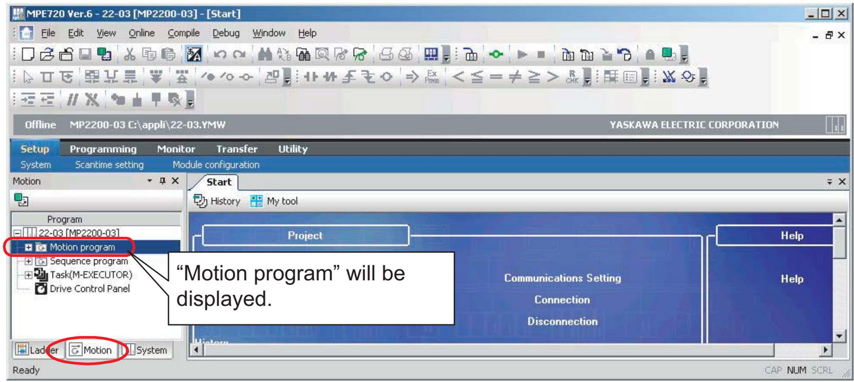
3. Specify the file name, location in which to save the file, and Controller model, and then click the **Create** Button.



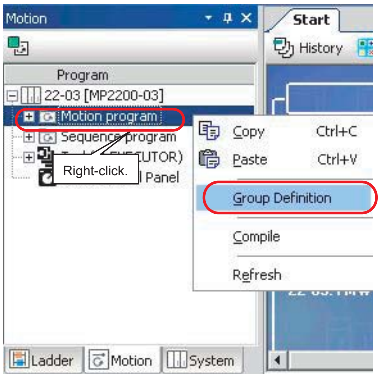
4.3.2 Creating Group Definitions

Before creating the motion program, organize the axes into groups according to the machine configuration.

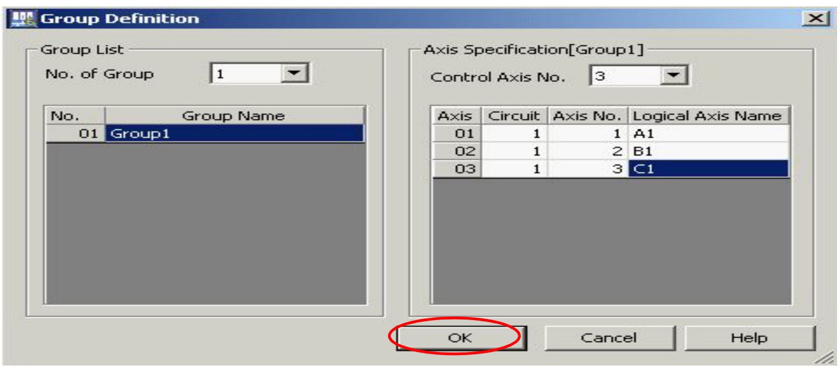
1. Click the **Motion** Tab in the subwindow to display **Motion program** in the subwindow.



2. Right-click **Motion program** in the left subwindow, and then select ***Group Definition*** from the pop-up menu.



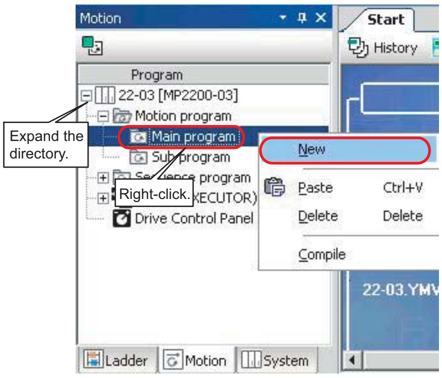
3. Click the **OK** Button. For more information, refer to *7.3 Group Definition* in the *Machine Controller MP2000 Series Users Manual Motion Programming* (manual no.: SIEP C880700 38).



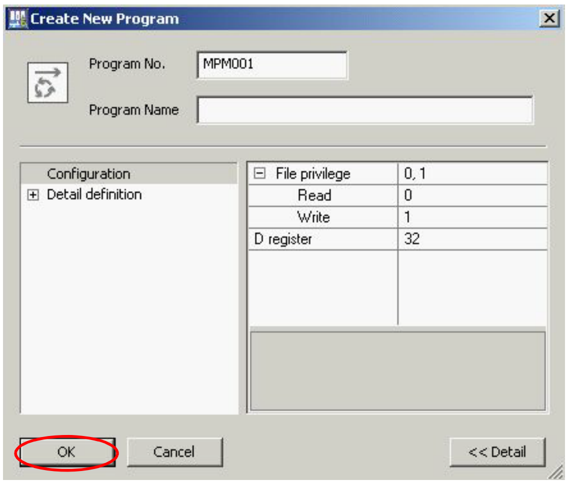
4.3.3 Creating Motion Programs

Start the Motion Editor to enter the motion program.

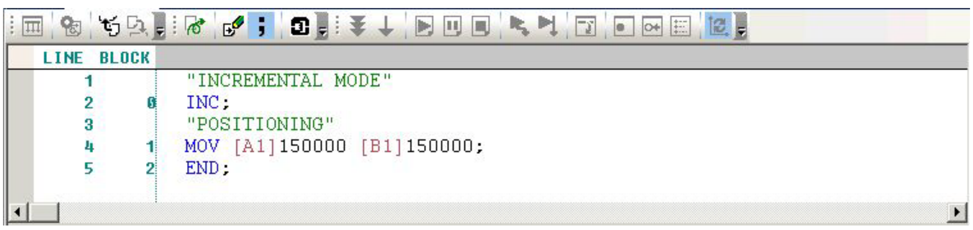
1. The motion program tree will expand in the subwindow. Right-click **Main program**, and then click **New** in the pop-up menu.



2. Click the **OK** Button.



3. Enter the motion program example given at the beginning of this section.



4. Click the  Icon in the Motion Editor to perform compiling. Once the motion program has been compiled, it will be automatically saved.

■ Caution

If the **Error List** Dialog Box is displayed when the motion program is being compiled, the program will not be saved.

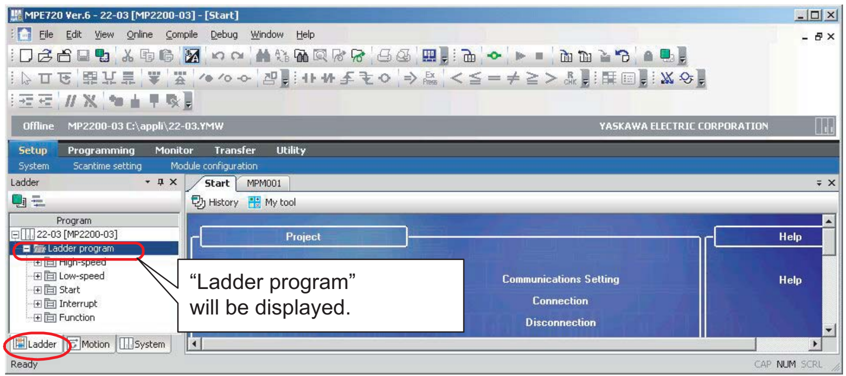
System Startup

4

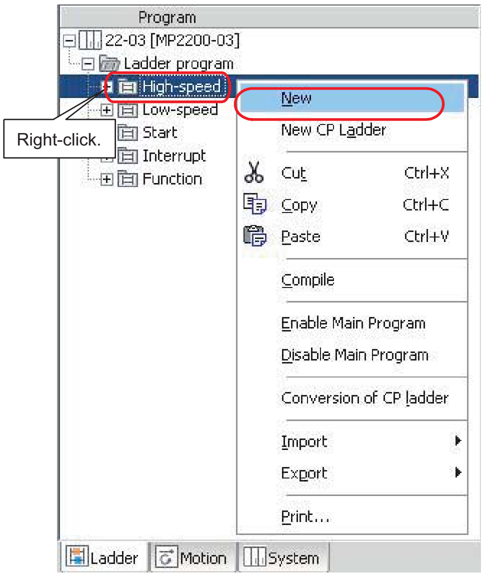
4.3.4 Registering a Motion Program for Execution

Call the motion program that you created from DWG.H by using an MSEE command. For details, refer to *5.2.3 (2) How to Run a Motion Program*.

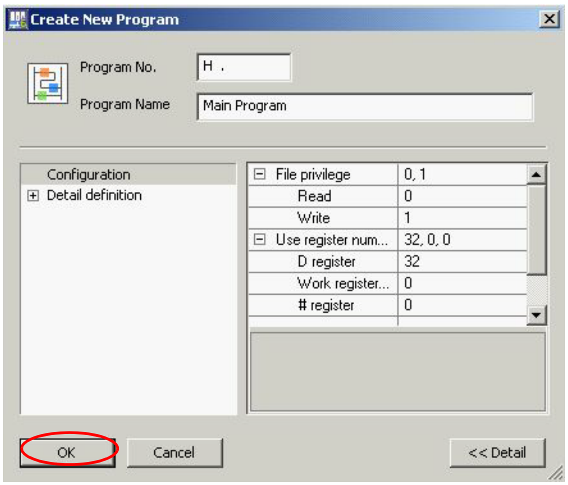
1. Click the **Ladder** Tab in the subwindow. **Ladder program** will be displayed in the subwindow.



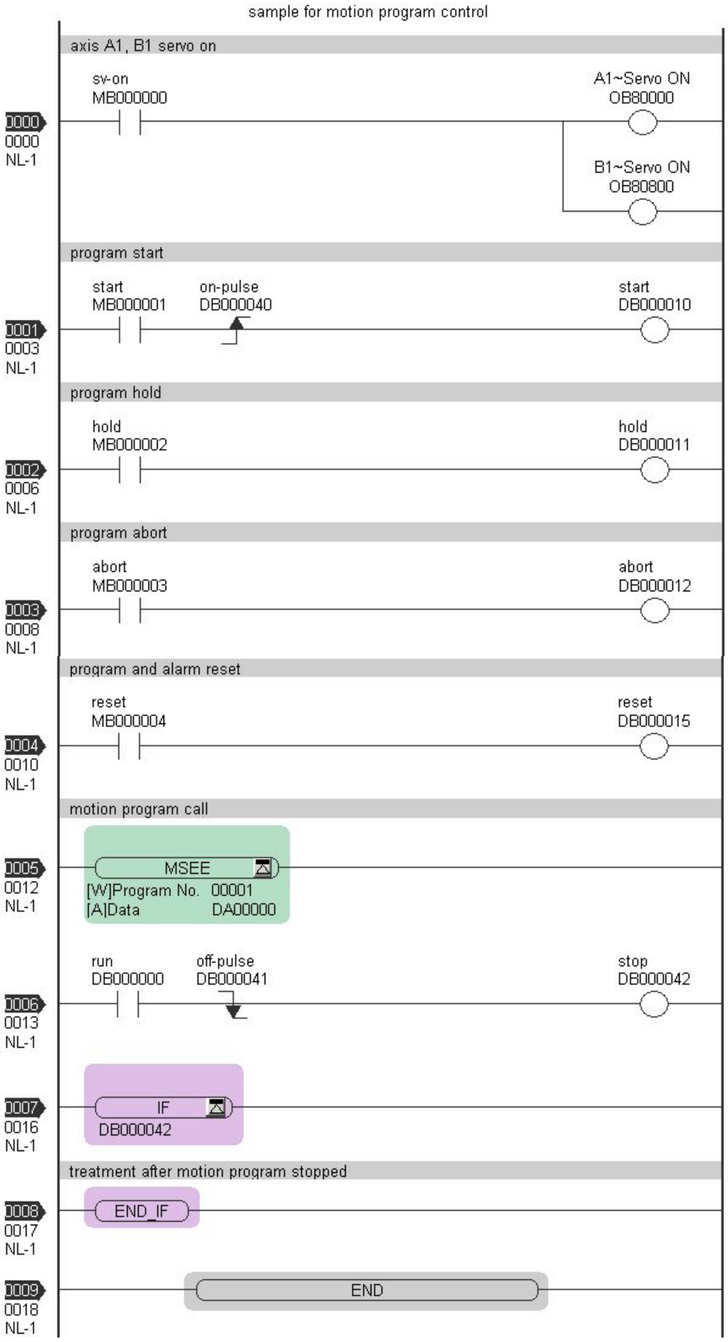
2. Right-click **High-speed** in the subwindow, and then click **New** in the pop-up menu.



3. Click the **OK** Button.



4. Create the following ladder program. After entering the program, compile it by pressing the F8 Key or clicking the  Icon.



■ Caution

- Confirm that Machine Controller Operation Ready (motion monitoring parameter IWxx00 bit 0) is ON, and then turn ON the Servo ON Command (MB000000).
- If Machine Controller Operation Ready is OFF, the Servo ON Command will not be accepted.

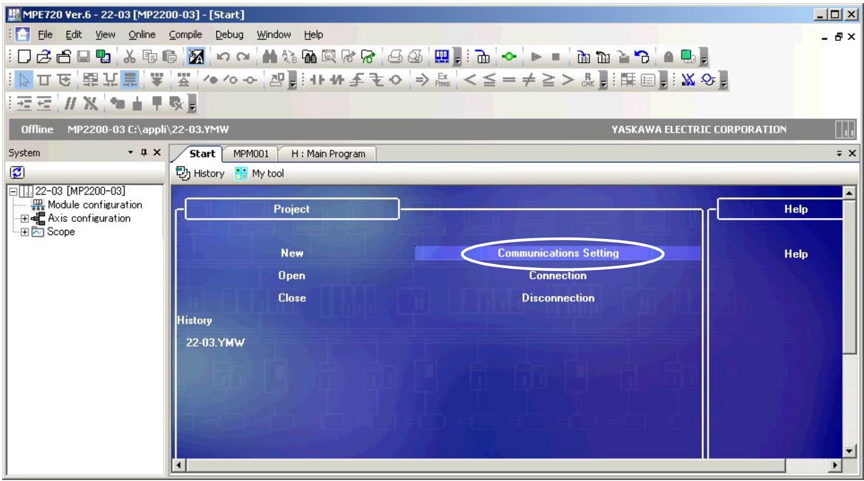
■ Complement

If you use the CPU-03 or CPU-04, you can register the program in the M-EXECUTOR program execution definitions rather than creating the ladder program described above. For information on the registration procedure, refer to *7.4.1 Registering Program Execution*.

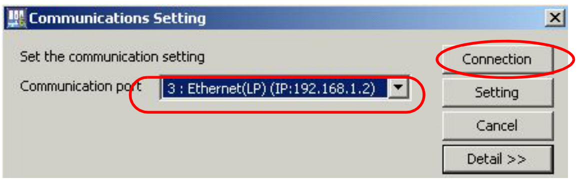
4.3.5 Transferring a Program

Transfer the motion program to the MP2000-series Controller. This procedure is not required if the motion program is made online.

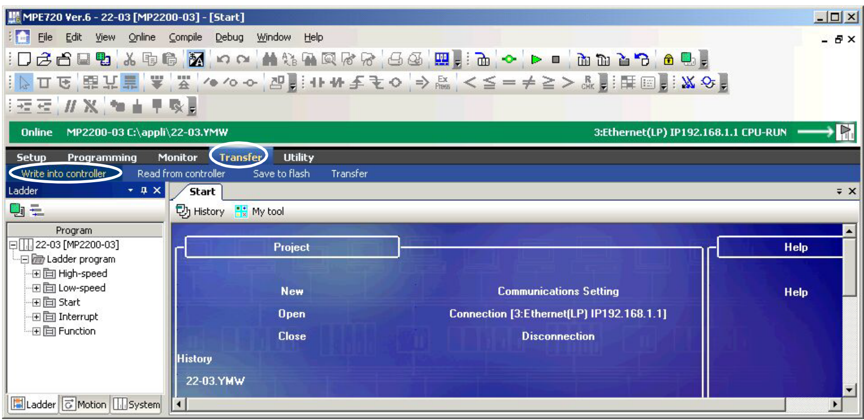
1. Click ***Communications Setting*** in the following window.



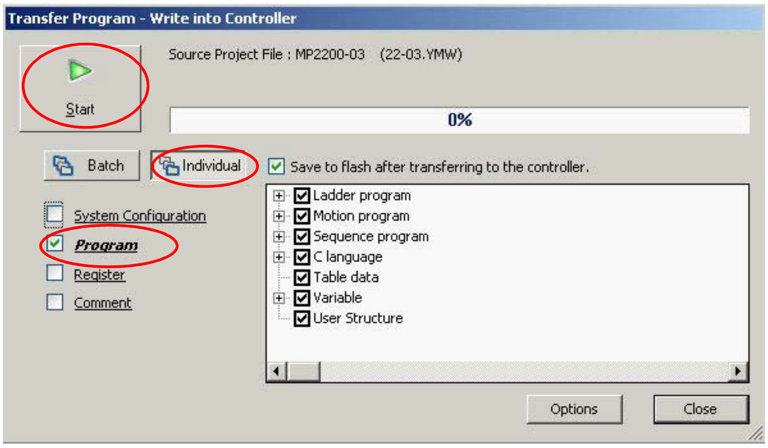
2. Select the communications port that was set in *4.2 Preparation*, and then click the **Connection** Button.



3. After you are online, click ***Transfer - Write into controller.***



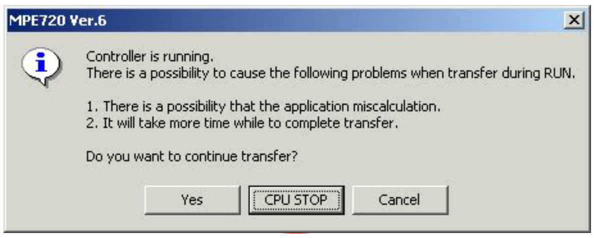
4. Click the **Individual** Button, and then select **Program** Check Box. Click the **Start** Button.



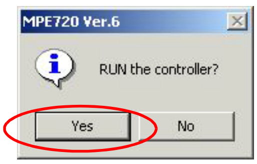
■ Caution

- If an individual transfer is performed, the same file in the Controller will be overwritten with the selected project file data.
- If batch transfer is performed, the RAM of the MP2000-series Controller will be cleared before transfer, and then the project file data will be batch-written.

5. Click the **CPU STOP** Button. The transfer will start.



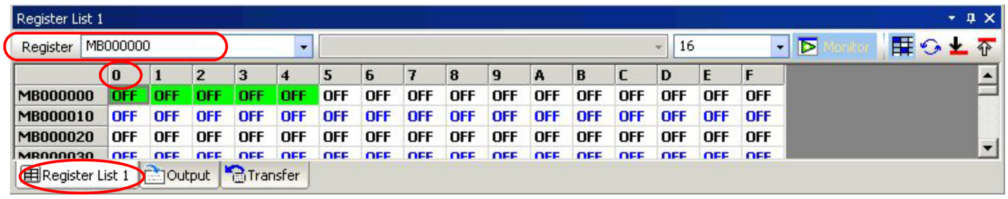
6. Select the **Yes** Button in the following dialog box to start the Controller.



4.3.6 Program Debugging

Debug the program that you created. For details on debugging, refer to *9.4 Debugging Operation in the Machine Controller MP2000 Series User's Manual Motion Programming* (manual no: SIEP C880700 38).

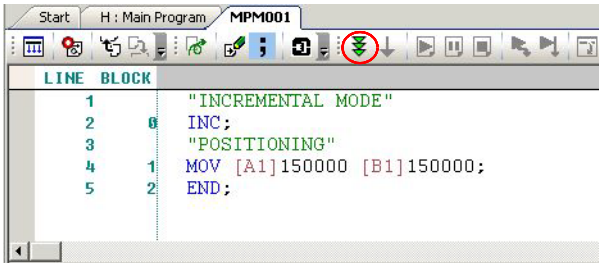
1. Double-click the **Register List 1** Tab to display the register list. Specify MB000000 for the register. Set MB000000 to ON as shown in the following figure to turn ON the servo.



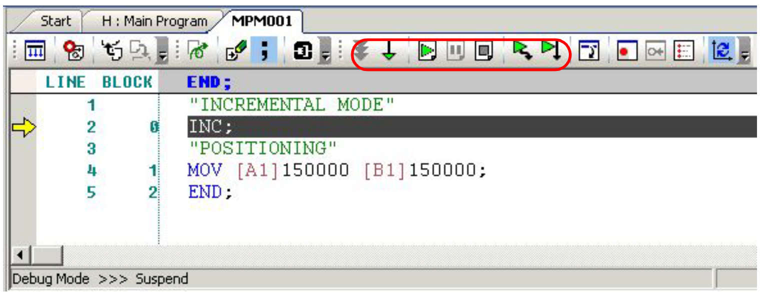
■ Complement

The CPU-03 and CPU-04 turn ON the servo by directly using the motion setting parameters if the M-EXECUTOR was used in the procedure given in *4.3.4 Registering a Motion Program for Execution*.

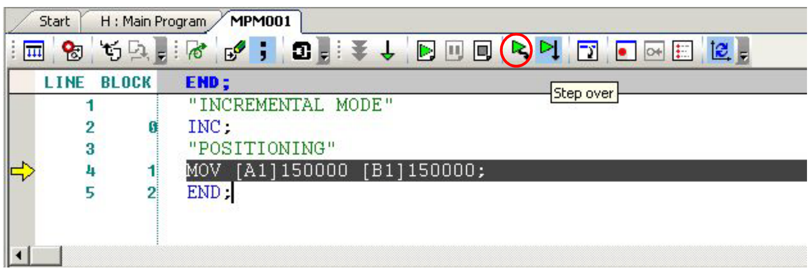
2. Click the (Debug Mode) Icon.



3. Debug Mode will be entered.



4. Click the (Step In) Icon to check program operation by executing program lines one at a time.

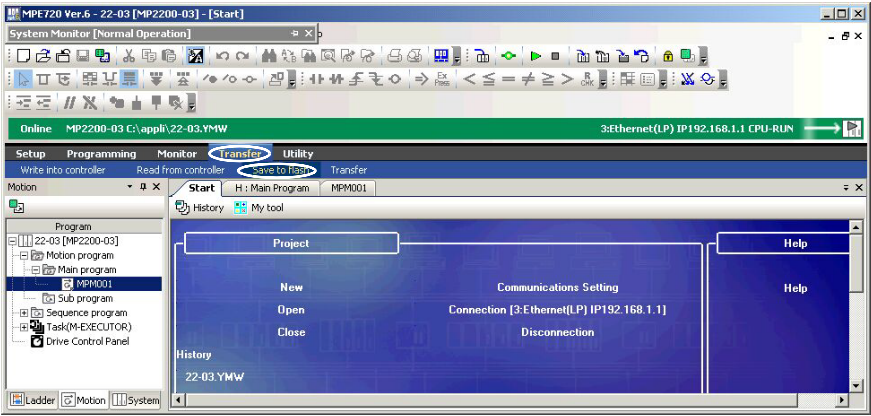


5. The program will be executed until the END command, and the servo will turn OFF after the debugging operation has been completed.

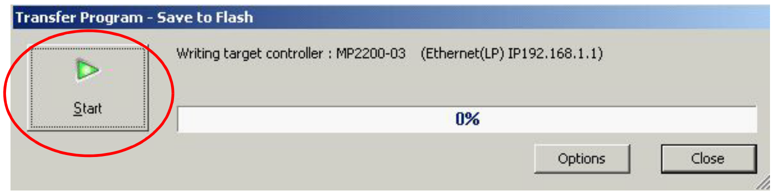
4.3.7 Saving a Program to Flash Memory

Use the following procedure to save the data in the RAM in the MP2000-series Controller to the flash memory in the MP2000-series Controller.

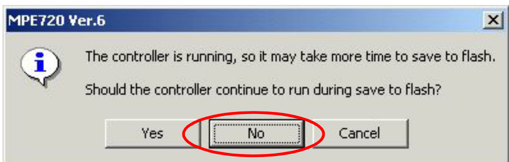
1. Click ***Transfer - Save to flash*** in the following window.



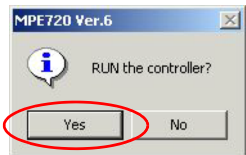
2. Click the **Start** Button.



3. Click the **No** Button. The transfer will start.



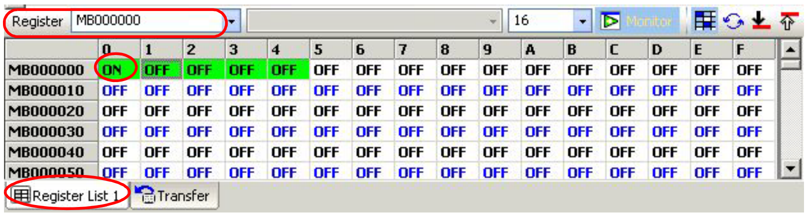
4. Select the **Yes** Button in the following dialog box to start the Controller.



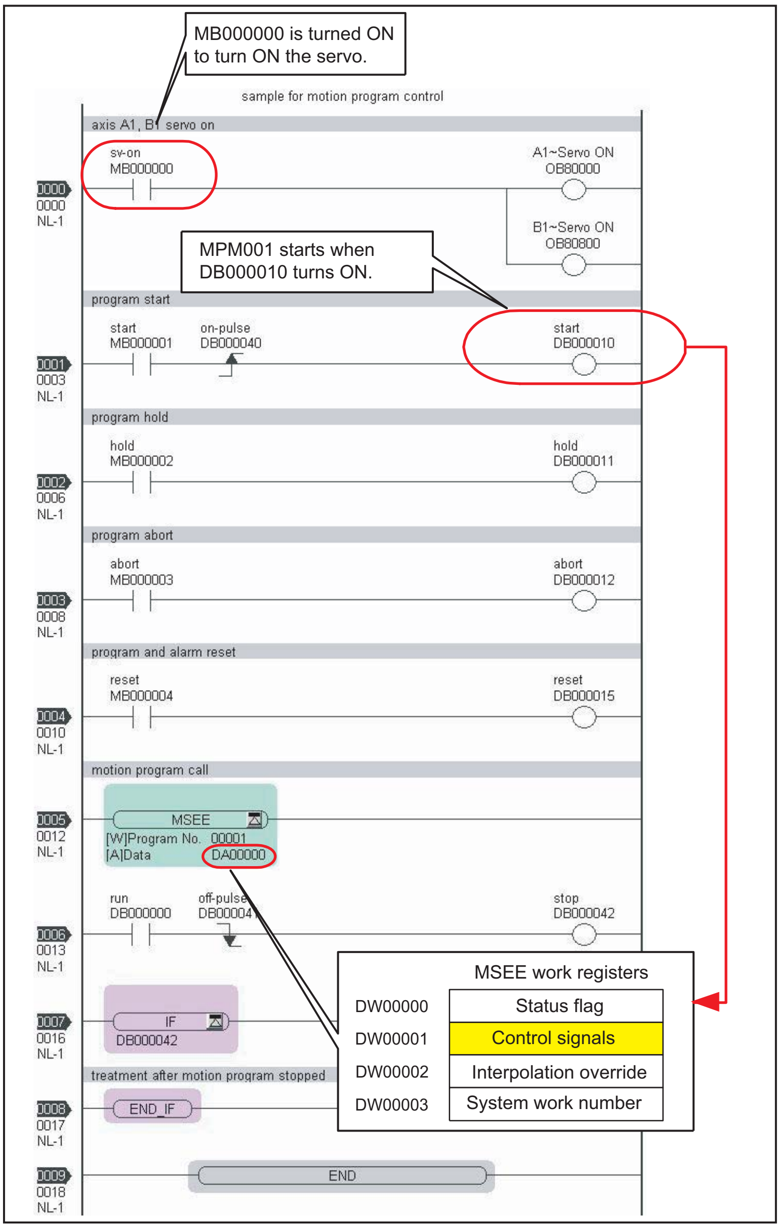
4.3.8 Executing Motions

Execute the program that you created using the actual system. To execute the motion program, turn ON a program operation start request by using the control signal.

1. Double-click the **Register List 1** Tab to display the register list. Specify MB000000 for the register. Set MB000000 to ON as shown in the following figure to turn ON the servo.



2. Set MB000001 to ON as shown in the following figure to execute motion program MPM001.



5

Overview of System Operation

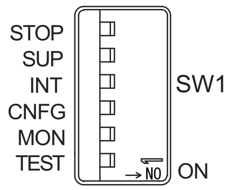
This chapter describes the basic operation of MP2200 Machine Control Systems and provides an outline of user programs and registers.

5.1 Startup Sequence and Basic Operation

This section describes the MP2200 startup sequence and basic operation together with the DIP switch settings, self-diagnosis at startup, and LED indicator patterns.

5.1.1 DIP Switch Settings

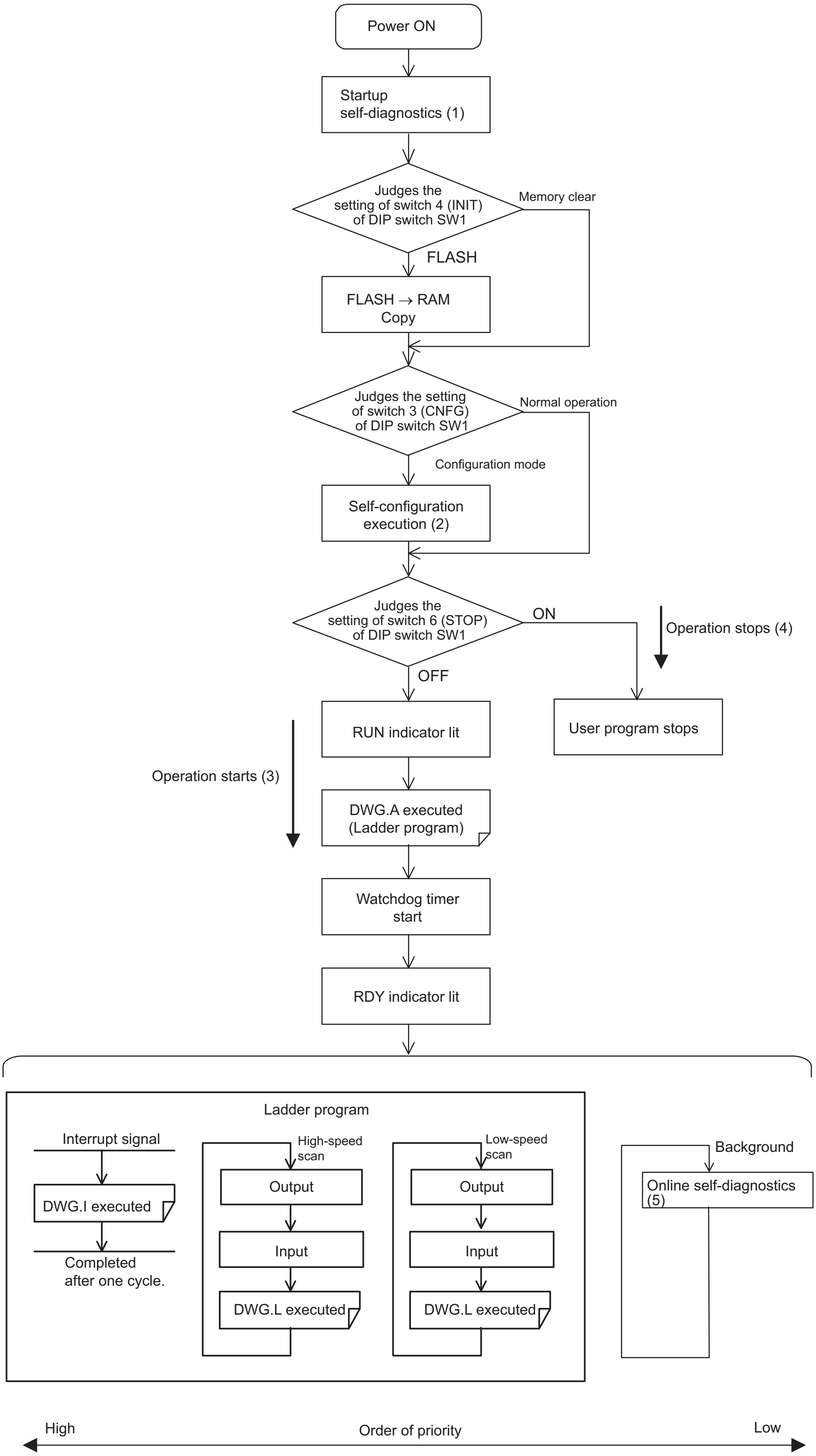
Set the DIP switch (SW1) on the Basic Module to control operations of the startup sequence. The six switches are provided on the DIP switch (SW1) on the Basic Module as shown in the following figure. The following table lists the functions of six switches.



No.	Switch Name	Status	Operating Mode	Default Setting	Remarks
S1-6	STOP	ON	User program stops	OFF	Set to ON to stop user program operation.
		OFF	User program operation		
S1-5	SUP	ON	System load	OFF	If set to ON, starts up in the mode that can renew the version of the farmware.
		OFF	Normal operation		
S1-4	INIT	ON	Memory clear	OFF	Set to ON to clear memory. Programs stored in flash memory will be run when this switch is set to OFF.
		OFF	Normal operation		
S1-3	CNFG	ON	Self-configuration mode	OFF	Set to ON for self-configuration of connected devices.
		OFF	Normal operation		
S1-2	MON	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
S1-1	TEST	ON	System use (Adjusted before shipment)	OFF	Always set to OFF.
		OFF	Normal operation		

5.1.2 Startup Sequence

The startup sequence for the MP2200 from the moment when the power has been turned ON is shown in the following flowchart.



Note: Refer to *5.1.3 Startup Sequence Operation Details* on the next page for details on (1) to (5).

5.1.3 Startup Sequence Operation Details

(1) Self-diagnosis at Startup

Self-diagnosis is performed on the following items after the power is turned ON.

- Read/write diagnosis of memory (RAM)
- System program (ROM) diagnosis
- Main processor (CPU) function diagnosis
- Floating Point Unit (FPU) function diagnosis

If diagnosis results in an error, the ALM and ERR LED indicators will blink red for the specified number of times. Refer to *5.1.4 LED Indicator Details*.

(2) Self-configuration

Self-configuration automatically recognizes the connected Optional Modules, and automatically creates a definitions file. For details, refer to *5.4 Self-configuration*.
The RUN LED indicator will blink green during execution of self-configuration.

(3) Operation Start

When the STOP switch is set to OFF (RUN), the CPU will execute the ladder program in DWG.A. (Refer to *5.2.2 Ladder Program* for information on DWG.A, the startup processing drawing.)
First scan processing is executed once DWG.A has been completed and the high-speed or low-speed scan time has elapsed. System I/O are executed from the first scan.

(4) Operation Stop

MP2200 stops motion control operation when the STOP switch is ON (STOP) and in the following circumstances.

Cause	Restart method
Power supply turned OFF	Turn ON the power again.
Power interruption	
Fatal error	Check the LED indicator for the cause of the error and then turn the power OFF then ON.
STOP executed from MPE720	Execute RUN from MPE720.

(5) Online Self-diagnosis

Self-diagnosis is performed on the following items during scan processing.

- System program (ROM) diagnosis
- Read/write diagnosis of memory (RAM)

If diagnosis results in an error, the ERR indicators will blink red. (Refer to the next page.)

5.1.4 LED Indicator Details

The MP2200 performs a variety of diagnostics at startup. If an error is found, the ERR LED indicator blinks red. The number of times the indicators blink differs depending on the error details, so error details can be determined from counting the number of blinks.The following table shows details of MP2200 LED indicator.

Note: 1. MPE720 cannot be operated when the indicators are blinking.
2. For information on errors and countermeasures, refer to *Chapter 8 Maintenance, Inspection, and Troubleshooting*.

Type	LED Indicator Name					Indicator Details	Remarks
	RDY	RUN	ALM	ERR	BAT		
Normal	Not lit	Not lit	Lit	Lit	Not lit	Hardware reset status	–
	Not lit	Not lit	Not lit	Not lit	Not lit	Initializing	
	Not lit	Lit	Not lit	Not lit	Not lit	Executing DWG.A	
	Lit	Not lit	Not lit	Not lit	Not lit	User program stopped (Offline stop mode)	User program stops when the DIP switch or MPE720 is used to execute the STOP operation.
	Lit	Lit	Not lit	Not lit	Not lit	User program executing normally (Online operation mode)	–
Error	Not lit	Not lit	Not lit	Lit	Not lit	Major damage has occurred	The ERR LED indicator is lit red when the CPU is down.
	Not lit	Not lit	Not lit	Blinking	Not lit	(Software error) No. of blinks 3: Address error (read) exception 4: Address error (write) exception 5: FPU exception 6: Illegal general command error 7: Illegal slot command error 8: General FPU inhibited error 9: Slot FPU inhibited error 10: TLB duplicated bit error 11: TLB mistake error (read) 12: TLB mistake error (write) 13: TLB protection violation error (read) 14: TLB protection violation error (write) 15: Initial page write error	The ERR LED indicator will blink red when an exception error has occurred.
	Not lit	Not lit	Blinking	Blinking	Not lit	(Hardware errors) No. of blinks 2: RAM diagnosis error 3: ROM diagnosis error 4: CPU function diagnosis error 5: FPU function diagnosis error	The ALM and ERR LED indicators will blink red if there is a self-diagnosis failure.
Alarm	–	–	–	–	Lit	Battery alarm	The BAT LED indicator will be lit when the battery voltage drops.
	Lit	Not lit	Lit	Not lit	Not lit	Operation error I/O error	The ALM LED indicator will be lit red when an operation or I/O error is detected.

5.2 User Programs

This section describes the basic operation and other information about user programs.

- For programming details, refer to the following manuals.
  - *Machine Controller MP900/MP2000 Series User's Manual Ladder Programming (manual no.: SIEZ-C887-1.2)*
  - *Machine Controller MP2000 Series User's Manual Motion Programming (manual no.: SIEP C880700 38)*
  - *Machine Controller MP900/MP2000 Series New Ladder Editor User's Manual Programming Instructions (manual no.: SIEZ-C887-13.1)*
  - *Machine Controller MP900/MP2000 Series New Ladder Editor User's Manual Operation (manual no.: SIEZ-C887-13.2)*
  - *Engineering Tool for MP2000 Series Machine Controller MPE720 Version 6 User's Manual (manual no.: SIEP C880700 30).*

5.2.1 Types and Execution Timing of User Program

The following table shows the types and execution timing of MP2200 user program.

User Program		Execution Timing
Motion Program	High-speed Scan Process	Turns ON the program operation start request of the control signal (runs when program operation start request is ON)
Sequence Program*	Startup Process	Power-up (during power-up, runs only once)
	High-speed Scan Process	Periodical startup (runs each time a high-speed scan is performed)
	Low-speed Scan Process	Periodical startup (runs each time a low-speed scan is performed)
Ladder Program	Startup Process	Power-up (during power-up, runs only once)
	Interrupt Process	Runs on external interrupt (runs on DI interrupt of Optional Module and counter match interrupt)
	High-speed Scan Process	Periodical startup (runs each time a high-speed scan is performed)
	Low-speed Scan Process	Periodical startup (runs each time a low-speed scan is performed)

* Sequence program is supported by the CPU-03 at CPU-04.

For more information on the user program, refer to the next page and after.

5.2.2 Ladder Program

(1) Types of Drawings (DWG)

Ladder programs are managed in units of ladder drawings, which are identified by drawing numbers. These drawings form the basis of user programs.
Ladder drawings include parent drawings, child drawings, grandchild drawings, and operation error processing drawings. In addition to drawings, there are functions that can be freely accessed from each drawing.

- Parent Drawings
  Parent drawings are automatically executed by the system program when the execution conditions, outlined in the table below, are met.
- Child Drawings
  Child drawings are accessed using a SEE command from a parent drawing.
- Grandchild Drawings
  Grandchild drawings are accessed using a SEE command from a child drawing.
- Operation Error Processing Drawings
  Operation error processing drawings are automatically executed by the system program when an operation error occurs.
- Functions
  Functions are accessed and executed from parent, child, and grandchild drawings using the FUNC command.

[a] Drawing Types and Order of Priority

Drawings are classified by their first letter (A, I, H, or L) based on the processing purpose. The following table outlines the order of priority and execution conditions for these drawings.

Type of Parent Drawing	Function	Priority	Execution Conditions	Max. No. of Drawings
DWG.A (Drawing A)	Startup processing	1	Power ON (Executed once only, when power turns ON)	64
DWG.I (Drawing I)	Interrupt processing	2	External interrupt (executed by Optional Module DI interrupt or counter match interrupt)	64
DWG.H (Drawing H)	High-speed scan processing	3	Periodical startup (Executed each high-speed scan)	200
DWG.L (Drawing L)	Low-speed scan	4	Periodical startup (Executed each low-speed scan)	500

Note: Lower values have higher order of priority.

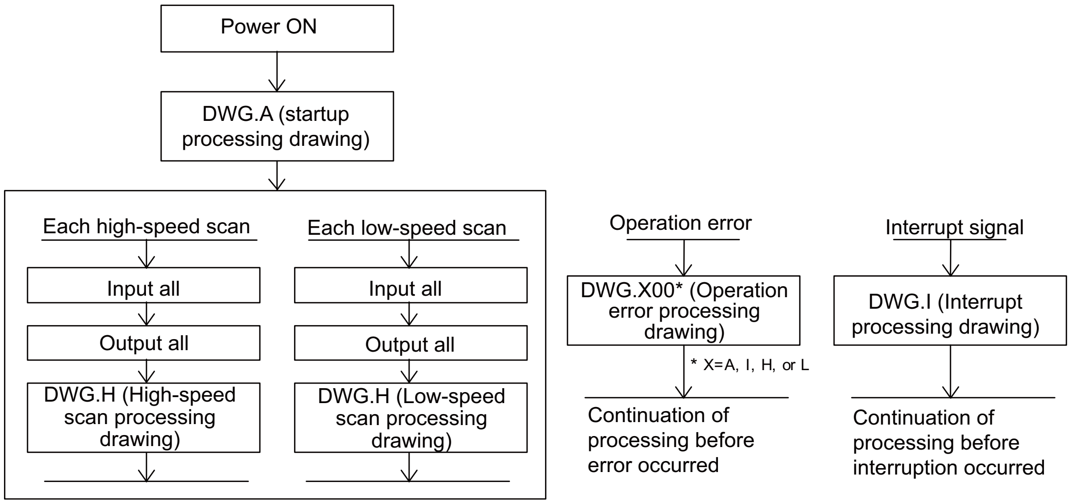
The following table provides details of the number of drawings for each drawing.

Drawing	Number of Drawings			
	DWG.A	DWG.I	DWG.H	DWG.L
Parent Drawings	1 (A)	1 (I)	1 (H)	1 (L)
Operation Error Processing Drawings	1 (A00)	1 (I00)	1 (H00)	1 (L00)
Child Drawings	Total: 62 max.	Total: 62 max.	Total: 198 max.	Total: 498 max.
Grandchild Drawings				

(2) Execution Control of Drawings

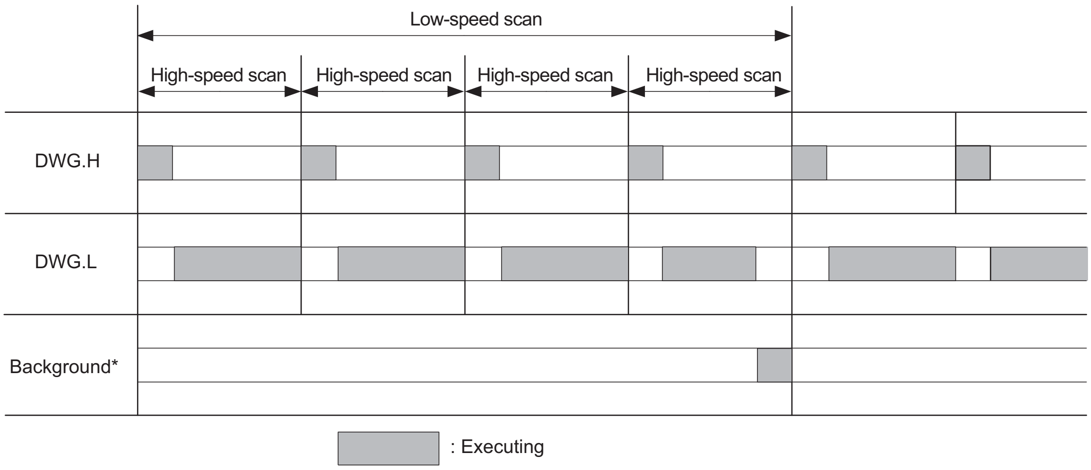
[a] Execution Control

The following table shows a flow chart of how each drawing is executed based on the order of priority.



[b] Execution Schedule for Scan Processing Drawings

The scan processing drawings are not executed simultaneously. As shown in the following figure, the execution of each drawing is scheduled based on the order of priority and time sharing.

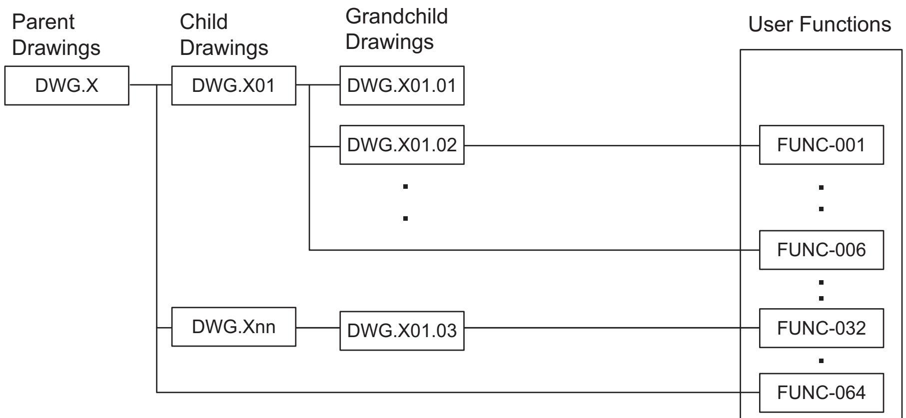


* Background processing is used to execute internal system processing, e.g., communication processing.

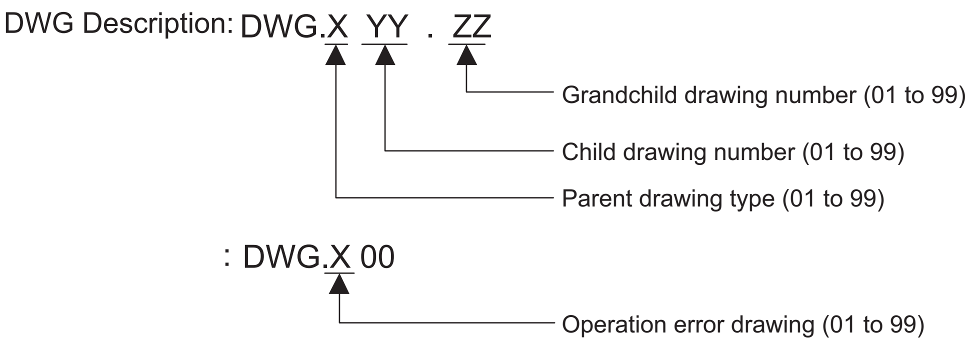
Low-speed scan processing is executed in spare processing time of the high-speed scan. Set the time of the high-speed scan to approximately double the total execution time for DWG.H.

[c] Hierarchical Structure of Drawings

Each processing program is made up of parent drawings, child drawings, and grandchild drawings. Parent drawings cannot call child drawings from a different type of drawing and child drawings cannot call grandchild drawings from a different type of drawing. Also, parent drawings cannot directly call grandchild drawings. Child drawings are always called from parent drawings and grandchild drawings are always called from child drawings. This is the hierarchical structure of drawings.
As shown in the following figure, each processing program is created from a hierarchy of parent, child, and grandchild drawings.

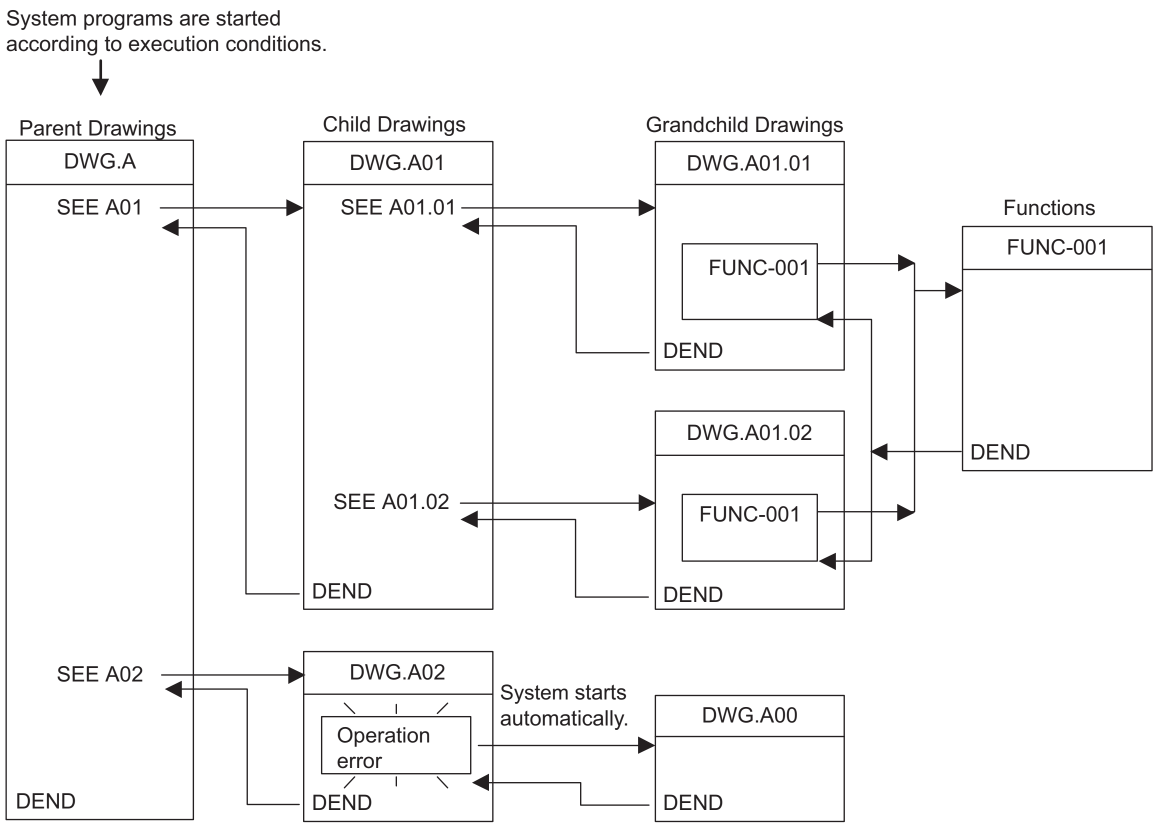


Note: X means A, I, H, or L.



[d] Drawing Execution Processing Method

The execution processing of hierarchical drawings are performed by calling lower-level drawings from higher-level drawings.
The following figure shows the execution processing for drawings, using DWG.A as an example.



Note: 1. Parent drawing is automatically called and executed by system. Therefore, you can execute the child and grandchild drawings by programming a DWG reference command (SEE command) in the parent and child drawings.
2. Functions can be referenced from any drawing. Functions can also be referenced from other functions.
3. When an operation error occurs, the operation error processing drawing for that drawing will be started.

(3) Functions

Functions are executed by calling them from parent, child, or grandchild drawings using the FUNC command. Functions can be called from any drawing, and the same function can be called at the same time from different types of drawings and from different levels of drawings. Another completed functions can also be called from functions.
Using functions has the following advantages.

- Easier creation of user program components
- Easier writing and maintenance of user programs

Functions include standard system functions that are already in the system and user functions that are defined by the user.

[a] Standard System Functions

The transmission and other functions listed below are already created as standard system functions. Standard system functions cannot be changed by users.

Type	Name	Symbol	Contents
System functions	Counter	COUNTER	Incremental/decremental counter
	First in/first out	FINFOUT	First in/first out
	Trace function	TRACE	Data trace execution control
	Data trace read	DTRC-RD	Reads data from data trace memory to user memory
	Inverter trace read function	ITRC-RD	Reads trace data from inverter trace memory to user memory
	Message send	MSG-SND	Sends messages to external communication devices
	Message receive	MSG-RCV	Receives messages from external communication devices

[b] User Functions

The functions (programs) and the function definitions can be changed (programmed) freely by users.
The maximum number of user functions that can be defined is 500 drawings.

- Refer to the following manual for information on defining functions.
  - *Machine Controller MP900/MP2000 Series User's Manual Ladder Programming (manual no.: SIEZ-C887-1.2)*
  - *Machine Controller MP2000 Series User's Manual Motion Programming (manual no.: SIEP C880700 38)*
  - *Machine Controller MP900/MP2000 Series New Ladder Editor User's Manual Programming Instructions (manual no.: SIEZ-C887-13.1)*
  - *Machine Controller MP900/MP2000 Series New Ladder Editor User's Manual Operation (manual no.: SIEZ-C887-13.2)*
  - *Engineering Tool for MP2000 Series Machine Controller MPE720 Version 6 User's Manual (manual no.: SIEP C880700 30).*

5.2.3 Motion Programs

Motion programs are programs written in a text-based language called motion language.
The following table shows the two types of motion programs.

Type	Specification Method	Features	No. of Programs
Main Program	MPM□□□ (□□□ = 1 to 256)	• Accessed from DWG.H • Accessed from M-EXECUTOR Program Execution Definitions*1	Up to 256 programs including the following programs can be created. • Motion main programs • Motion subprograms • Sequence main programs* • Sequence subprograms*
Subprogram	MPS□□□ (□□□ = 1 to 256)	Accessed from main programs	

* Sequence program is supported by the CPU-03 at CPU-04.

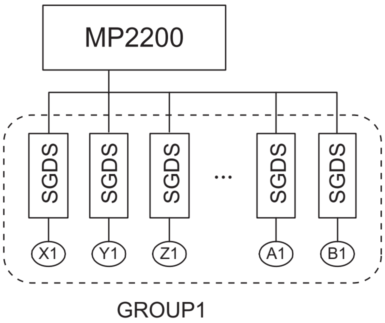
- The program numbers of motion programs are managed in the same manner as the sequence program numbers. (See note.) Assign a unique number for each program number.
  - Program number of Motion program MPM □□□, MPS □□□
  - Program number of Sequence program SPM □□□, SPS □□□ (See note 1.)

  Note: Sequence program is supported by the CPU-03 at CPU-04.
- The MP2200 can execute up to 16 motion programs simultaneously. An alarm (no system work error) will occur if 17 or more programs are executed simultaneously.
  - No system work error: Bit E of the leading word in the MSEE work registers

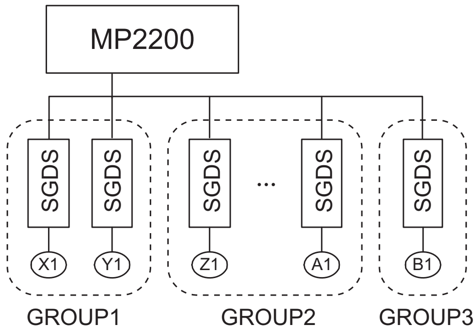
(1) Groups

A group of axes with related operations can be treated as one group by motion programs and programs can be executed for each group. This allows one MP2200 to independently control multiple machines using group operation. Group operation can be single group operation or multiple group operation.
Definitions for axes to be grouped together are made under *Group Definitions*.

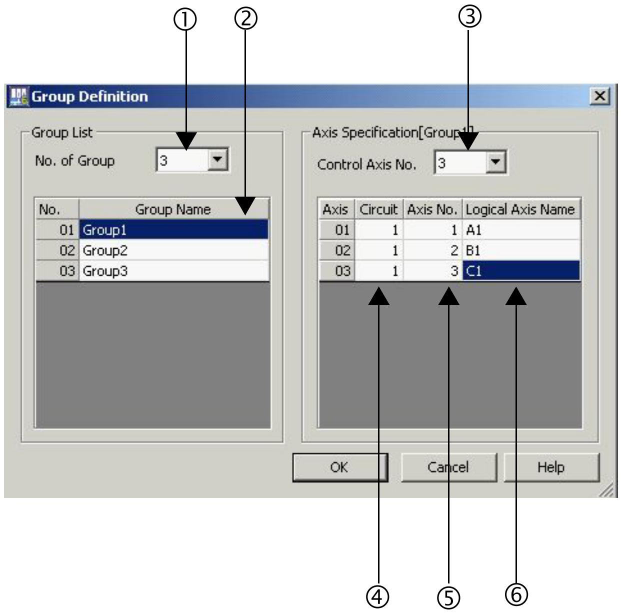
(a) Single Group Operation



(b) Multiple Group Operation



This section explains the **Group Definition** window.



① No. of Group

Set a number for the group operation.

Set it to 1 for the single group operation.

Set it to the number of group operation for the multiple group operation.

② Group Name

Define a group name.

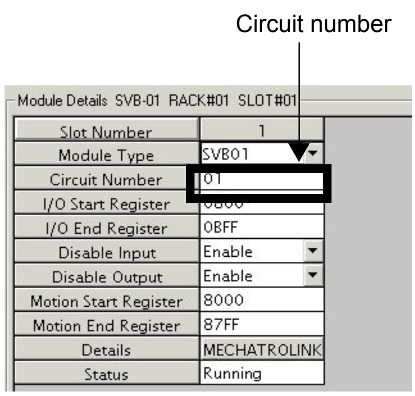
③ Control Axis No.

Set the number of axes controlled in the group.

④ Circuit

Set a circuit number for the used motion module.

The circuit number can be checked in the **Module Configuration** Window.



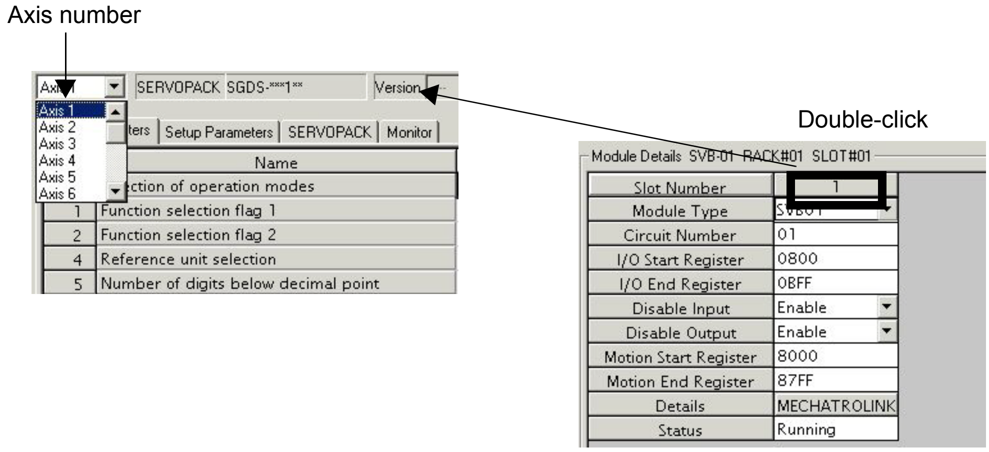
Overview of System Operation

5

⑤ Axis No.

Set an axis number for the used axis.

The axis number can be checked in the detailed window of the used motion module.



⑥ Logical Axis Name

Define a name for the specified axis number.

The name defined here is used when programming a motion program.

MVS [A1]1000 [B1]2000 [C1]3000 F1000;

Logical axis name

(2) How to Run a Motion Program

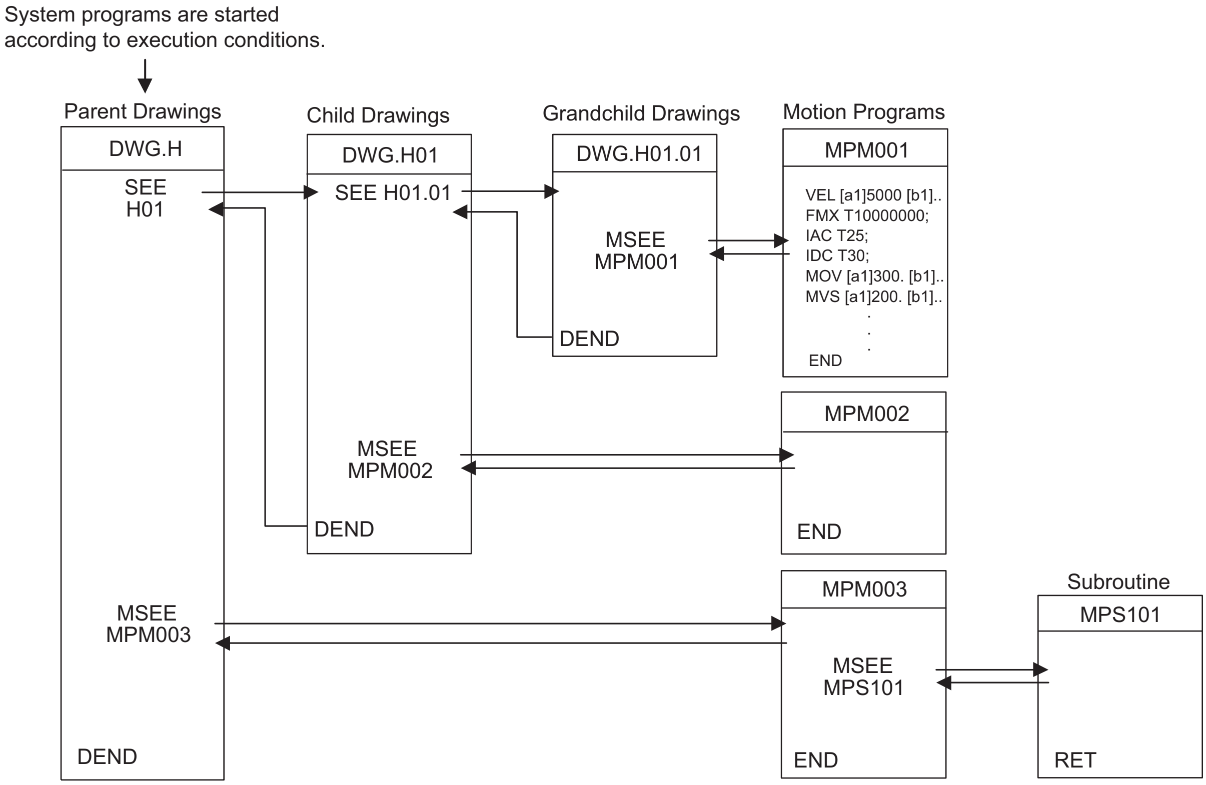
The following two methods are available for running a motion program.

- Executing programs using a MSEE command from a ladder program of H drawing
- Registering programs in the **Program definition** Tab Page of the **M-EXECUTOR** Window (Supported by CPU-03 and CPU-04.)

Now, this section explains each way to run a motion program:

[a] Executing H Drawings by MSEE Command

Motion programs are always called from H drawings using the MSEE command (motion program call command). Motion programs can be called from any parent, child, or grandchild drawing in an H drawing.
The following figure shows an example of motion program execution.



H drawing ladder commands are executed in hierarchical order i.e., parent drawings, child drawings, then grandchild drawings each high-speed scan cycle.
The above method is a preparation for running a motion program. When a MSEE command is built in, the motion program does not start up. To start up the motion program, after the MSEE command is incorporated, use a control signal to turn on the request for the program operation startup.
Motion programs are also called each scan cycle, but unlike ladder programs, all motion programs cannot be executed in one scan. For this reason, motion programs are executed and controlled by special system's motion management function.

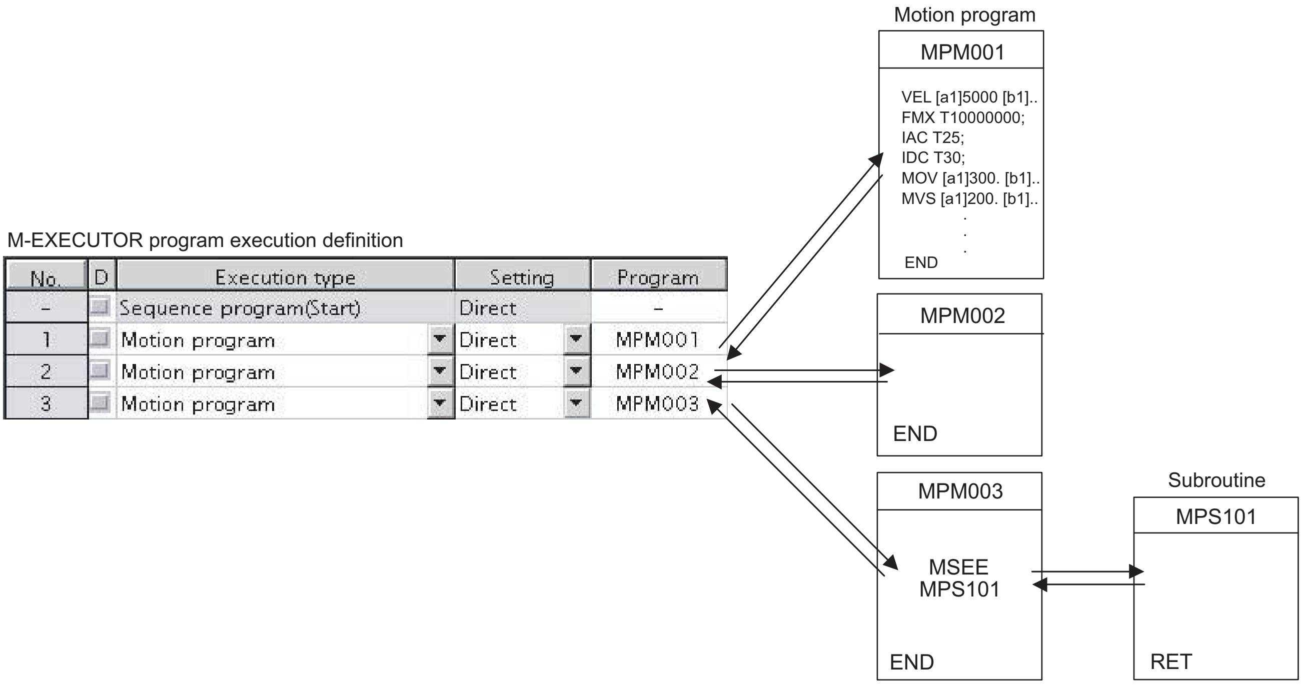
■ Caution

When running a motion program, pay attention to the followings:

- Multiple motion programs with the same number cannot be executed using a MSEE command.
- A subroutine (MPS □□□) cannot be executed from a MSEE command in a ladder.
  It can only be referenced from a motion program (MPM □□□, MPS □□□).
- The same subroutine cannot be referenced at the same time.
- A motion program registered in M-EXECUTOR cannot be executed using a MSEE command (CPU-03/CPU-04).
- Sequence programs (SPM□□□ and SPS□□□) cannot be executed using a MSEE command (CPU-03/CPU-04).

[b] Registering Programs in the M-EXECUTOR Program Execution Definitions (Supported by CPU-03 and CPU-04.)

After creating a motion program, register it in the **Program definition** Tab Page of the **M-EXECUTOR** Window.
The programs registered in the **Program definition** Tab Page of the **M-EXECUTOR** Window are executed in ascending numeric order.
The execution example is shown in the figure below.



The above method is a preparation for running a motion program. When registered in the M-EXECUTOR program execution definition, a motion program does not start up. To start up the motion program, after the motion program registration, use a control signal to turn ON the request for the program operation startup.
The motion program registered in M-EXECUTOR is executed at a scan cycle, but similar to a ladder, the whole program cannot be executed at a single scan. In case of the motion program, a motion management function in the system carries out an execution control exclusive for the motion programs.

■ Caution

When registering a motion program to M-EXECUTOR, pay attention to the followings:
- Multiple motion programs with the same number cannot be registered.
- Multiple motion programs with the same number cannot be referenced using an indirect designation.

(3) How to Designate a Motion Program

The following two methods are available for designating a motion program.

- Using a direct designation to call a motion program
- Using a indirect designation to call a motion program

Now, this section explains each way to designate a motion program.

[a] Using a Direct Designation to Call a Motion Program

A direct designation method designates a motion program to call using a program number (MPM □□□).

■ Motion Program Referenced by a MSEE Command from a Ladder Program

Set a program number to Program No. (□□□□□) in the MSEE command.

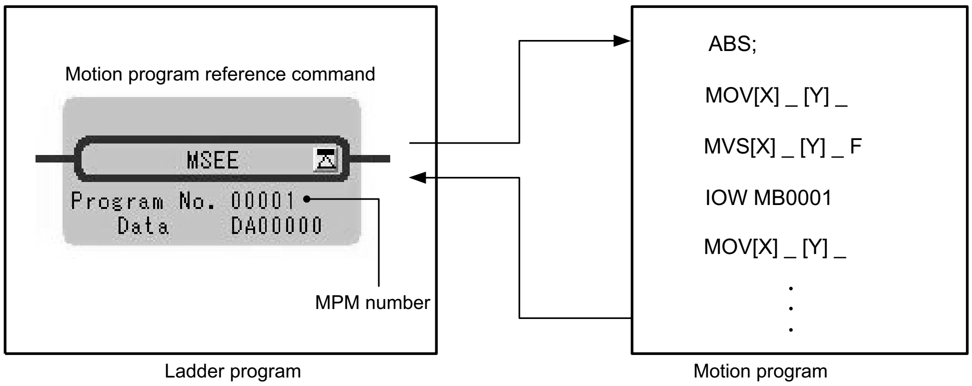


Fig. 5.1 Calling a Motion Program Using a Direct Designation -1

■ Motion Program Registered in M-EXECUTOR Program Execution Definitions (Supported by CPU-03 and CPU-04.)

Select ***Direct*** for the Setting and set a program number (MPM □□□).

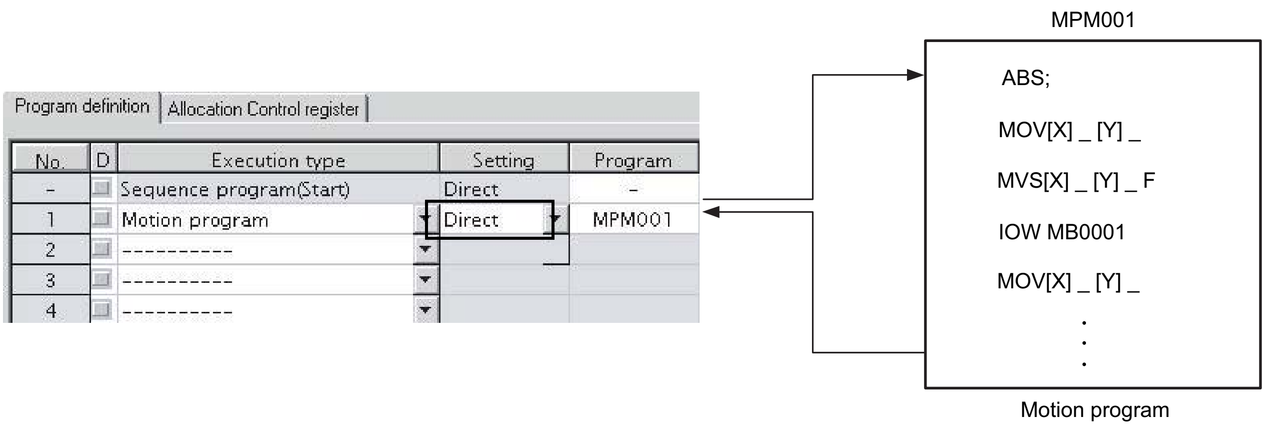


Fig. 5.2 Calling a Motion Program Using a Direct Designation - 2

[b] Using an Indirect Designation to Call a Motion Program

An indirect designation method designates a motion program to call using a register.
In this method, a program (MPM □□□) coinciding with value stored in the register is called.

■ Motion Program Referenced by a MSEE Command from a Ladder Program

Specify any register (M or D register) used for an indirect designation for Program No. in the MSEE command.

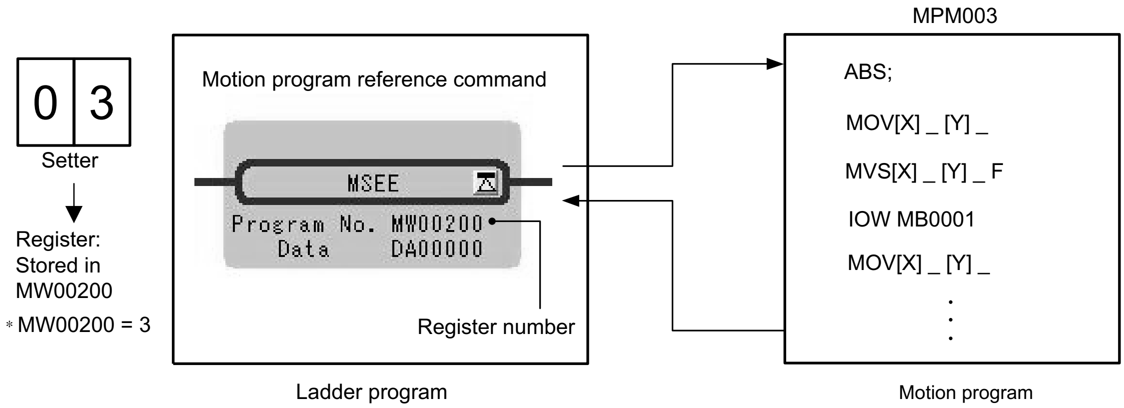


Fig. 5.3 Calling a Motion Program Using an Indirect Designation -1

■ Motion Program Registered in M-EXECUTOR Program Execution Definitions (Supported by CPU-03 and CPU-04.)

Select ***Indirect*** for the Setting. A register for the indirect designation is automatically mapped.

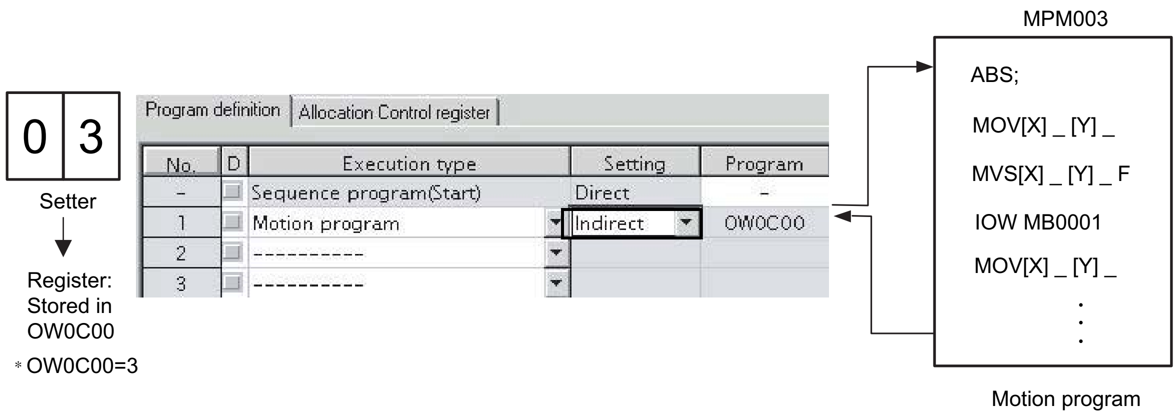


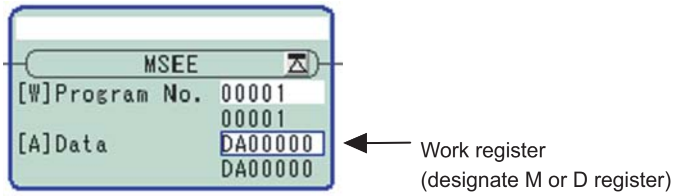
Fig. 5.4 Calling a Motion Program Using an Indirect Designation - 2

(4) Work Register

Configure and monitor a motion program via a work register.
The work register constitution for motion programs referenced by a MSEE command from a ladder program differs from that for motion programs registered in the M-EXECUTOR program execution definition.
The work register constitution in each case is as follows:

■ Motion Program Executed by a MSEE Command from a Ladder Program

Use a MSEE command of a ladder program to designate the work register (M or D register).
The work register constitution is as follows:



Work Register	Description
First word	Status flag of a motion program
Second word	Control signal of a motion program
Third word	Override for interpolation
Fourth word	System work number

■ Motion Program Registered in M-EXECUTOR Program Execution Definitions (Supported by CPU-03 and CPU-04.)

The work register is assigned to a M-EXECUTOR control register. (automatically defined by system)
The M-EXECUTOR control register constitution is as follows:

No.	Item	M-EXECUTOR Control register
1	Program number	OW0C00
	Status	IW0C00
	Control signal	OW0C01
	Override	OW0C02

M-EXECUTOR Control Register	Description
Status	Status flag of a motion program
Control Signal	Control Signal of a motion program
Override	Override for interpolation

For more information on the work register, refer to the subsequent pages.

Overview of System Operation
5

[a] Status Flag of a Motion Program

The motion program status flag shows the execution status of the motion program.
The following table shows details of status flag.

[Status Flag]

Bit No.	Status
0	Program running
1	Program paused
2	Program stopped by stop request (used by system)
3	(Reserved)
4	Single program block operation stopped
5	(Reserved)
6	(Reserved)
7	(Reserved)
8	Program alarm
9	Stopped by brake point
A	(Reserved)
B	In debug mode (EWS debugging operation)
C	Program type 0: Motion program
D	Start request signal history
E	No system work error
F	Main program number exceeded error

- When program alarm has occurred, the error details of the motion program are stored in the error information window and S registers.

[b] Control Signal

Program control signals (e.g., program operation start requests and program stop requests) need to be entered to execute the motion program.
The following types of signals for controlling motion programs are available.

Bit No.	Signal Name	Signal Type
0	Program operation start request	Differential or NO contact input
1	Program pause request	NO contact
2	Program stop request	NO contact
3	Program single block mode selection	NO contact
4	Program single block start request	Differential or NO contact input
5	Alarm reset request	NO contact
6	Program continuous operation start request	Differential or NO contact input
7	(Reserved)	
8	Skip 1 information	NO contact
9	Skip 2 information	NO contact
A	(Reserved)	
B	(Reserved)	
C	(Reserved)	
D	System work number setting*1	NO contact
E	Override setting for interpolation*2	NO contact
F	(reserved)	

* 1. System work number setting
- When a motion program is registered in M-EXECUTOR:
  Unable to designate it. The same system work number as No. defined in the system is used.
- When a motion program is called by a MSEE command from a ladder program:
  OFF: A system work automatically retrieved by system is used. The system work number may differ in each case.
  ON: A work with the designated system work number is used.
  However, when a work occupied by M-EXECUTOR is designated, “BitE: No system work error” is reported to the status.

* 2. Override setting for interpolation
OFF: 100% fixed at an override for interpolation
ON: Depends on the designated override for interpolation.

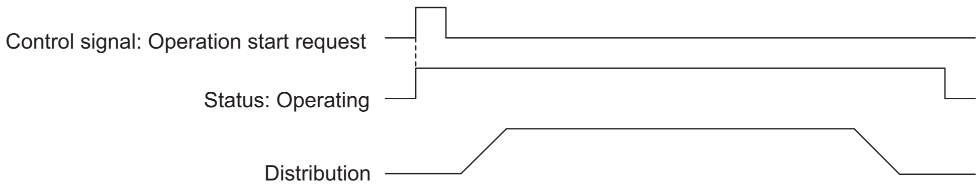
Use signals conforming to the above signal types when writing ladder programs.

Note: Motion programs are executed if the program operation start request signal is ON when the power is turned ON.

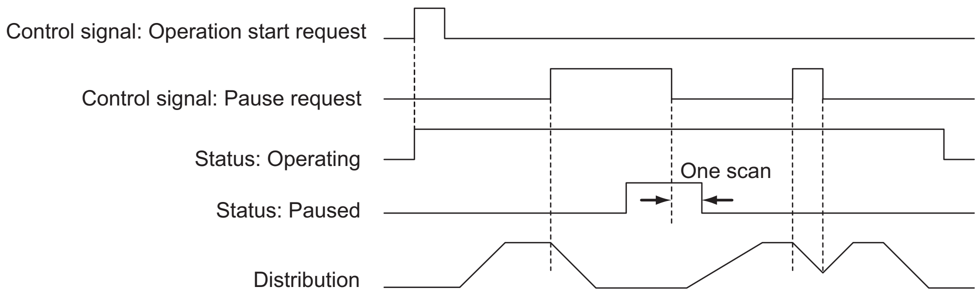
■ Timing Chart for Motion Program Control Signals

The following figure shows an example of a timing chart for motion program control signals.

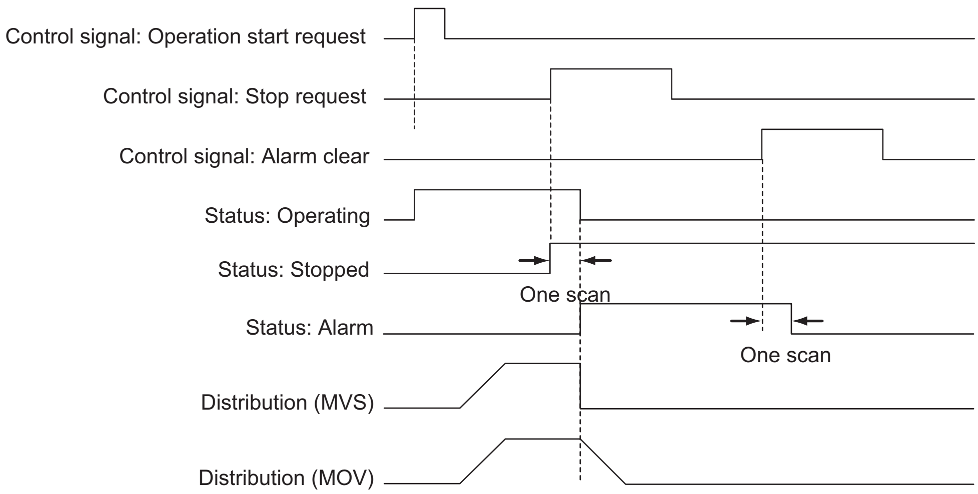
・Program Operation Start Request



・Pause Request



・Stop Request



Note: An alarm will occur if the stop request is turned ON during axis operation using a motion command.

[c] Interpolation Override

The override when executing interpolation move commands (setting; unit: 1 = 0.01%) is set.
This interpolation override is enabled only when the motion program control signal bit E (interpolation override setting) is ON.

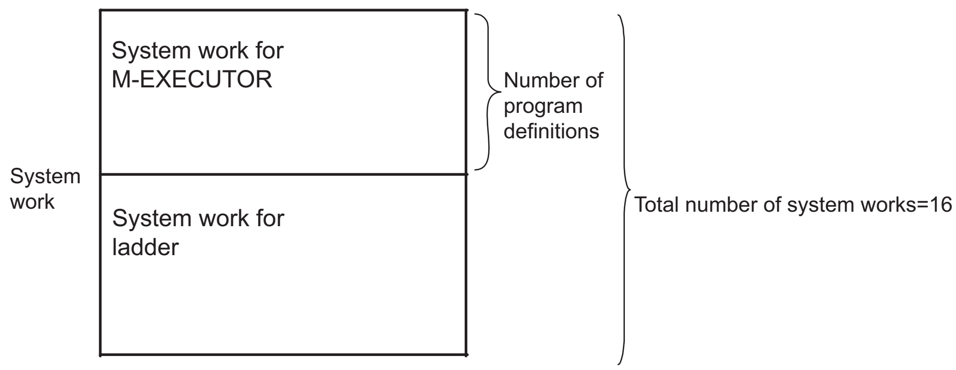
[d] System Work Number

If a motion program is executed by a ladder MSEE command, the system work number n (setting range: 1 to 16) used when executing motion programs can be set.

Note: System work numbers cannot be set for motion programs registered in the M-EXECUTOR detailed definition window (supported by CPU-03 and CPU-04) .

■ System Works for M-EXECUTOR (Supported by CPU-03 and CPU-04)

The total number of system works of a motion program is 16 for both M-EXECUTOR and ladder. The number set for **Program definition number** in the M-EXECUTOR detailed definition window is the number of the system works for M-EXECUTOR.



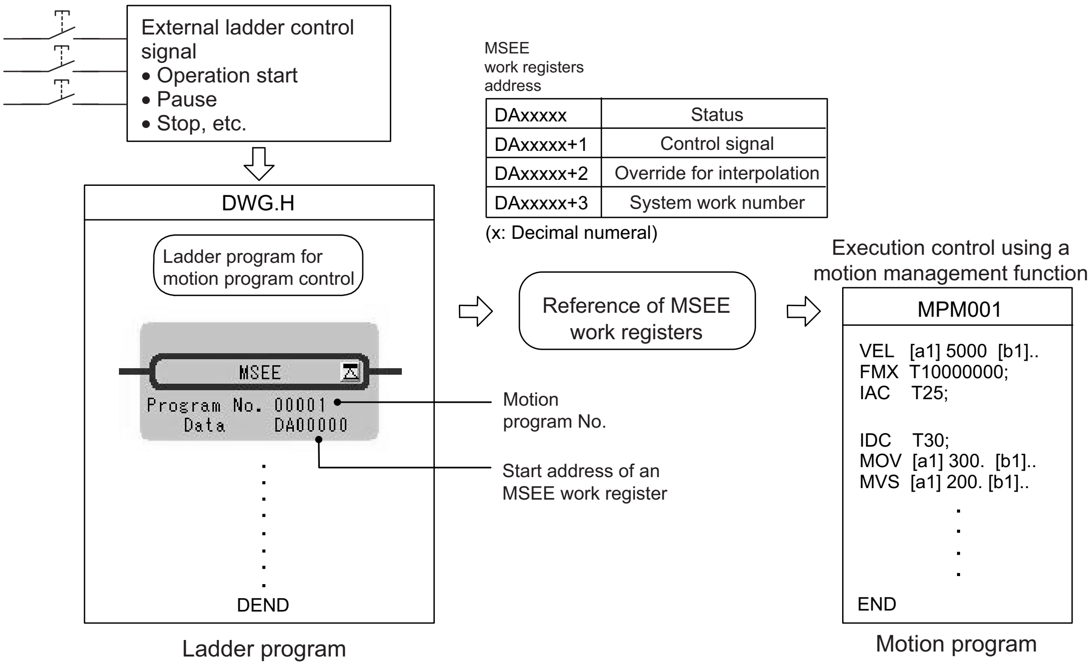
1. A work not occupied by M-EXECUTOR can execute a motion program using a MSEE command from a ladder program.
2. When the number of a system work occupied by M-EXECUTOR is specified by a ladder program, an alarm (No system work error) occurs. Therefore, when the number of program definitions is set to 16 in the M-EXECUTOR detailed definition window, a motion program cannot be executed by a ladder MSEE command.
   - No system work error: Status flag Bit E of the motion program

(5) How to Operate a Work Register

The way to operate a work register of a motion program referenced by a MSEE command from a ladder program differs from that of a motion program registered in the M-EXECUTOR program execution definition.
The way to operate it in each case is shown as follows:

[a] Motion Program Referenced by a MSEE Command from a Ladder Program

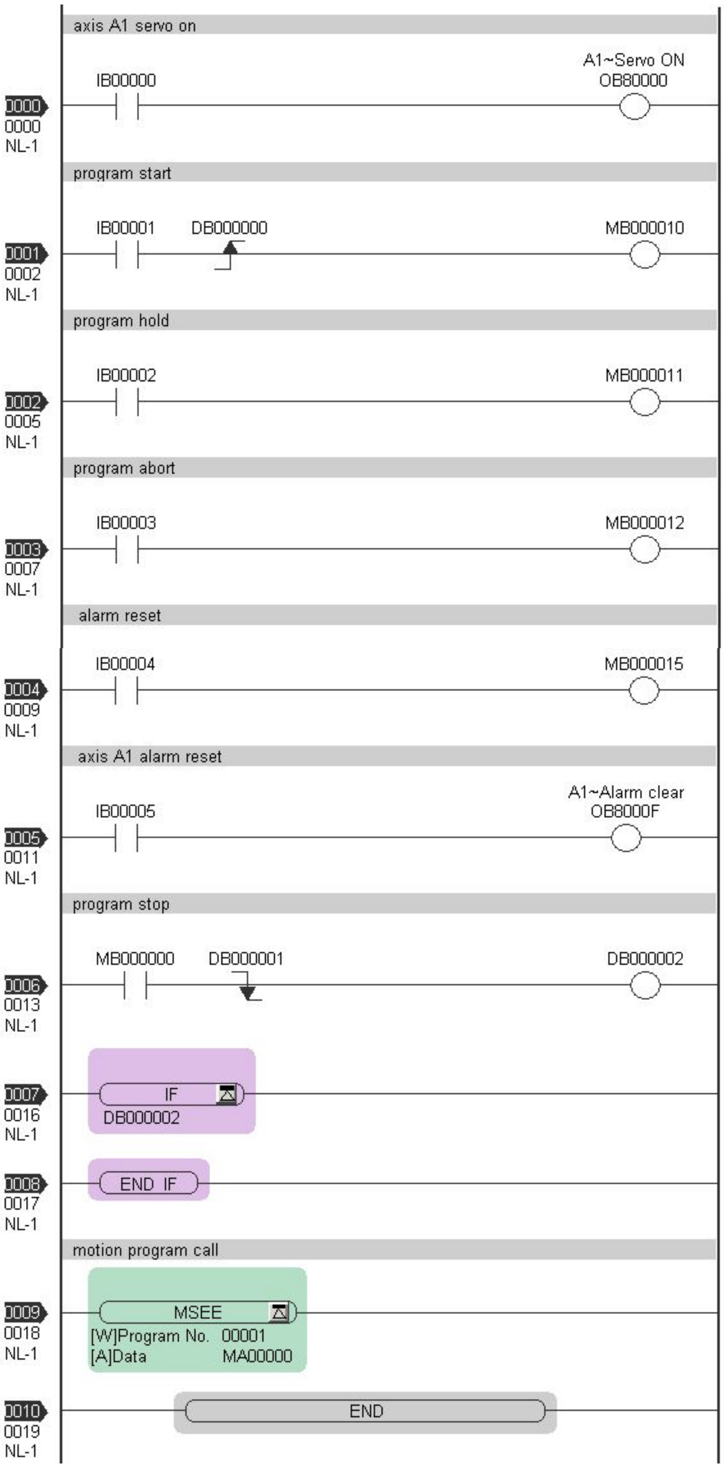
When a motion program is referenced by a MSEE command from a ladder program, the motion program is controlled by a sequence or ladder program.
To use this execution processing, incorporate a MSEE command in the ladder H drawing. In this case, MSEE work register configures and monitors the motion program.
The following figure shows a setting example in this method.



- For information about the meaning and estimation of the register number, refer to *5.3 Registers*.

An example of using a ladder program to control motion program is provided on the next page.

■ Example using a ladder program



[b] Motion Program Registered in M-EXECUTOR Program Execution Definitions (Supported by CPU-03 and CPU-04.)

When a motion program is registered in M-EXECUTOR program execution definition, select one from the following two execution processings.

- A way to immediately control a motion program from external device
- A way to control a motion program via a sequence or ladder program

Now, this section explains each execution processing in the subsequent pages.

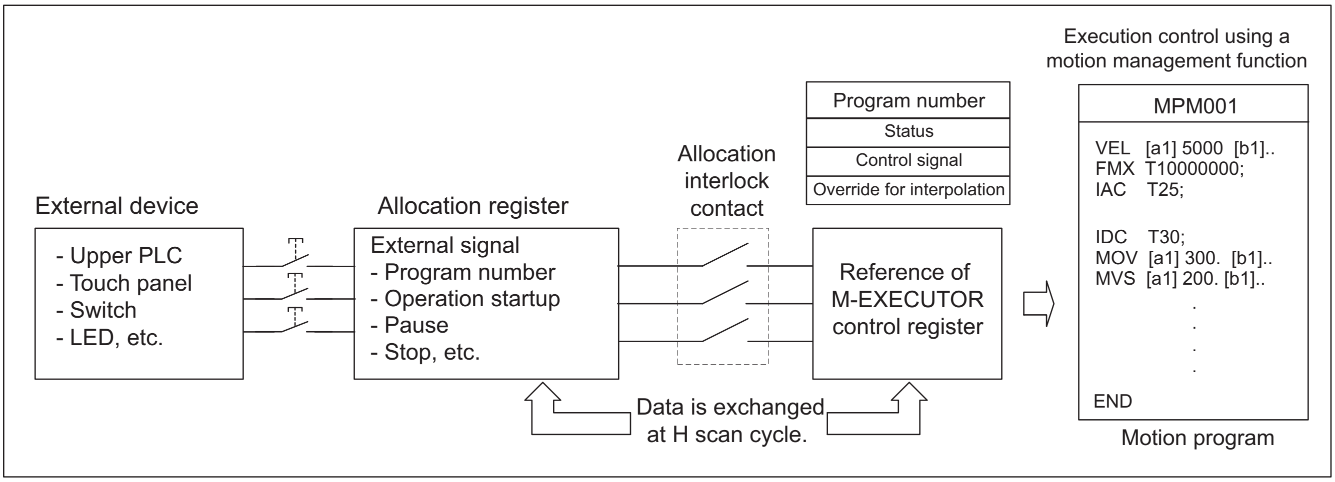
■ A Way to Immediately Control a Motion Program from External Device

M-EXECUTOR has a function which allocates any register to an M-EXECUTOR control register.
Using this function allows you to automatically exchange data between an M-EXECUTOR control register and an I/O register connected to an external device. This allows you to immediately control the motion program from the external device.
The following figure shows a setting example in this method.

Window for allocating the M-EXECUTOR register

No.	Item	M-EXECUTOR Control register	Allocation Disable	Direction	Allocation register	Allocation Contact interlock
1	Program number	MPM001				
	Status	IW0C00	☐	->	OW0000	IB00020
	Control signal	OW0C01	☐	<-	IW0000	IB00020
	Override	OW0C02	☐	<-	IW0001	IB00020

Specify any registers for allocation register and mapping interlock contact.



■ An allocation contact interlock is used to interlock the operation of a motion program. When setting an allocation register, be sure to set an allocation contact interlock.

It is processed, as shown below, by turning ON/OFF an allocation contact interlock:

- When an allocation contact interlock contact is ON, data is exchanged between an allocation register and M-EXECUTOR control register at H scan cycle. Now, the motion program becomes executable.
- When an allocation contact interlock is OFF, data is not exchanged between an allocation register and M-EXECUTOR control register. Now, the motion program becomes unexecutable.
- When an allocation contact interlock is switched from ON to OFF while running a motion program, the running motion program stops and an axis in operation also stops. Now, the motion program falls into the alarm “1Bh: Executing an emergency stop command” state, and the status “Bit8: Program alarm is occurring” is turned ON.
  Again, to execute a motion program, follow the procedure below for operation:

1. Switch the interlock contact from OFF to ON.
2. Turn ON a control signal “Bit5: Alarm reset request.”
3. Make sure that the status “Bit8: Program alarm is occurring” is turned OFF.
4. Turn OFF the control signal “Bit5: Alarm reset request.”
5. Turn ON a control signal “Bit0: Request for the program operation startup.”

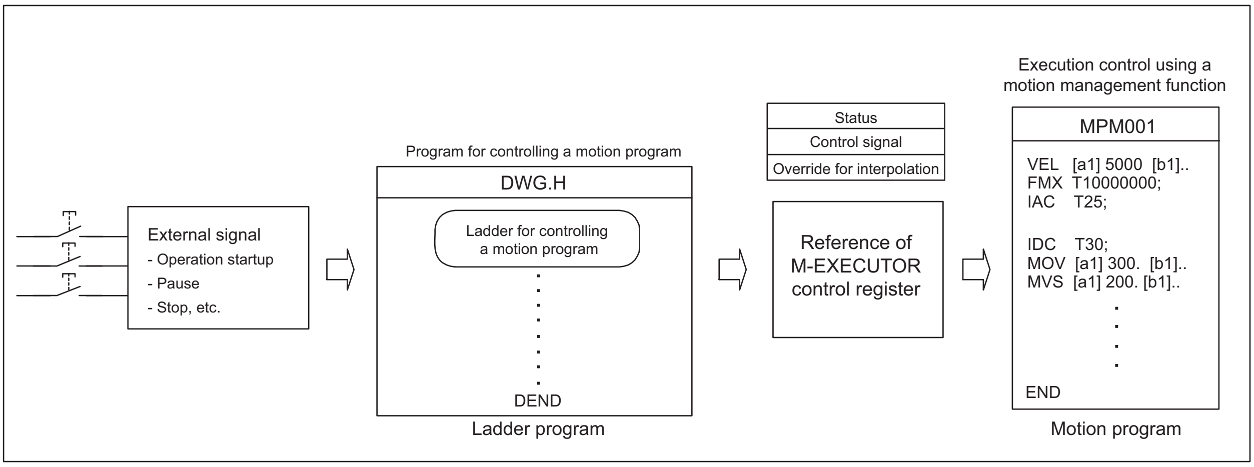
■ A Way to Control a Motion Program via a Sequence or Ladder Program

Without using the allocating function of the above mentioned M-EXECUTOR control register, controls a motion program via a sequence or ladder program.
To use this execution processing, save the blank Allocation register and the blank Allocation interlock contact as a blank.
In this case, set and monitor the motion program by using the M-EXECUTOR control register configures.
The following figure shows a setting example in this method.

M-EXECUTOR program execution definition

No.	Item	M-EXECUTOR Control register	Allocation Disable	Direction	Allocation register	Allocation Contact interlock
1	Program number	MPM001				
	Status	IW0C00		->		
	Control signal	OW0C01		<-		
	Override	OW0C02		<-		

Save the allocation register and the mapping interlock contact as a blank.



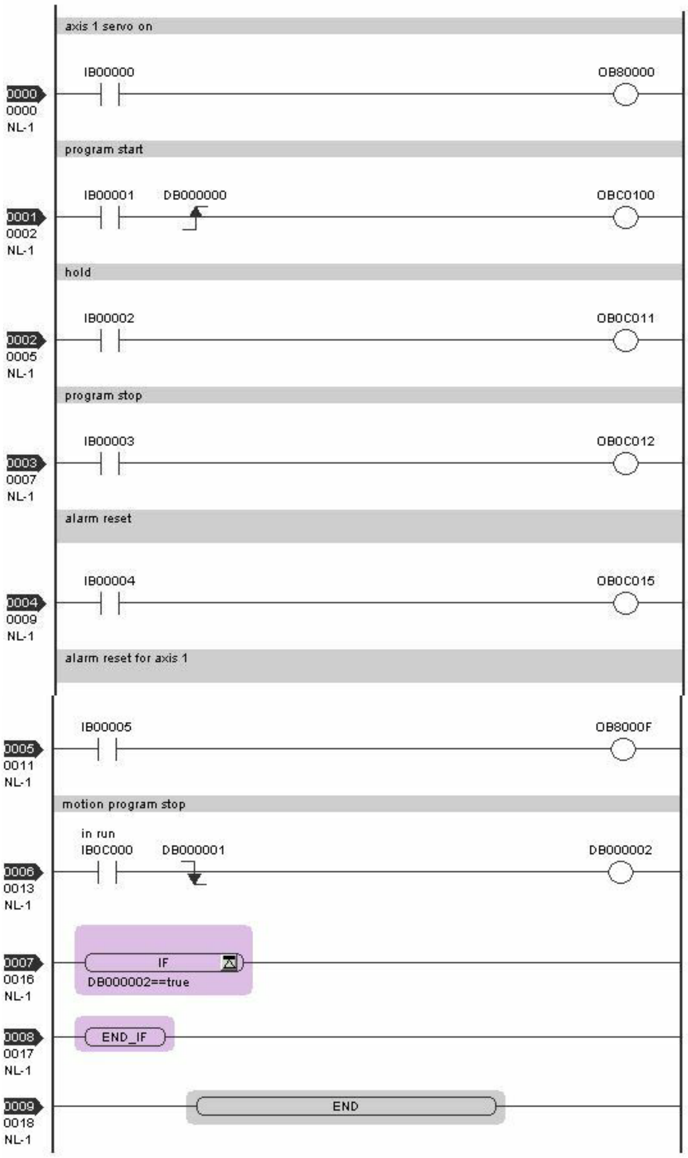
Each example which uses a sequence program and ladder program respectively as a program for controlling the motion program is shown as follows:

1. Example using a sequence program

```
OB80000 = IB00000;                    "axis 1 servo on"
OB0C010 = PON( IB00001 DB000000 );    "program start"
OB0C011 = IB00002;                    "hold"
OB0C012 = IB00003;                    "program stop"
OB0C015 = IB00004;                    "alarm reset"
OB8000F = IB00005;                    "Turn ON a single axis servo"
IF NON( IB0C000 DB000001 ) == 1;      "Is the program operation OFF?"
;                                     "Process when program operation is stopped"
IEND;

END;
```

2. Example using a ladder program



(6) Monitor the Motion Program Execution Information using S register

Using S register (SW03200 to SW04191) allows you to monitor the motion program execution information.
The way to monitor the execution information for a motion program referenced by an MSEE command from a ladder program differs from that for a motion program registered in the M-EXECUTOR program execution definition (supported by CPU-03 and CPU-04).
The way to monitor it in each case is shown as follows:

[a] Motion Program Referenced by an MSEE Command from a Ladder Program

When a motion program is referenced by an MSEE command from a ladder program, the way differs, depending on the "BitD" setting (system work number setting) of the motion program control signal, as follows:

■ The motion program control signal "BitD, System Work Number Setting" = ON

The execution information is reported to "Program Information Using Work n" register (SW03264-SW04191).
For example, when "System Work Number"=1, the motion program execution information can be monitored in SW03264-SW03321 "Program Information Using Work 1".

■ The motion program control signal "BitD, System Work Number Setting" = OFF

The used system work is automatically decided by system. Thus, to check which work is used, refer to "Running Program Number" (=SW03200 to SW03215).
For example, when you want to monitor the motion program MPM001 and SW03202=001, as used the work number=3, the execution information for the motion program MPM001 can be monitored in "Program Information Using Work 3" (=SW03380 to SW03437).

[b] Motion Program Registered in M-EXECUTOR Program Execution Definitions (Supported by CPU-03 and CPU-04.)

When a motion program is registered in the M-EXECUTOR program execution definition, the same system work number as the definition No. is used.
For example, a motion program is registered as "Definition No." =3, the used system work number is "System Work"=3. In this case, the execution information for the motion program can be monitored in "Program Information Using Work 3" (=SW03380 to SW03437).

For more information on the register area of the motion program execution information, refer to the subsequent pages.

■ Register Areas for Motion Program Execution Information

Motion program execution information

SW03200	Executing program number (No. of main program currently executing)	16W
SW03216	Reserved by the system.	16W
SW03232	Executing Program Bit (Executing when corresponding bit is ON)	16W
SW03248	Reserved by the system.	16W
SW03264	Program information used by work 1	58W
SW03322	Program information used by work 2	58W
SW03380	Program information used by work 3	58W
SW03438	Program information used by work 4	58W
SW03496	Program information used by work 5	58W
SW03554	Program information used by work 6	58W
SW03612	Program information used by work 7	58W
SW03670	Program information used by work 8	58W
SW03728	Program information used by work 9	58W
SW03786	Program information used by work 10	58W
SW03844	Program information used by work 11	58W
SW03902	Program information used by work 12	58W
SW03960	Program information used by work 13	58W
SW04018	Program information used by work 14	58W
SW04076	Program information used by work 15	58W
SW04134	Program information used by work 16	58W
SW04192	Reserved by the system.	928W
SW05120	Reserved by the system.	64W

Executing program number

SW03200	Program number used by work 1
SW03201	Program number used by work 2
SW03202	Program number used by work 3
SW03203	Program number used by work 4
SW03204	Program number used by work 5
SW03205	Program number used by work 6
SW03206	Program number used by work 7
SW03207	Program number used by work 8
SW03208	Program number used by work 9
SW03209	Program number used by work 10
SW03210	Program number used by work 11
SW03211	Program number used by work 12
SW03212	Program number used by work 13
SW03213	Program number used by work 14
SW03214	Program number used by work 15
SW03215	Program number used by work 16

Executing program bit

SW03232	MP☐016 (Bit15) to MP☐001 (Bit0)
SW03233	MP☐032 (Bit15) to MP☐017 (Bit0)
SW03234	MP☐048 (Bit15) to MP☐033 (Bit0)
SW03235	MP☐054 (Bit15) to MP☐049 (Bit0)
SW03236	MP☐080 (Bit15) to MP☐055 (Bit0)
SW03237	MP☐096 (Bit15) to MP☐081 (Bit0)
SW03238	MP☐112 (Bit15) to MP☐097 (Bit0)
SW03239	MP☐128 (Bit15) to MP☐113 (Bit0)
SW03240	MP☐144 (Bit15) to MP☐129 (Bit0)
SW03241	MP☐160 (Bit15) to MP☐145 (Bit0)
SW03242	MP☐176 (Bit15) to MP☐161 (Bit0)
SW03243	MP☐192 (Bit15) to MP☐177 (Bit0)
SW03244	MP☐208 (Bit15) to MP☐193 (Bit0)
SW03245	MP☐224 (Bit15) to MP☐209 (Bit0)
SW03246	MP☐240 (Bit15) to MP☐225 (Bit0)
SW03247	MP☐256 (Bit15) to MP☐241 (Bit0)

Note: ☐ indicates M or S.

■ Details of Program Information Used by Work n

Program information used by work n

Offset	Content	Size
+0	Program status	
+1	Program control signal	
+2	Parallel 0 information	3W
+5	Parallel 1 information	3W
+8	Parallel 2 information	3W
+11	Parallel 3 information	3W
+14	Parallel 4 information	3W
+17	Parallel 5 information	3W
+20	Parallel 6 information	3W
+23	Parallel 7 information	3W
+26	Logical axis #1 program current position	2W
+28	Logical axis #2 program current position	2W
+30	Logical axis #3 program current position	2W
+32	Logical axis #4 program current position	2W
+34	Logical axis #5 program current position	2W
+36	Logical axis #6 program current position	2W
+38	Logical axis #7 program current position	2W
+40	Logical axis #8 program current position	2W
+42	Logical axis #9 program current position	2W
+44	Logical axis #10 program current position	2W
+46	Logical axis #11 program current position	2W
+48	Logical axis #12 program current position	2W
+50	Logical axis #13 program current position	2W
+52	Logical axis #14 program current position	2W
+54	Logical axis #15 program current position	2W
+56	Logical axis #16 program current position	2W

Executing program number
Executing block number
Error code

5.2.4 Sequence Program (Supported by CPU-03 and CPU-04.)

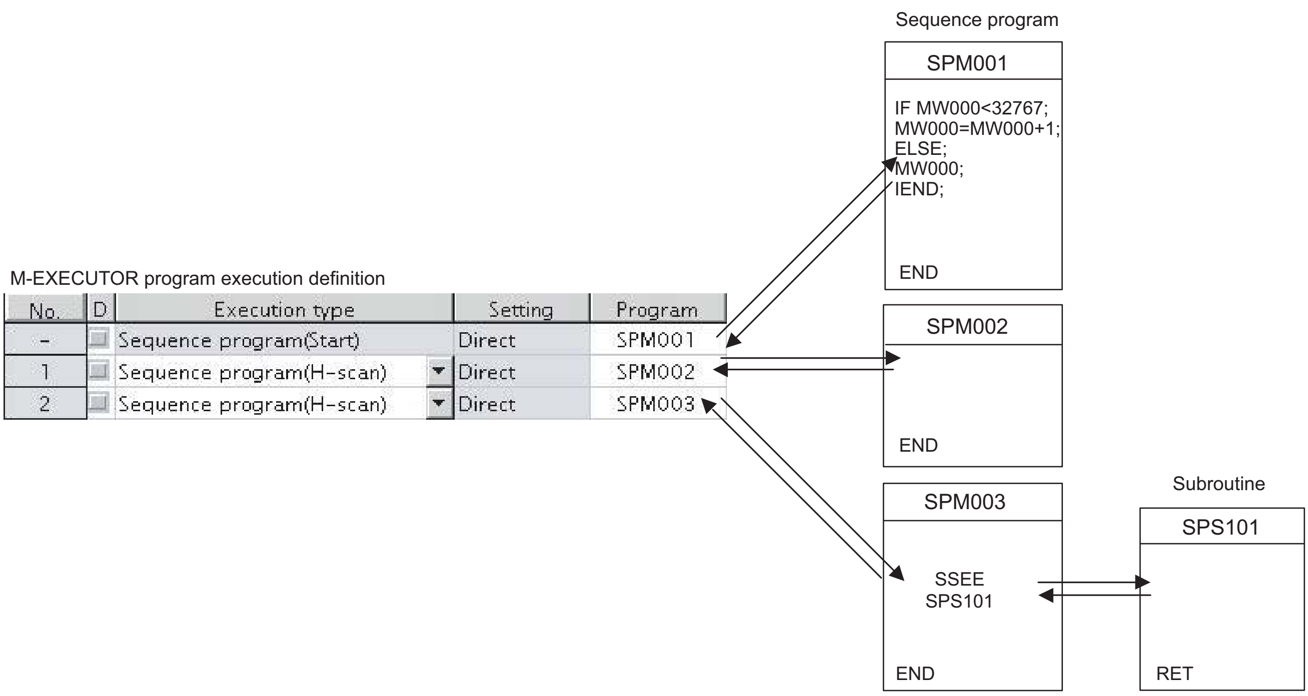
A sequence program is a program described with motion language of text format.
The following table shows two types of sequence programs.

Category	Designation Method	Features	Number of Programs
Main program	SPM□□□ (□□□=1 to 256)	Calling from the M-EXECUTOR program execution definition	Up to 256 programs of the following types can be created: • Motion main program • Motion subprogram • Sequence main program • Sequence subprogram
Subprogram	SPS□□□ (□□□=1 to 256)	Calling from the main program	

- The program numbers of sequence programs are managed in the same manner as the motion program numbers. Assign a different number for each program number.
  - Motion program MPM□□□: Program number of MPS□□□
  - Sequence program SPM□□□: Program number of SPS□□□

(1) How to Run a Sequence Program

A sequence program is executed by registering it in the M-EXECUTOR program execution definition.
Sequence programs are executed in ascending numeric order.
The following figure shows an execution example.



When the execution type is set to "**Sequence program (H scan)**" or "**Sequence program (L scan)**", the program is executed at the time the definition is saved. When the execution type is set to **Sequence program (Start)**, the program is executed when the power supply is turned ON again next time.

(2) How to Designate a Sequence Program

You can only designate a sequence program directly. Indirect designation is unavailable.
Use the program number (SPM□□□) when designating a sequence program to execute.

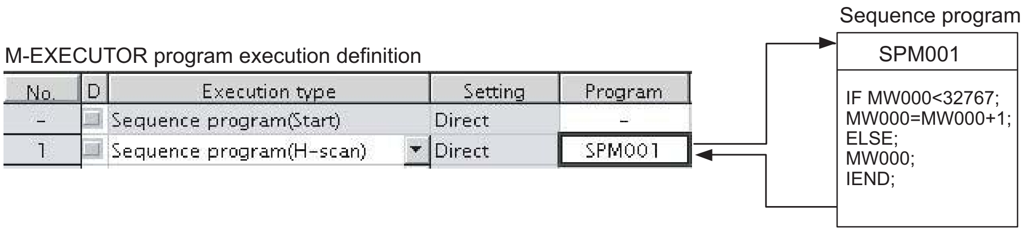


Fig. 5.5 Calling a Sequence Program

(3) Work Register

Monitor a sequence program through a work register.
A work register, similar to the motion program registered in M-EXECUTOR, has status flags in the M-EXECUTOR control register.
The following table shows the work register configuration of the sequence program.

Work Register	Content
Status	Status flag of a sequence program

[a] Status Flag of Sequence Program

The status flags of a sequence program allow you to know the execution status of the sequence program.
The following table explains the detailed contents of status flags.
[Status]

Bit No.	Status
0	Program running
1	(Reserved)
2	(Reserved)
3	(Reserved)
4	(Reserved)
5	(Reserved)
6	(Reserved)
7	(Reserved)
8	Program alarm is occurring
9	Stopping at breakpoint
A	(Reserved)
B	In debug mode (EWS debug operation)
C	Program type 1: Sequence program
D	Start request history
E	(Reserved)
F	(Reserved)

■ Sequence program alarm

When referencing a sequence subprogram (SSEE command execution) and an error is detected, “Bit8: Program alarm is occurring” is turned ON. If the error is cleared, it is turned OFF.
Error details are as follows:

Error Details
Called program is unregistered
Called program is not a sequence program
Called program is not a sub program (main program is called)
Called program number is over
Nest over error

5.3 Registers

This section describes the types of registers used in MP2200 user programs (mainly ladder programs) and how to use them.

5.3.1 Types of Registers

(1) DWG Registers

Registers used by ladder programs (ladder drawings; DWG). Each drawing can use the registers outlined in the following table.

Type	Name	Specification Method	Range	Details	Characteristics
S	System registers	SB, SW, SL, SFnnnnn (SAnnnnn)	SW00000 to SW08191	Registers provided by the system. SW00000 to SW00049 are cleared to all zeros when the system starts.	Common to all drawings
M	Data registers	MB, MW, ML, MFnnnnn (MAnnnnn)	MW00000 to MW65534	Registers shared by all drawings. Used, e.g., as an interface between drawings.	
I	Input registers	IB, IW, IL, IFhhhh (IAhhhh)	IW0000 to IW7FFF	Registers used for input data.	
O	Output registers	OB, OW, OL, OFhhhh (OAhhhh)	OW0000 to OW7FFF	Registers used for output data.	
C	Constants registers	CB, CW, CL, CFnnnnn (CAnnnnn)	CW00000 to CW16383	Registers that can only be called from programs.	
#	# registers	#B, #W, #L, #Fnnnnn (#Annnnn)	#W00000 to #W16383	Call-only registers Can be called only by corresponding drawing. The usage range is set by the user using MPE720.	Unique to each drawing
D	D registers	DB, DW, DL, DFnnnnn (DAnnnnn)	DW00000 to DW16383	Internal registers unique to each drawing. Can be used only by corresponding drawing. The usage range is set by the user using MPE720.	

Note: 1. n: Decimal number; h: Hexadecimal number
2. B, W, L, F, and A: Data type (B: Bit, W: Integer, L: Double-length integer, F: Real number, A: Address. Refer to *5.3.2 Data Types*.)
3. Up to 32 D registers (32 words, DW0000 to DW0031) can be used when creating drawings, but this can be changed in the MPE720 Drawings Properties Window. Refer to the *Machine Controller MP900/MP2000 Series User's Manual MPE720 Software for Programming Device (*manual no.: *SIEP C880700 05)* or, refer to *Engineering Tool for MP2000 Series Machine Controller MPE720 Version 6 User's Manual* (manual no.: SIEP C880700 30) for details.
4. S and M register data has a battery backup to ensure the data is held even if the MP2200 power is turned OFF and ON.
It is recommended that data to be held regardless of whether or not the power is turned OFF and ON should be written to M registers if possible.

(2) Function Registers

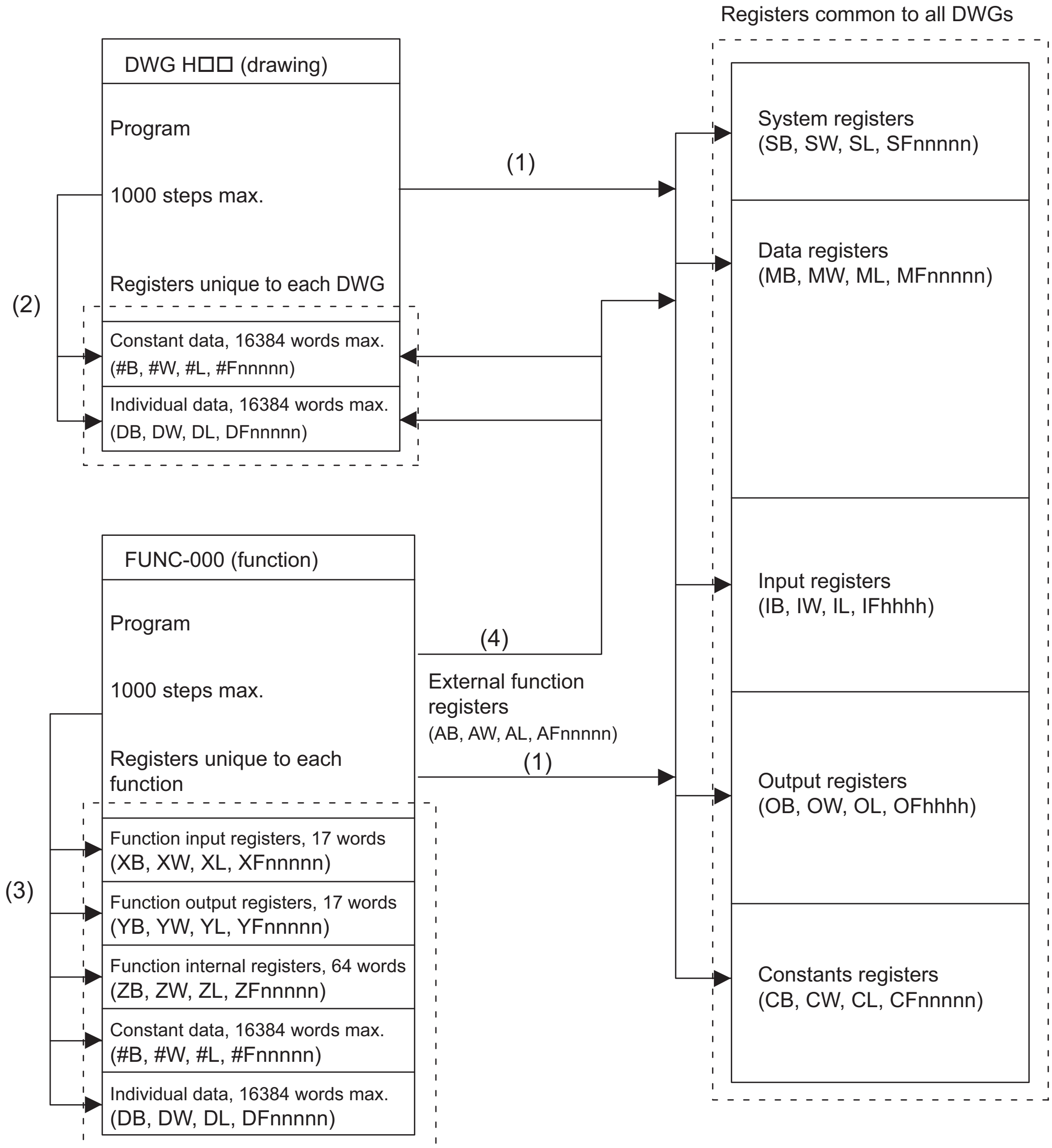
The following table shows the registers that can be used with each function.

Type	Name	Specification Method	Range	Details	Characteristics
X	Function input registers	XB, XW, XL, XFnnnnn	XW00000 to XW00016	Input to functions Bit input: XB000000 to XB00000F Integer input: XW00001 to XW00016 Double-length integer input: XL00001 to XL00015	Unique to each function
Y	Function output registers	YB, YW, YL, YFnnnnn	YW00000 to YW00016	Output from functions Bit output: YB000000 to YB00000F Integer output: YW00001 to YW00016 Double-length integer output: YL00001 to YL00015	
Z	Internal function registers	ZB, ZW, ZL, ZFnnnnn	ZW0000 to ZW00063	Internal registers unique to each function Can be used for function internal processing.	
A	External function registers	AB, AW, AL, AFhhhh	AW0000 to AW32767	External registers with the address input value as the base address. For linking with S, M, I, O, #, and DAnnnnn.	
#	# registers	#B, #W, #L, #Fnnnnn (#Annnnn)	#W00000 to #W16383	Call-only registers Can be called only from the relevant function. The usage range is set by the user using MPE720.	
D	D registers	DB, DW, DL, DFnnnnn (DAnnnnn)	DW00000 to DW16383	Internal registers unique to each function. Can be called only the relevant function. The usage range is set by the user using MPE720.	
S	System registers	SB, SW, SL, SFnnnnn (SAnnnnn)	Same as DWG registers. Note: These registers are shared by drawings and functions. Pay attention to how these registers are to be used when calling the same function from a drawing of a different priority level.		
M	Data registers	MB, MW, ML, MFnnnnn (MAnnnnn)			
I	Input registers	IB, IW, IL, IFhhhh (IAhhhh)			
O	Output registers	OB, OW, OL, OFhhhh (OAhhhh)			
C	Constants registers	CB, CW, CL, CFnnnnn (CAnnnnn)			

Note: 1. n: Decimal number; h: Hexadecimal number
2. B, W, L, F, and A: Data type (B: Bit, W: Integer, L: Double-length integer, F: Real number, A: Address. Refer to *5.3.2 Data Types*.)
3. SA, MA, IA, OA, DA, #A, and CA registers can be used within functions.

(3) Register Ranges in Programs

The following figure shows DWG programs, function programs, and register call ranges.

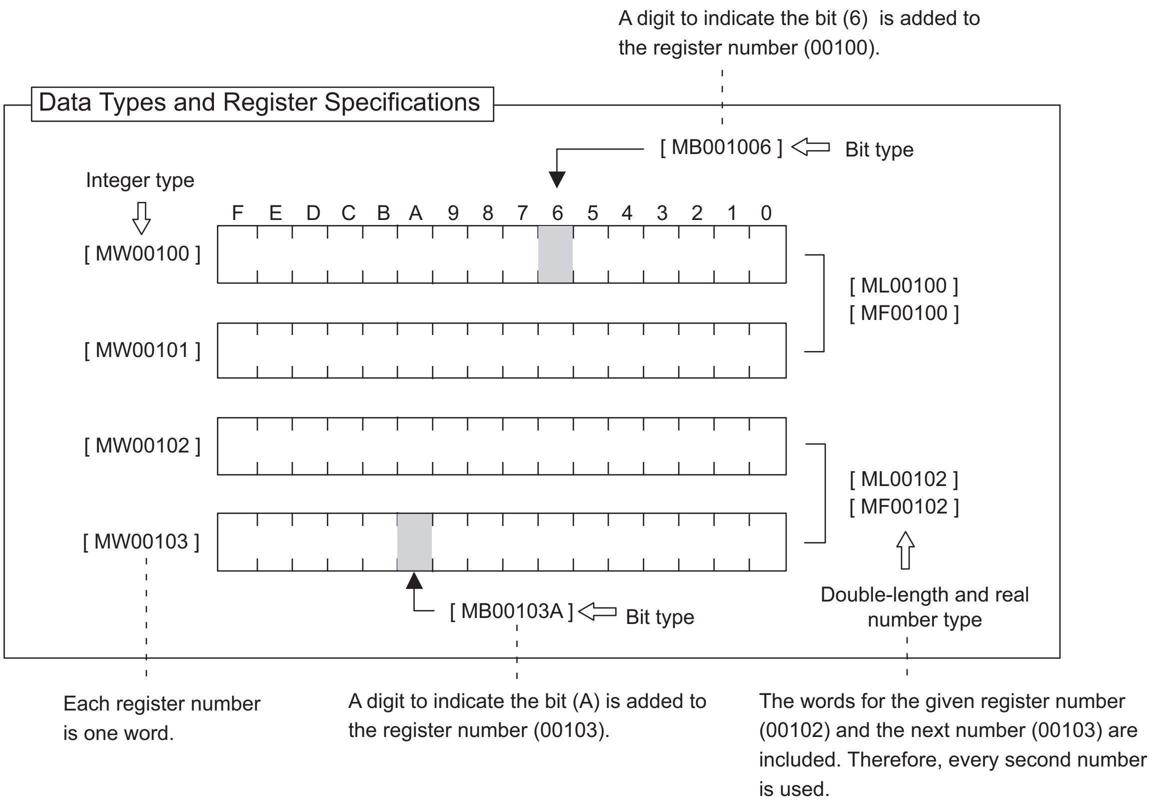


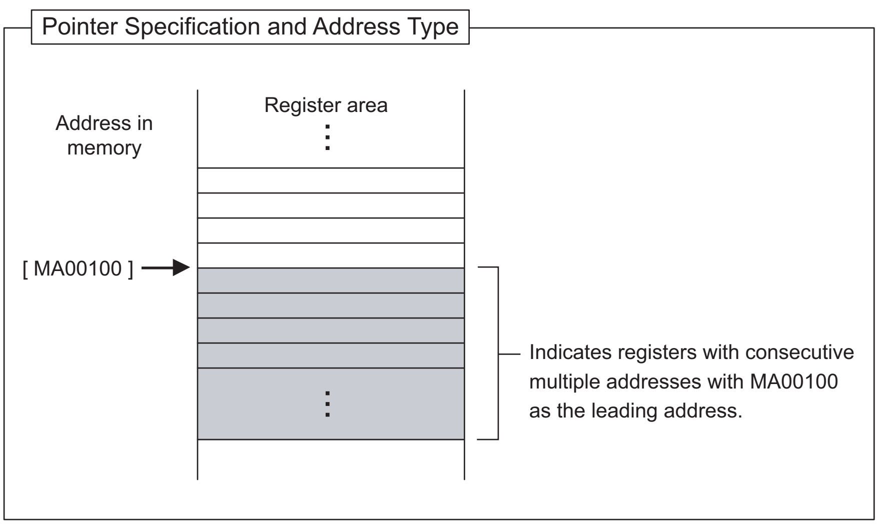
(1): Registers that are common to all drawings can be called from any drawing or function.
(2): Registers that are unique to each drawing can be called only from within the drawing.
(3): Registers that are unique to each function can be called only from within the function.
(4): Registers that are common to all drawings and registers that are unique to each drawing can be called from functions using the external function registers.

5.3.2 Data Types

There are five kinds of data: Bit, integer, double-length integer, real number, and address data. Each is used differently depending on the application. Address data, however, is used only inside functions when specifying pointers.
The following table shows the types of data.

Type	Data types	Numeric Value Range	Remarks
B	Bit	ON, OFF	Used by relay circuits.
W	Integer	–32768 to +32767 (8000H) (7FFFH)	Used for numeric value operations. The values in parentheses ( ) are for use with logical operations.
L	Double-length integer	–2147483648 to +2147483647 (80000000H) (7FFFFFFFH)	Used for numeric value operations. The values in parentheses ( ) are for use with logical operations.
F	Real number	± (1.175E-38 to 3.402E+38), 0	Used for numeric value operations.
A	Address	0 to 32767	Used only when specifying pointers.

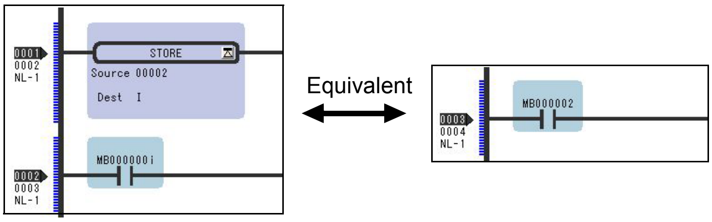




5.3.3 How to Use Subscripts i, j

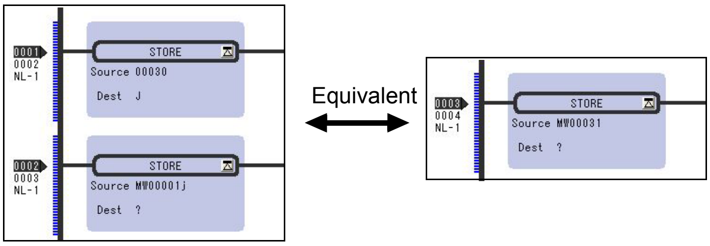
Two type of registers (i, j) are available as dedicated registers to modify the relay and register numbers. Both i and j have the same function. They are used when you want to handle a register number as a variable.
An example for each register data type is given as explanation.

(1) Bit Type Attached with a Subscript



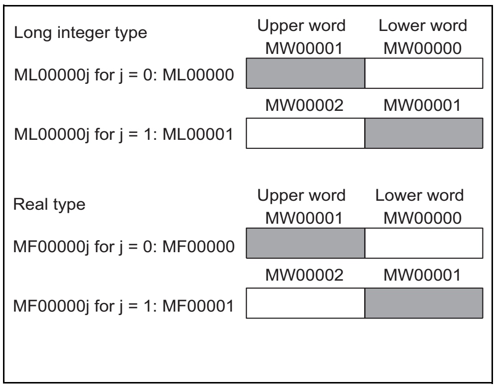
The result is a relay number added with i or j value.
For example, MB000000i for i=2 is the same as MB000002.
Also, MB000000j for j=27 is the same as MB00001B.

(2) Integer Type Attached with a Subscript



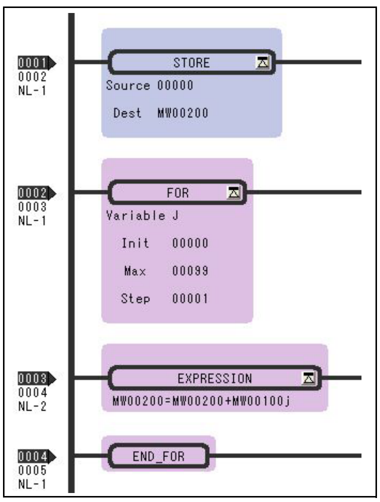
The result is a register number added with i or j value.
For example, MW00010i for i=3 is the same as MW00013.
Also, MW00001j for j=30 is the same as MW00031.

(3) Double-length Integer or Real Number Type Attached with a Subscript



The result is a register number with an added i or j value.
For example, “ML00000j for j=1” is the same as ML00001. Also, “MF00000j for j=1” is the same as MF00001.
However, as a word indicated by a register number is the lower word for a double-length integer/real number type, for the same ML00001 and MF00001, be aware that an upper/lower word of ML00001 and MF00001 for j=0 may differ from those of. ML00001 and MF00001 for j=1.

Program example using subscript

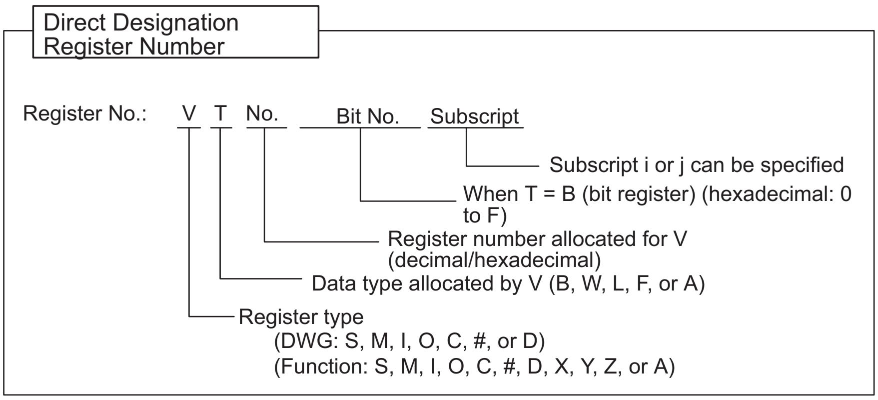


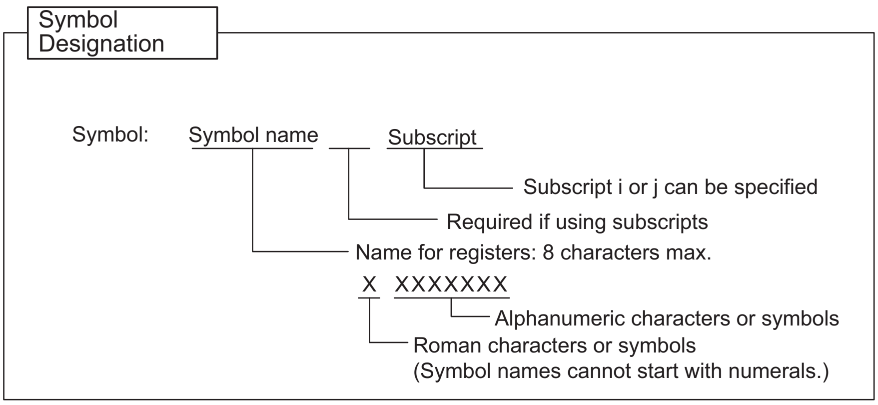
The left program uses a subscript j and calculates the total amount of a hundred registers from MW00100 to MW00199, and stores the total amount in MW00200.

5.3.4 Register Designation

Registers can be specified directly by register number or by symbol (register name). A combination of both of these register designation methods can be used in ladder programs.
When using the symbol specification method, the relationship between symbols and register numbers must be defined.
The following table shows the register specification methods.

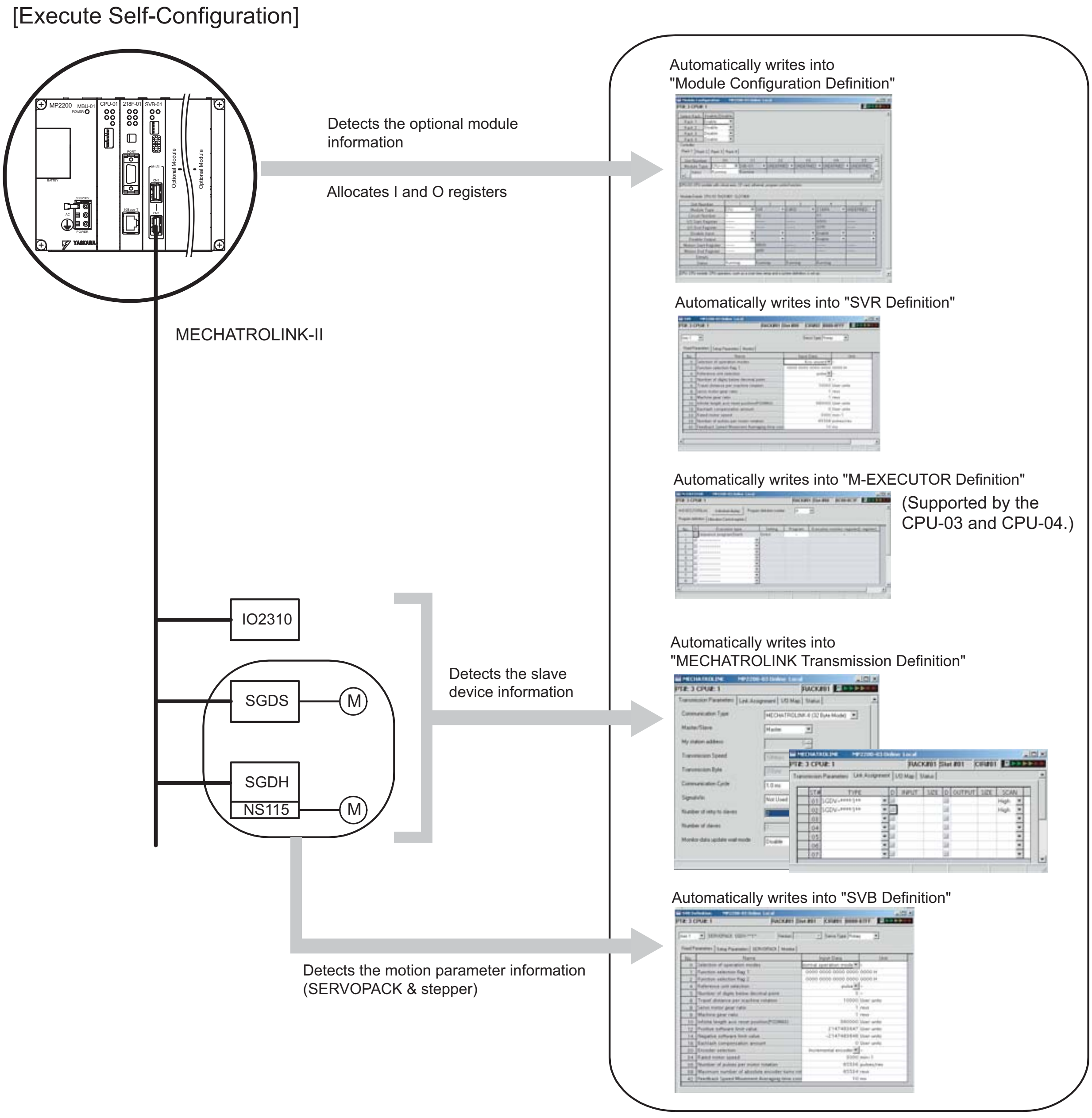
Designation Method	Designation Example for Each Data Type	
Register Number Direct Designation	Bit register: Integer register: Double-length integer register: Real number register: Address register:	MB00100AX MW00100X ML00100X MF00100X MA00100X
	X: When specifying subscripts, subscript i or j is added after the register number.	
Symbol Designation	Bit register: Integer register: Double-length integer registers: Real number registers: Address registers:	RESET1-A.X STIME-H.X POS-REF.X IN-DEF.X PID-DATA.X ↓ 8 alphanumeric characters max.
	X: When specifying subscripts, a period (.) is added after the symbol (8 alphanumeric characters max.) and then a subscript i or j is added.	





5.4 Self-configuration

The self-configuration function automatically recognizes the Optional Modules mounted to the MP2200 Basic Module and all slave data for slaves connected to the MECHATROLINK network, and automatically generates a definition file. Self-configuration greatly simplifies the procedure needed to start the system.
Refer to *5.4.2 Definition Information Updated with Self-Configuration* for items that are automatically generated.



5.4.1 How to Execute Self-Configuration

The following two methods are available for executing the self-configuration.

- Execute the self-configuration (from DIP switch)
- Execute the self configuration (from MPE720)

Now, this section explains each way to execute the self-configuration:

(1) Procedure Using the DIP Switch

The following two methods are available for executing the self-configuration.

- When executing the self-configuration first time after connecting a device
- When executing the self-configuration after adding devices such as SERVOPACKs

Self-configuration can be executed from the Basic Module DIP switch.
Now, this section explains each way to execute the self-configuration:

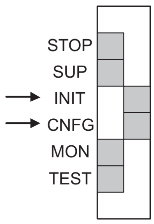
[a] When Executing the Self-Configuration First Time after Connecting a Device

By performing the operation below, the self-configuration for all modules is newly executed, and all new definition files are created.
Before performing the operation, turn ON the power supply of a device such as SERVOPACK.

■ Caution

Note that this operation can clear the following data in MP2200.

- All definition files, all user programs, and all registers

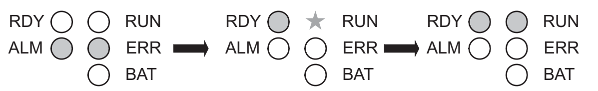


1. Turn OFF the power supply.
   Turn OFF the 24-VDC or 100/200-VAC power supply to the MP2200.

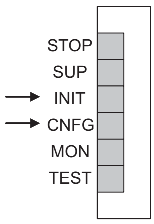
2. Set the DIP switch.
   Set the switches INIT and CNFG of the DIP switch SW1 on the MP2200 Basic Module to ON.

3. Turn ON the power supply.
   Turn ON the 24-VDC or 100/200-VAC power supply to the MP2200.

4. Check the LED indicators.
   Check that the LED indicators on the MP2200 Basic Module change as follows.



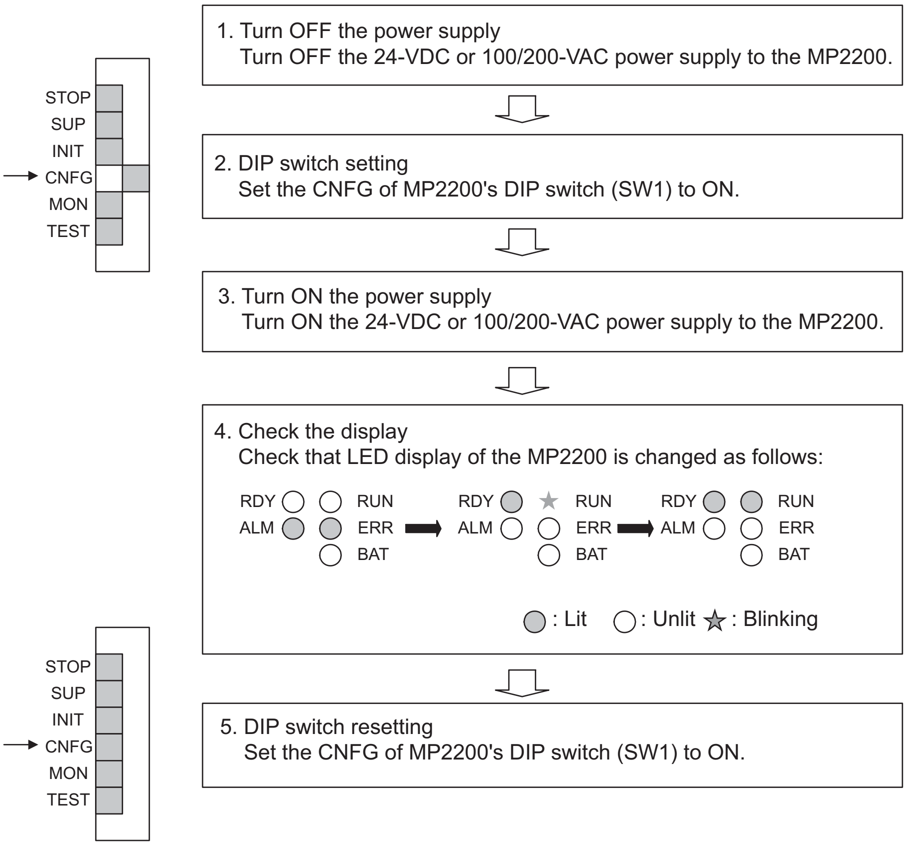
: Lit : Unlit : Blinking



5. Reset the DIP siwtch.
   Set the switches INIT and CNFG of the DIP switch SW1 on the MP2200 Basic Module to OFF.

[b] When Executing the Self-Configuration after Adding Devices Such as SERVOPACKs

By performing the following operation, a definition for an axis newly detected in the MECHATROLINK transmission is created. The definitions for already mapped axes are not updated.
Before performing the operation, turn ON the power supply of devices such as SERVOPACK.



Note: Since a register mapping was manually changed after the self-configuration was last executed last time, input/output addresses may be changed by executing subsequent self-configurations.
Also, when SVR is set to Disable, SVR may be reset to Enable.
To retain the changed register mapping, etc., manually map a register to the additional devices instead of using self-configuration, and then update the definition file.

■ INIT Switch and RAM Data

RAM data will be cleared if the INIT switch of the DIP switch on the MP2200 Basic Module is ON and the power is turned ON. Flash memory data is read and overwritten when the INIT switch is OFF and the power is turned ON. Therefore, to protect RAM data, always save data to the MP2200 flash memory before turning OFF the power when writing or editing programs.

■ Turning OFF Power After Executing Self-configuration

Do not turn OFF the 24-VDC or 100/200-VAC power supply to the MP2200 after executing self-configuration until the definitions data has been saved to flash memory in the MP2200. If the power is somehow turned OFF before the data is saved to flash memory, re-execute the self-configuration.

---

(2) Procedure Using MPE720

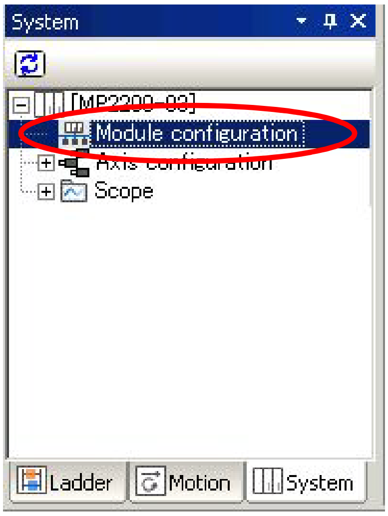
Executing self-configuration from MPE720 allows self-configuration for individual Modules as well as for all modules.
When self-configuration is carried out from MPE720, a definition for any axis newly detected in the MECHATROLINK transmission is created. The definitions for already mapped axes are not updated.

This section explains each way to execute the self-configuration:

[a] Self-configuration for All the Modules

By performing the following operation, the self-configuration for MP2200 Basic Units and Optional Modules is executed.
Before performing the operation, turn ON the power supply of devices such as SERVOPACK.

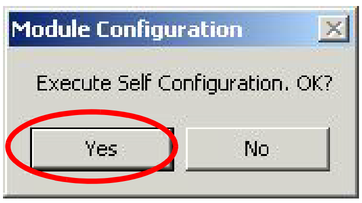
1. Double-click ***System - Module configuration***.



The **Engineering Manager** Window will open and the **Module Configuration** Window will appear.

2. Select ***Order - Self Configure All Modules*** to execute self-configuration.

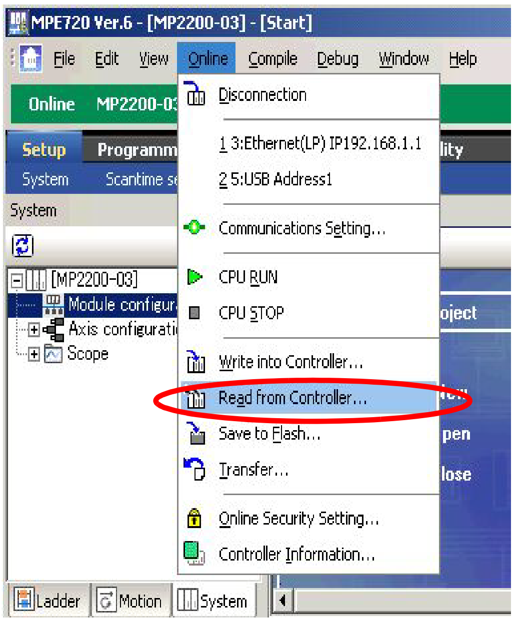
3. Click **Yes** for the following message.



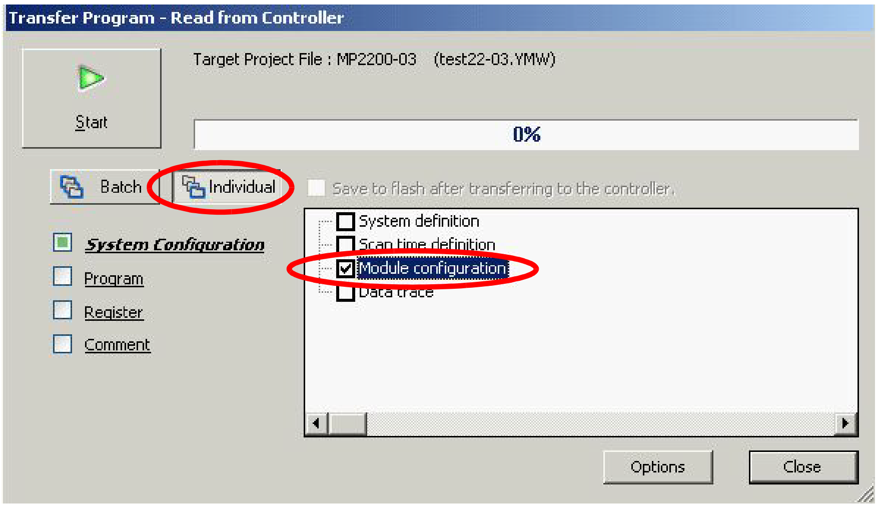
4. While running the self-configuration, the following message is shown.

5. If the following warning message is shown after performing step 4, the module configuration definitions for CPU and MPE720 may differ from each other. Continue to perform step 6. When the message is not shown, go to step 9.

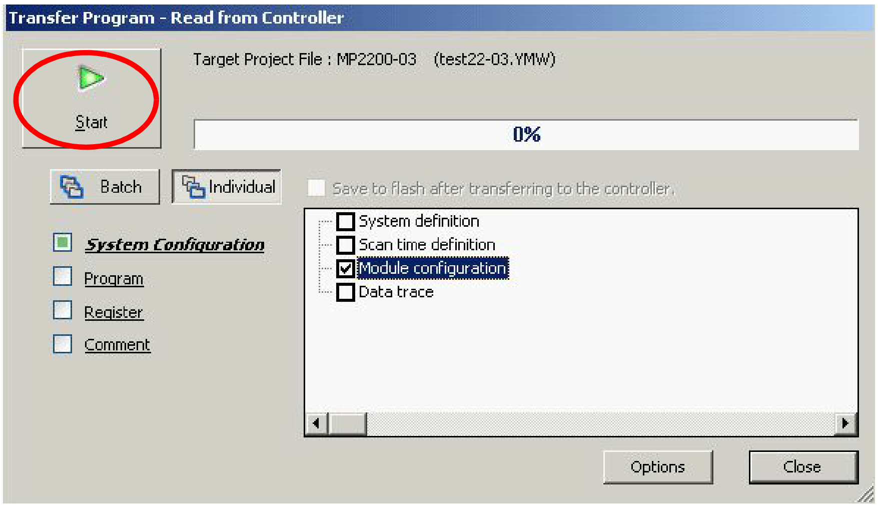
6. Select ***Online(O) - Read from Controller(A)***.



7. Click **Individual**, and only check **Module configuration**.



8. Click **Start** to read the module configuration definition from a controller.



9. Click the **Save & FLASH Save** Button to flash save the definition information.

10. Check that the definition is successfully created in the **Module Configuration** Window.

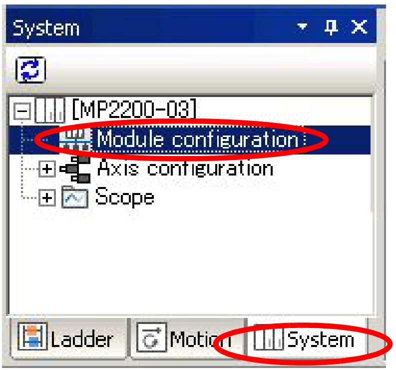
Overview of System Operation

5

[b] Self Configuration of Each Module

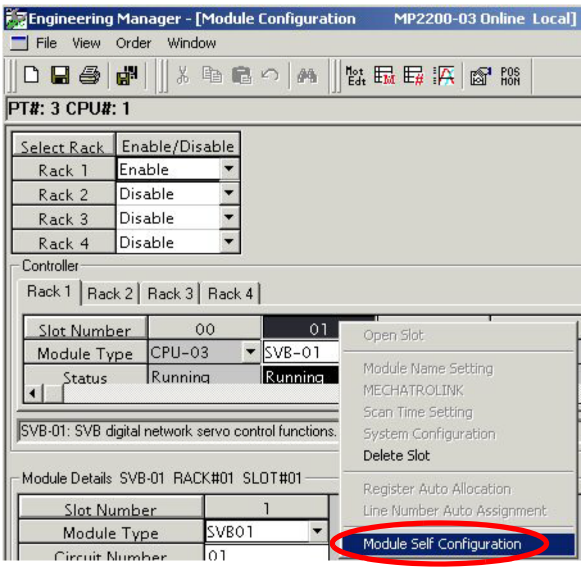
If modules or devices are added, self-configuration can be executed separately for the Module (port) that has been changed.
By performing the following operation, self-configuration will be executed for the selected slot.
Before performing the operation, turn ON the power supply of devices such as SERVOPACK.

1. Double-click ***System - Module configuration***.

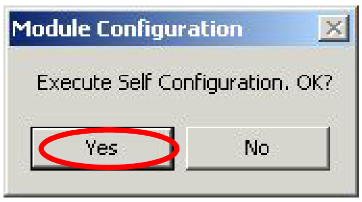


The **Engineering Manager** Window will start and the **Module Configuration** Window will appear.

2. Right-click the Module for which devices have been added and select **Module Self Configuration** from the pop menu to execute self-configuration.



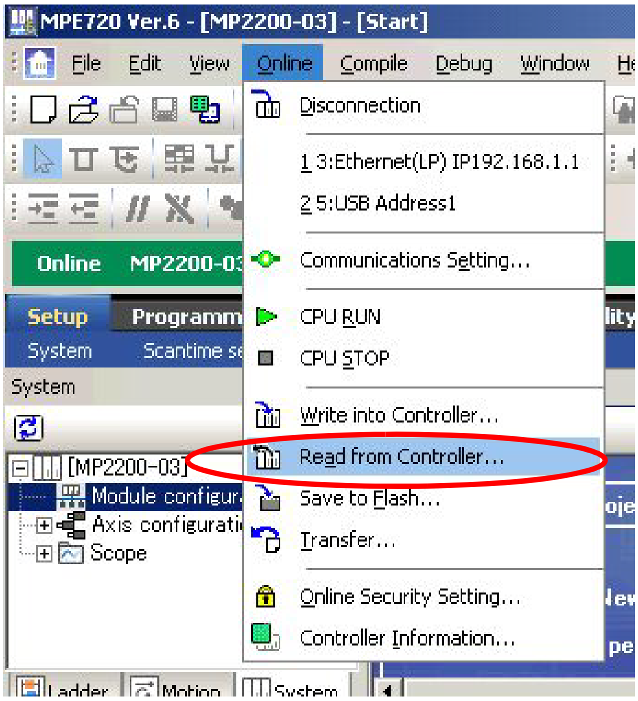
3. Click **Yes** for the following message.



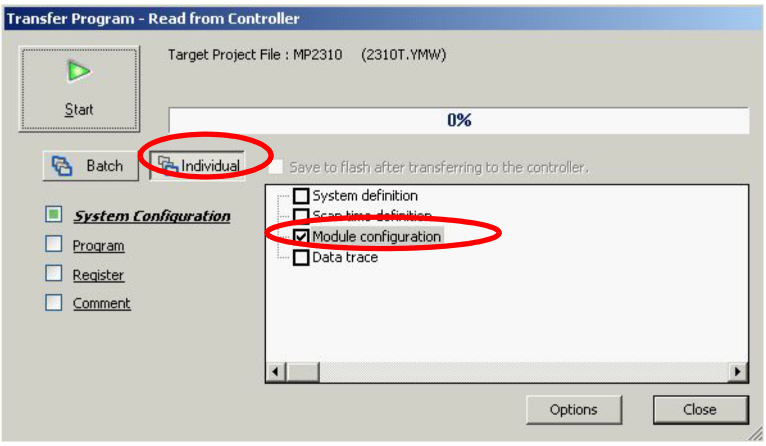
4. While running the self-configuration, the following message is shown.

5. If the following warning message is shown after performing step 4, the module configuration definitions for CPU and MPE720 may differ from each other. Continue to perform step 6. When the message is not shown, go to step 9.

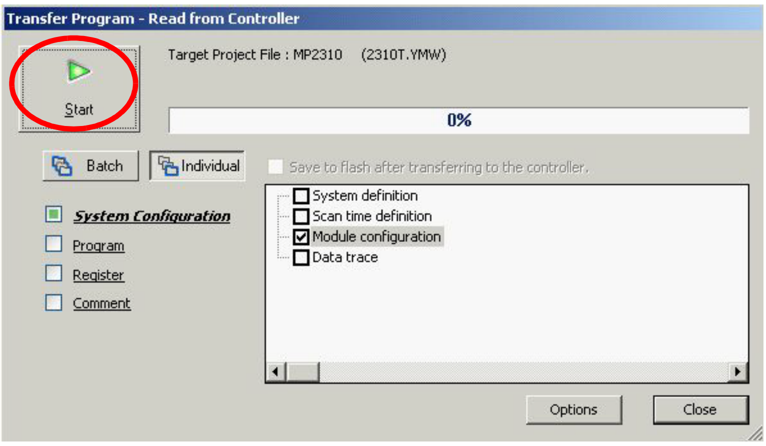
6. Select ***Online(O) - Read from Controller(A)***.



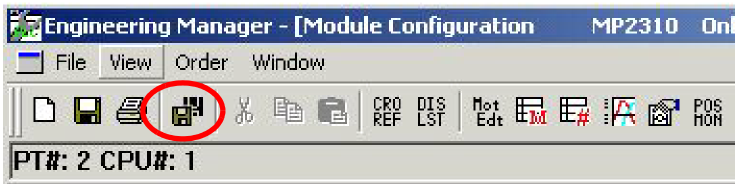
7. Click **Individual**, and only check **Module configuration**.



8. Click **Start** to read the module configuration definition from a controller.



9. Click the **Save & FLASH Save** Button to flash save the definition information.



10. In the **Module Configuration** Window, check that the definition has been created.

5.4.2 Definition Information Updated with Self-Configuration

Now, the definition information updated during executing the self-configuration and the module configuration definition example based on the module combination are as follows:

(1) Definition Information of CPU-01 and CPU-02 Module

[a] I/O Allocations

Item		Allocation
SVR	Motion Parameter	The last circuit number of the Motion Module will be allocated, and the input and output registers will be allocated according to the results. Example 1: SVA-01, SVB-01, SVC-01, and PO-1 Not Mounted • Start motion register: IW8000/OW8000 • End motion register: IW87FF/OW87FF (Input register: IW8000 to IW87FF Output register: OW8000 to OW87FF) Example 2: One SVB-01 Mounted • Start motion register: IW8800/OW8800 • End motion register: IW8FFF/OW8FFF (Input register: IW8800 to IW8FFF Output register: OW8800 to OW8FFF)

[b] SVR Definition

Type	No.	Name	Allocation
Fixed Parameter	0	Selection of Operation Modes	Axis unused
	1	Function Selection Flag 1	0000h
	4	Reference Unit Selection	pulse
	5	Number of Digits below Decimal Point	3
	6	Travel Distance per Machine Rotation	10000 reference unit
	8	Servo Motor Gear Ratio	1 rev (rotation)
	9	Machine Gear Ratio	1 rev (rotation)
	10	Infinite Length Axis Reset Position (POSMAX)	360000 reference unit
	34	Rated Motor Speed	3000 min^{-1}
	36	Number of Pulses per Motor Rotation	65536 pulse/rev
	42	Feedback Speed Movement Averaging Time Constant	10 ms
Setting Parameter	OW□□00	RUN Command Setting	0000h
	OW□□03	Function Setting 1	0011h
	OW□□08	Motion Command	0: No command
	OW□□09	Motion Command Control Flag	0000h
	OW□□0A	Motion Subcommand	0: No command
	OL□□0C	Torque/Thrust Reference Setting	0.00 %
	OL□□10	Speed Reference Setting	3000 × 10**n reference unit/min
	OL□□16	Secondly Speed Compensation	0.00 %
	OL□□1C	Position Reference Setting	0 reference unit
	OW□□31	Speed Compensation	0.00 %
	OL□□36	Straight Line Acceleration/ Acceleration Time Constant	0 ms
	OL□□38	Straight Line Deceleration/ Deceleration Time Constant	0 ms
	OW□□3A	Filter Time Constant	0.0 ms
	OW□□3B	Bias Speed for Index Deceleration/ Acceleration Filter	0 reference unit/s
	OW□□3D	Width of Starting Point Position Output	100 reference unit
	OL□□44	STEP Travel Distance	1000 reference unit
	OL□□48	Zero Point Position in Machine Coordinate System Offset	0 reference unit
	OL□□4A	Work Coordinate System Offset	0 reference unit
	OL□□4C	Number of POSMAX Turns Presetting Data	0 turn
	OW□□5C	Fixed Parameter Number	0

(2) Definition Information of CPU-03 and CPU-04 Module

[a] I/O Allocations

Item		Allocation
SVR	Motion Parameter	The last circuit number of the Motion Module will be allocated, and the input and output registers will be allocated according to the results. Example 1: SVA-01, SVB-01, SVC-01, and PO-1 Not Mounted • Start motion register: IW8000/OW8000 • End motion register: IW87FF/OW87FF (Input register: IW8000 to IW87FF Output register: OW8000 to OW87FF) Example 2: One SVB-01 Mounted • Start motion register: IW8800/OW8800 • End motion register: IW8FFF/OW8FFF (Input register: IW8800 to IW8FFF Output register: OW8800 to OW8FFF)
218IFA/218IFC		• Start I/O register: IW0000/OW0000 • End I/O register: IW07FF/OW07FF (Input register: IW0000 to IW07FF Output register: OW0000 to OW07FF)
M-EXECUTOR		UNDEFINED

[b] SVR Definition

Type	No.	Name	Allocation
Fixed Parameter	0	Selection of Operation Modes	Axis unused
	1	Function Selection Flag 1	0000h
	4	Reference Unit Selection	pulse
	5	Number of Digits below Decimal Point	3
	6	Travel Distance per Machine Rotation	10000 reference unit
	8	Servo Motor Gear Ratio	1 rev (rotation)
	9	Machine Gear Ratio	1 rev (rotation)
	10	Infinite Length Axis Reset Position (POSMAX)	360000 reference unit
	34	Rated Motor Speed	3000 min^{-1}
	36	Number of Pulses per Motor Rotation	65536 pulse/rev
	42	Feedback Speed Movement Averaging Time Constant	10 ms
Setting Parameter	OW□□00	RUN Command Setting	0000h
	OW□□03	Function Setting 1	0011h
	OW□□08	Motion Command	0: No command
	OW□□09	Motion Command Control Flag	0000h
	OW□□0A	Motion Subcommand	0: No command
	OL□□0C	Torque/Thrust Reference Setting	0.00 %
	OL□□10	Speed Reference Setting	3000×10^n reference unit/min
	OL□□16	Secondly Speed Compensation	0.00 %
	OL□□1C	Position Reference Setting	0 reference unit
	OW□□31	Speed Compensation	0.00 %
	OL□□36	Straight Line Acceleration/ Acceleration Time Constant	0 ms
	OL□□38	Straight Line Deceleration/ Deceleration Time Constant	0 ms
	OW□□3A	Filter Time Constant	0.0 ms
	OW□□3B	Bias Speed for Index Deceleration/ Acceleration Filter	0 reference unit/s
	OW□□3D	Width of Starting Point Position Output	100 reference unit
	OL□□44	STEP Travel Distance	1000 reference unit
	OL□□48	Zero Point Position in Machine Coordinate System Offset	0 reference unit
	OL□□4A	Work Coordinate System Offset	0 reference unit
	OL□□4C	Number of POSMAX Turns Presetting Data	0 turn
	OW□□5C	Fixed Parameter Number	0

[c] 218IFA/218IFC Definition

Item	Allocation
IP Address	192.168.1.1
Subnet Mask	255.255.255.0
Gateway IP Address	0.0.0.0
Equipment Name	“CONTROLLER NAME”
Engineering Port	9999 (UDP)
Response Time (Check & Monitor Time of MEMOBUS response)	0 s
Count or Retry	0

Note: The self-configuration allows you to connect with MPE720 for engineering transmission. In order to carry out MEMOBUS message transmission, manually use an automatic reception and I/O message communication separately, or MSG-SND/MSG-RCV functions are required.

[d] M-EXECUTOR Definition

Note: M-EXECUTOR is not defined for use with the MP2200. For details on how to define the M-EXECUTOR, refer to *7.3.1 Initializing the M-EXECUTOR Module*.

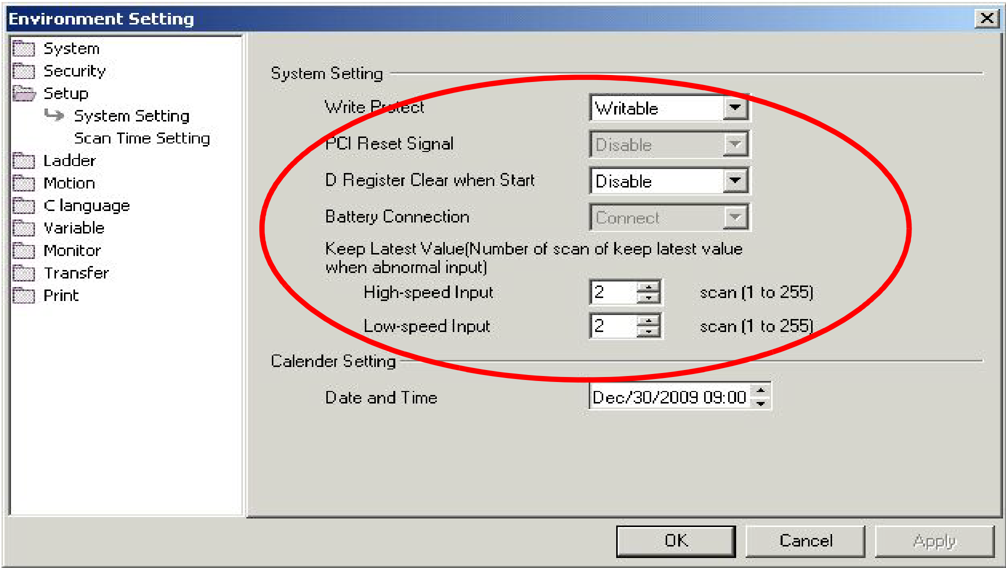
5.5 Application Precautions

This section explains precautions when a user definition file is configured/changed and when setting a scan time.

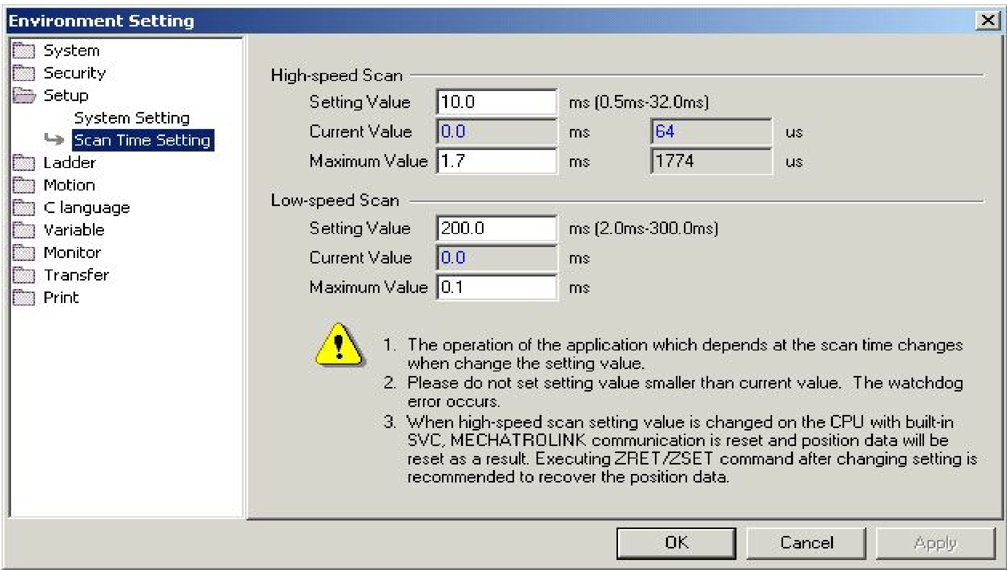
5.5.1 Setting or Changing User Definition Files

System settings, scan time settings, and module configuration definitions must be saved in flash memory (flash save). When a system setting, scan time setting, or module configuration definition is configured/changed, be sure to use MPE720 to flash save it. Note that when the MP2200 power supply is turn ON again without flash saving, the configured/changed data may be lost.

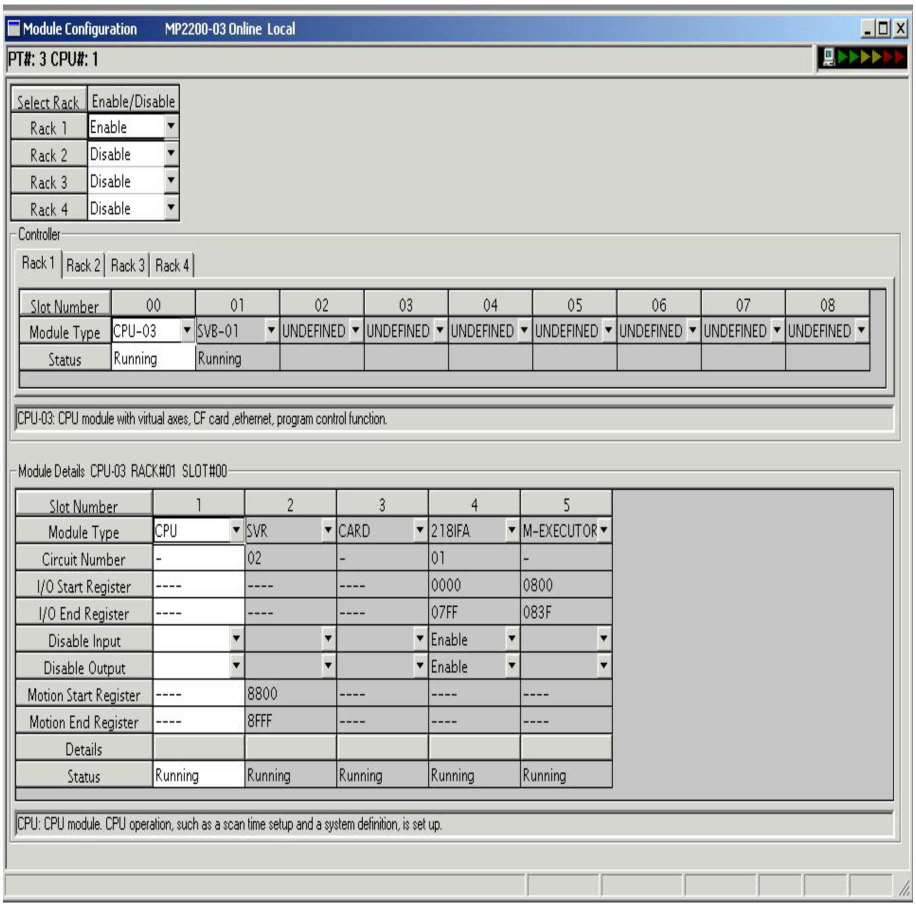
- System Setting



- Scan Time Setting



• Module Configuration Definition



5.5.2 Setting or Changing Module Configuration Definition Files

Observe the following precautions when setting or changing module configuration definition files.

- Always check to make sure that the mounted Module is the one that is defined.
- Be sure to save any new settings or changes to flash memory.
- After the settings or changes have been completed, turn the power supply to the MP2200 OFF and ON.

5.5.3 Setting and Changing the Scan Time

(1) Precautions When Setting or Changing the Scan Time

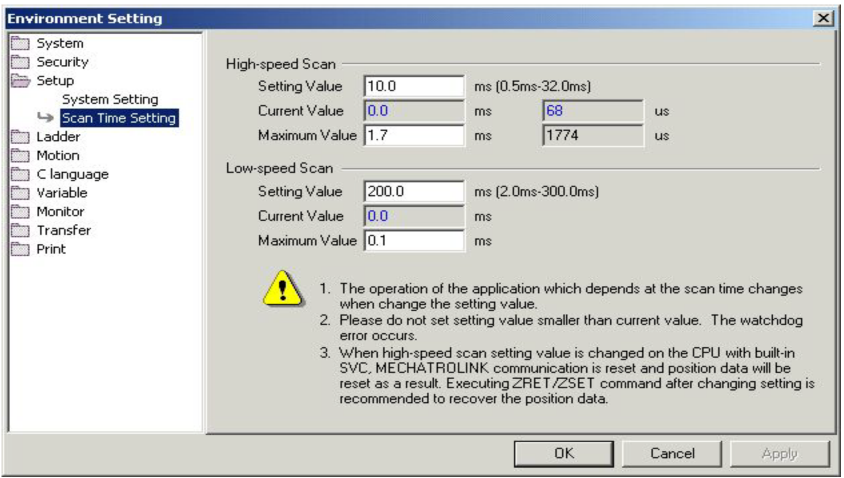
The scan time can be set and changed in the **Scan Time Setting** Window in the **Environmental Setting** Dialog Box on the MPE720.

Observe the following precautions when setting or changing the scan time.

- Set the set values of the scan time for both the high-speed (H) and low-speed (L) scans to at least the maximum time required to execute the scans. We recommend setting the set values of the scan time using the formula (set value − maximum time to execute scan) ≥ (0.2 × set values of the scan time), i.e., setting the set values of the scan time to at least 1.25 times the maximum times required to execute the scans.

Note: If the scan time is set too close to the maximum execution time for the scan, the refresh time for the window on the MPE720 will be very slow and communication timeouts may occur. If the maximum execution time exceeds the scan time set value, a watchdog timer timeout error will occur and the MP2200 system will stop.

- Do not change the scan time set value while the Servo is ON. Never change the setting while the axis is moving (while the motor is running). Otherwise an error may occur during motor operation (e.g., high-speed rotation).
- When the scan time is set or changed, be sure to save the data to flash memory.



Overview of System Operation

5

(2) Scan Time Set Value Examples (Settings Are the Same for the Low-speed Scan.)

■ 0.8-ms Maximum Scan Time and 1-ms Communication Cycle (MECHATROLINK-II Only)

High-speed scan set value ≥ (1.25 × 0.8) = 1 ms
High-speed scan set value = 1 ms, 2 ms, 3 ms . . . (integers starting from 1)

■ 1.4-ms Maximum Scan Time and 1-ms Communication Cycle (MECHATROLINK-II Only)

High-speed scan set value ≥ (1.25 × 1.4) = 1.75 ms
High-speed scan set value = 2 ms, 3 ms . . . (integers starting from 2)

■ 0.8-ms Maximum Scan Time and 2-ms Communication Cycle (MECHATROLINK-I/MECHATROLINK-II Only)

High-speed scan set value ≥ 1.25 × 0.8) = 1 ms
High-speed scan set value = 1 ms, 2 ms, 4 ms . . . (1 ms and integer multiples of 2 ms starting from 2 ms)

■ 1.4-ms Maximum Scan Time and 2-ms Communication Cycle (MECHATROLINK-I/MECHATROLINK-II Only)

High-speed scan set value ≥ (1.25 × 1.4) = 1.75 ms
High-speed scan set value = 2 ms, 4 ms . . . (integer multiples of 2 ms starting from 2 ms)

6

Built-in Ethernet Communication (Supported by the CPU-03 and CPU-04)

This chapter explains how to communicate with devices (PLCs, touch panels, etc.) connected to the CPU-03/CPU-04 in an MP2200-series Controller by Ethernet.

6.1 Communication Methods

The following table provides the appropriate mode of communication for each remote device and purpose.

Remote Device	Purpose	Communication Method	Remarks
Other MP Series	When other MP series device reads/writes the coil state or register content of CPU-03/ CPU-04	Uses the Extended MEMOBUS communication protocol. The remote device (master) side creates a ladder program using a MSG-SND function. The CPU-03/CPU-04 (slave) side uses an automatic receive function. (You do not need to create a ladder program.) ⇒ Refer to *6.2.1 When the CPU-03/CPU-04 Acts as Slave (automatic receive function is used)*	CPU-03/CPU-04 can communicate with only one master using the automatic receive function.
		Uses the Extended MEMOBUS communication protocol. The remote device (master) side creates a ladder program using a MSG-SND function. The CPU-03/CPU-04 (slave) side creates a ladder program using a MSG-RCV function. ⇒ Refer to *6.2.2 When the CPU-03/CPU-04 Acts as a Slave (ladder program which uses a MSG-RCV function)*	Communication with multiple masters is possible.
	When CPU-03/CPU-04 reads/writes the coil state or register content of other MP series equipment	Uses the Extended MEMOBUS communication protocol. The CPU-03/CPU-04 (master) side uses an I/O message communication function. (You do not need to create a ladder program.) The remote device (slave) side creates a ladder program using a MSG-RCV function. ⇒ Refer to *6.2.3 When CPU-03/CPU-04 Acts as Master (I/O message communication function is used)*	Only the holding register (M register) is capable of reading/ writing using an I/O message communication function. It can communicate with only one slave.
		Uses the Extended MEMOBUS communication protocol. A ladder program is created in the CPU-03/CPU-04 (master) using the MSG-SND function. The remote device (slave) side creates a ladder program using a MSG-RCV function. ⇒ Refer to *6.2.4 When the CPU-03/CPU-04 Acts as Master (ladder program which uses MSG-SND function)*	Registers other than the holding register are capable of reading/ writing. Communication with multiple slaves is enabled.
Touch Panel	When a touch panel reads/writes the coil state or register content of CPU-03/CPU-04	Uses the Extended MEMOBUS communication protocol. Set the protocol for the touch panel side to the Extended MEMOBUS protocol. The CPU-03/CPU-04 (slave) side uses an automatic receive function. (You do not need to create a ladder program.) ⇒ Refer to *6.3 Communication with Touch Panel*.	The CPU-03/CPU-04 can communicate with only one master using the automatic reception function.
PLC Manufactured by Mitsubishi Electric Corporation	When a PLC Manufactured by Mitsubishi Electric Corporation reads/writes the CPU-03/CPU-04 register content.	Uses the MELSEC communication protocol. The remote device (master) side creates a ladder program using a BUFSND function. The CPU-03/CPU-04 (slave) side uses an automatic receive function. (You do not need to create a ladder program.) ⇒ Refer to *6.4.1 When the CPU-03/CPU-04 Acts as Slave (automatic receive function is used)*	The CPU-03/CPU-04 can communicate with only one master when using the automatic receive function.
	When an CPU-03/ CPU-04 reads/writes the relay state or register content of PLC Manufactured by Mitsubishi Electric Corporation.	Uses the MELSEC communication protocol. The CPU-03/CPU-04 (master) side uses an I/O message communication function. (You do not need to create a ladder program.) The remote device (slave) side needs to set the network parameters. (You do not need to create a ladder program.) ⇒ Refer to *6.4.2 When the CPU-03/CPU-04 Acts as Master (I/O message communication function is used)*	The MP2200 can communicate with only one slave when using the I/O message communication function.

Note: For information on *applications where the remote device* is a PLC or Windows computer, refer to the *Machine Controller MP2300S/MP2310/MP2400 Basic Module Supplement for Ethernet Communications* (manual no: SIEP C880700 37). You can download this manual from the Yaskawa e-mechatronics website (http://www.e-mechatronics.com).

6.2 Communication with Other MP Series

When Ethernet communication is carried out between the CPU-03/CPU-04 and other MP series, the Extended MEMOBUS protocol is used as a communication protocol. The Extended MEMOBUS protocol allows the master to read/write the slave register contents.
This section explains communications when the CPU-03/CPU-04 acts as a slave and a master respectively.
When the CPU-03/CPU-04 acts as a slave, this chapter explains communications using an automatic receive function and a ladder program with the MSG-RCV function.
When the CPU-03/CPU-04 acts as a master, this chapter explains communications using an I/O message communication function and a ladder program with the MSG-SND function.

6.2.1 When the CPU-03/CPU-04 Acts as Slave (automatic receive function is used)

This section explains how to communicate with the MP2300 message send function (MSG-SND) using the CPU-03/CPU-04 automatic receive function.

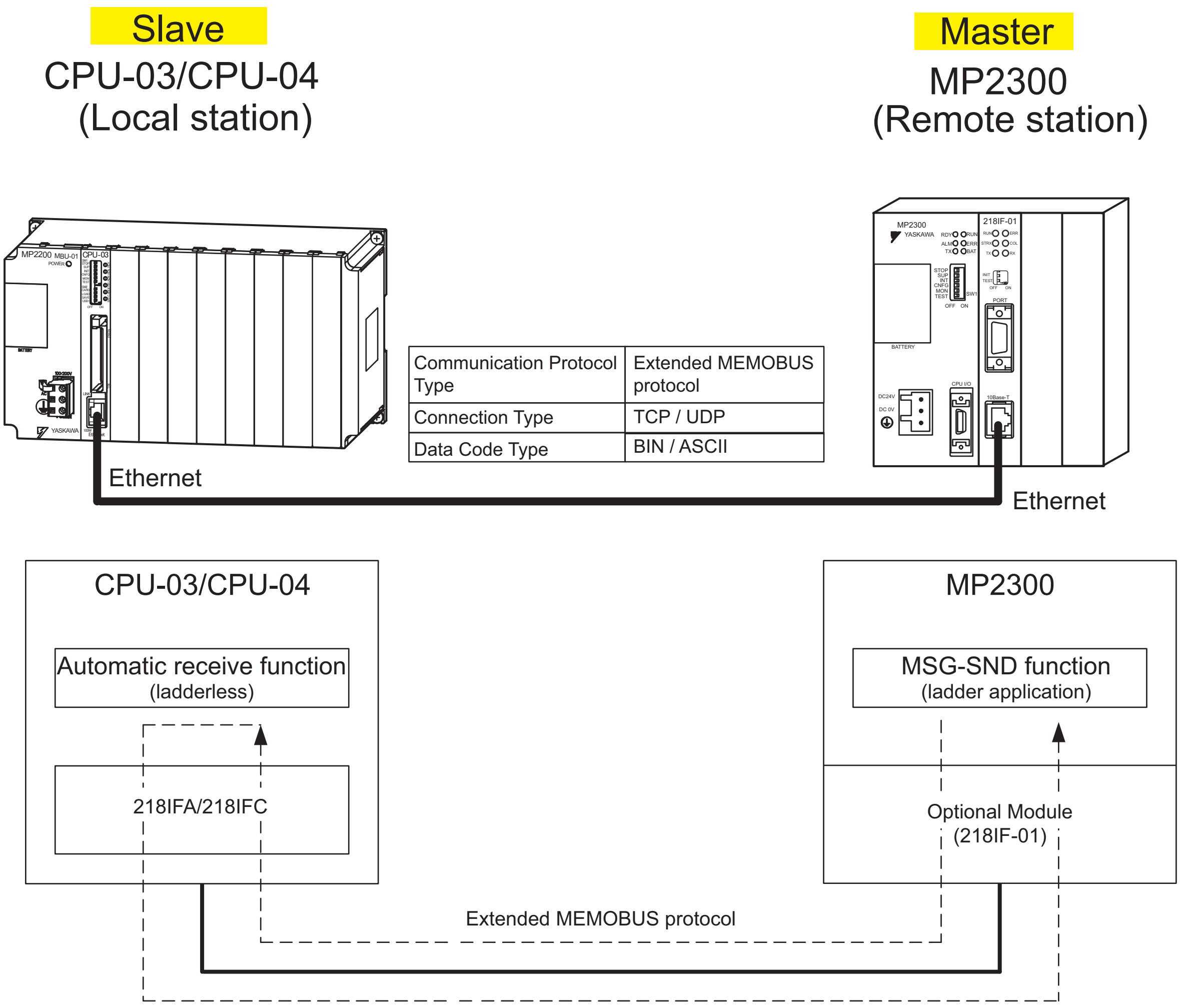
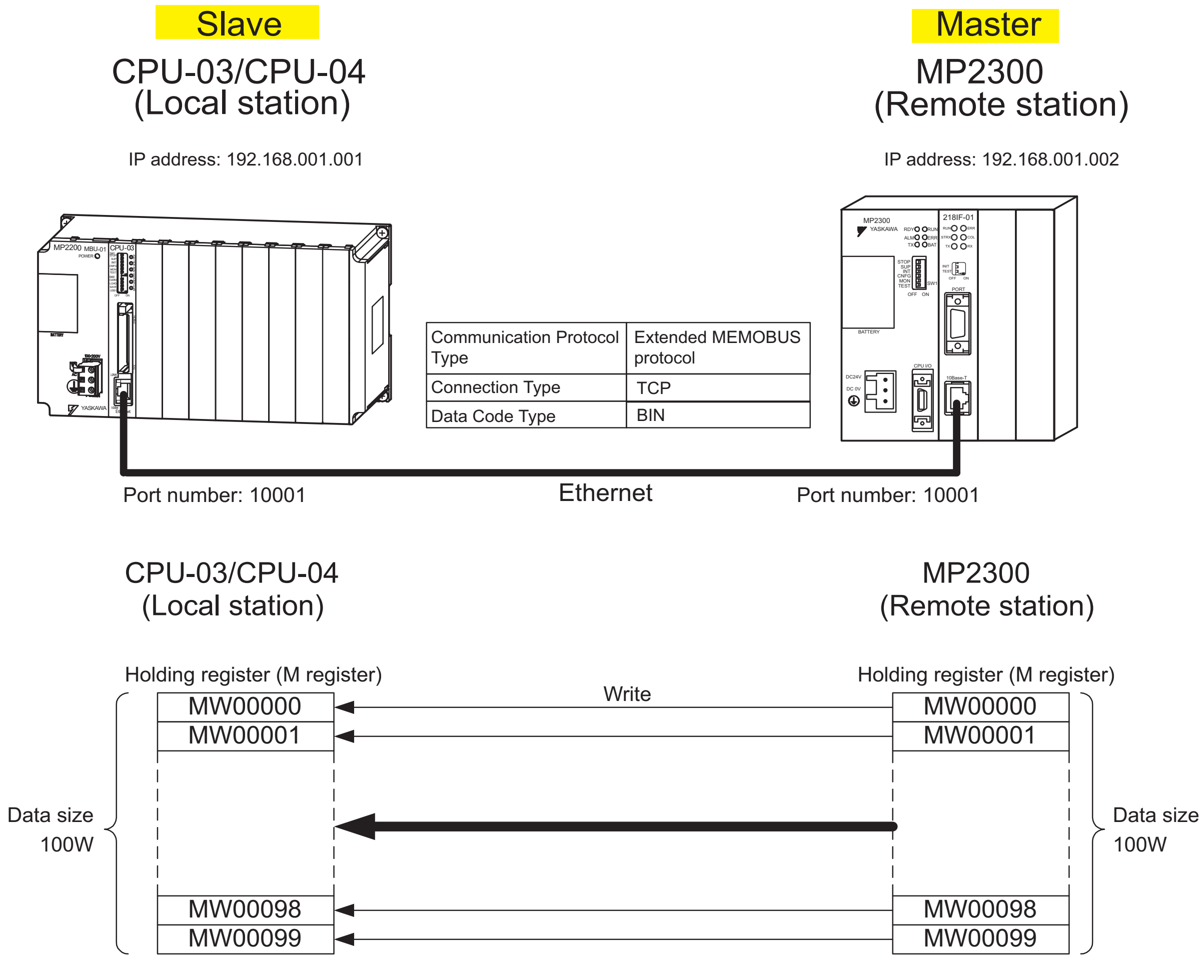


Fig. 6.1 Message Flow with MP2300 when Automatic Receive Function Is Used

■ Setting Example

The following figure illustrates how the content of the MP2300 (master) holding register (MW00000 to MW00099) is written into the CPU-03/CPU-04 (slave) holding register (MW00000 to MW00099).

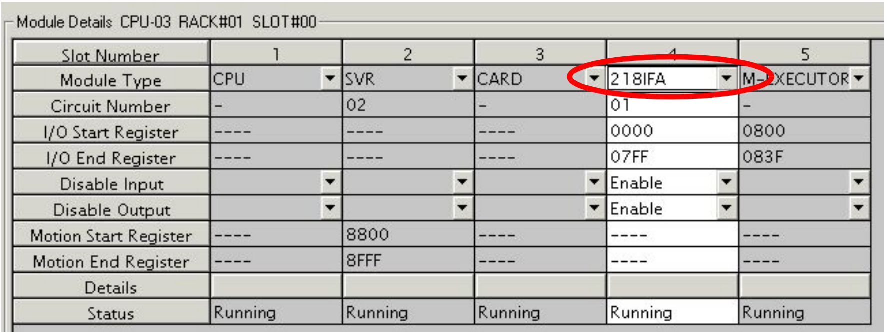


The setup procedure is explained in the following pages.

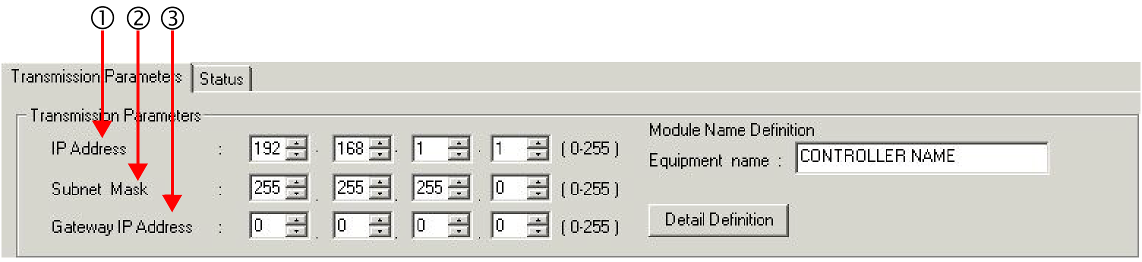
(1) How to Set up the CPU-03/CPU-04 Side

If the setting of transmission parameters (IP address, subnet mask) is already completed, start from step 3.

1. Double-click the **218IFA** Tab (**218IFC** Tab for the CPU-04) in the **Module Details** Area in the CPU-03 **Module Configuration** Window.



2. Set transmission parameters.



■ How to set up transmission parameters

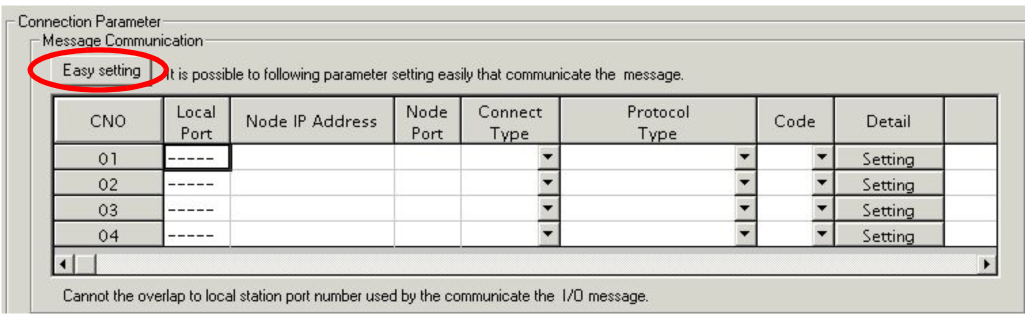
① Set **IP Address** (to “192.168.001.001,” for example).
② Set **Subnet Mask** (to “255.255.255.000,” for example).
③ Set **Gateway IP Address** (to “000.000.000.000,” for example).

■ Caution

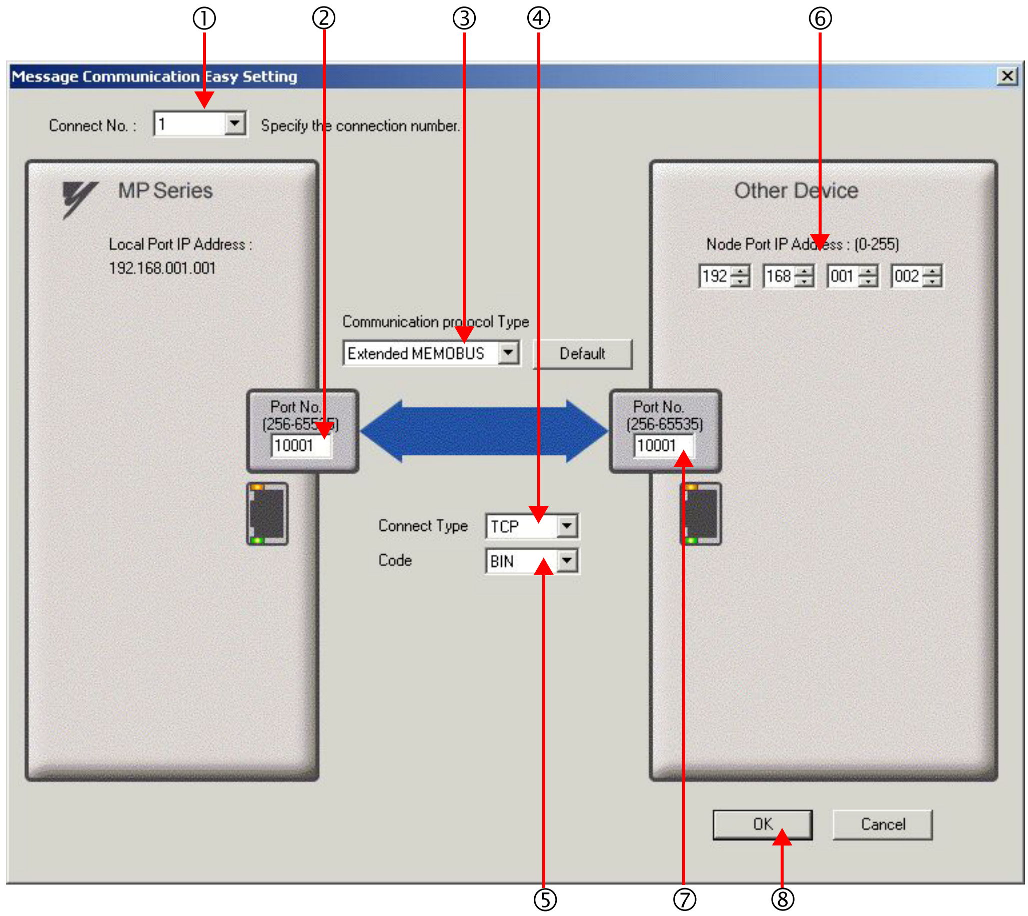
Set up a unique IP address in the network.
For the IP address, check with your network administrator.

---

3. Click the **Easy Setting** Button in the **Message Communication** Area of the connection parameter setting.



4. Set a communication setting in the **Message Communication Easy Setting** Dialog Box.



■ How to set up in the **Message Communication Easy Setting** Dialog Box

① When automatic receive function is used, select "1" for the **Connect No.**
② Set **Port No.** of the CPU-03/CPU-04 side ("10001," for example).
③ Select **Extended MEMOBUS** for the **Communication protocol Type**, and click **Default** Button.
④ Select **Connect Type** (TCP, for example).
⑤ Select **Code** (BIN, for example).
⑥ Set **Node Port IP Address** of the other device (MP2300) to be connected (to "192.168.001.002," for example).
⑦ Set **Port No.** of the other device (MP2300) to be connected (to "10001," for example).
⑧ Click **OK** Button.

■ Caution

When message functions (MSG-SND, MSG-RCV) are used with the connection number 01, disable the automatic receive function. If message functions are used while the automatic receive function is enabled, communications will not function properly.

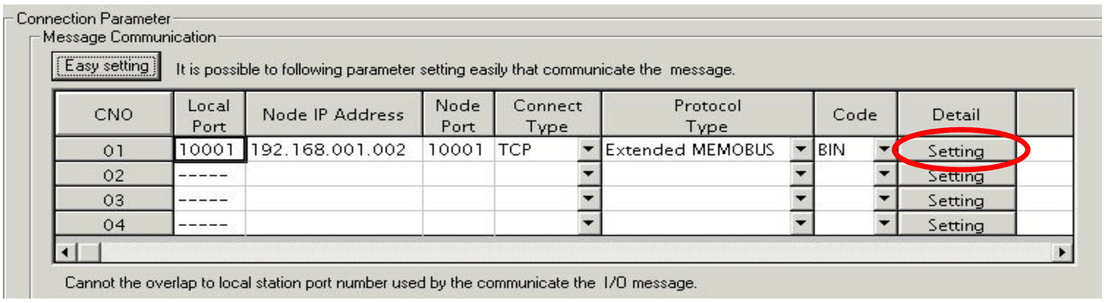
Note: The automatic receive function with a connection number 01 is set to "Enable" by default.

5. Click **Yes** in the confirmation dialog of the parameter setting.

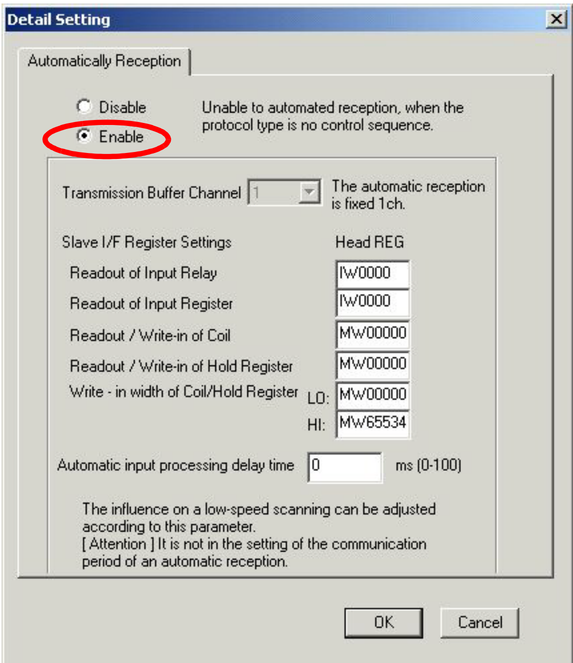
■ Caution

Note that when a parameter with the same connection number is already set and you click **Yes** in the confirmation dialog of the parameter setting, the setting will be overwritten by the parameter configured in the **Message Communication Easy Setting** Dialog Box.

6. Check the setting value and click the **Setting** Button in the **Detail** Column.



7. Select the **Enable** Option on the **Automatically Reception** Tab Page of the **Detail Setting** Dialog Box and then click the **OK** Button.



Note: For more information on **Slave I/F Register Settings** and **Automatic input precessing delay time**, refer to *2.5.6 (4) [b] ■ Automatically Reception Dialog Box for Message Communication*.

Now, the automatic receive function is set up when the CPU-03/CPU-04 acts as a slave.

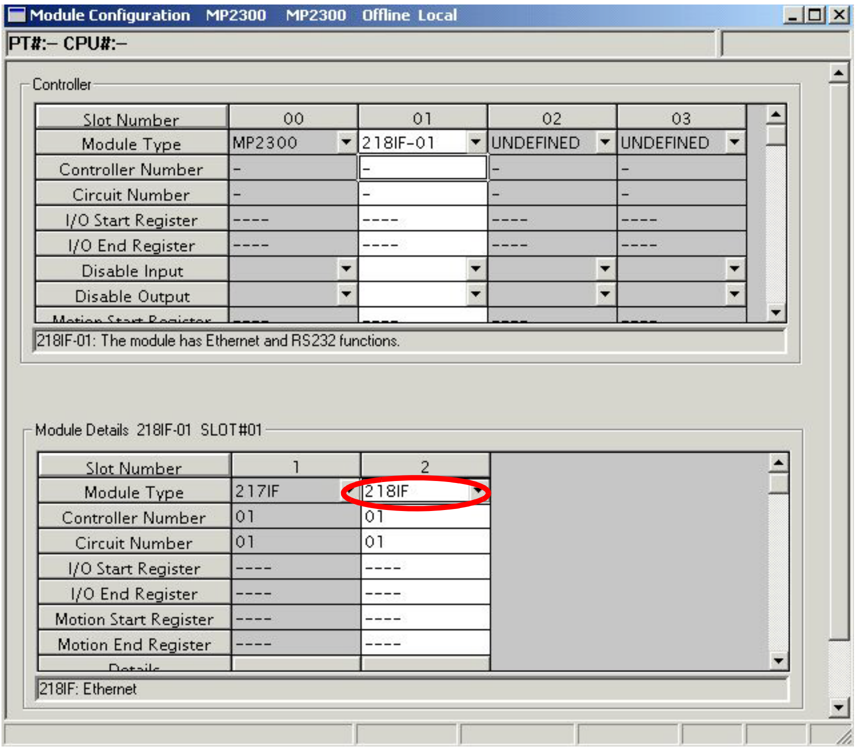
■ Caution

When any transmission or connection parameter is changed, the change will be reflected after it has been saved in flash memory and the power is turned ON again.

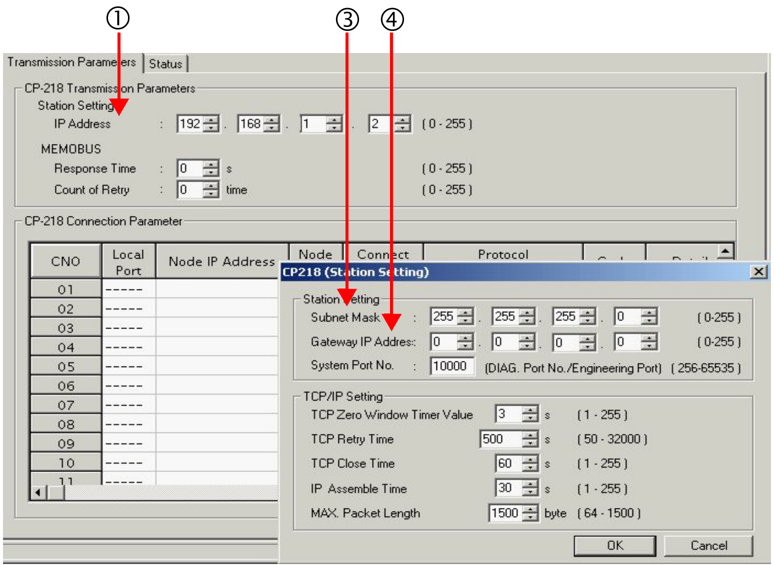
(2) How to Set up the Remote Device (MP2300) to Be Connected

If the setting of transmission parameters (IP address, subnet mask) is already completed, start from step 3.

1. Double-click the **218IF** Tab in the **Module Details** Area of the **Module Configuration** Window.



2. Set transmission parameters.



■ How to set up transmission parameters

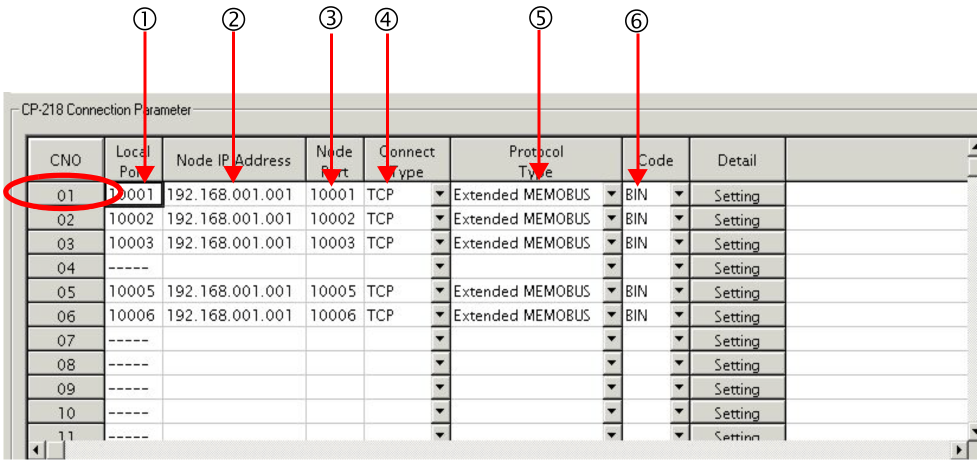
① Set **IP Address** (“192.168.001.001,” for example).
② Select ***Edit - Local Station: TCP/IP Setting*** in the Engineering Manager Window the **Detail Setting** Dialog Box will appear.
③ Set **Subnet Mask** (“255.255.255.000,” for example).
④ Set **Gateway IP Address** (“000.000.000.000,” for example).

■ Caution

Set up a unique IP address in the network.
For the IP address, check with your network administrator.

---

3. Set connection parameters.



■ How to set up with a connection number 01 in the **CP-218 Connection Parameter** Area

① Set **Local Port** to the port number used in the MP2300 side (“10001,” for example).
② Set **Node IP Address** to the IP address configured in the CPU-03/CPU-04 side.
③ Set **Node Port** to the port number configured in the CPU-03/CPU-04 side (“10001,” for example).
④ Select **Connect Type** (TCP, for example).
⑤ Select **Extended MEMOBUS** for **Protocol Type**.
⑥ Select **Code** (BIN, for example).

■ Caution

When any transmission or connection parameter is changed, the change will be reflected after it has been saved in flash memory and the power turned is ON again.

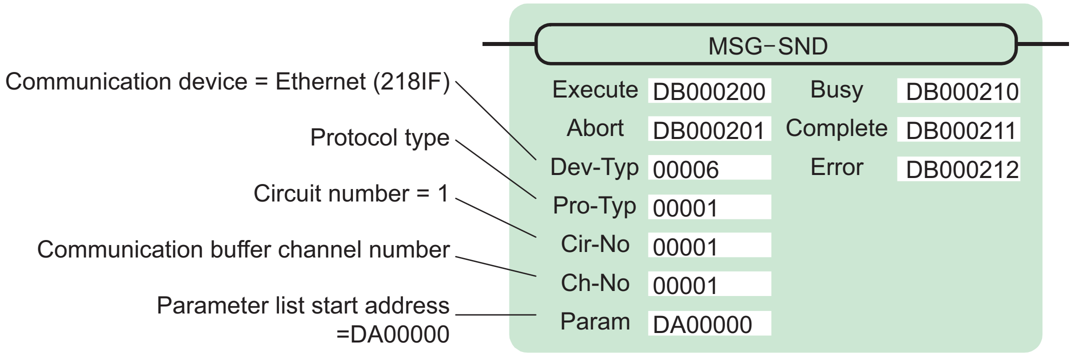
---

4. Create a ladder program with a message send function (MSG-SND).

A ladder program for transmitting messages to/from the remote device (MP2300) side is shown as follows:

■ Message send function (MSG-SND)

This system function is required for transmitting messages. Message transmission is carried out by describing and executing this message send function in a ladder program.



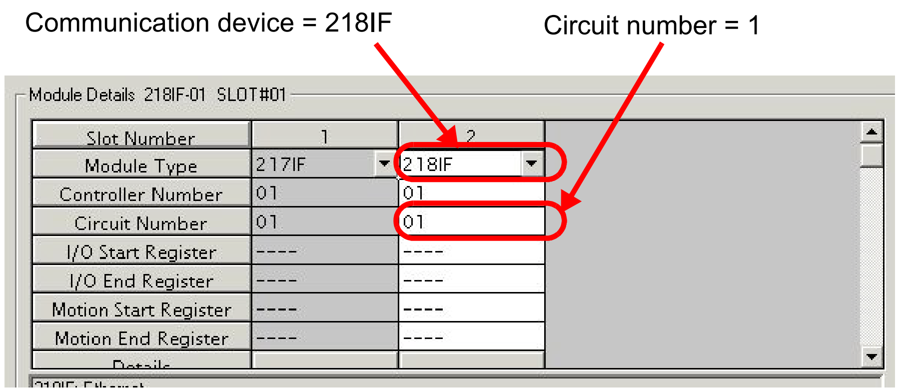


Fig. 6.2 MPE720 Module Configuration Window

■ Input/output definitions for message send function

The input/output definitions for the message send function are explained as follows:

Table 6.1 Input/Output Definitions for Message Send Function

I/O Definition	No.	Name	Setting Example	Explanation
Input Item	1	Execute	DB000200	Executes a transmission When the Execute is ON, the message is transmitted.
	2	Abort	DB000201	Aborts a transmission When the Abort is ON, the message transmission is forcibly stopped.
	3	Dev-Typ	00006	Communication device type Specify the type of the communication device used in transmission. When Ethernet (218IF) is used, specify “6”.
	4	Pro-Typ	00001	Communication protocol Specify the type of the communication protocol. MEMOBUS(*1) = 1, non-procedure 1(*2) = 2, non-procedure 2(*2) = 3
	5	Cir-No	00001	Circuit number Specify the circuit number of the communication device. Specify it in accordance with the circuit number displayed in the MPE720 **Module Configuration** Window.
	6	Ch-No	00001	Communication buffer channel number Specify the channel number of the communication buffer. When Ethernet (218IF) is used, specify it in the range between “1” and “10”. Note: Set up a unique channel number in the circuit.
	7	Param	DA00000	Parameter list start address Specify the start address of the parameter list. For the Parameter List, 17 words are automatically assigned from the configured address.
Output Item	1	Busy	DB000210	In process Busy is turned ON while executing a message transmission or forced abort process.
	2	Complete	DB000211	Process completed When a message transmission or abort process is properly completed, Complete will turn ON only for one scan.
	3	Error	DB000212	Error occurred When an error occurs, the Error will turn ON only for one scan.

* 1. When transmitting in MEMOBUS, Extended MEMOBUS, MELSEC, or MODBUS/TCP protocol, set the communication protocol (Pro-Typ) to MEMOBUS (=1). The communication device automatically converts the protocol.

* 2. Non-procedure 1: In non-procedural communications, data is transmitted on a per-word basis.
Non-procedure 2: In non-procedural communications, data is transmitted on a per-byte basis.

■ Parameter list setting example for the message send function

An example of a parameter list setting when writing 100 words of data from MW00000 to the destination using the connection with a connection number = 1 follows:

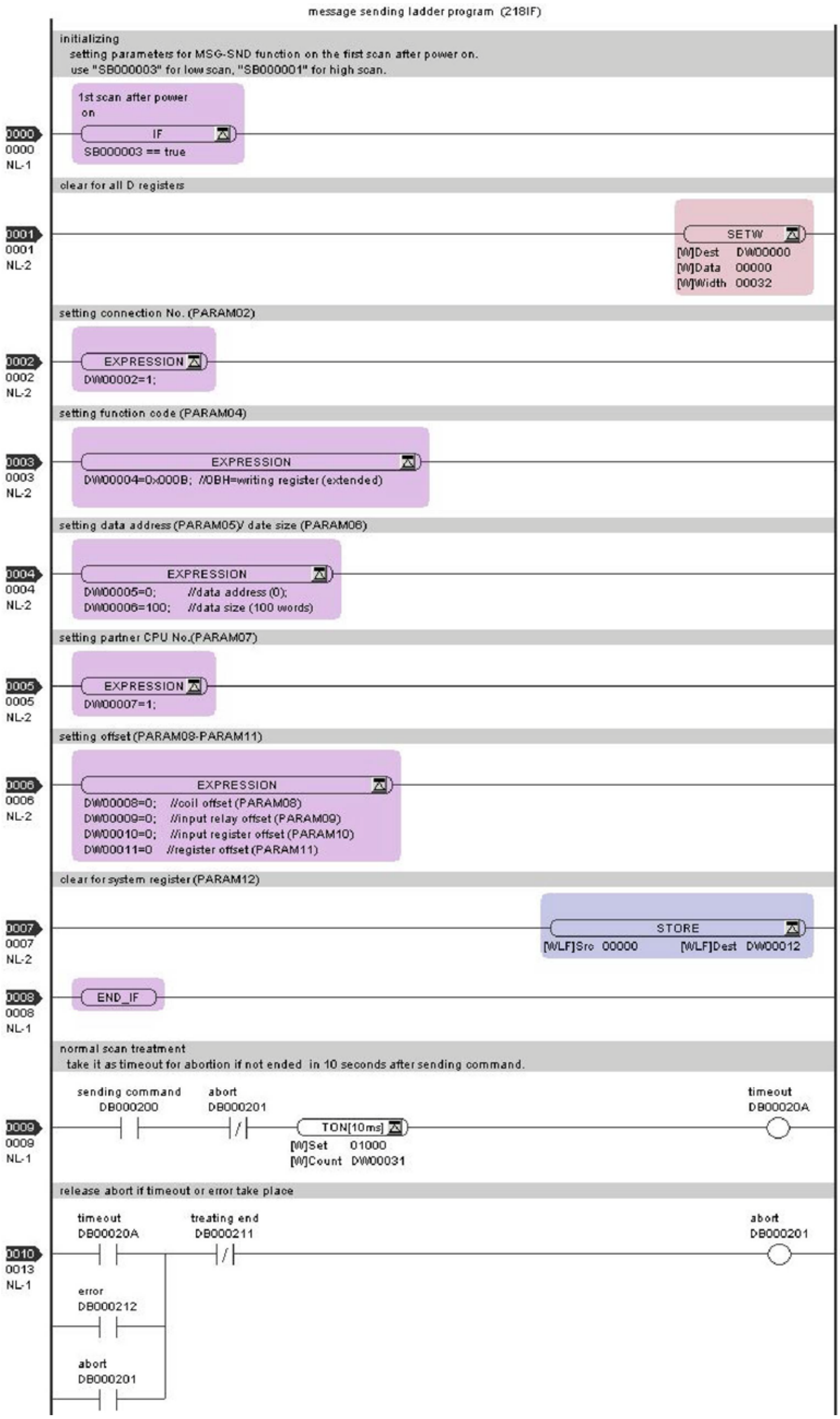
Table 6.2 Sample Parameter List Setting (parameter list start address Param=DA00000)

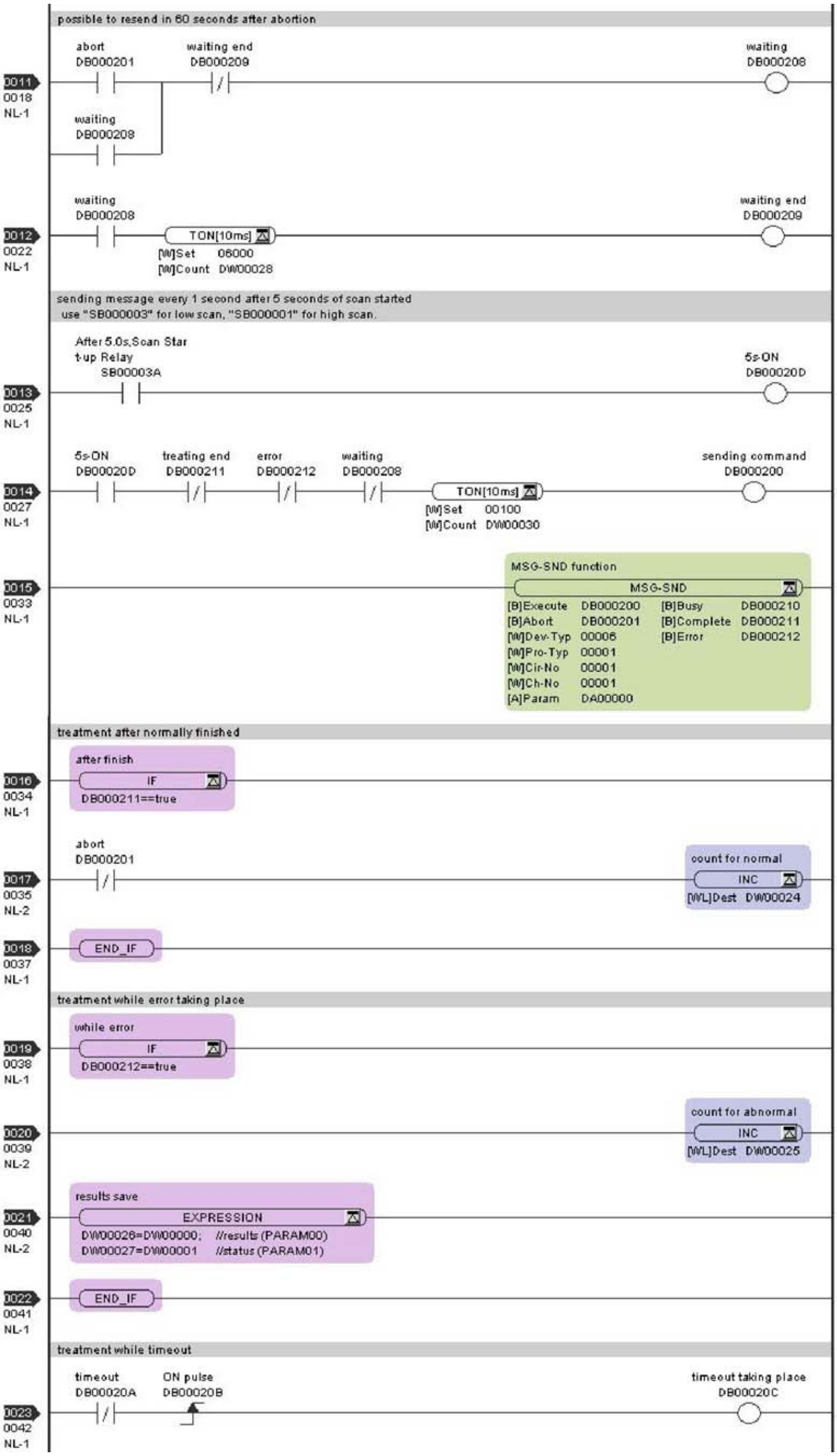
Register Number	Setting Value	Parameter Number	IN/OUT	Remarks
DW00000	–	PARAM00	OUT	Process result
DW00001	–	PARAM01	OUT	Status
DW00002	00001	PARAM02	IN	Connection number = 1
DW00003	–	PARAM03	IN	Option (Setting unnecessary)
DW00004	000BH	PARAM04	IN	Function code = 0BH (Writes to holding register)
DW00005	00000	PARAM05	IN	Data address = 0 (Starting from MW00000)
DW00006	00100	PARAM06	IN	Data size = 100 (100 words)
DW00007	00001	PARAM07	IN	Remote CPU number = 1
DW00008	00000	PARAM08	IN	Coil offset = 0 word
DW00009	00000	PARAM09	IN	Input relay offset = 0 word
DW00010	00000	PARAM10	IN	Input register offset = 0 word
DW00011	00000	PARAM11	IN	Holding register offset = 0 word
DW00012	–	PARAM12	SYS	Reserved by the system. (Zero clear at startup)
DW00013	–	PARAM13	SYS	Reserved by the system.
DW00014	–	PARAM14	SYS	Reserved by the system.
DW00015	–	PARAM15	SYS	Reserved by the system.
DW00016	–	PARAM16	SYS	Reserved by the system.

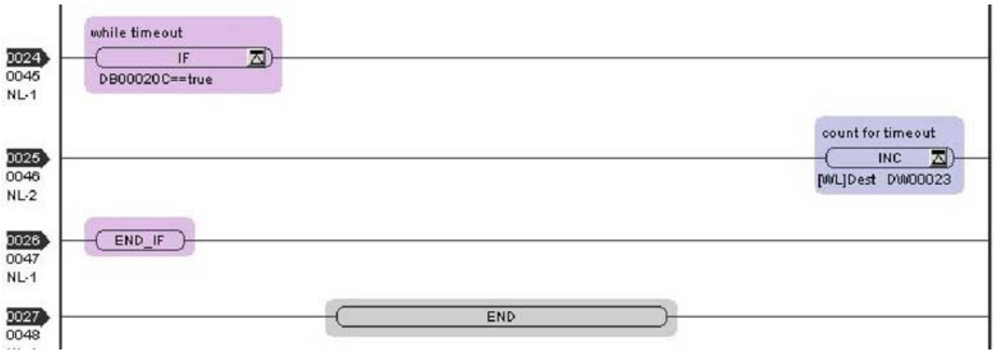
Note: IN: Input, OUT: Output, SYS: For system use

■ Example of Using the Message Send Function in a Ladder Program

Here is one example of the message send function through Ethernet (218IF).




possible to resend in 60 seconds after abortion
abort DB000201
waiting end DB000209
waiting DB000208
0011 0018 NL-1
waiting DB000208
waiting DB000208
TON[10ms]
[W]Set 06000
[W]Count DW00028
waiting end DB000209
0012 0022 NL-1
sending message every 1 second after 5 seconds of scan started
use "SB000003" for low scan, "SB000001" for high scan.
After 5.0s,Scan Start-up Relay SB00003A
5s-ON DB00020D
0013 0025 NL-1
5s-ON DB00020D
treating end DB000211
error DB000212
waiting DB000208
TON[10ms]
[W]Set 00100
[W]Count DW00030
sending command DB000200
0014 0027 NL-1
MSG-SND function
MSG-SND
[B]Execute DB000200
[B]Abort DB000201
[W]Dev-Typ 00006
[W]Pro-Typ 00001
[W]Cir-No 00001
[W]Ch-No 00001
[A]Param DA00000
[B]Busy DB000210
[B]Complete DB000211
[B]Error DB000212
0015 0033 NL-1
treatment after normally finished
after finish
IF
DB000211==true
0016 0034 NL-1
abort DB000201
count for normal
INC
[WL]Dest DW00024
0017 0035 NL-2
END_IF
0018 0037 NL-1
treatment while error taking place
while error
IF
DB000212==true
0019 0038 NL-1
count for abnormal
INC
[WL]Dest DW00025
0020 0039 NL-2
results save
EXPRESSION
DW00026=DW00000; //results (PARAM00)
DW00027=DW00001 //status (PARAM01)
0021 0040 NL-2
END_IF
0022 0041 NL-1
treatment while timeout
timeout DB00020A
ON pulse DB00020B
timeout taking place DB00020C
0023 0042 NL-1



The communication setting and the ladder program creation are now finished, when MP2300 acts as a master.

(3) How to Start Communications

1. The CPU-03/CPU-04 side starts to receive the messages.

When the automatic receive function is used, the message receive operation starts automatically.

2. Turn Execute ON for the message send function in the MP2300 side to transmit messages.

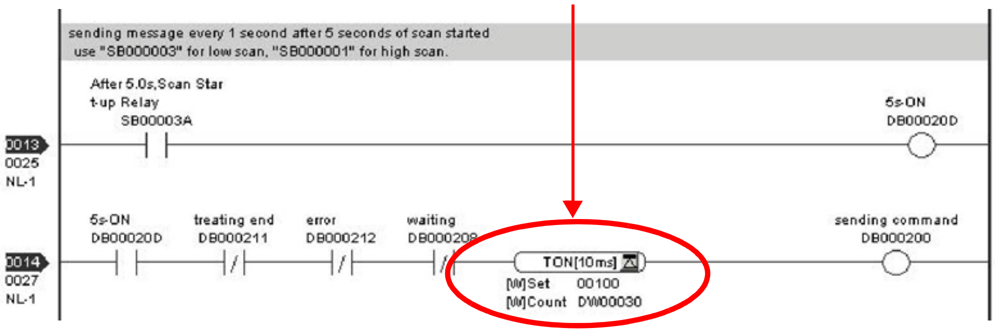
Messages are transmitted by turning ON the register (DB000200, for example), configured in Execute of the message send function, and communication with the CPU-03/CPU-04 starts.

Table 6.3 Input/Output Definition for Message Send Function

I/O Definition	No.	Name	Setting Example	Content
Input Item	1	Execute	DB000200	Executes a transmission When Execute is ON, the message transmission will be carried out.

The sample ladder program is created to transmit a message every one second when five seconds have elapsed after the low-speed scan (or high-speed scan) startup.
To change the message transmission interval, change the timer value ①.

①



6.2.2 When the CPU-03/CPU-04 Acts as a Slave (ladder program which uses a MSG-RCV function)

The CPU-03/CPU-04 can communicate with only one master when using the automatic receive function.
To communicate with more than one master, use a ladder program with a message receive function (MSG-RCV) at the CPU-03/CPU-04 end. You can use the message receive function (MSG-RCV) as well as the automatic receive function by keeping connections separate from each other.

This section explains how to communicate with an CPU-03/CPU-04 message send function (MSG-SND) using the MP2300 message receive function (MSG-RCV).

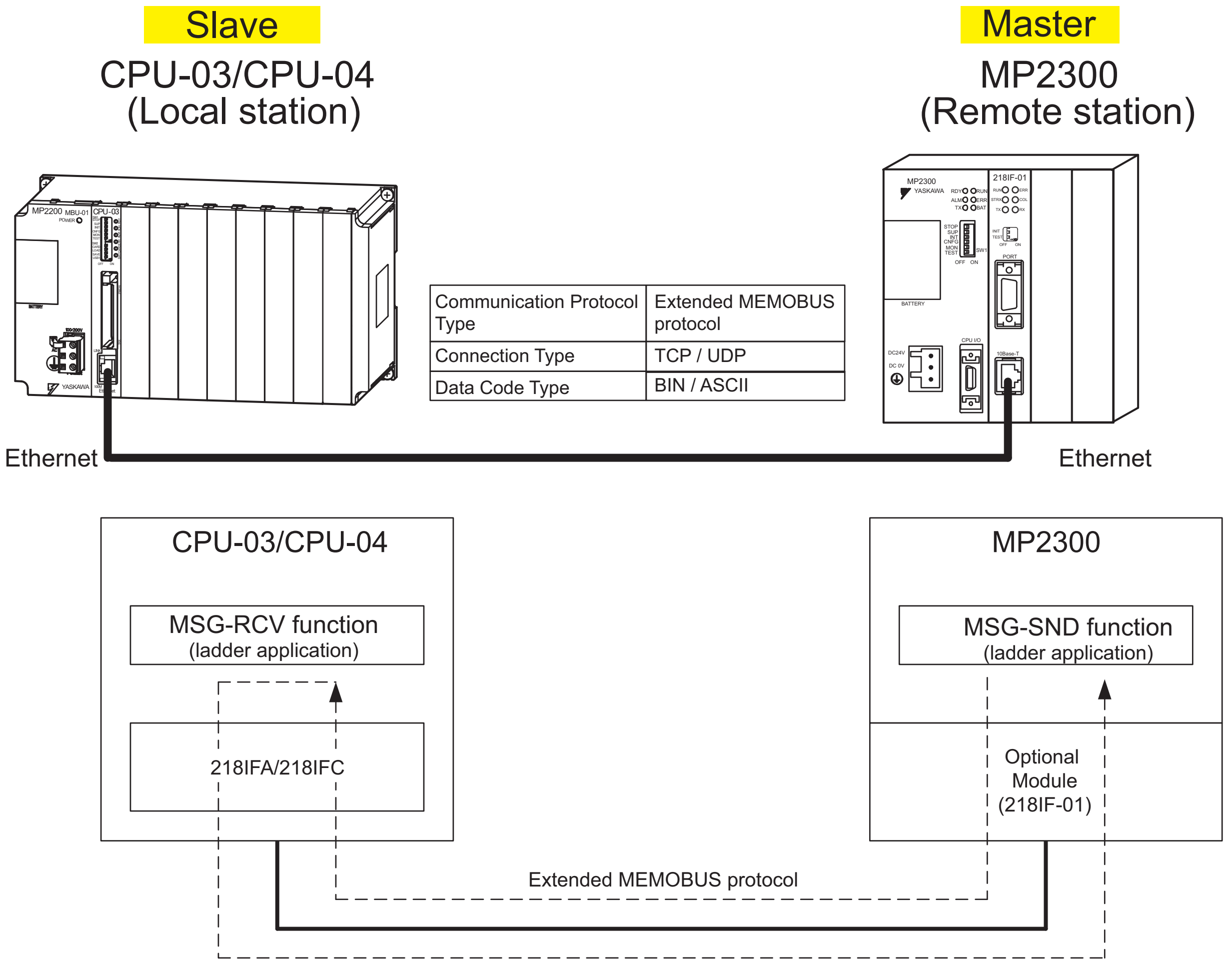
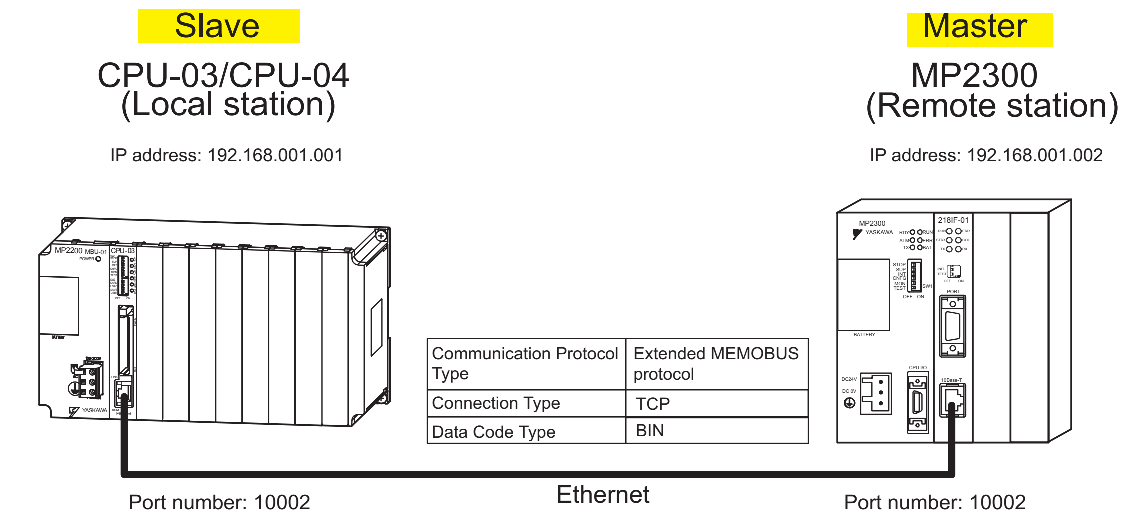


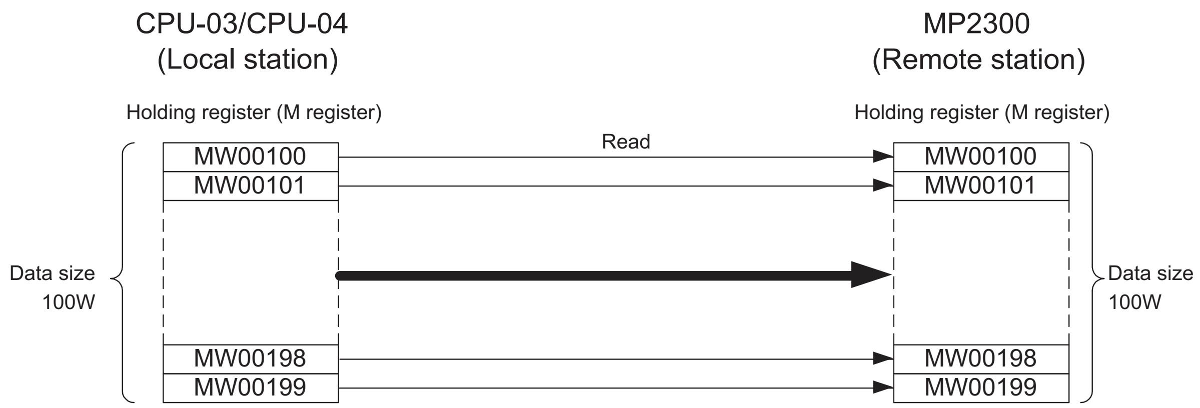
Fig. 6.3 Message Flow with MP2300 when Message Receive Function (MSG-RCV) Is Used

■ Setting Example

The following figure illustrates one example of writing the contents of the MP2300 (master) holding register (MW00100 to MW00199) into the CPU-03/CPU-04 (slave) holding register (MW00100 to MW00199).



Communication Protocol Type	Extended MEMOBUS protocol
Connection Type	TCP
Data Code Type	BIN

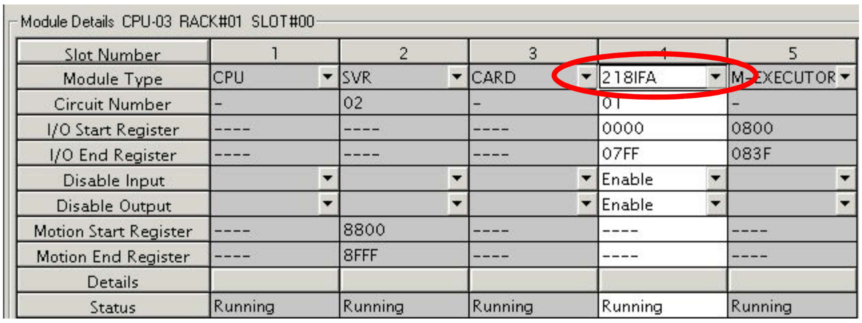


The setup procedure is explained in the following pages.

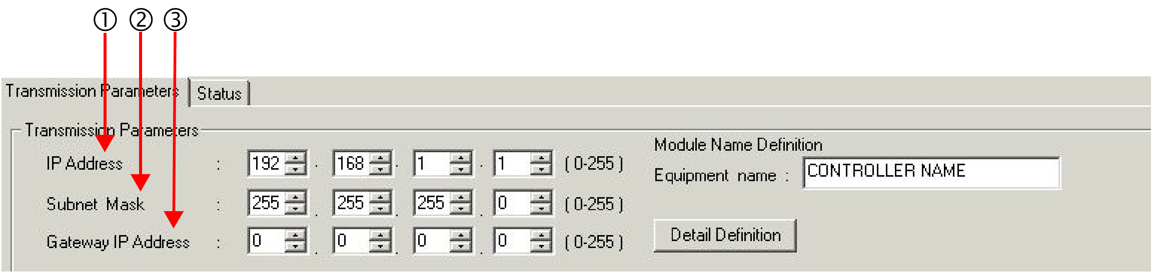
(1) How to Set up the CPU-03/CPU-04 Side

If the setting of transmission parameters (IP address, subnet mask) is already completed, start from step 3.

1. Double-click the **218IFA** Tab (**218IFC** Tab for the CPU-04) in the **Module Details** Area in the CPU-03 **Module Configuration** Window.



2. Set transmission parameters.



■ How to set up transmission parameters

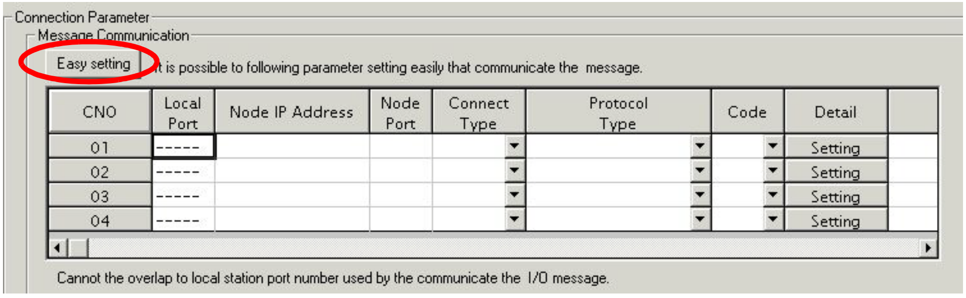
① Set **IP Address** (“192.168.001.001,” for example).
② Set **Subnet Mask** (“255.255.255.000,” for example).
③ Set **Gateway IP Address** (“000.000.000.000,” for example).

■ Caution

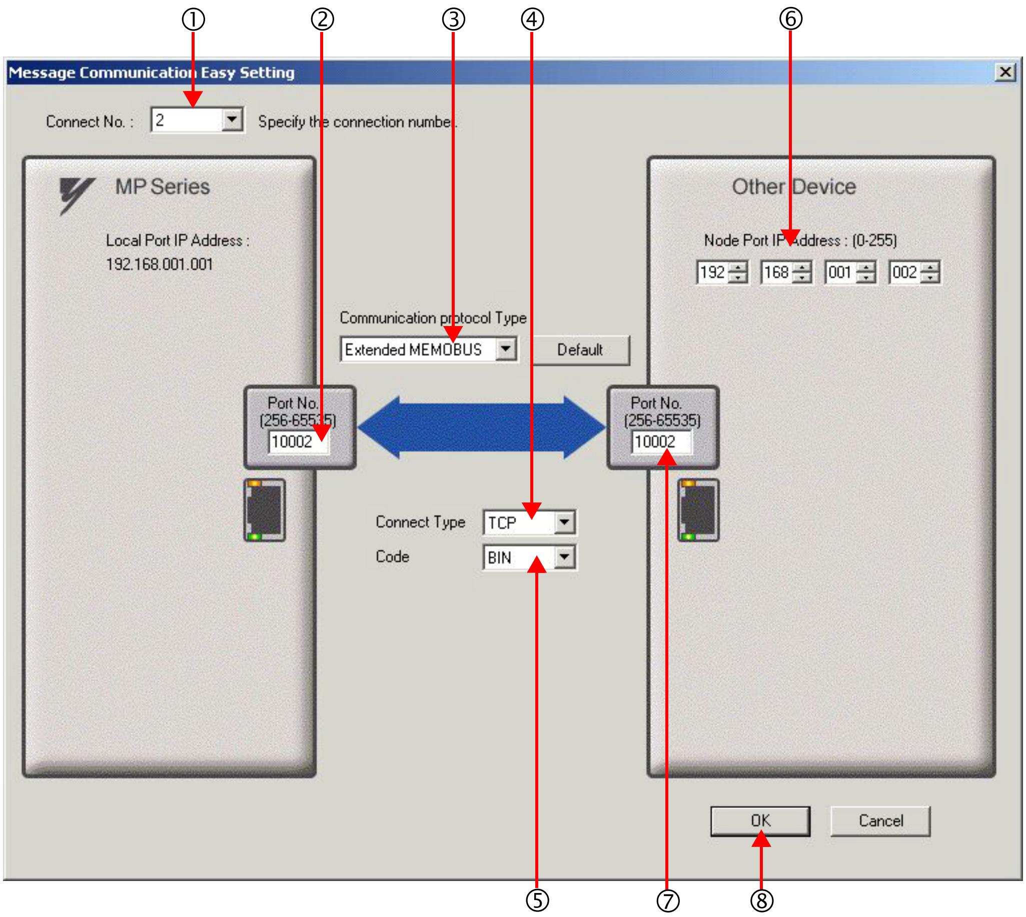
Set up a unique IP address in the network.
For the IP address, check with your network administrator.

---

3. Click the **Easy Setting** Button in the **Message Communication** Area of the connection parameter setting.



4. Set a communication setting in the **Message Communication Easy Setting** Dialog Box.



■ How to set up in the **Message Communication Easy Setting** Dialog Box

① When automatic receive function is used, select “2” for the **Connect No.**
② Set **Port No.** of the CPU-03/CPU-04 side (“10002,” for example).
③ Select **Extended MEMOBUS** for **Communication protocol Type**, and click the **Default** Button.
④ Select **Connect Type** (TCP, for example).
⑤ Select **Code** (BIN, for example).
⑥ Set **Node Port IP Address** of the other device (MP2300) to be connected (to “192.168.001.002,” for example).
⑦ Set **Port No.** of the other device (MP2300) to be connected (to “10002,” for example).
⑧ Click **OK**.

■ Caution

When message functions (MSG-SND, MSG-RCV) are used with the connection number 01, disable the automatic receive function. If message functions are used while the automatic receive function is enabled, communications will not function properly.

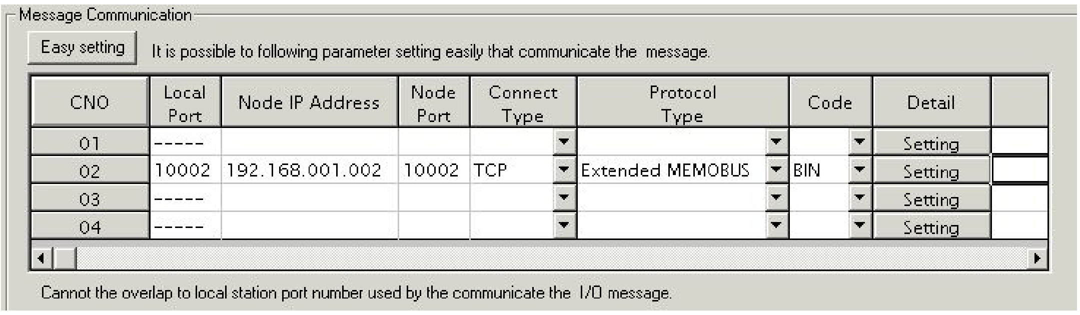
Note: The automatic receive function with a connection number 01 is set to “Enable” by default.

5. Click **Yes** in the parameter setting confirmation dialog.

■ Caution

Note that when a parameter with the same connection number is already set and you click **Yes** in the parameter setting confirmation dialog, the setting will be overwritten by the parameter configured in the **Message Communication Easy Setting** Dialog Box.

6. Check the setting values.



■ Caution

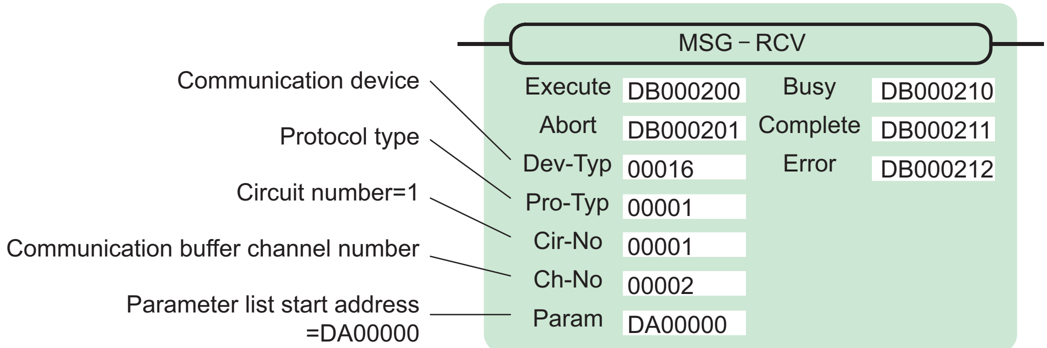
When any transmission or connection parameter is changed, the change will be reflected after it has been saved in flash memory and the power supply is turned ON again.

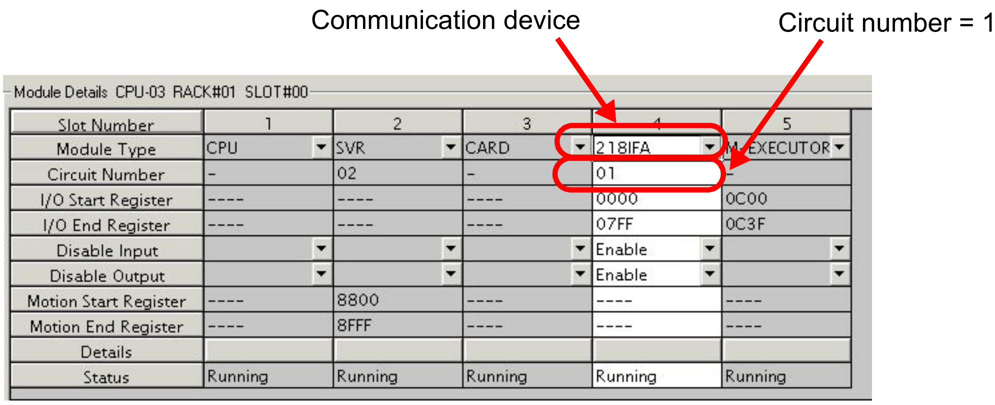
7. Create a ladder program with a message receive function (MSG-RCV) in it.

An example of a ladder program for receiving messages in the CPU-03/CPU-04 side is as follows:

■ Message receive function (MSG-RCV)

This system function is required for receiving messages. A message reception is carried out by describing and executing this message receive function in a ladder program.





Slot Number	1	2	3	4	5
Module Type	CPU	SVR	CARD	218IFA	M-EXECUTOR
Circuit Number	-	02	-	01	-
I/O Start Register	----	----	----	0000	0C00
I/O End Register	----	----	----	07FF	0C3F
Disable Input				Enable	
Disable Output				Enable	
Motion Start Register	----	8800	----	----	----
Motion End Register	----	8FFF	----	----	----
Details					
Status	Running	Running	Running	Running	Running

Fig. 6.4 MPE720 Module Configuration Window

Built-in Ethernet Communication (Supported by the CPU-03 and CPU-04)

6

■ Input/output definition contents for message receive function

The input/output definition content for message receive function is as follows:

Table 6.4 Input/Output Definitions for Message Receive Function

I/O Definition	No.	Name	Setting Example	Contents
Input Item	1	Execute	DB000200	Executes a reception When Execute is ON, message reception will be carried out.
	2	Abort	DB000201	Aborts a reception When Abort is ON, message reception is forcibly stopped.
	3	Dev-Typ	00016	Communication device type Specify the type of the communication device used in reception. When Ethernet (218IFA/218IFC) is used, specify "16."
	4	Pro-Typ	00001	Communication protocol Specify the type of the communication protocol. MEMOBUS(*1) = 1, non-procedure 1(*2) = 2, non-procedure 2(*2) = 3
	5	Cir-No	00001	Circuit number Specify a circuit number of the communication device. Specify it in accordance with the circuit number displayed in the MPE720 **Module Configuration** Window.
	6	Ch-No	00002	Communication buffer channel number Specify the channel number of the communication buffer. • When Ethernet (218IFA) is used, specify it in the range between "1" and "4." • When Ethernet (218IFC) is used, specify in the range between "1" and "10." Note: Set up a unique channel number in the circuit.
	7	Param	DA00000	Parameter list start address Specify the start address of the parameter list. For the Parameter List, 17 words are automatically assigned from the configured address.
Output Item	1	Busy	DB000210	In process Busy will be ON while executing a message reception or forced abort process.
	2	Complete	DB000211	Process completed When a message reception or forced abort process is properly completed, Complete will turn ON only for one scan.
	3	Error	DB000212	Error When an error occurs, Error will turn ON only for one scan.

* 1. When transmitting in MEMOBUS, Extended MEMOBUS, MELSEC, or MODBUS/TCP protocol, set the communication protocol (Pro-Typ) to MEMOBUS (=1). The communication device automatically converts the protocol.

* 2. Non-procedure 1: In non-procedural communication, data is received on a per-word basis.
Non-procedure 2: In non-procedural communication, data is received on a per-byte basis.

■ Parameter list setting example for message receive function

An example of a parameter list setting when receiving messages from a transmit source using the connection with a connection number = 2 follows:

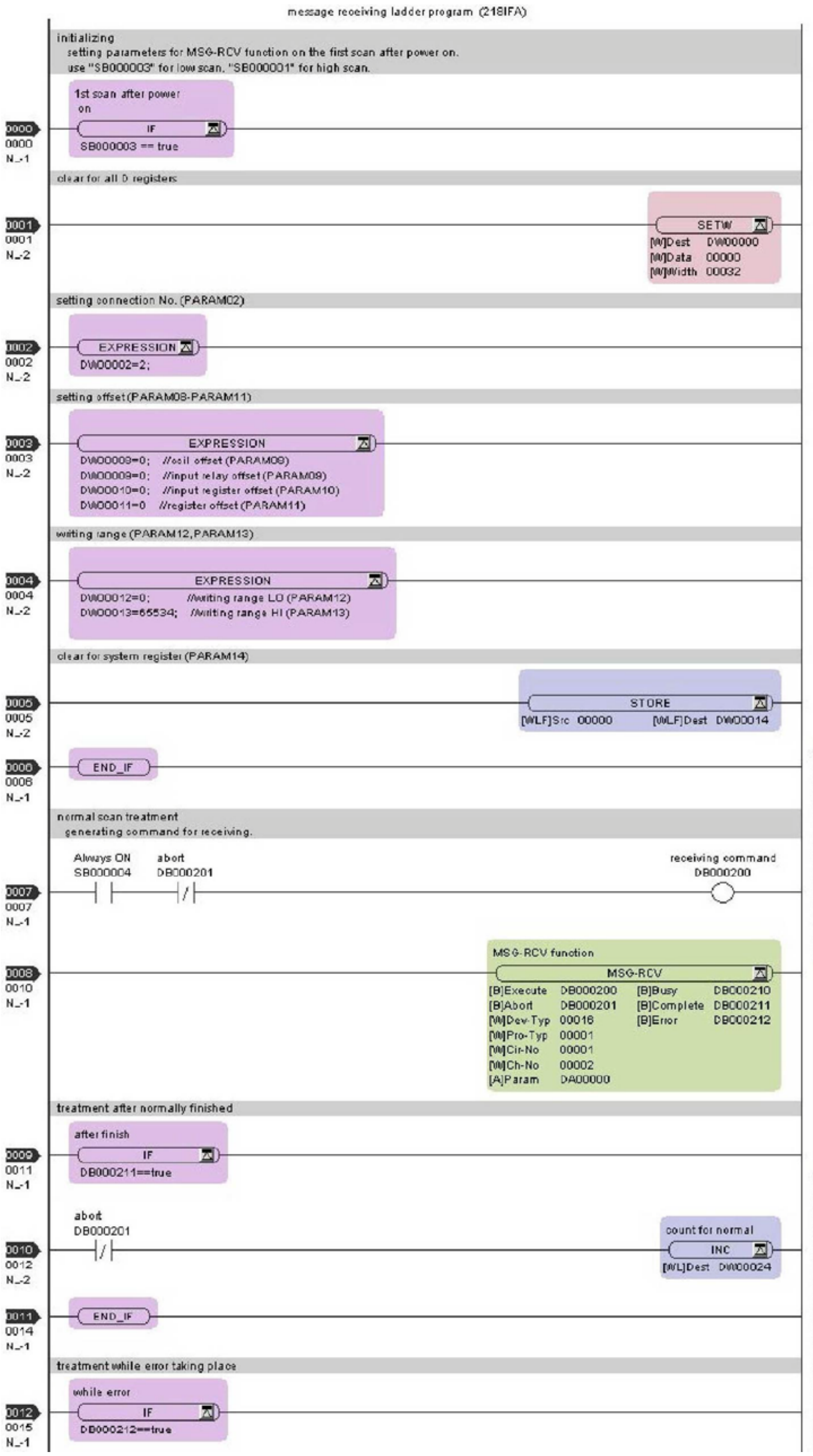
Table 6.5 Parameter List Setting Example (parameter list start address Param=DA00000)

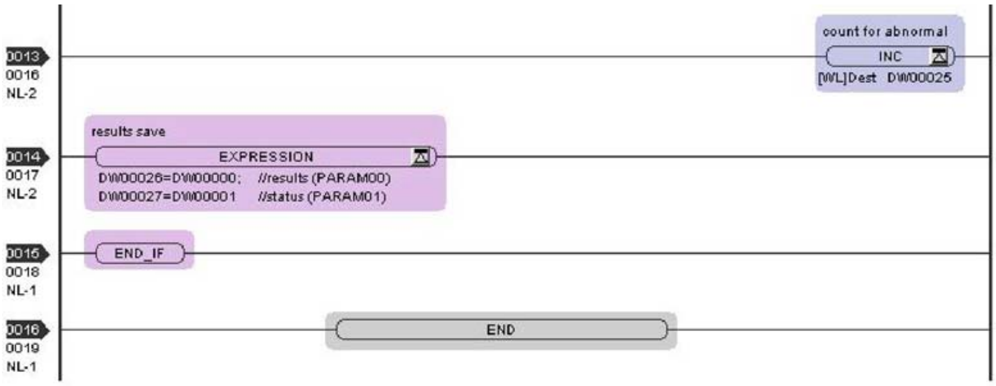
Register Number	Setting Value	Parameter Number	IN/OUT	Remarks
DW00000	-	PARAM00	OUT	Process result
DW00001	-	PARAM01	OUT	Status
DW00002	00002	PARAM02	IN	Connection number = 2
DW00003	-	PARAM03	OUT	Option
DW00004	-	PARAM04	OUT	Function code
DW00005	-	PARAM05	OUT	Data address
DW00006	-	PARAM06	OUT	Data size
DW00007	-	PARAM07	OUT	Remote CPU number
DW00008	00000	PARAM08	IN	Coil offset = 0 word
DW00009	00000	PARAM09	IN	Input relay offset = 0 word
DW00010	00000	PARAM10	IN	Input register offset = 0 word
DW00011	00000	PARAM11	IN	Holding register offset = 0 word
DW00012	00000	PARAM12	IN	Writable address lower limit = MW00000
DW00013	65534	PARAM13	IN	Writable address upper limit = MW65534
DW00014	-	PARAM14	SYS	Reserved by the system. (Zero clear at startup)
DW00015	-	PARAM15	SYS	Reserved by the system.
DW00016	-	PARAM16	SYS	Reserved by the system.

Note: IN: Input, OUT: Output, SYS: For system use

■ Example of Using the Message Receive Function in a Ladder Program

Here is an example of the message receive function through Ethernet (218IFA/218IFC).



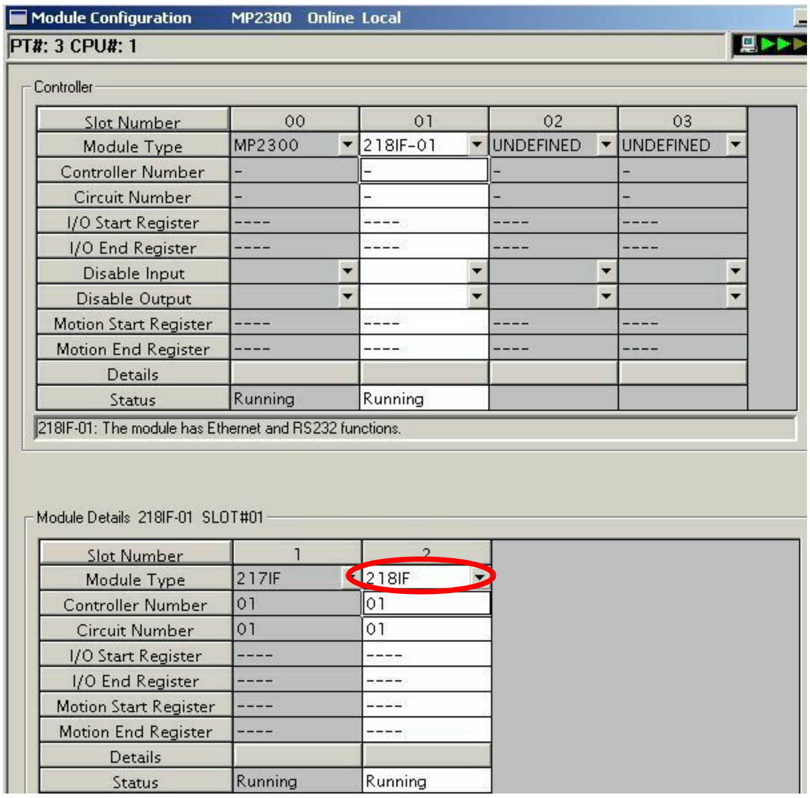


The communication setting and the ladder program creation are now finished, when the CPU-03/CPU-04 acts as a slave.

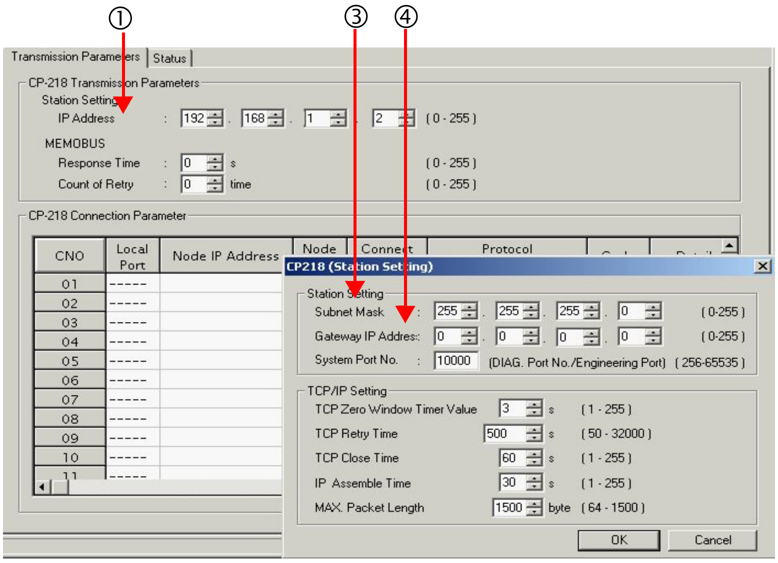
(2) How to Set up the Remote Device (MP2300) to Be Connected

If the setting of transmission parameters (IP address, subnet mask) is already completed, start from step 3.

1. Double-click the **218IF** Tab in the **Module Details** Area of the **Module Configuration** Window.



2. Set transmission parameters.



■ How to set up transmission parameters

① Set **IP Address** (“192.168.001.001,” for example).

② Select ***Edit - Local Station: TCP/IP Setting*** in the Engineering Manager Window the **Detail Setting** Dialog Box will appear.

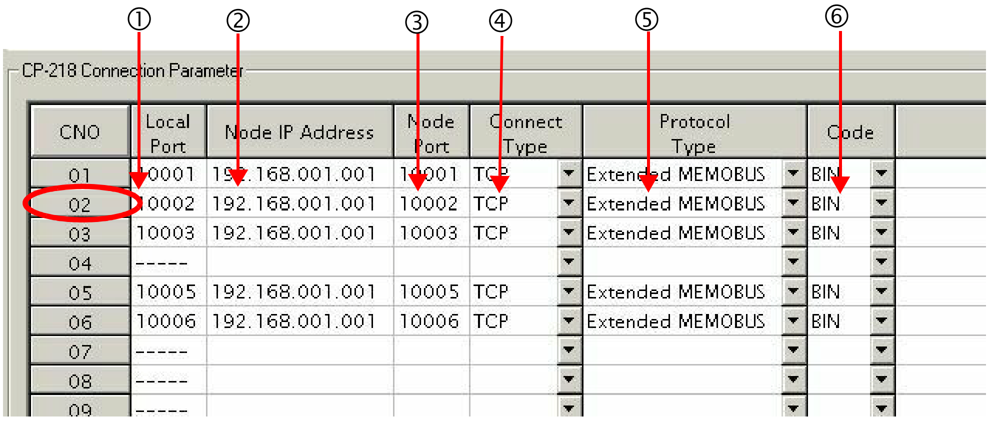
③ Set **Subnet Mask** (“255.255.255.000,” for example).

④ Set **Gateway IP Address** (“000.000.000.000,” for example).

■ Caution

Set up a unique IP address in the network.
For the IP address, check with your network administrator.

3. Set connection parameters.



CNO	Local Port	Node IP Address	Node Port	Connect Type	Protocol Type	Code
01	10001	192.168.001.001	10001	TCP	Extended MEMOBUS	BIN
02	10002	192.168.001.001	10002	TCP	Extended MEMOBUS	BIN
03	10003	192.168.001.001	10003	TCP	Extended MEMOBUS	BIN
04	-----					
05	10005	192.168.001.001	10005	TCP	Extended MEMOBUS	BIN
06	10006	192.168.001.001	10006	TCP	Extended MEMOBUS	BIN
07	-----					
08	-----					
09	-----					

■ How to set up with a connection number 02 in the **CP-218 Connection Parameter** Area

① Set **Local Port** to the port number used in the MP2300 side ("10002" for example).
② Set the **Node IP Address** to the IP address configured in the CPU-03/CPU-04 side ("192.168.001.001" for example).
③ Set the **Node Port** to the port number configured in the CPU-03/CPU-04 side ("10002," for example).
④ Select **Connect Type** (TCP, for example).
⑤ Select **Extended MEMOBUS** for **Protocol Type**.
⑥ Select **Code** (BIN, for example).

■ Caution

When any transmission or connection parameter is changed, the change will be reflected after it has been saved in flash memory and the power supply is turned ON again.

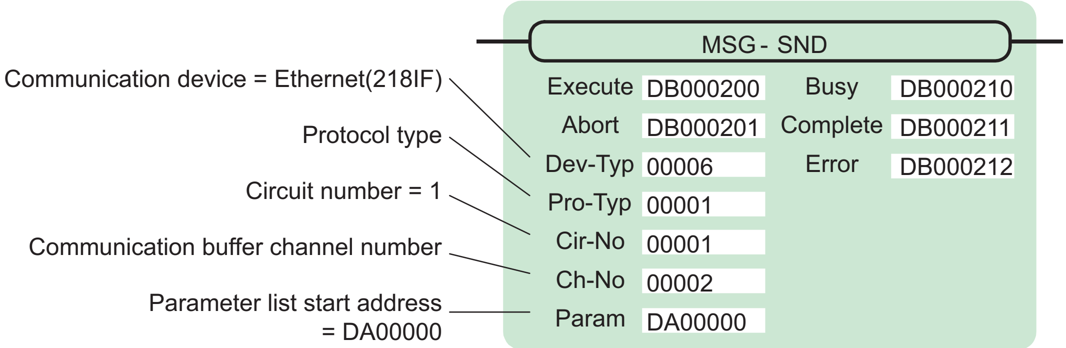
---

4. Create a ladder program containing a message send function (MSG-SND).

An example of a ladder program for transmitting messages to/from the remote device (MP2300) side follows:

■ Message send function (MSG-SND)

This system function is required for transmitting messages. Message transmission is carried out by describing and executing this message send function in a ladder program.



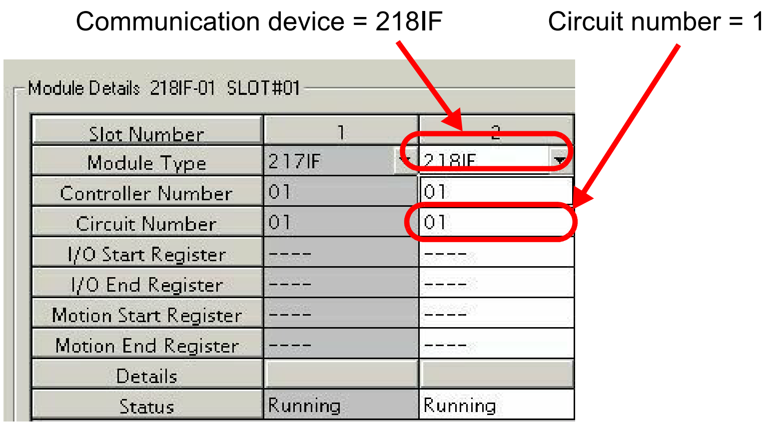


Fig. 6.5 MPE720 Module Configuration Window

■ Input/output definitions contents for message send function

The input/output definition content for the message send function is as follows:

Table 6.6 Input/Output Definitions for Message Send Function

I/O Definition	No.	Name	Setting Example	Contents
Input Item	1	Execute	DB000200	Executes a transmission When Execute is ON, the message transmission will be carried out.
	2	Abort	DB000201	Forcibly aborts a transmission When Abort is ON, the message transmission is forcibly stopped.
	3	Dev-Typ	00006	Communication device type Specify the type of the communication device used in transmission. When Ethernet (218IF) is used, specify "6."
	4	Pro-Typ	00001	Communication protocol Specify the type of the communication protocol. MEMOBUS(*1) = 1, non-procedure 1(*2) = 2, non-procedure 2(*2) = 3
	5	Cir-No	00001	Circuit number Specify a circuit number of the communication device. Specify it in accordance with the circuit number displayed in the MPE720 **Module Configuration** Window.
	6	Ch-No	00002	Communication buffer channel number Specify the channel number of the communication buffer. • When Ethernet (218IF) is used, specify it in the range between "1" and "10." • When Ethernet (218IFC) is used, specify it in the range between "1" and "10." Note: Set up a unique channel number in the circuit.
	7	Param	DA00000	Parameter list start address Specify the start address of the parameter list. For the Parameter List, 17 words are automatically assigned from the configured address.
Output Item	1	Busy	DB000210	In process Busy will be ON while executing a message transmission or forced abort process.
	2	Complete	DB000211	Process completed When a message transmission or forced abort process is properly completed, Complete will turn ON only for one scan.
	3	Error	DB000212	Error When an error occurs, Error will turn ON only for one scan.

* 1. When transmitting in MEMOBUS, Extended MEMOBUS, MELSEC, or MODBUS/TCP protocol, set the communication protocol (Pro-Typ) to MEMOBUS(=1). The communication device automatically converts the protocol.

* 2. Non-procedure 1: In non-procedural communication, data is transmitted on a per-word basis.
Non-procedure 2: In non-procedural communication, data is transmitted on a per-byte basis.

■ Parameter list setting for the message send function

An example of a parameter list setting when reading 100 words of data from MW00100 from the destination using the connection with a connection number = 2 follows:

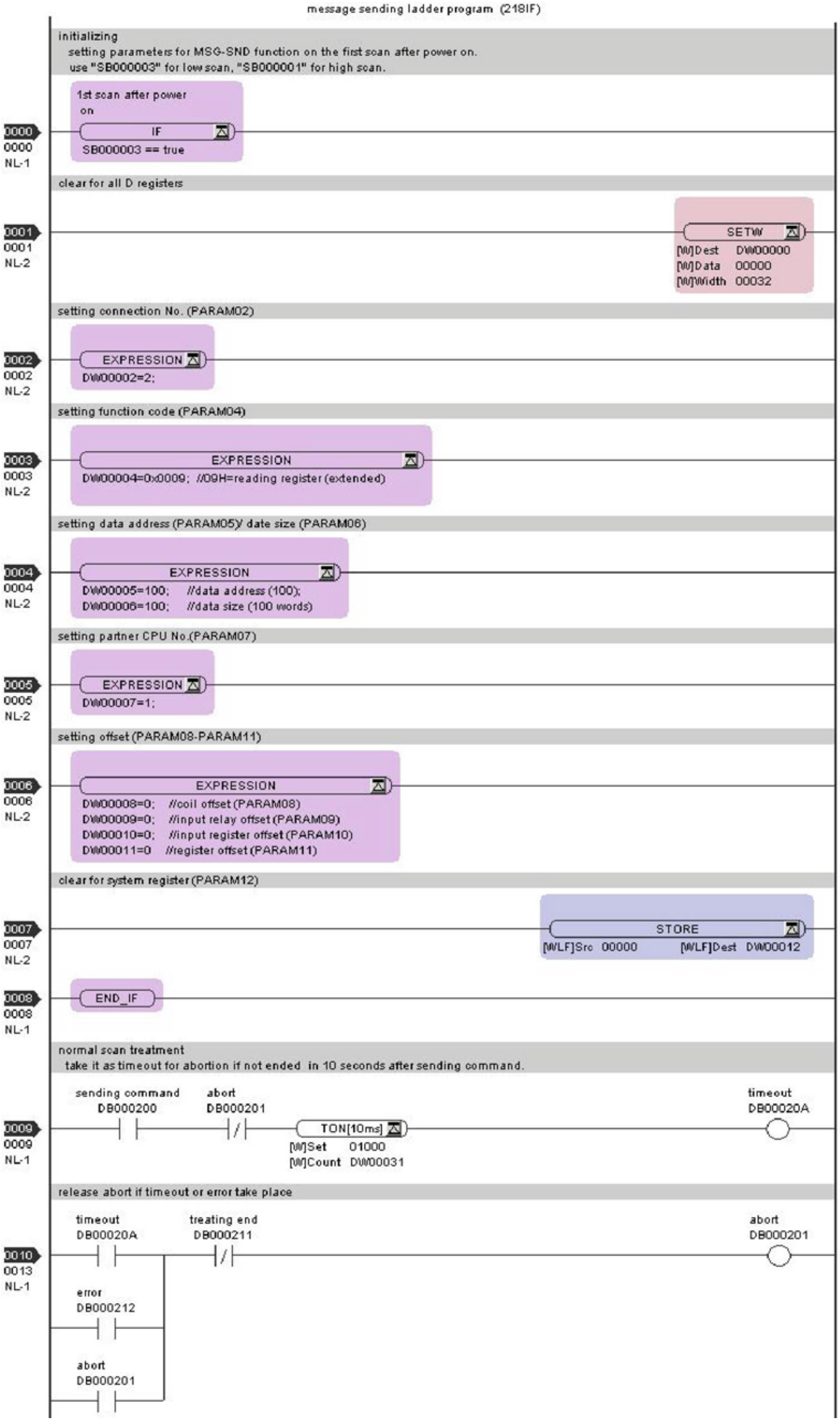
Table 6.7 Parameter List Setting Example (parameter list start address Param=DA00000)

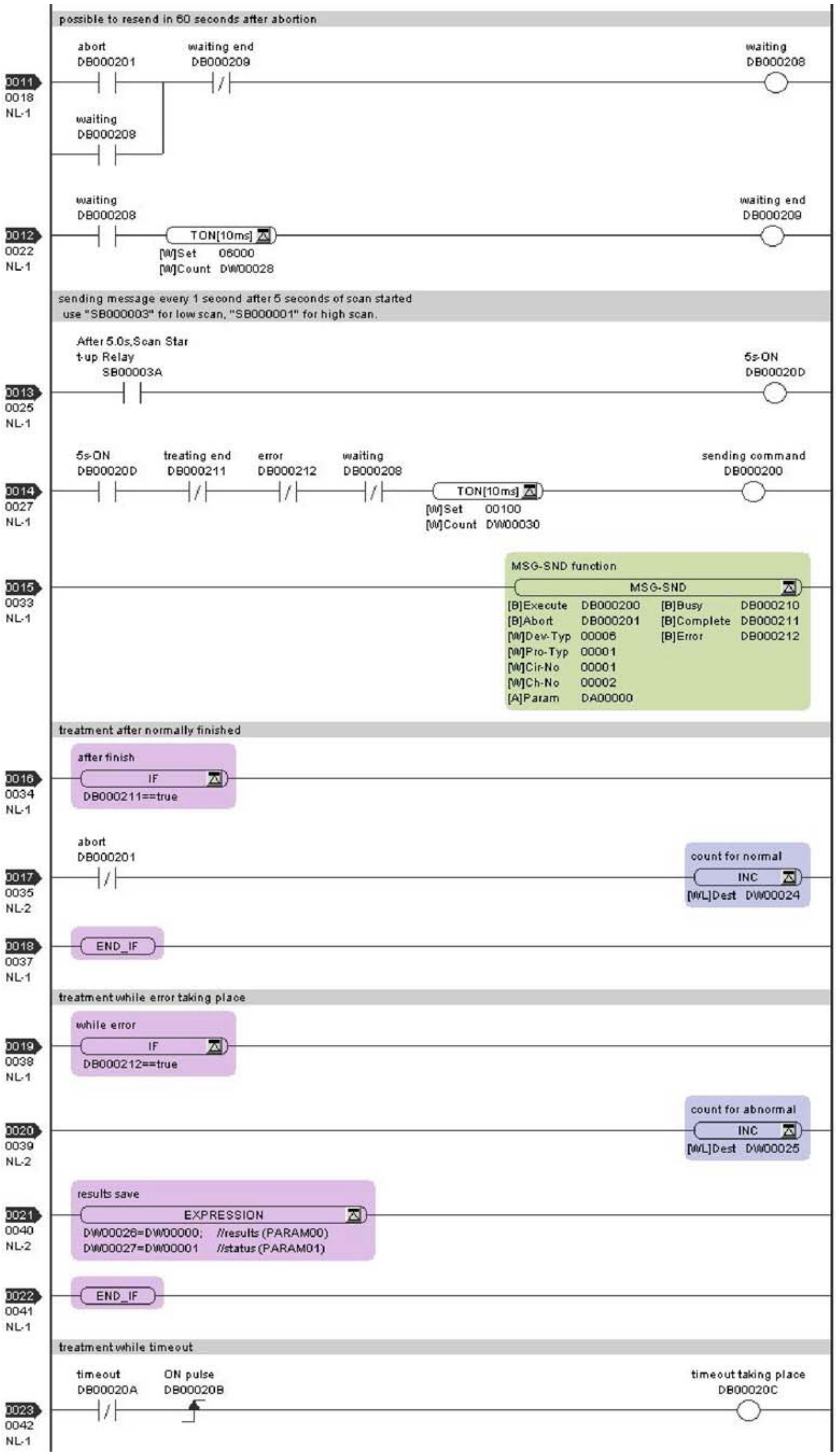
Register Number	Setting Value	Parameter Number	IN/OUT	Remarks
DW00000	–	PARAM00	OUT	Process result
DW00001	–	PARAM01	OUT	Status
DW00002	00002	PARAM02	IN	Connection number = 2
DW00003	–	PARAM03	IN	Option (Setting unnecessary)
DW00004	0009H	PARAM04	IN	Function code = 09H (Reads a holding register)
DW00005	00100	PARAM05	IN	Data address = 100 (Starting from MW00100)
DW00006	00100	PARAM06	IN	Data size = 100 (100 words)
DW00007	00001	PARAM07	IN	Remote CPU number = 1
DW00008	00000	PARAM08	IN	Coil offset = 0 word
DW00009	00000	PARAM09	IN	Input relay offset = 0 word
DW00010	00000	PARAM10	IN	Input register offset = 0 word
DW00011	00000	PARAM11	IN	Holding register offset = 0 word
DW00012	–	PARAM12	SYS	Reserved by the system. (Zero clear at startup)
DW00013	–	PARAM13	SYS	Reserved by the system.
DW00014	–	PARAM14	SYS	Reserved by the system.
DW00015	–	PARAM15	SYS	Reserved by the system.
DW00016	–	PARAM16	SYS	Reserved by the system.

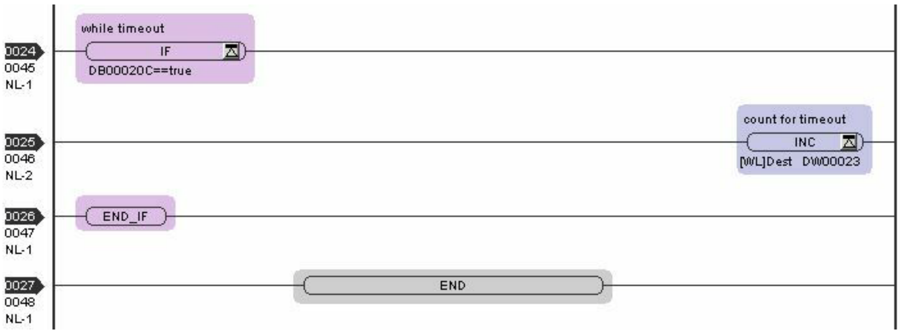
Note: IN: Input, OUT: Output, SYS: For system use

■ Example of Using the Message Send Function in a Ladder Program

Here is one example of the message send function through Ethernet (218IF).




possible to resend in 60 seconds after abortion
abort DB000201
waiting end DB000209
waiting DB000208
0011 0018 NL-1
waiting DB000208
waiting DB000208
TON[10ms]
[W]Set 06000
[W]Count DW00028
waiting end DB000209
0012 0022 NL-1
sending message every 1 second after 5 seconds of scan started
use "SB000003" for low scan, "SB000001" for high scan.
After 5.0s,Scan Star t-up Relay SB00003A
5s-ON DB00020D
0013 0025 NL-1
5s-ON DB00020D
treating end DB000211
error DB000212
waiting DB000208
TON[10ms]
[W]Set 00100
[W]Count DW00030
sending command DB000200
0014 0027 NL-1
MSG-SND function
MSG-SND
[B]Execute DB000200
[B]Abort DB000201
[W]Dev-Typ 00006
[W]Pro-Typ 00001
[W]Cir-No 00001
[W]Ch-No 00002
[A]Param DA00000
[B]Busy DB000210
[B]Complete DB000211
[B]Error DB000212
0015 0033 NL-1
treatment after normally finished
after finish
IF
DB000211==true
0016 0034 NL-1
abort DB000201
count for normal
INC
[WL]Dest DW00024
0017 0035 NL-2
END_IF
0018 0037 NL-1
treatment while error taking place
while error
IF
DB000212==true
0019 0038 NL-1
count for abnormal
INC
[WL]Dest DW00025
0020 0039 NL-2
results save
EXPRESSION
DW00026=DW00000; //results (PARAM00)
DW00027=DW00001 //status (PARAM01)
0021 0040 NL-2
END_IF
0022 0041 NL-1
treatment while timeout
timeout DB00020A
ON pulse DB00020B
timeout taking place DB00020C
0023 0042 NL-1



The communication setting and the ladder program creation are now finished, when MP2300 acts as a master.

(3) How to Start Communications

1. The CPU-03/CPU-04 side starts to receive the messages.

As the sample ladder program automatically starts the message receive operation just after system startup, you are not required to do anything. In normal operation, accept the default.

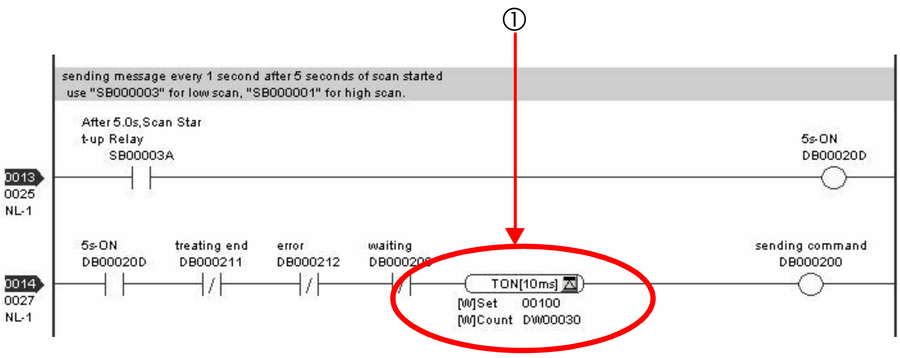
2. Turn Execute ON for the message send function in the MP2300 side to transmit messages.

Messages are transmitted by turning on the register DB000200 configured in Execute of the message send function, for example, and communication with the CPU-03/CPU-04 starts.

Table 6.8 Input/Output Definition for Message Send Function

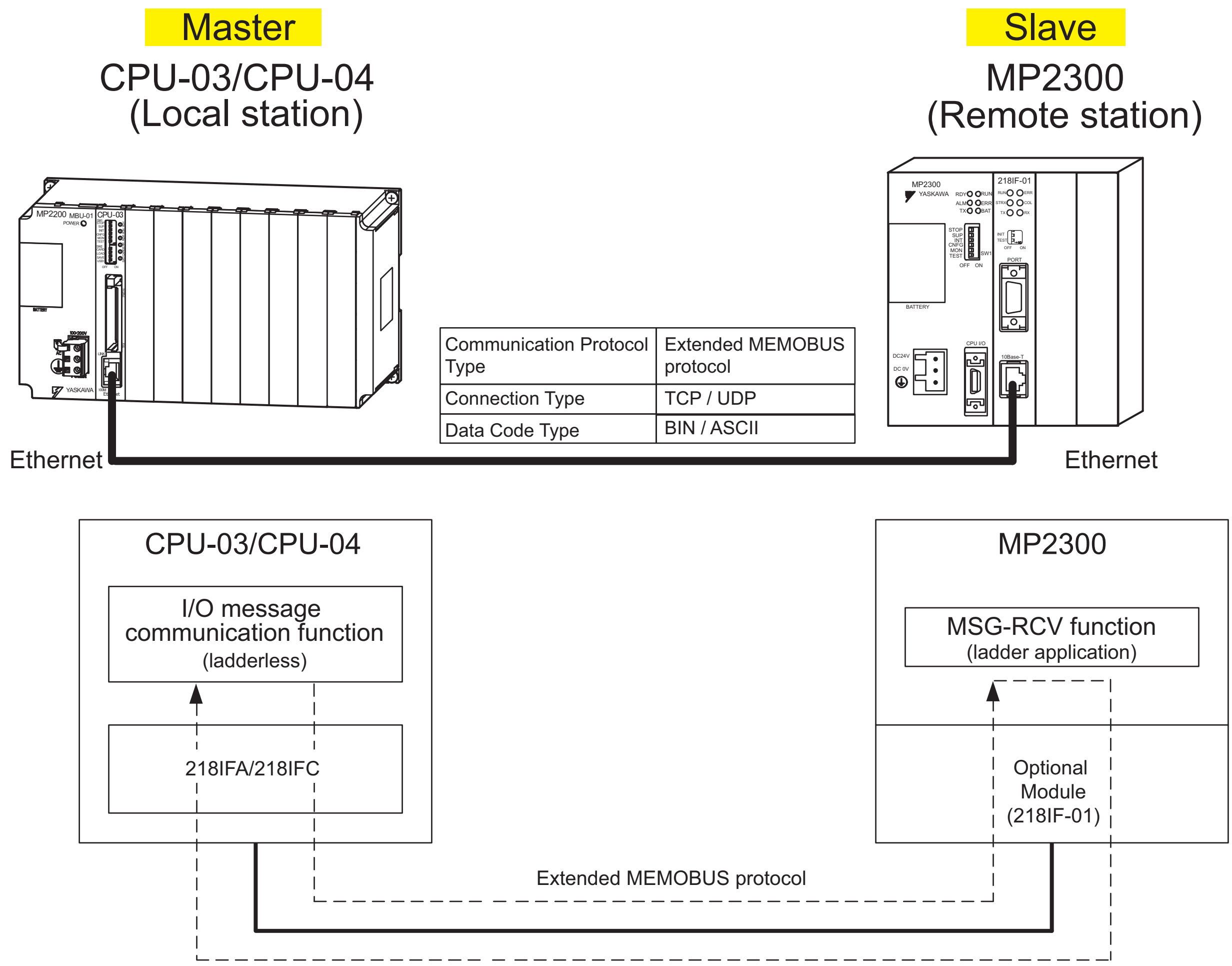
I/O Definition	No.	Name	Setting Example	Contents
Input Item	1	Execute	DB000200	Executes a transmission When Execute is ON, the message transmission is carried out.

The sample ladder program is created to transmit a message every one second when five seconds have elapsed after the low-speed scan (or high-speed scan) startup.
To change the message transmission interval, change the timer value ①.



6.2.3 When CPU-03/CPU-04 Acts as Master (I/O message communication function is used)

This section explains how to communicate with the CPU-03/CPU-04 message receive function (MSG-RCV) using the MP2300 I/O message communication function.



Communication Protocol Type	Extended MEMOBUS protocol
Connection Type	TCP / UDP
Data Code Type	BIN / ASCII

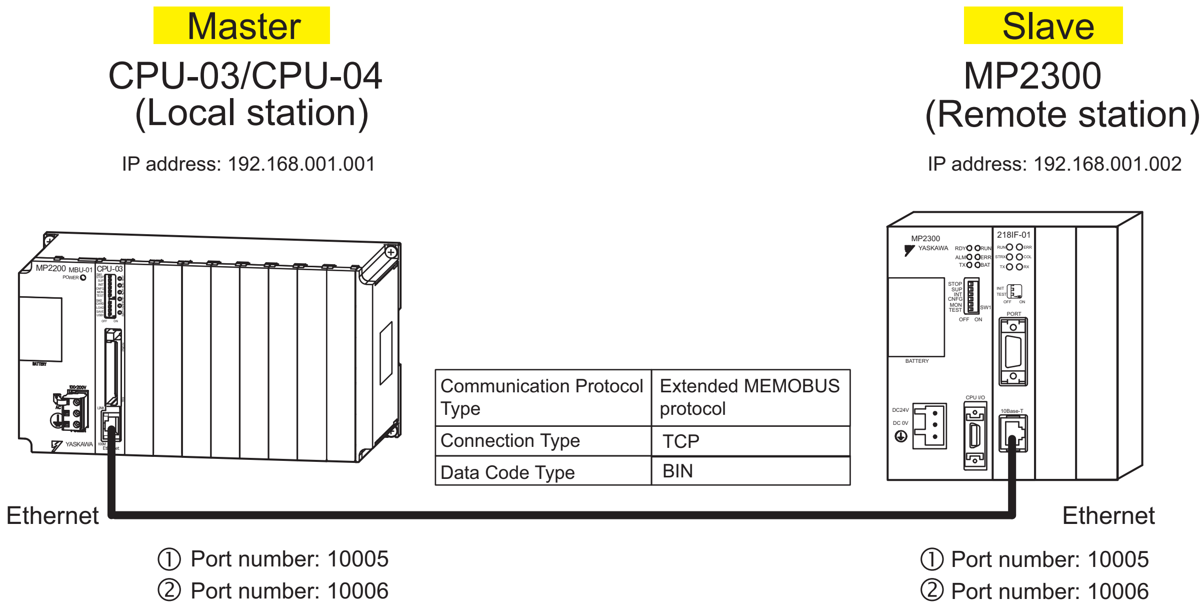
Fig. 6.6 Message Flow with MP2300 when I/O Message Communication Function Is Used

■ I/O Message Communication

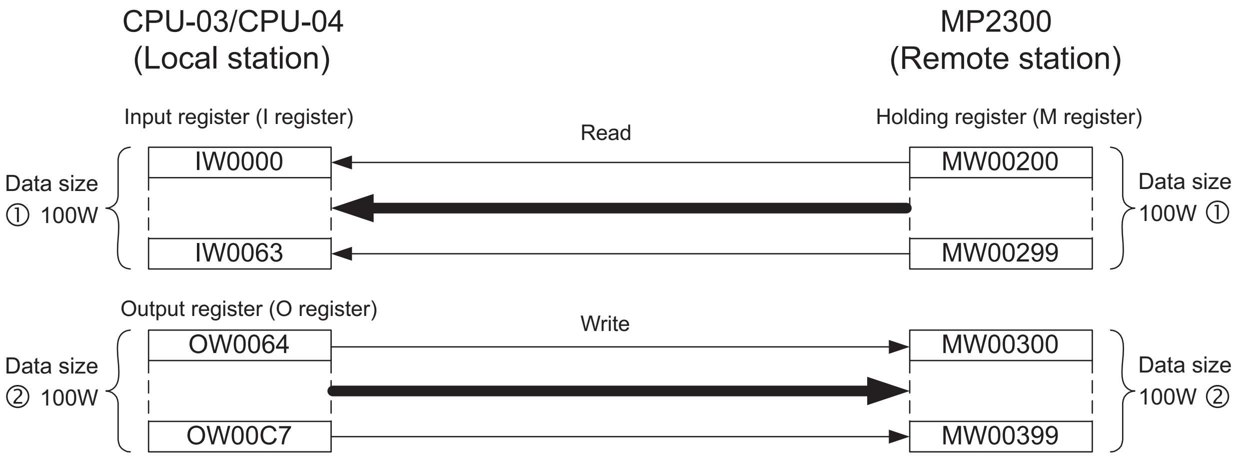
I/O message communication implements out 1:1 communication.
In addition, you can read and write only the holding register in the case of “Communication Protocol Type: Extended MEMOBUS” used in the communication with MP series.
When you communicate with multiple remote devices, or when you need to read a coil state or input relay, or change a coil state as well as read/write a holding register, use the message send function (MSG-SND).

■ Setting Example

The following figure illustrates one example of reading the contents of the holding register (MW00200 to MW00299) of MP2300 (slave) into an input register (IW0000 to IW0063) of the CPU-03/CPU-04 (master) and writing the contents of an output register (OW0064 to OW00C7) of the CPU-03/CPU-04 (master) into a holding register (MW00300 to MW00399) of MP2300 (slave).



Communication Protocol Type	Extended MEMOBUS protocol
Connection Type	TCP
Data Code Type	BIN

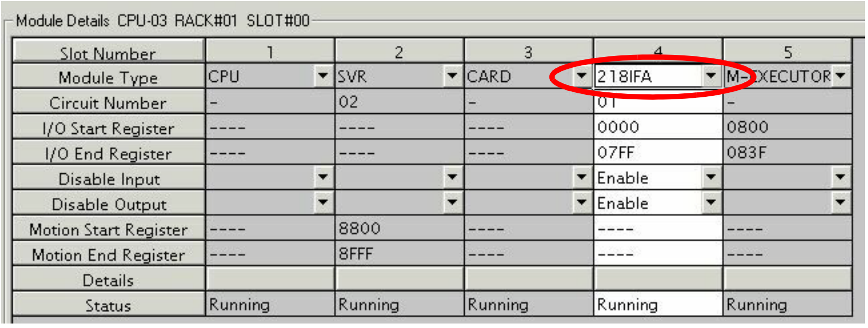


The setup procedure is explained in the following pages.

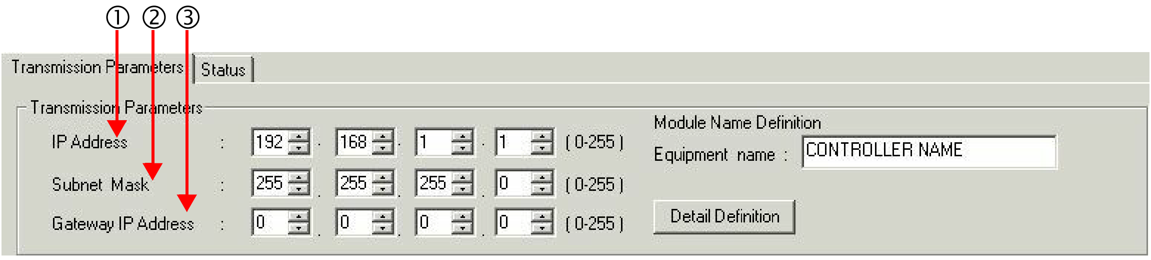
(1) How to Set up the CPU-03/CPU-04 Side

If the setting of transmission parameters (IP address, subnet mask) is already completed, start from step 3.

1. Double-click the **218IFA** Tab (**218IFC** Tab for the CPU-04) in the **Module Details** Area in the CPU-03 **Module Configuration** Window.



2. Set transmission parameters.



■ How to set up transmission parameters

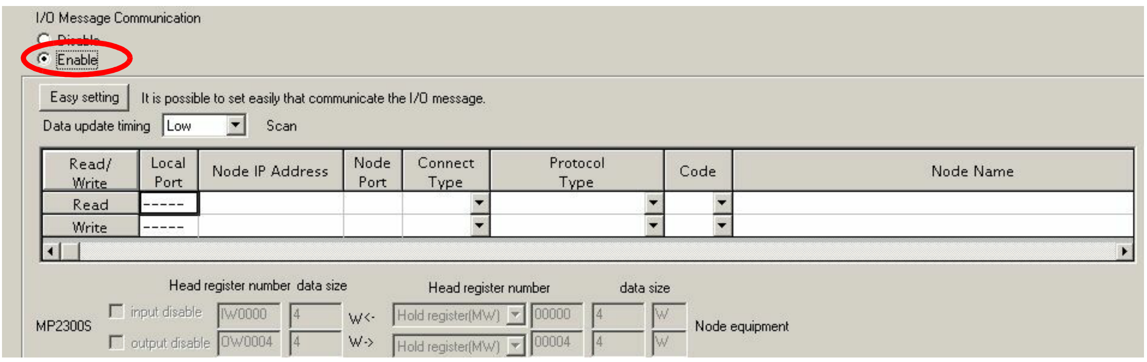
① Set **IP Address** ("192.168.001.001," for example).
② Set **Subnet Mask** ("255.255.255.000," for example).
③ Set **Gateway IP Address** ("000.000.000.000," for example).

■ Caution

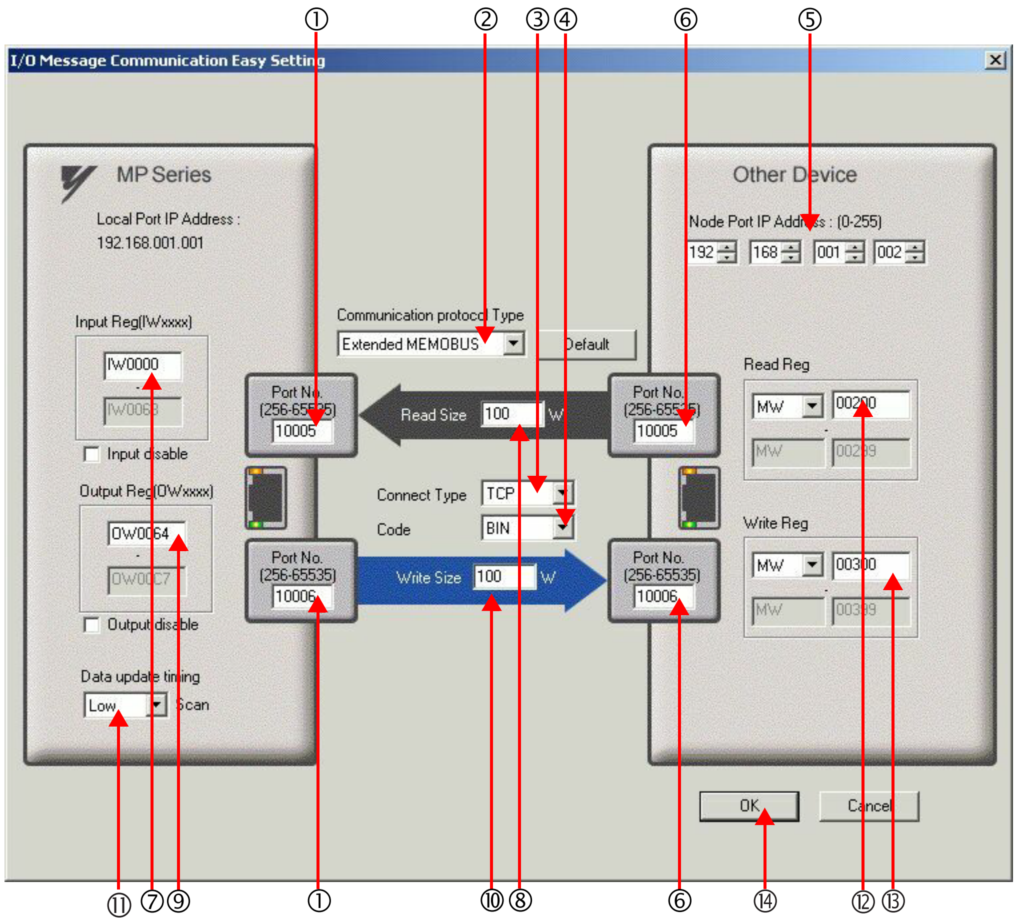
Set up a unique IP address in the network.
For the IP address, check with your network administrator.

---

3. Select the **Enable** Option in the **I/O Message Communication** Area of the connection parameter setting.



4. Set a communication setting in the **I/O Message Communication Easy Setting** Dialog Box.



■ How to set up in the **I/O Message Communication Easy Setting** Dialog Box

① Set **Port No.** of the CPU-03/CPU-04 side (“10005, 10006,” for example).

② Select **Extended MEMOBUS** for **Communication Protocol Type**, and click the **Default** Button.

■ Caution

When the communication protocol is Extended MEMOBUS, the register type that can select both read and write is fixed at the Holding Register (MW).

---

③ Select **Connect Type** (TCP, for example).

④ Select **Code** (BIN, for example).

⑤ Set **Remote IP Address** of the other device (MP2300) to be connected (“192.168.001.002,” for example).

⑥ Set **Port No.** of the other device (MP2300) to be connected (“10005, 10006,” for example).

■ Caution

In I/O message communication, as a message is transmitted from each port number for register read/write, a connected remote device needs the message receive functions to receive two messages.

---

⑦ Set a storage area (**Input Reg**) of data read by CPU-03/CPU-04 (IW0000, for example).
⑧ Set the **Read Size** of data to be read by the CPU-03/CPU-04 (“100” W, for example).
⑨ Set a storage area (**Output Reg**) of data written by the CPU-03/CPU-04 (OW0064, for example).
⑩ Set the **Write Size** of data written by the CPU-03/CPU-04 (“100” W, for example).
⑪ Set an I/O data update timing (**Data update timing**) for CPU and built-in Ethernet (“Low” scan, for example).

■ Data Update Timing

Data update timing indicates when to send and receive data between the CPU and built-in Ethernet. Communication with the remote device is carried out asynchronously, so note that a message is not necessarily transmitted to the remote equipment at each set data update time.

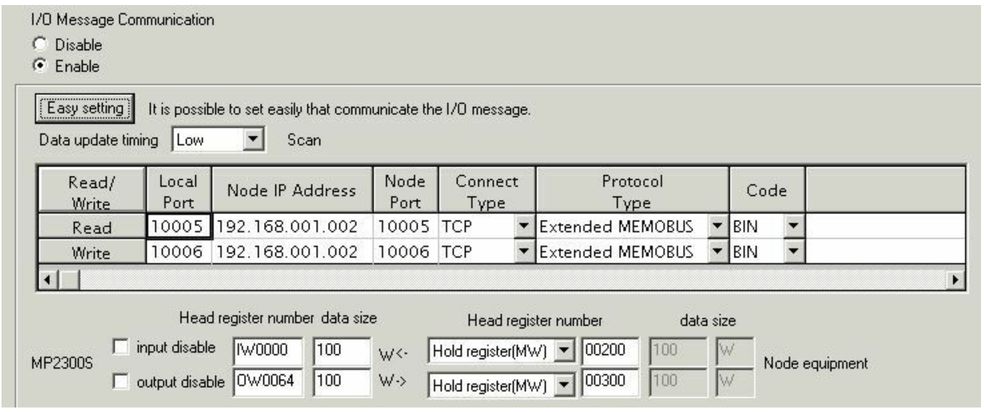
⑫ Set the register type and start address (**Read Reg**) of the remote device (MP2300) read by the CPU-03/CPU-04 (“MW00200,” for example).
⑬ Set the register type and start address (**Write Reg**) of the remote device (MP2300) written by the CPU-03/CPU-04 (“MW00300,” for example).
⑭ Click **OK**.

5. Click **Yes** in the parameter setting confirmation window.

■ Caution

Note that when a parameter with the same connection number is already set and you click **Yes** in the parameter setting confirmation window, the setting will be overwritten by the parameter configured in the **Message Communication Easy Setting** Dialog Box.

6. Check the setting values.



The I/O message communication is now set up, when the CPU-03/CPU-04 acts as a master.

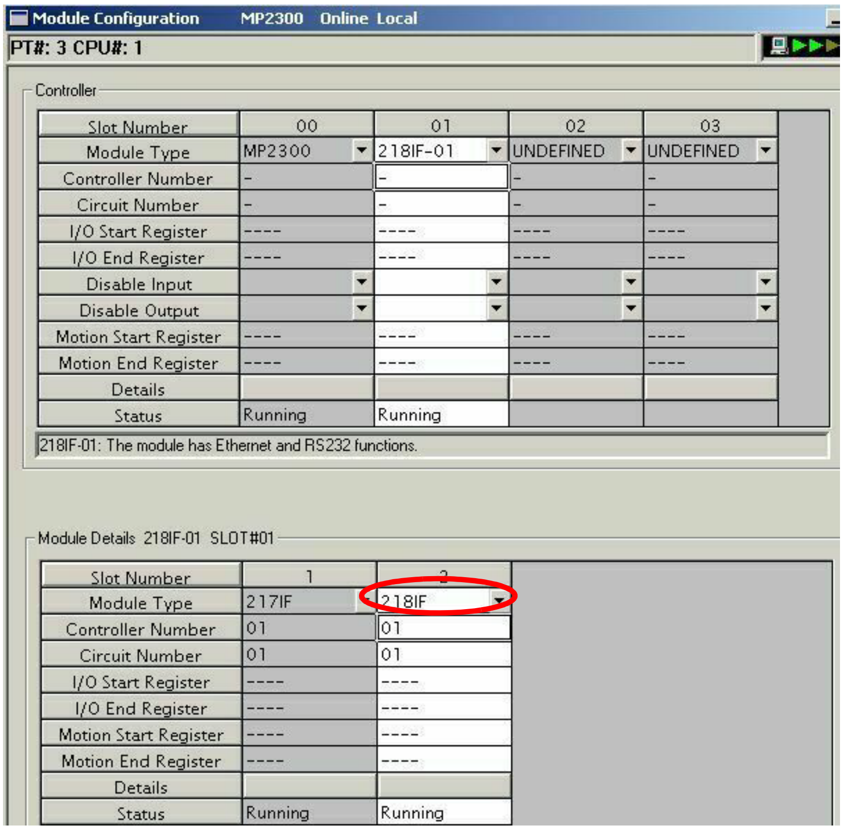
■ Caution

When any transmission or connection parameter is changed, the change will be not reflected after it has been saved in flash memory and the power supply is turned ON again.

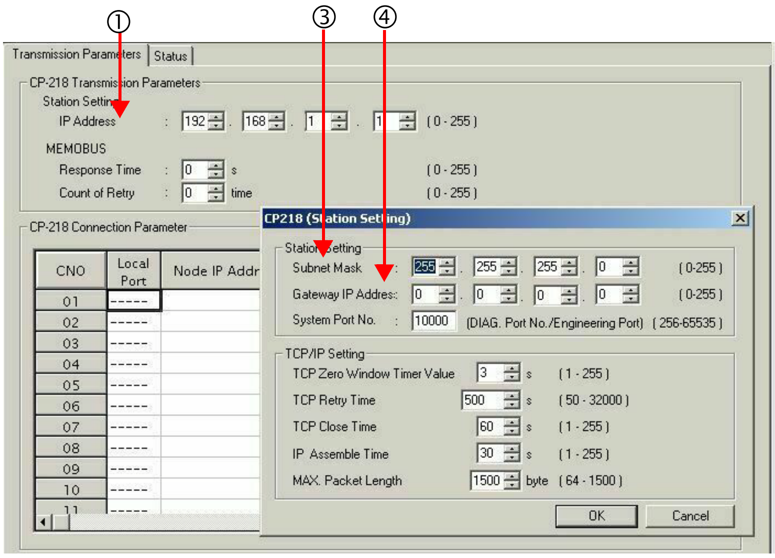
(2) How to Set up the Remote Device (MP2300) to Be Connected

When the setting of transmission parameters (IP address, subnet mask) is already completed, start from step 3.

1. Double-click the **218IF** Tab in the **Module Details** of the **Module Configuration** Window.



2. Set transmission parameters.



■ How to set up transmission parameters

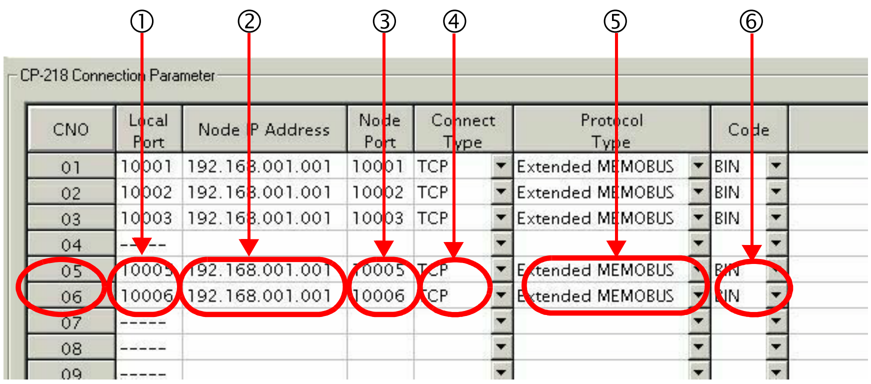
① Set **IP Address** (“192.168.001.001”, for example).
② Select ***Edit - Local Station: TCP/IP Setting*** in the Engineering Manager Window the **Detail Setting** Dialog Box will appear.
③ Set **Subnet Mask** (“255.255.255.000”, for example).
④ Set **Gateway IP Address** (“000.000.000.000”, for example).

■ Caution

Set up a unique IP address in the network.
For the IP address, check with your network administrator.

---

3. Set connection parameters.



CNO	Local Port	Node IP Address	Node Port	Connect Type	Protocol Type	Code
01	10001	192.168.001.001	10001	TCP	Extended MEMOBUS	BIN
02	10002	192.168.001.001	10002	TCP	Extended MEMOBUS	BIN
03	10003	192.168.001.001	10003	TCP	Extended MEMOBUS	BIN
04	-----					
05	10005	192.168.001.001	10005	TCP	Extended MEMOBUS	BIN
06	10006	192.168.001.001	10006	TCP	Extended MEMOBUS	BIN
07	-----					
08	-----					
09	-----					

■ How to set up with a connection numbers 05, 06 in the **CP-218 Connection Parameter** Area

① Set **Local Port** to the port number used in the MP2300 side (“10005, 10006,” for example).
② Set **Node IP Address** to the IP address configured in the CPU-03/CPU-04 side (“192.168.001.001,” for example).
③ Set **Node Port** to the port number configured in the CPU-03/CPU-04 side (“10005, 10006,” for example).
④ Select **Connect Type** (TCP, for example).
⑤ Select **Extended MEMOBUS** for **Protocol Type**.
⑥ Select **Code** (BIN, for example).

■ Caution

When any transmission or connection parameter is changed, the change will be reflected after it has been saved in flash memory and the power supply is turned ON again.

---

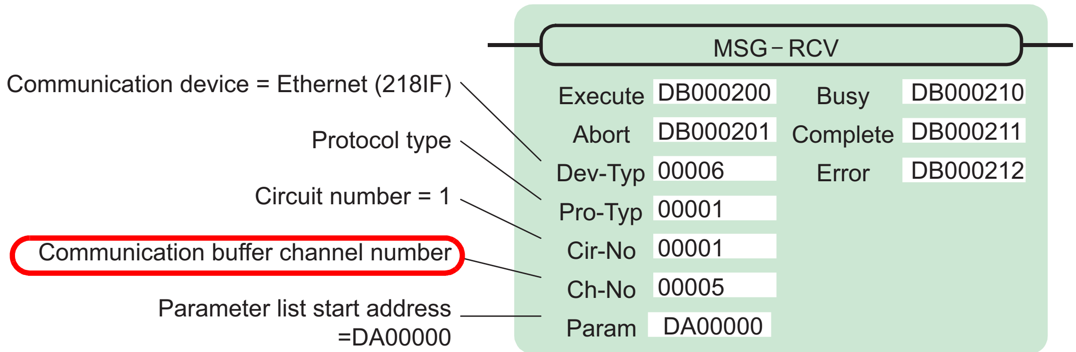
4. Create a ladder program with a message receive function (MSG-RCV) in it.

An example of a ladder program for receiving messages in the remote device (MP2300) side follows:

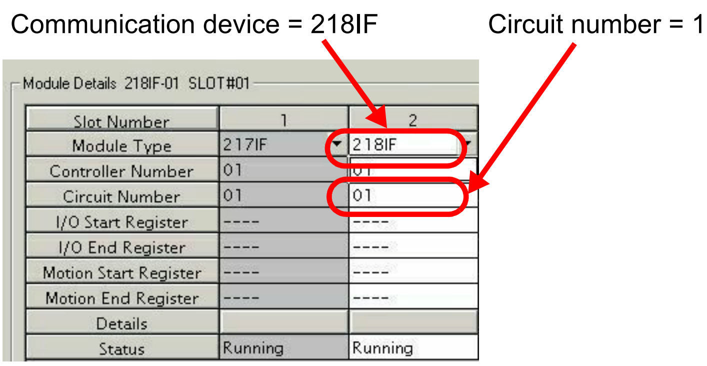
■ Message receive function (MSG-RCV)

This system function is required for receiving messages. Message reception is carried out by describing and executing this message receive function in a ladder program.

In addition, in order to support Read and Write by the CPU-03/CPU-04, two message receive functions should be provided. Here, the input item and parameters (Communication buffer channel number and Connection number) of the message receive function need to accord with the CPU-03/CPU-04 side settings.



Note: Similarly, a message receive function with the communication buffer channel number = 6 is required.



Slot Number	1	2
Module Type	217IF	218IF
Controller Number	01	01
Circuit Number	01	01
I/O Start Register	----	----
I/O End Register	----	----
Motion Start Register	----	----
Motion End Register	----	----
Details		
Status	Running	Running

Fig. 6.7 MPE720 Module Configuration Window

■ Input/output definition contents for message receive function

The input/output definition content for message receive function is as follows:

Table 6.9 Input/Output Definitions for Message Receive Function

I/O Definition	No.	Name	Setting Example	Content
Input Item	1	Execute	DB000200	Executes a reception When Execute is ON, message reception is carried out.
	2	Abort	DB000201	Forcibly aborts a reception When Abort is ON, the message reception is forcibly stopped.
	3	Dev-Typ	00006	Communication device type Specify the type of the communication device used in reception. When Ethernet (218IF) is used, specify “6.”
	4	Pro-Typ	00001	Communication protocol Specify the type of the communication protocol. MEMOBUS(*1) = 1, non-procedure 1(*2) = 2, non-procedure 2(*2) = 3
	5	Cir-No	00001	Circuit number Specify a circuit number of the communication device. Specify it in accordance with the circuit number displayed in the MPE720 **Module Configuration** Window.
	6	Ch-No	00005 & 00006	Communication buffer channel number Specify the channel number of a communication buffer. When Ethernet (218IF) is used, specify it in the range between “1” and “10.” Note: Set up a unique channel number in the circuit.
	7	Param	DA00000	Parameter list start address Specify the start address of the parameter list. For the Parameter List, 17 words are automatically assigned from the configured address.
Output Item	1	Busy	DB000210	In process Busy will be ON while executing a message reception or forced abort process.
	2	Complete	DB000211	Process completed When a message reception or forced abort process is properly completed, Complete will turn ON only for one scan.
	3	Error	DB000212	Error When an error occurs, Error will turn ON only for one scan.

* 1. When transmitting in MEMOBUS, Extended MEMOBUS, MELSEC, or MODBUS/TCP protocol, set the communication protocol (Pro-Typ) to MEMOBUS (=1). The communication device automatically converts the protocol.
* 2. Non-procedure 1: In non-procedural communication, data is received on a per-word basis.
Non-procedure 2: In non-procedural communication, data is received on a per-byte basis.

■ Parameter list setting example for message receive function

An example of a parameter list setting when receiving messages from a transmit source using the connection with connection numbers = 5 and 6 follows:

Table 6.10 Parameter List Setting Example 1 (parameter list start address Param = DA00000)

Register Number	Setting Value	Parameter Number	IN/OUT	Remarks
DW00000	–	PARAM00	OUT	Process result
DW00001	–	PARAM01	OUT	Status
DW00002	00005	PARAM02	IN	Connection number = 5 (For receiving read operation)
DW00003	–	PARAM03	OUT	Option
DW00004	–	PARAM04	OUT	Function code
DW00005	–	PARAM05	OUT	Data address
DW00006	–	PARAM06	OUT	Data size
DW00007	–	PARAM07	OUT	Remote CPU number
DW00008	00000	PARAM08	IN	Coil offset = 0 word
DW00009	00000	PARAM09	IN	Input relay offset = 0 word
DW00010	00000	PARAM10	IN	Input register offset = 0 word
DW00011	00000	PARAM11	IN	Holding register offset = 0 word
DW00012	00000	PARAM12	IN	Writable address lower limit = MW00000
DW00013	65534	PARAM13	IN	Writable address upper limit = MW65534
DW00014	–	PARAM14	SYS	Reserved by the system. (Zero clear at startup)
DW00015	–	PARAM15	SYS	Reserved by the system.
DW00016	–	PARAM16	SYS	Reserved by the system.

Note: IN: Input, OUT: Output, SYS: For system use

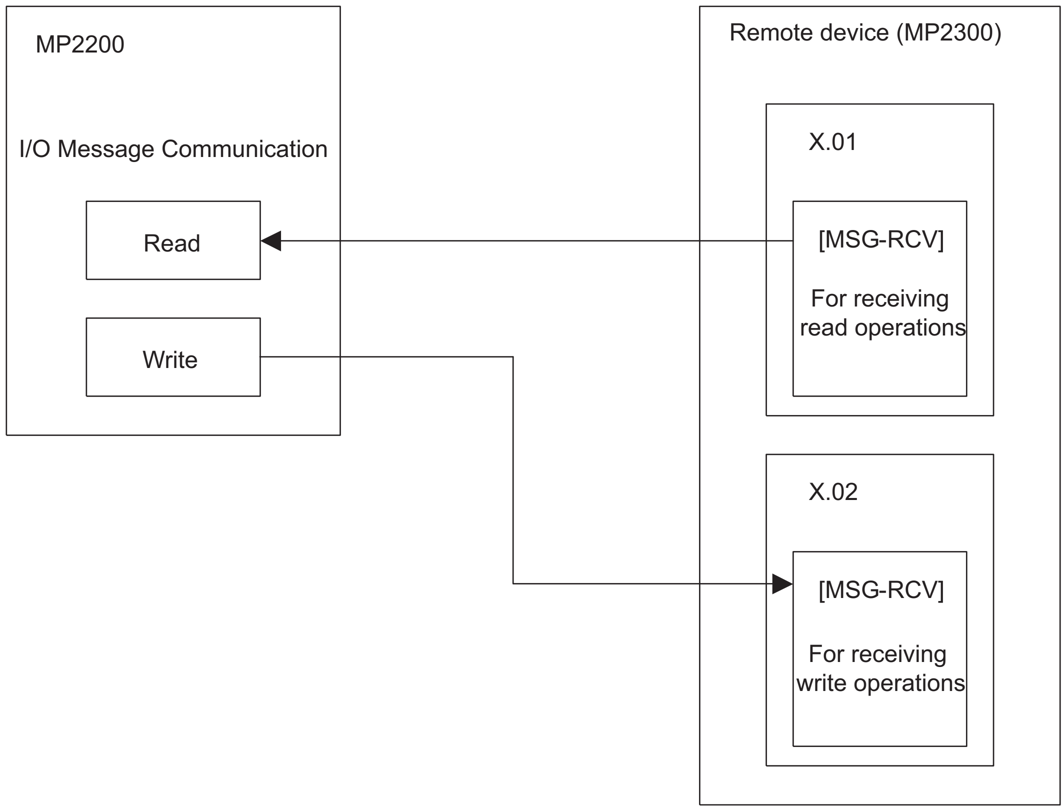
Table 6.11 Parameter List Setting Example 2 (parameter list start address Param = DA00000)

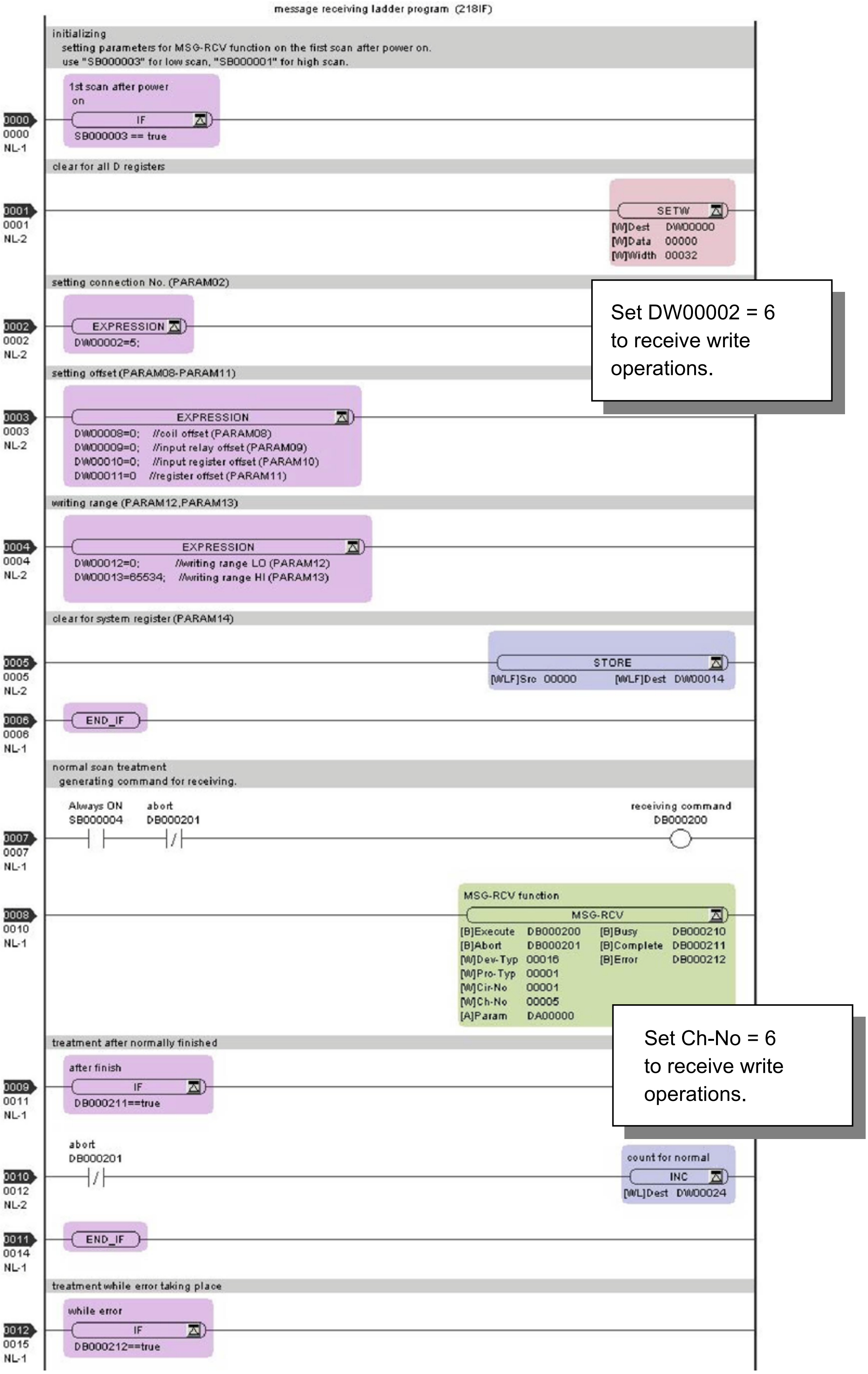
Register Number	Setting Value	Parameter Number	IN/OUT	Remarks
DW00000	–	PARAM00	OUT	Process result
DW00001	–	PARAM01	OUT	Status
DW00002	00006	PARAM02	IN	Connection number = 6 (For receiving write operation)
DW00003	–	PARAM03	OUT	Option
DW00004	–	PARAM04	OUT	Function code
DW00005	–	PARAM05	OUT	Data address
DW00006	–	PARAM06	OUT	Data size
DW00007	–	PARAM07	OUT	Remote CPU number
DW00008	00000	PARAM08	IN	Coil offset = 0 word
DW00009	00000	PARAM09	IN	Input relay offset = 0 word
DW00010	00000	PARAM10	IN	Input register offset = 0 word
DW00011	00000	PARAM11	IN	Holding register offset = 0 word
DW00012	00000	PARAM12	IN	Writable address lower limit = MW00000
DW00013	65534	PARAM13	IN	Writable address upper limit = MW65534
DW00014	–	PARAM14	SYS	Reserved by the system. (Zero clear at startup)
DW00015	–	PARAM15	SYS	Reserved by the system.
DW00016	–	PARAM16	SYS	Reserved by the system.

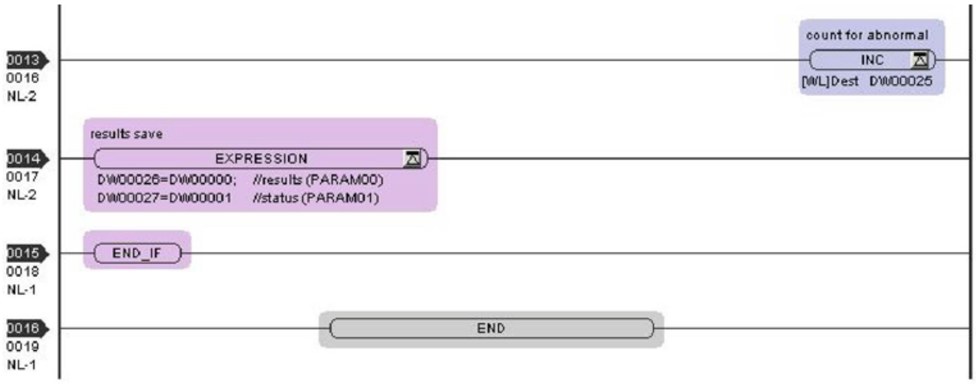
Note: IN: Input, OUT: Output, SYS: For system use

■ Example of Using the Message Receive Function in a Ladder Program

Here is one example of using the message receive function through Ethernet (218IF). In addition, this ladder program is for receiving read operation. A ladder program for receiving write operations is required separately.




message receiving ladder program (218IF)
initializing
setting parameters for MSG-RCV function on the first scan after power on.
use "SB000003" for low scan, "SB000001" for high scan.
1st scan after power on
IF
SB000003 == true
clear for all D registers
SETW
[W]Dest DW00000
[W]Data 00000
[W]Width 00032
setting connection No. (PARAM02)
EXPRESSION
DW00002=5;
Set DW00002 = 6 to receive write operations.
setting offset (PARAM08-PARAM11)
EXPRESSION
DW00008=0; //coil offset (PARAM08)
DW00009=0; //input relay offset (PARAM09)
DW00010=0; //input register offset (PARAM10)
DW00011=0 //register offset (PARAM11)
writing range (PARAM12,PARAM13)
EXPRESSION
DW00012=0; //writing range LO (PARAM12)
DW00013=65534; //writing range HI (PARAM13)
clear for system register (PARAM14)
STORE
[WLF]Src 00000 [WLF]Dest DW00014
END_IF
normal scan treatment
generating command for receiving.
Always ON SB000004
abort DB000201
receiving command DB000200
MSG-RCV function
MSG-RCV
[B]Execute DB000200
[B]Abort DB000201
[W]Dev-Typ 00016
[W]Pro-Typ 00001
[W]Cir-No 00001
[W]Ch-No 00005
[A]Param DA00000
[B]Busy DB000210
[B]Complete DB000211
[B]Error DB000212
Set Ch-No = 6 to receive write operations.
treatment after normally finished
after finish
IF
DB000211==true
abort DB000201
count for normal
INC
[WL]Dest DW00024
END_IF
treatment while error taking place
while error
IF
DB000212==true



The communication setting and the ladder program creation are now finished, when MP2300 acts as a slave.

(3) How to Start Communication

1. The MP2300 side starts to receive the messages.

As the sample ladder program starts the message receive operation just after the system startup, you are not required to do anything. In normal operation, accept the default.

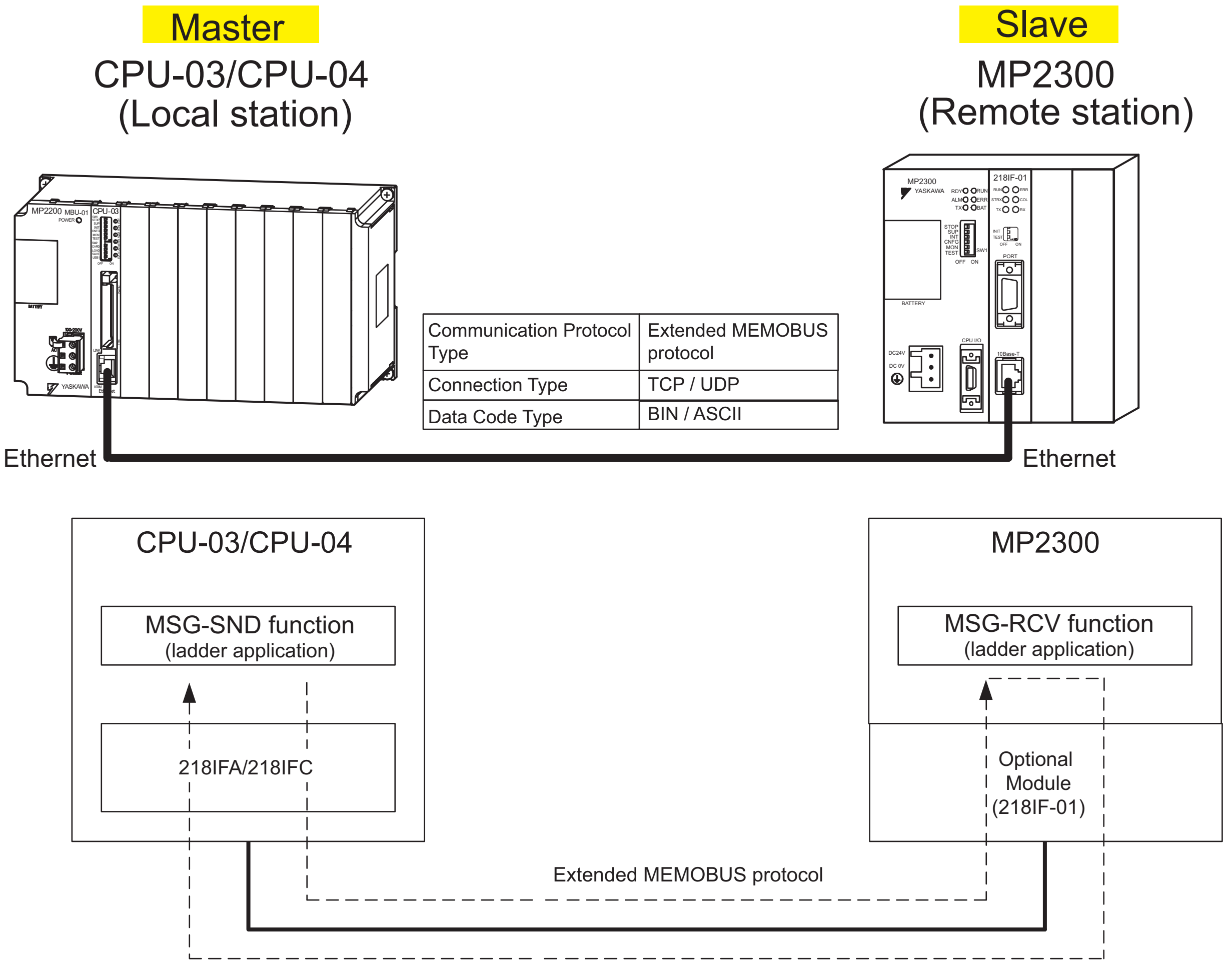
2. The CPU-03/CPU-04 side transmits messages.

When an I/O message communication function is used, message transmit operation status automatically.

6.2.4 When the CPU-03/CPU-04 Acts as Master (ladder program which uses MSG-SND function)

The I/O message communication function cannot operate any registers other than the holding register (M register). Note that it can communicate with only one slave.
To communicate with more than one slave, use a ladder program with a message send function (MSG-SND) on the CPU-03/CPU-04 side. You can use the message transmit (MSG-SND) function as well as the I/O message communication function by keeping connections separate from each other.

This section explains how to communicate with the CPU-03/CPU-04 message receive function (MSG-RCV) using the MP2300 message send function (MSG-SND).

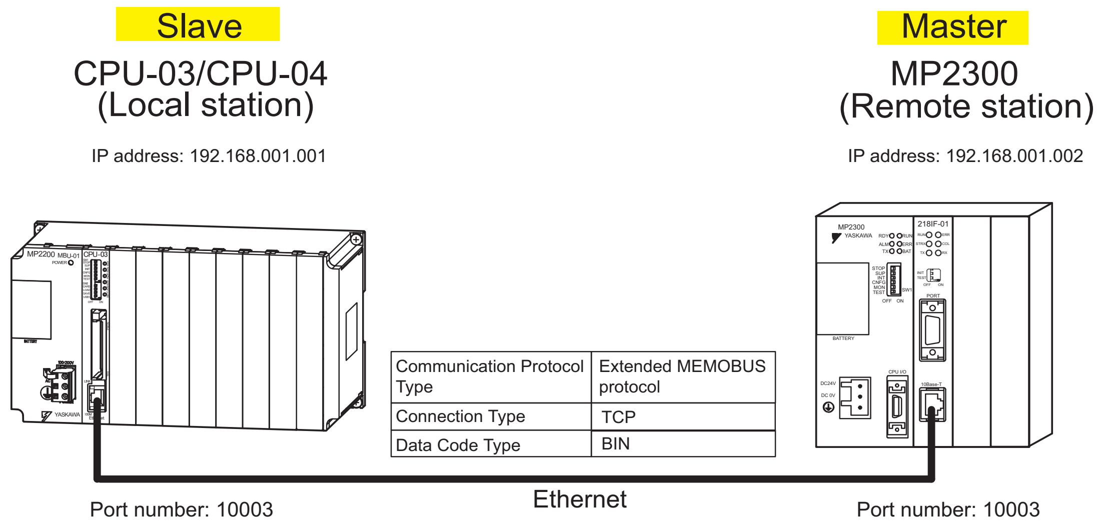


Communication Protocol Type	Extended MEMOBUS protocol
Connection Type	TCP / UDP
Data Code Type	BIN / ASCII

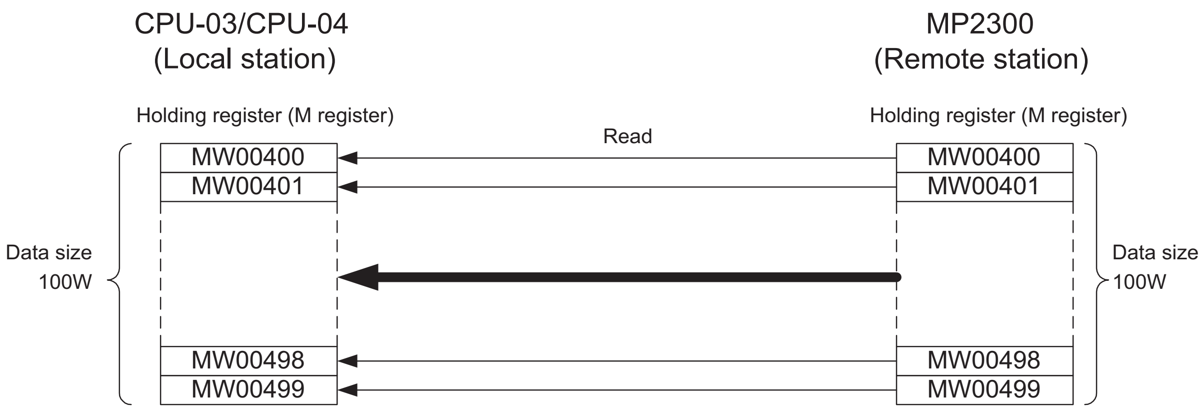
Fig. 6.8 Message Flow with MP2300 when Message Send Function (MSG-SND) Is Used

■ Setting Example

The following figure illustrates one example of reading the content of the MP2300 (slave) holding register (MW00400 to MW00499) into the CPU-03/CPU-04 (master) holding register (MW00400 to MW00499).



Communication Protocol Type	Extended MEMOBUS protocol
Connection Type	TCP
Data Code Type	BIN

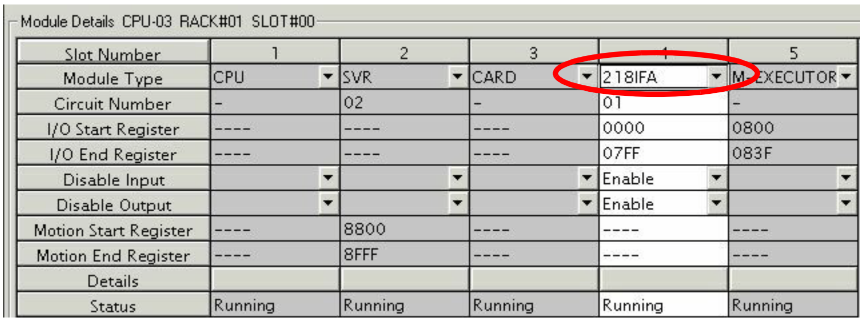


The setup procedure is explained in the following pages.

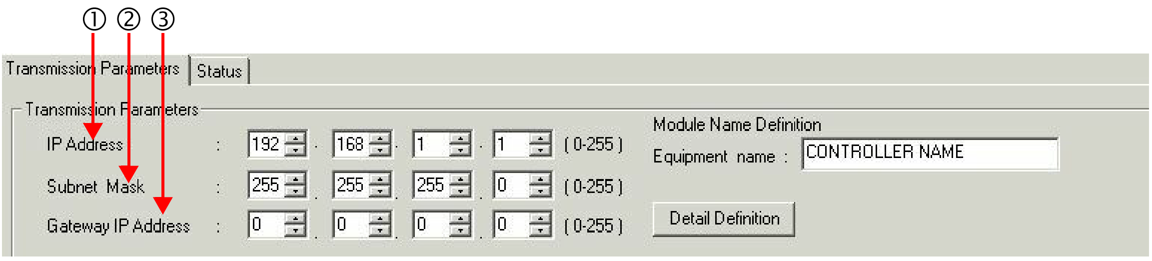
(1) How to Set up the CPU-03/CPU-04 Side

When the setting of transmission parameters (IP address, subnet mask) is already completed, start from step 3.

1. Double-click the **218IFA** Tab (**218IFC** Tab for the CPU-04) in the **Module Details** Area in the CPU-03 **Module Configuration** Window.



2. Set transmission parameters.



■ How to set up transmission parameters

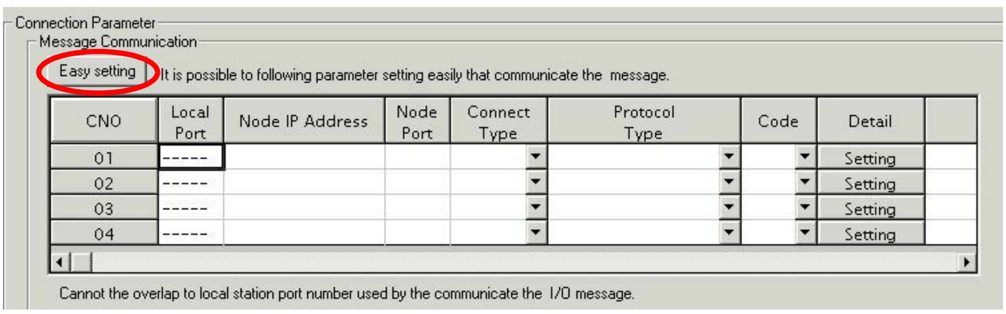
① Set **IP Address** (“192.168.001.001,” for example).
② Set **Subnet Mask** (“255.255.255.000,” for example).
③ Set **Gateway IP Address** (“000.000.000.000,” for example).

■ Caution

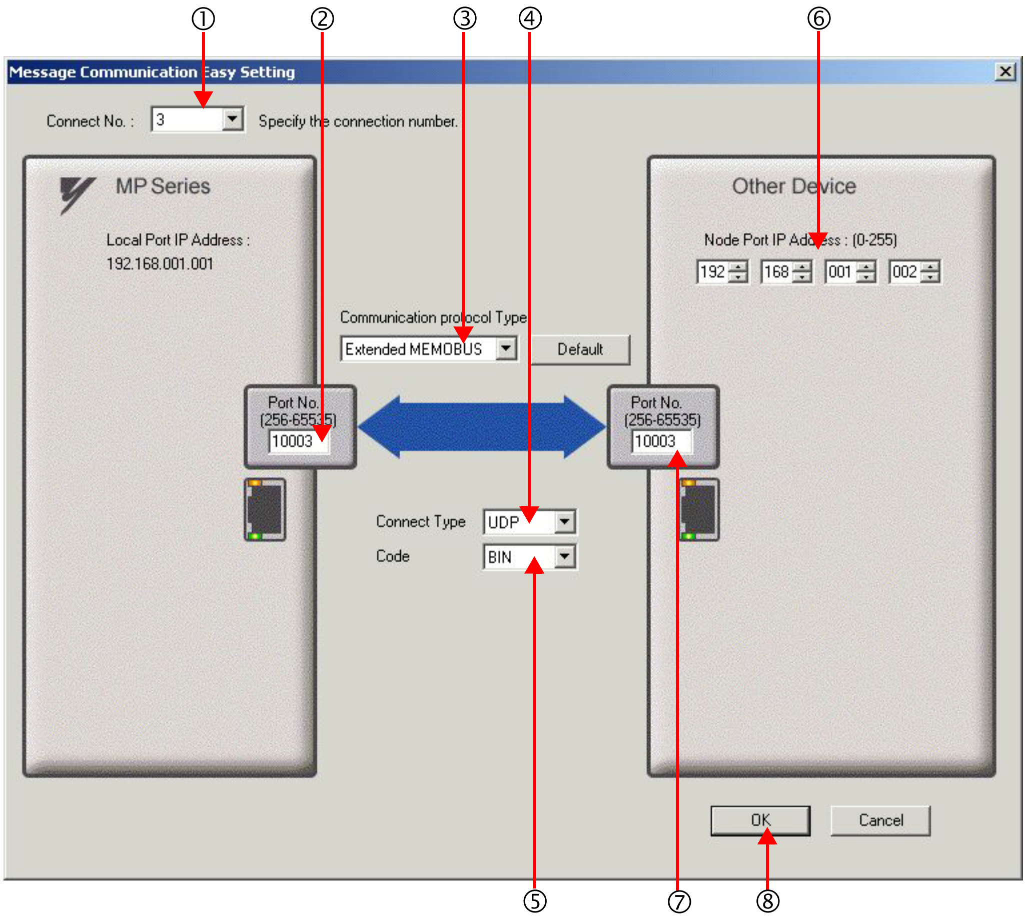
Set up a unique IP address in the network.
For the IP address, check with your network administrator.

---

3. Click the **Easy setting** Button in the **Message Communication** Area of the connection parameter setting.



4. Set a communication setting in the **Message Communication Easy Setting** Dialog Box.



■ How to set up in the **Message Communication Easy Setting** Dialog Box

① When automatic receive function is used, select "3" for the **Connect No.**
② Set **Port No.** of the CPU-03/CPU-04 side ("10003," for example).
③ Select **Extended MEMOBUS** for **Communication protocol Type**, and click the **Default** Button.
④ Select **Connect Type** (TCP, for example).
⑤ Select **Code** (BIN, for example).
⑥ Set **Node Port IP Address** of the other device (MP2300) to be connected ("192.168.001.002," for example).
⑦ Set **Port No.** of the other device (MP2300) to be connected ("10003," for example).
⑧ Click **OK**.

■ Caution

When message functions (MSG-SND, MSG-RCV) are used with the connection number 01, disable the automatic receive function. If message functions are used while the automatic receive function is enabled, the communication will not function properly.

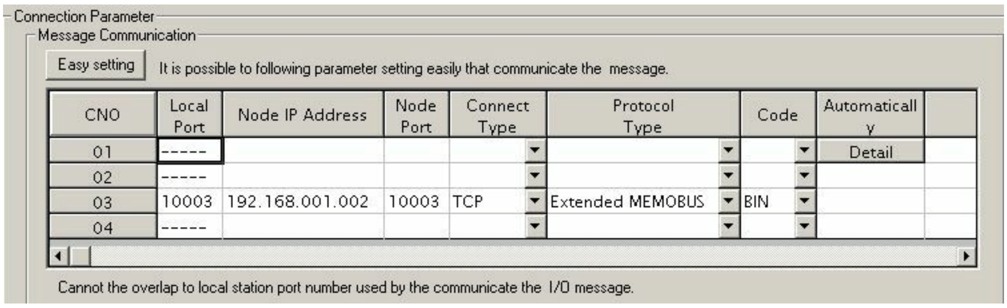
Note: The automatic receive function with a connection number 01 is set to "Enable" by default.

5. Click **Yes** in the parameter setting confirmation dialog box.

■ Caution

Note that when a parameter with the same connection number is already set and you click **Yes** in the parameter setting confirmation dialog, the setting will be overwritten by the parameter configured in the **Message Communication Easy Setting** Dialog Box.

6. Check the setting values.



Connection Parameter
Message Communication
Easy setting It is possible to following parameter setting easily that communicate the message.

CNO	Local Port	Node IP Address	Node Port	Connect Type	Protocol Type	Code	Automatically
01	-----						Detail
02	-----						
03	10003	192.168.001.002	10003	TCP	Extended MEMOBUS	BIN	
04	-----						

Cannot the overlap to local station port number used by the communicate the I/O message.

■ Caution

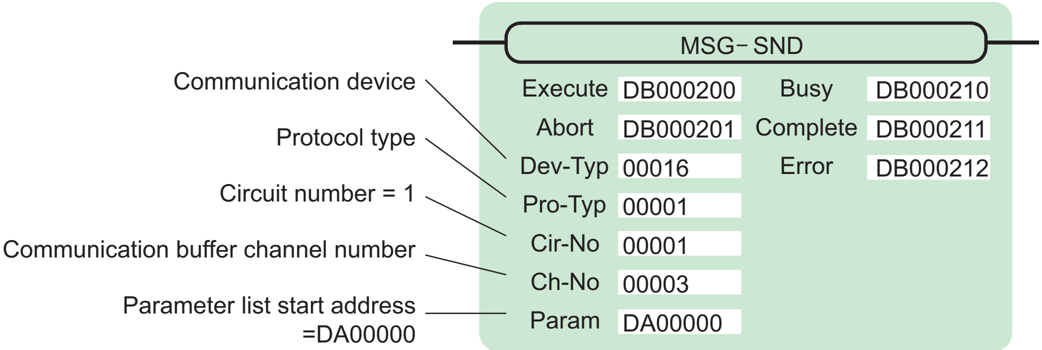
When any transmission or connection parameter is changed, the change will be reflected after it has been saved in flash memory and the power supply is turned ON again.

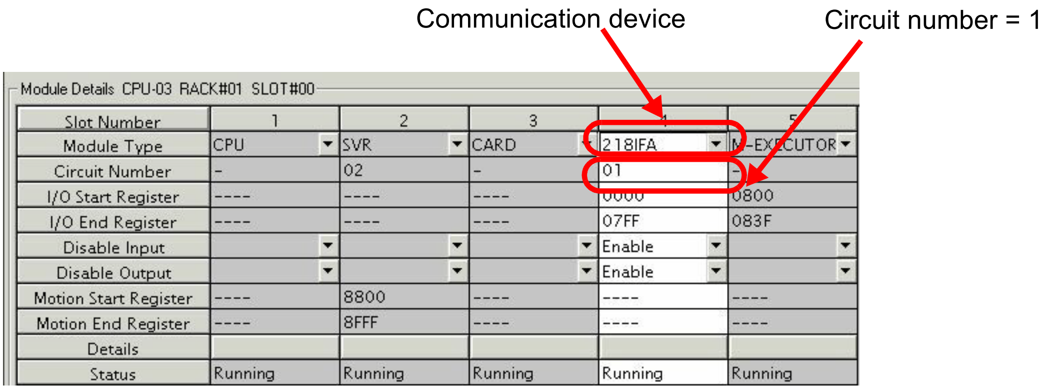
7. Create a ladder program containing a message send function (MSG-SND).

An example of a ladder program example for transmitting messages from the CPU-03/CPU-04 side follows:

■ Message send function (MSG-SND)

This system function is required for transmitting messages. A message transmission is carried out by describing and executing this message send function in a ladder program.





Slot Number	1	2	3	4	5
Module Type	CPU	SVR	CARD	218IFA	M-EXECUTOR
Circuit Number	-	02	-	01	-
I/O Start Register	----	----	----	0000	0800
I/O End Register	----	----	----	07FF	083F
Disable Input				Enable	
Disable Output				Enable	
Motion Start Register	----	8800	----	----	----
Motion End Register	----	8FFF	----	----	----
Details					
Status	Running	Running	Running	Running	Running

Fig. 6.9 MPE720 Module Configuration Window

■ Input/output definition contents for message send function

The input/output definition content for the message send function is as follows:

Table 6.12 Input/Output Definitions for Message Send Function

I/O Definition	No.	Name	Setting Example	Contents
Input Item	1	Execute	DB000200	Executes a transmission When Execute is ON, the message transmission is carried out.
	2	Abort	DB000201	Aborts a transmission When the Abort is ON, message transmission is forcibly stopped.
	3	Dev-Typ	00016	Communication device type Specify the type of the communication device used in reception. When Ethernet (218IFA/218IFC) is used, specify 16.
	4	Pro-Typ	00001	Communication protocol Specify the type of the communication protocol. MEMOBUS(*1) = 1, non-procedure 1(*2) = 2, non-procedure 2(*2) = 3
	5	Cir-No	00001	Circuit number Specify a circuit number of the communication device. Specify it in accordance with the circuit number displayed in the MPE720 **Module Configuration** Window.
	6	Ch-No	00003	Communication buffer channel number Specify the channel number of a communication buffer. • When Ethernet (218IFA) is used, specify a number between 1 and 4. • When Ethernet (218IFC) is used, specify a number between 1 and 10. Note: Set up a unique channel number in the circuit.
	7	Param	DA00000	Parameter list start address Specify the start address of the parameter list. For the Parameter List, 17 words are automatically assigned from the configured address.
Output Item	1	Busy	DB000210	In process Busy is ON while executing a message transmission or forced abort process.
	2	Complete	DB000211	Process completed When a message transmission or abort process is properly completed, Complete will turn ON only for one scan.
	3	Error	DB000212	Error occurred When an error occurs, Error will turn ON only for one scan.

* 1. When transmitting in MEMOBUS, Extended MEMOBUS, MELSEC, or MODBUS/TCP protocol, set the communication protocol (Pro-Typ) to MEMOBUS (=1). The communication device automatically converts the protocol.

* 2. Non-procedure 1: In non-procedural communication, data is transmitted on a per-word basis.
Non-procedure 2: In non-procedural communication, data is transmitted on a per-byte basis.

■ Parameter list setting example for the message send function

An example of parameter list settings when writing 100 words of data from MW00400 to the destination using a connection with a connection number of 3 follows:

Table 6.13 Parameter List Setting Example (parameter list start address Param = DA00000)

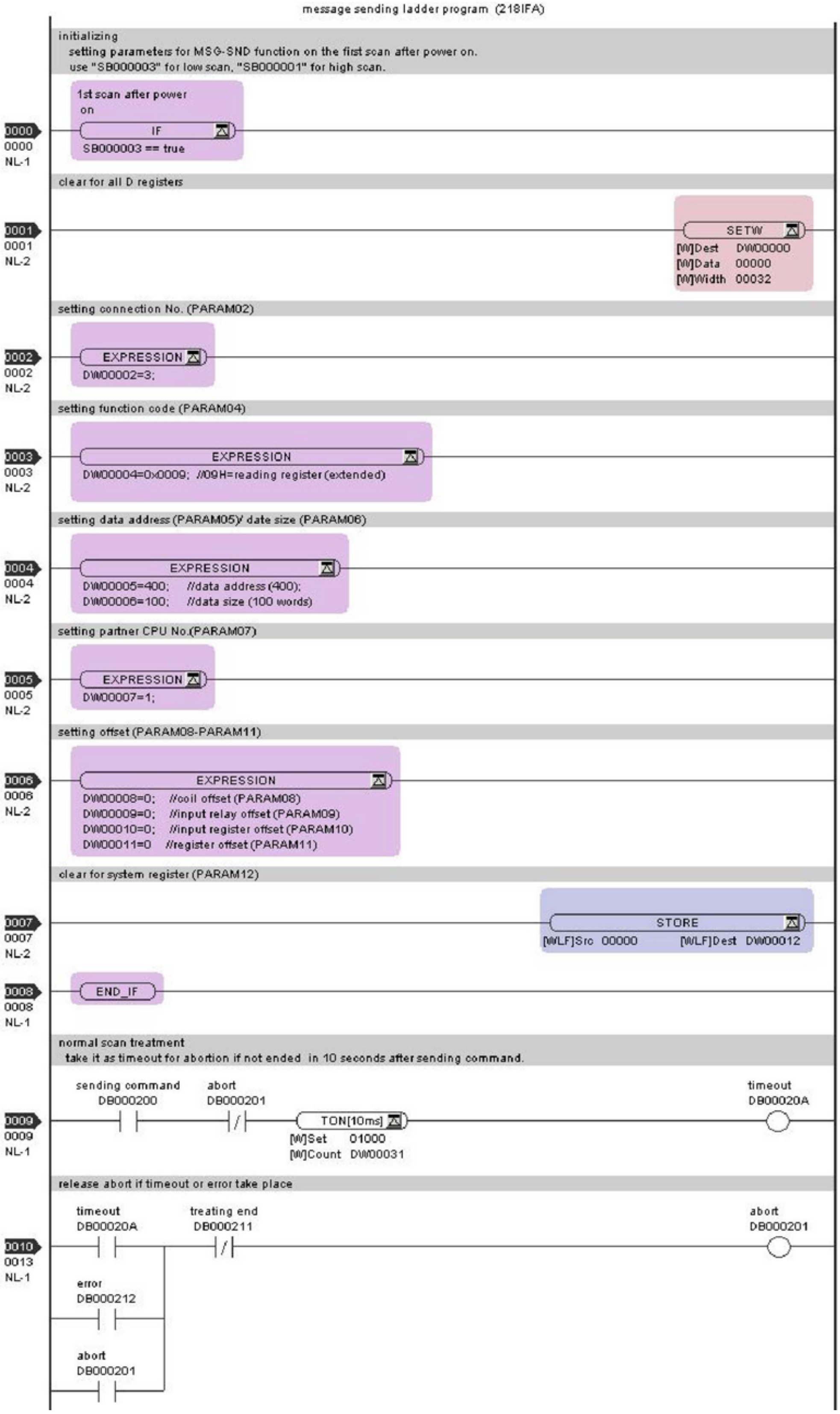
Register Number	Setting Value	Parameter Number	IN/OUT	Remarks
DW00000	–	PARAM00	OUT	Process result
DW00001	–	PARAM01	OUT	Status
DW00002	00003	PARAM02	IN	Connection number = 3
DW00003	–	PARAM03	IN	Option (Setting unnecessary)
DW00004	0009H	PARAM04	IN	Function code = 09H (Reads a holding register)
DW00005	00400	PARAM05	IN	Data address = 400 (Starting from MW00400)
DW00006	00100	PARAM06	IN	Data size = 100 (100 words)
DW00007	00001	PARAM07	IN	Remote CPU number = 1
DW00008	00000	PARAM08	IN	Coil offset = 0 word
DW00009	00000	PARAM09	IN	Input relay offset = 0 word
DW00010	00000	PARAM10	IN	Input register offset = 0 word
DW00011	00000	PARAM11	IN	Holding register offset = 0 word
DW00012	–	PARAM12	SYS	Reserved by the system. (Zero clear at startup)
DW00013	–	PARAM13	SYS	Reserved by the system.
DW00014	–	PARAM14	SYS	Reserved by the system.
DW00015	–	PARAM15	SYS	Reserved by the system.
DW00016	–	PARAM16	SYS	Reserved by the system.

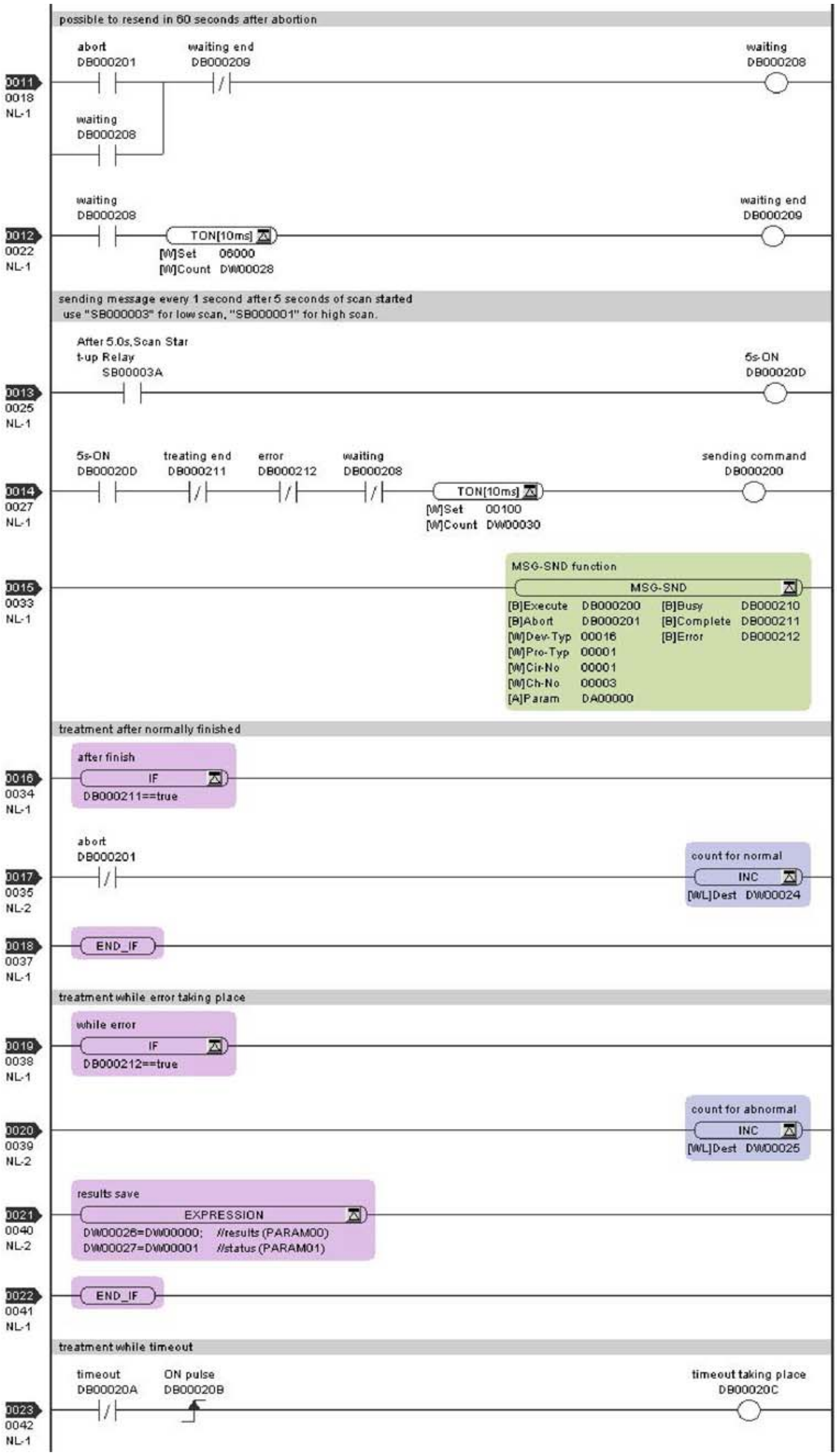
Note: IN: Input, OUT: Output, SYS: For system use

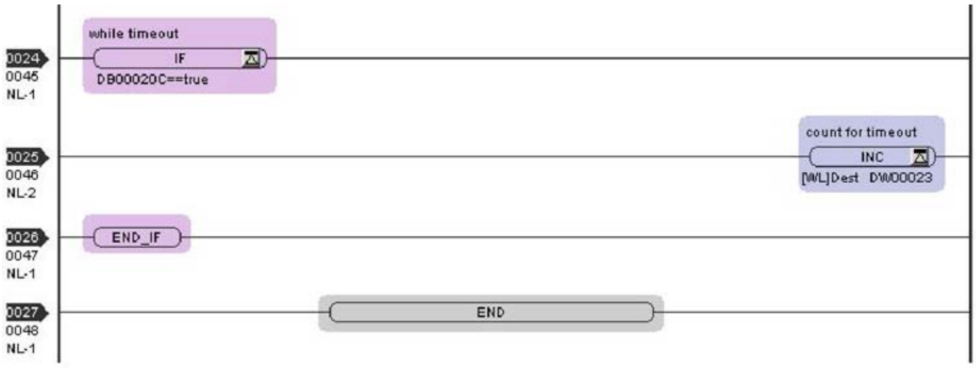
■ Example of Using the Message Send Function in a Ladder Program

Here is one example of using the message send function through Ethernet (218IFA/218IFC).

message sending ladder program (218IFA)




possible to resend in 60 seconds after abortion
abort DB000201
waiting end DB000209
waiting DB000208
0011 0018 NL-1
waiting DB000208
waiting DB000208
waiting end DB000209
0012 0022 NL-1
TON[10ms]
[W]Set 06000
[W]Count DW00028
sending message every 1 second after 5 seconds of scan started
use "SB000003" for low scan, "SB000001" for high scan.
After 5.0s,Scan Star t-up Relay SB00003A
5s-ON DB00020D
0013 0025 NL-1
5s-ON DB00020D
treating end DB000211
error DB000212
waiting DB000208
sending command DB000200
0014 0027 NL-1
TON[10ms]
[W]Set 00100
[W]Count DW00030
MSG-SND function
0015 0033 NL-1
MSG-SND
[B]Execute DB000200
[B]Abort DB000201
[W]Dev-Typ 00016
[W]Pro-Typ 00001
[W]Cir-No 00001
[W]Ch-No 00003
[A]Param DA00000
[B]Busy DB000210
[B]Complete DB000211
[B]Error DB000212
treatment after normally finished
after finish
0016 0034 NL-1
IF
DB000211==true
abort DB000201
0017 0035 NL-2
count for normal
INC
[WL]Dest DW00024
0018 0037 NL-1
END_IF
treatment while error taking place
while error
0019 0038 NL-1
IF
DB000212==true
0020 0039 NL-2
count for abnormal
INC
[WL]Dest DW00025
results save
0021 0040 NL-2
EXPRESSION
DW00026=DW00000; //results (PARAM00)
DW00027=DW00001 //status (PARAM01)
0022 0041 NL-1
END_IF
treatment while timeout
timeout DB00020A
ON pulse DB00020B
timeout taking place DB00020C
0023 0042 NL-1

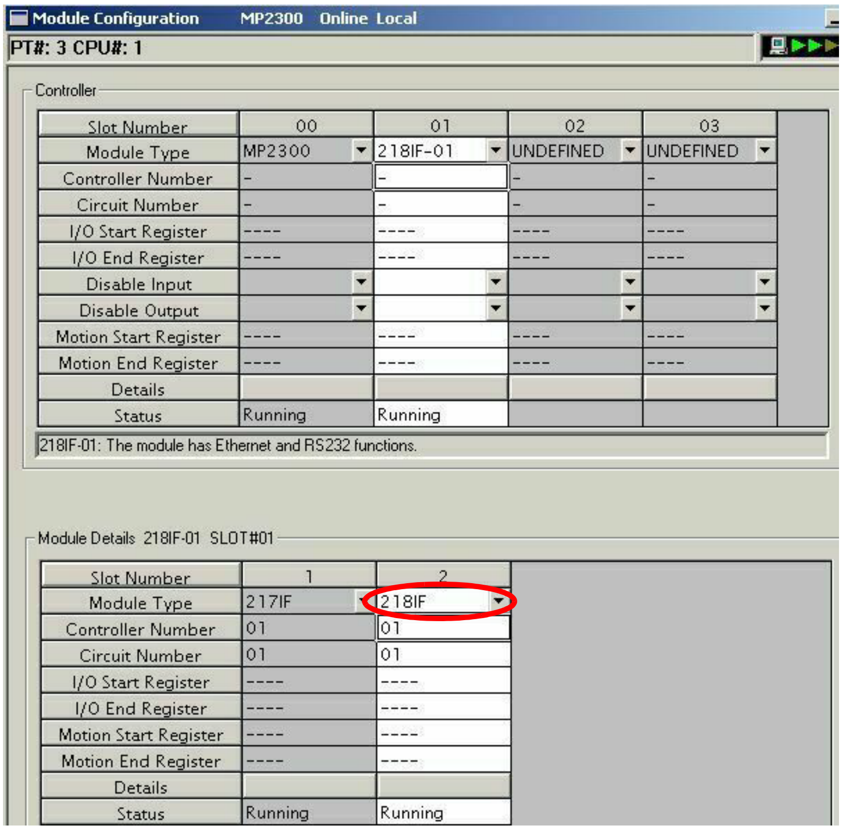


The communication setting and the ladder program creation are now finished, when the CPU-03/CPU-04 acts as a master.

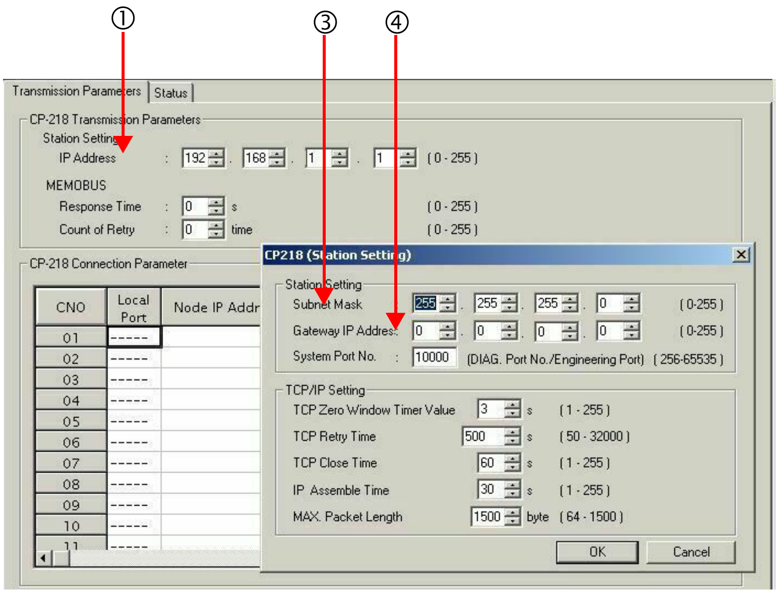
(2) How to Set up the Remote Device (MP2300) to Be Connected

If the setting of transmission parameters (IP address, subnet mask) is already completed, start from step 3.

1. Double-click the **218IF** Tab in the **Module Details** Area of the **Module Configuration** Window.



2. Set transmission parameters.



■ How to set up transmission parameters

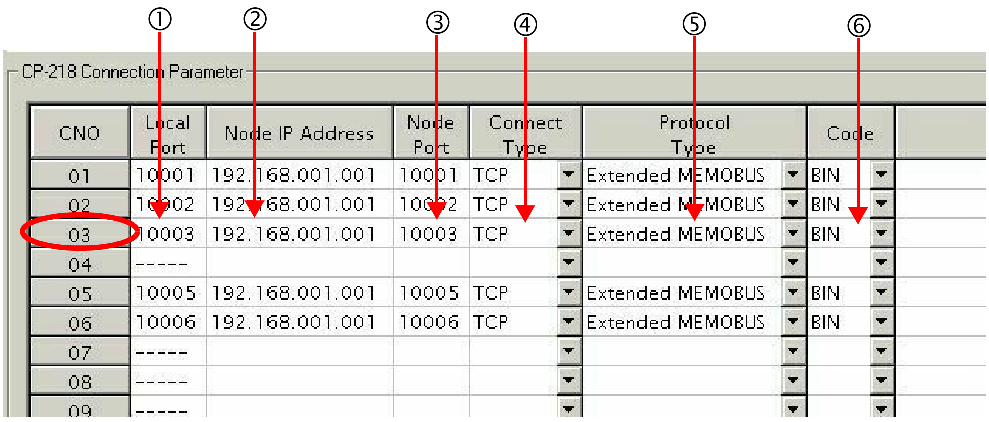
① Set **IP Address** (“192.168.001.001,” for example).
② Select ***Edit - Local Station: TCP/IP Setting*** in the Engineering Manager Window the **Detail Setting** Dialog Box will appear.
③ Set **Subnet Mask** (“255.255.255.000,” for example).
④ Set **Gateway IP Address** (“000.000.000.000,” for example).

■ Caution

Set up a unique IP address in the network.
For the IP address, check with your network administrator.

---

3. Set connection parameters.



CNO	Local Port	Node IP Address	Node Port	Connect Type	Protocol Type	Code
01	10001	192.168.001.001	10001	TCP	Extended MEMOBUS	BIN
02	10002	192.168.001.001	10002	TCP	Extended MEMOBUS	BIN
03	10003	192.168.001.001	10003	TCP	Extended MEMOBUS	BIN
04	-----					
05	10005	192.168.001.001	10005	TCP	Extended MEMOBUS	BIN
06	10006	192.168.001.001	10006	TCP	Extended MEMOBUS	BIN
07	-----					
08	-----					
09	-----					

■ Procedure to set up with a connection number 03 in the **CP-218 Connection Parameter** Area

① Set **Local Port** to the port number used in the MP2300 side (“10003,” for example).
② Set **Node IP Address** to the IP address configured in the CPU-03/CPU-04 side (“192.168.001.001,” for example).
③ Set **Node Port** to the port number configured in the CPU-03/CPU-04 side (“10003,” for example).
④ Select **Connect Type**, (TCP, for example).
⑤ Select **Extended MEMOBUS** for **Protocol Type**.
⑥ Select **Code** (BIN, for example).

■ Caution

When any transmission or connection parameter is changed, the change will be reflected after it has been saved in flash memory and the power supply is turned ON again.

---

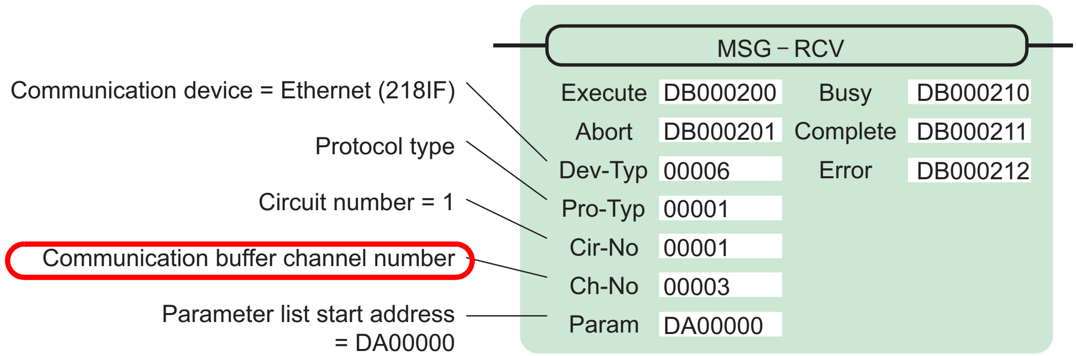
4. Create a ladder program with a message receive function (MSG-RCV) in it.

An example of a ladder program for receiving messages in the remote device (MP2300) side follows:

■ Message receive function (MSG-RCV)

This system function is required for receiving messages. Message reception is carried out by describing and executing this message receive function in a ladder program.

In addition, in order to support Read and Write by CPU-03/CPU-04, two message receive functions should be provided. Here, the input item and parameters (communication buffer channel number and connection number) of the message receive function need to accord with the CPU-03/CPU-04 side settings.



Note: Similarly, a message receive function with the communication buffer channel number=6 is required.

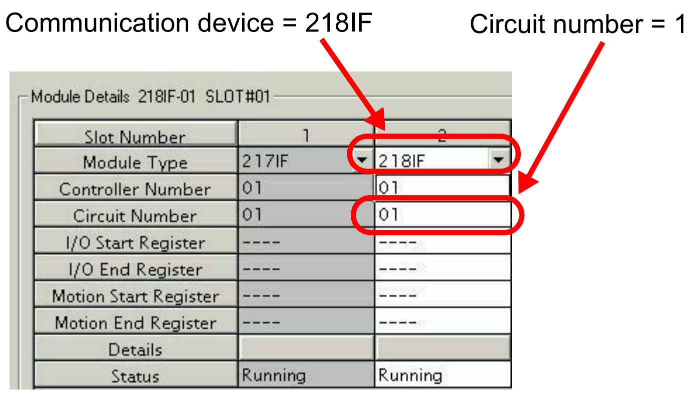


Fig. 6.10 MPE720 Module Configuration Window

■ Input/output definitions contents for message receive function

The input/output definition content for message receive function is as follow:

Table 6.14 Input/Output Definitions for Message Receive Function

I/O Definition	No.	Name	Setting Example	Contents
Input Item	1	Execute	DB000200	Executes a reception When Execute turns ON, the message reception is carried out.
	2	Abort	DB000201	Aborts a reception When Abort turns ON, the message reception is forcibly stopped.
	3	Dev-Typ	00006	Communication device type Specify the type of the communication device used in reception. When Ethernet (218IF) is used, specify “6.”
	4	Pro-Typ	00001	Communication protocol Specify the type of the communication protocol. MEMOBUS(*1) = 1, non-procedure 1(*2) = 2, non-procedure 2(*2) = 3
	5	Cir-No	00001	Circuit number Specify a circuit number of the communication device. Specify it in accordance with the circuit number displayed in the MPE720 **Module Configuration** Window.
	6	Ch-No	00003	Communication buffer channel number Specify the channel number of a communication buffer. When Ethernet (218IF) is used, specify it in the range between “1” and “10.” Note: Set up a unique channel number in the circuit.
	7	Param	DA00000	Parameter list start address Specify the start address of the parameter list. For the Parameter List, 17 words are automatically assigned from the configured address.
Output Item	1	Busy	DB000210	In process Busy will be ON while executing a message reception or forced abort process.
	2	Complete	DB000211	Process completed When a message reception or forced abort process is properly completed, Complete will turn ON only for one scan.
	3	Error	DB000212	Error occurred When an error occurs, Error will turn ON only for one scan.

* 1. When transmitting in MEMOBUS, Extended MEMOBUS, MELSEC, or MODBUS/TCP protocol, set the communication protocol (Pro-Typ) to MEMOBUS(=1). The communication device automatically converts the protocol.

* 2. Non-procedure 1: In non-procedural communication, data is received on a per-word basis.
Non-procedure 2: In non-procedural communication, data is received on a per-byte basis.

■ Parameter list setting example for message receive function

A parameter list setting example is as follows when receiving messages from a transmit source using the connection with a connection number = 3:

Table 6.15 Parameter List Setting Example1 (parameter list start address Param = DA00000)

Register Number	Setting Value	Parameter Number	IN/OUT	Remarks
DW00000	–	PARAM00	OUT	Process result
DW00001	–	PARAM01	OUT	Status
DW00002	00003	PARAM02	IN	Connection number = 3 (For receiving read operation)
DW00003	–	PARAM03	OUT	Option
DW00004	–	PARAM04	OUT	Function code
DW00005	–	PARAM05	OUT	Data address
DW00006	–	PARAM06	OUT	Data size
DW00007	–	PARAM07	OUT	Remote CPU number
DW00008	00000	PARAM08	IN	Coil offset = 0 word
DW00009	00000	PARAM09	IN	Input relay offset = 0 word
DW00010	00000	PARAM10	IN	Input register offset = 0 word
DW00011	00000	PARAM11	IN	Holding register offset = 0 word
DW00012	00000	PARAM12	IN	Writable address lower limit = MW00000
DW00013	65534	PARAM13	IN	Writable address upper limit = MW65534
DW00014	–	PARAM14	SYS	Reserved by the system. (Zero clear at startup)
DW00015	–	PARAM15	SYS	Reserved by the system.
DW00016	–	PARAM16	SYS	Reserved by the system.

Note: IN: Input, OUT: Output, SYS: For system use

■ Example of Using the Message Receive Function in a Ladder Program

A ladder program for receiving messages in the remote device (MP2300) side is similar to that in *6.2.3 (2) How to Set up the Remote Device (MP2300) to Be Connected*.
However, change Communication buffer channel number and Connection number respectively based on the I/O definition and parameter list settings.

(3) How to Start Communication

1. MP2300 side starts to receive the messages.

As the sample ladder program starts the message receive operation just after the system startup, you don't need to operate it particularly. Normally, accept the default.

2. Turn Execute ON for the message send function in the CPU-03/CPU-04 side to transmit messages.

The sample ladder program is created to transmit a message every one second when five seconds elapsed after the low-speed scan (or high-speed scan) startup.
The way to change the message transmission interval is similar to that in *6.2.1 (3) How to Start Communications*.

6.3 Communication with Touch Panel

This section explains how to communicate with a touch panel that supports extended MEMOBUS protocol using the CPU-03/CPU-04 automatic receive function.
In this section, the GP3000 series touch panel manufactured by Digital Electronics Corp. is used.

6.3.1 When CPU-03/CPU-04 Acts as Slave (automatic receive function is used)

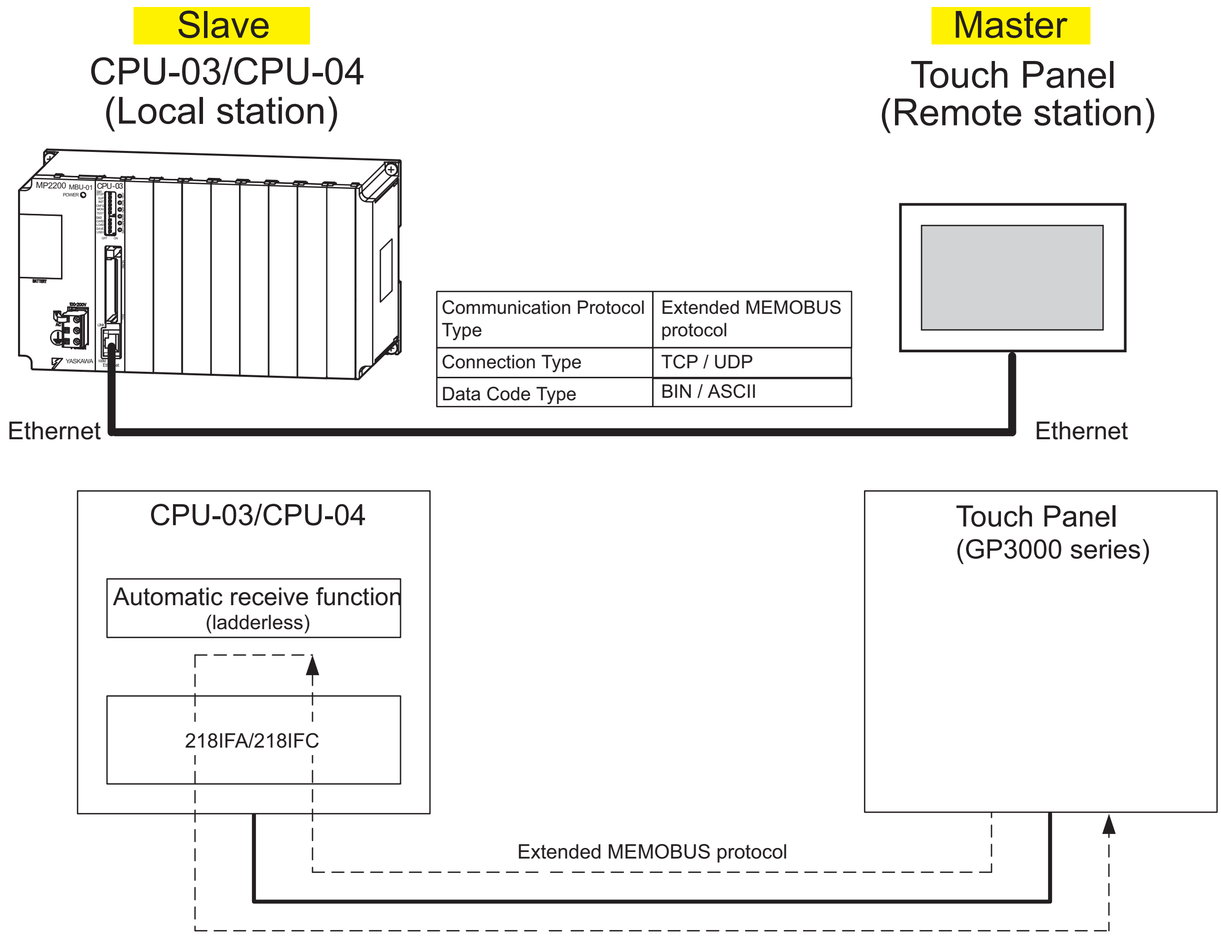
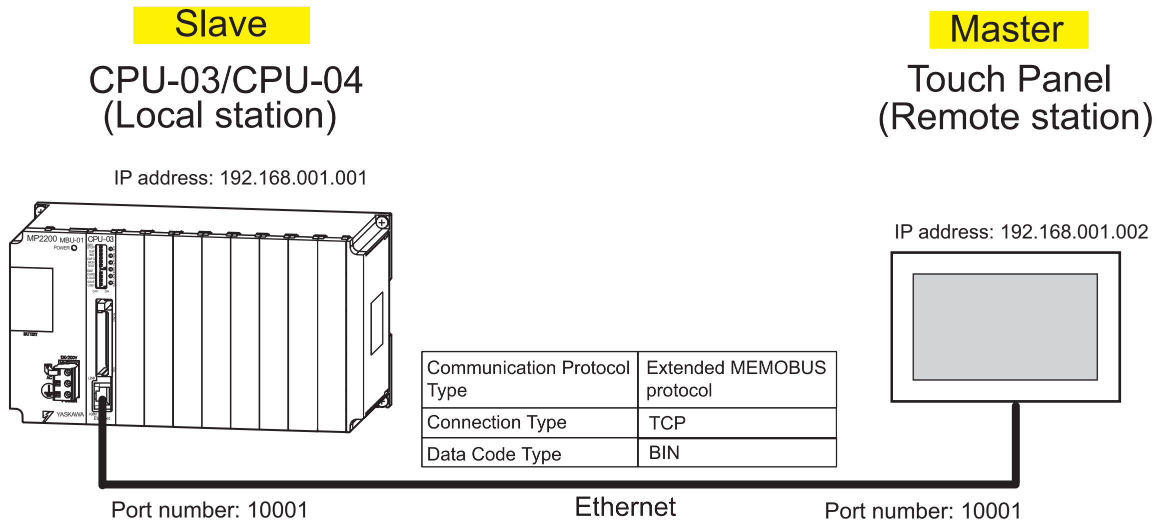


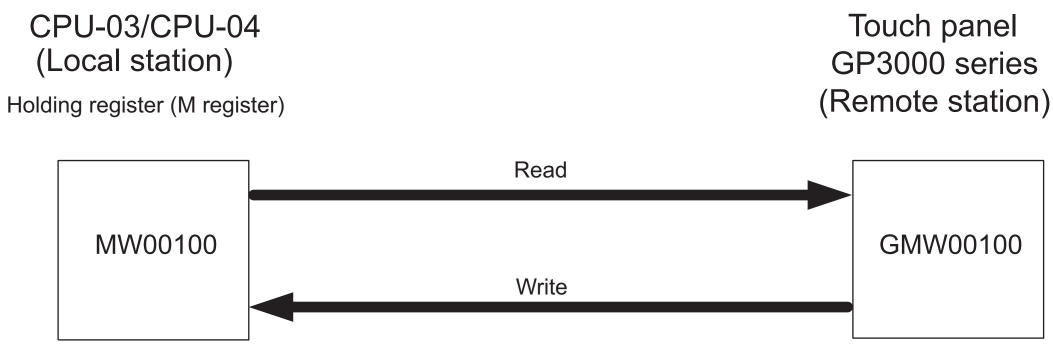
Fig. 6.11 Message Flow with Touch Panel (GP3000 series) when Automatic Receive Function Is Used

Note: Here, communication with the touch panel is carried out using the automatic receive function, but it can also use the message receive function (MSG-RCV). For information on how to set up when the message receive function (MSG-RCV) is used, refer to *6.2.2 (1) How to Set up the CPU-03/CPU-04 Side*.

■ Setting Example

The following figure shows an example which displays the content of the CPU-03/CPU-04 (slave) holding register (MW00100) on a touch panel and writes values from the touch panel to the same register.

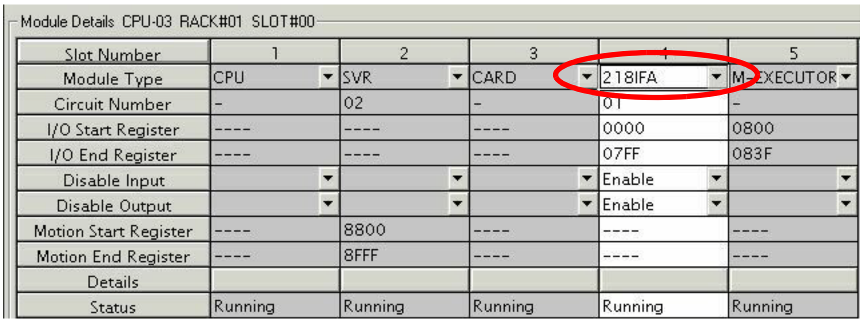




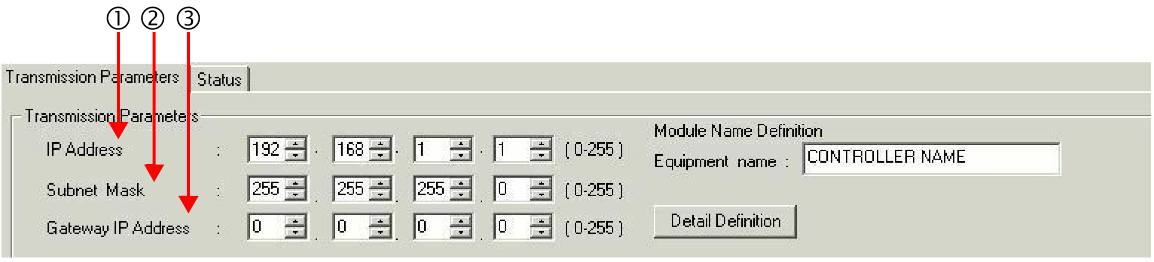
The setup procedure is explained in the following pages.

(1) How to Set up the CPU-03/CPU-04 Side

1. Double-click the **218IFA** Tab (**218IFC** Tab for the CPU-04) in the **Module Details** Area in the CPU-03 **Module Configuration** Window.



2. Set transmission parameters.



■ How to set up transmission parameters

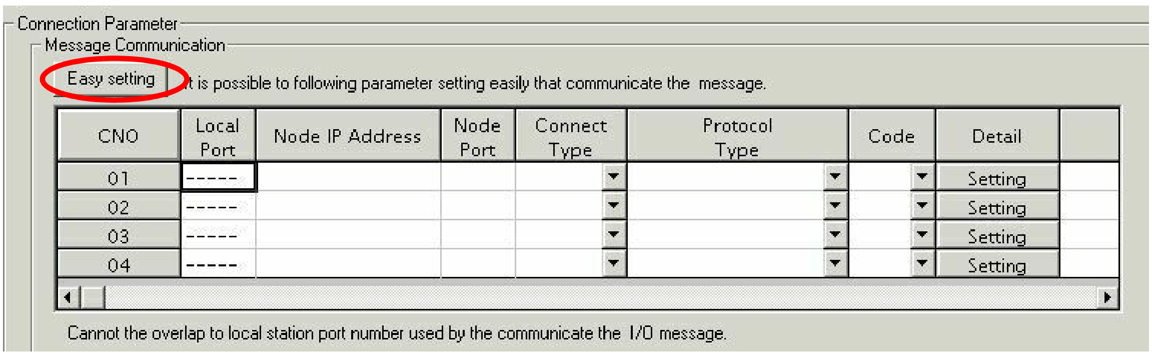
① Set **IP Address** (“192.168.001.001,” for example).
② Set **Subnet Mask** (“255.255.255.000,” for example).
③ Set **Gateway IP Address** (“000.000.000.000,” for example).

■ Caution

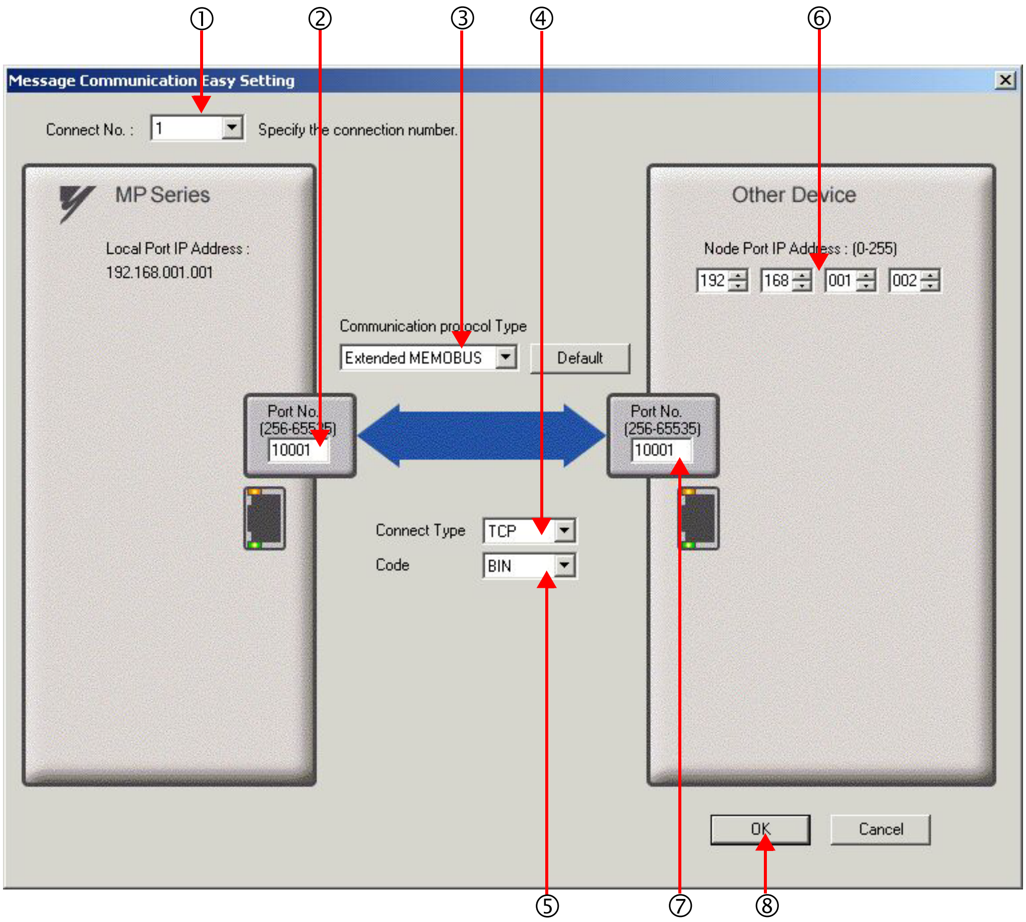
Set up a unique IP address in the network.
For the IP address, check with your network administrator.

---

3. Click the **Easy Setting** Button in the **Message Communication** Area of the connection parameter setting.



4. Set a communication setting in the **Message Communication Easy Setting** Dialog Box.



■ How to set up in the **Message Communication Easy Setting** Dialog Box

① When automatic receive function is used, select "1" for the **Connect No.**
② Set **Port No.** of CPU-03/CPU-04 side ("10001," for example).
③ Select **Extended MEMOBUS** for **Communication protocol Type**, and click the **Default** Button.
④ Select **Connect Type** (TCP, for example).
⑤ Select **Code** (BIN, for example).
⑥ Set **Node Port IP Address** of the other device (touch panel) to be connected ("192.168.001.002," for example).
⑦ Set **Port No.** of the other device (touch panel) to be connected ("10001," for example).
⑧ Click **OK**.

■ Caution

When message functions (MSG-SND, MSG-RCV) are used with the connection number 01, disable the automatic receive function. If message functions are used while the automatic receive function is enabled, the communications will not function properly.

Note: The automatic receive function with a connection number 01 is set to "Enable" by default.

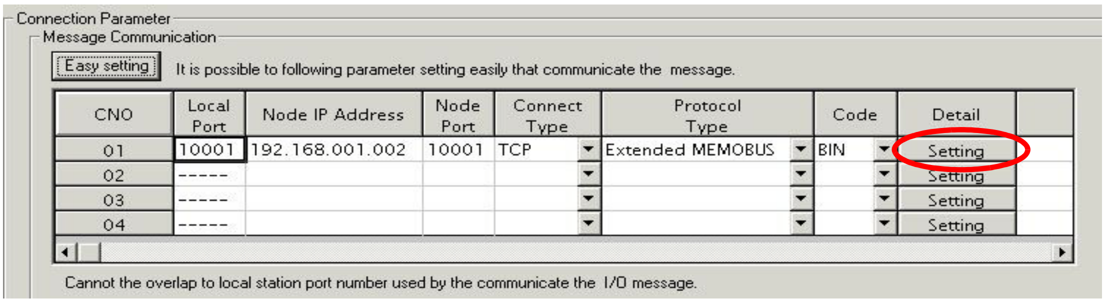
5. Click **Yes** in the parameter setting confirmation dialog box.

■ Caution

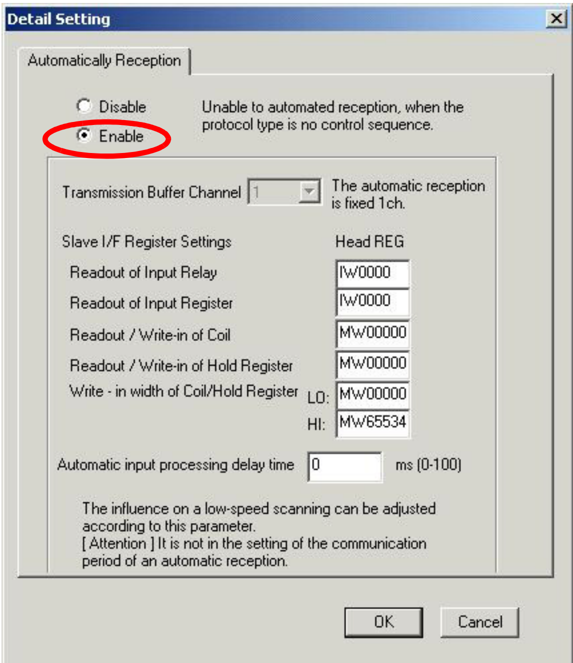
Note that when a parameter with the same connection number is already set and you click **Yes** in the parameter setting confirmation dialog, the setting will be overwritten by the parameter configured in the **Message Communication Easy Setting** Dialog Box.

---

6. Check the setting value and click the **Setting** Button in the **Detail** Column.



7. Select the **Enable** Option in the **Automatically Reception** Tab Page of the **Detail Setting** Dialog Box and then click the **OK** Button.



Note: For more information on Slave Side I/F Register Setting and Automatic Receive Process Delay Time, refer to *2.5.6 (4) [b] ■ Automatically Reception Dialog Box for Message Communication*.

The automatic receive function for connecting the CPU-03/CPU-04 to the touch panel is now set up.

■ Caution

When any transmission or connection parameter is changed, the change will be reflected after it has been saved in flash memory and the power supply is turned ON again.

---

(2) How to set up a touch panel

This section explains the GP-Pro EX side procedure to set up for connecting the CPU-03/CPU-04 to an indicator (GP3000 series) and the window creation example.

Note: The indicator (GP3000 series) and GP-Pro EX are manufactured by Digital Electronics Corp. Contact Digital Electronics Corp. for more information.

[a] How to Set up GP-Pro EX

1. Start up GP-Pro EX.

2. Create a new project.

3. Set its indicator type. Set the indicator type in accordance with the model in use.

Here, we explain the setting when AGP-3600T is used.

Table 6.16 Indicator Type Setting (example)

Series	GP3000 series
	AGP33** series
Model	AGP-3600T
Installation Method	Horizontal model

4. Set up connected equipment.

Table 6.17 Connected Equipment

Manufacturer	YASKAWA Electric Corporation
Series	MEMOBUS Ethernet

5. Set up the way to connect.

Table 6.18 Connection Method

Port	Ethernet (TCP)

6. Select the **Connected Equipment Setting** Window from the **System** Tab to display the connected equipment setting window.

7. Set the communication setting.

Table 6.19 Communication Setting

Port Number*	10001
Timeout	3(sec)
Retry	0
Transmit Weight	0(ms)

* For more information on the port number, refer to the following.

■ Port Number

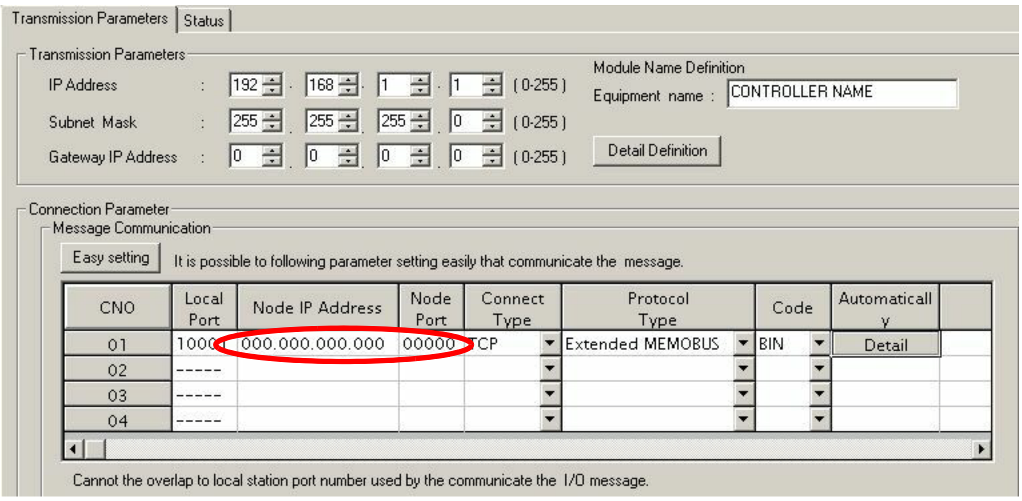
- If you don't select **Automatic Assignment** of the port number in the communication setting window, the automatic assignment will be disabled, and the GP3000 series port number will be fixed at the setting value.

- If you select **Automatic Assignment** of the port number in the communication setting window, automatic assignment will be enabled, and the GP3000 series port number will be assigned in each case.
  When you use Automatic Assignment, set *Unpassive open mode* in the 218IFA/218IFC detailed definition window of MPE720.
  For more information about *Unpassive open mode*, refer to *2.5.6 (4) [b] 218IFA Module Detailed Definition Window*.
  For information on the relationship between GP-Pro EX and MPE720 settings, see the table below.

MPE720 Side Setting / GP-Pro EX Side Setting	Unpassive open Mode	Fixed Value Setting
Automatic Assignment Enable	√	–
Automatic Assignment Disable	√	√

Note: √: connectable, – : unconnectable

- How to set up *Unpassive open mode* of the CPU-03/CPU-04 (reference)
  Set **Node IP Address** to 000.000.000.000 and the **Node Port** to 0 to enter into the *Unpassive open mode*.



8. Click the setup button of the connected PLC1 for each device setting to display the setting window for each device.

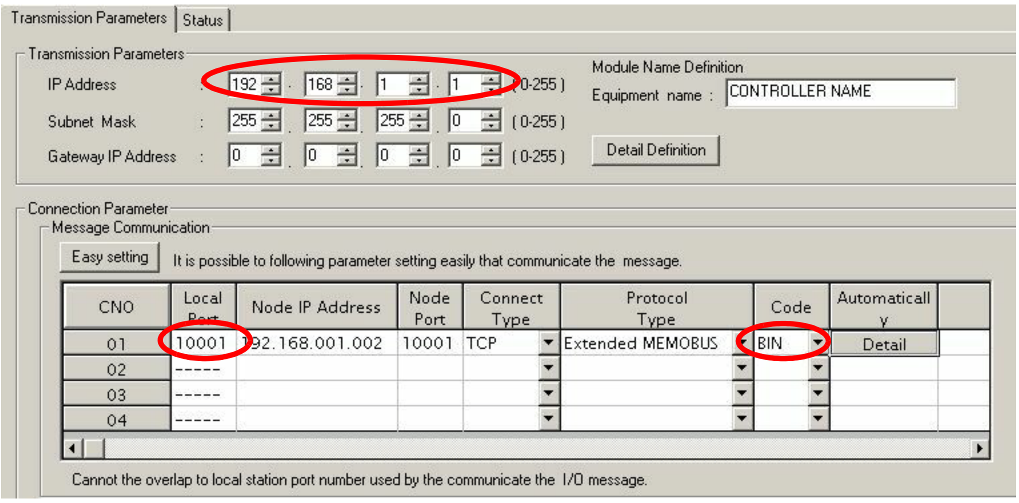
9. Set up the setting window for each device.

In the setting window for each device, set up a connected device (in this case, the CPU-03/CPU-04).
Set the IP address, port number, and data code in the same manner as in the 218IFA window for the CPU-03 (or the 218IFC window for the CPU-04) of the MP2200.

Table 6.20 Each Device Setting

IP Address	192.168.001.001
Port Number	10001
Data Code	binary

- 218IFA/218IFC Window (reference)



The setting is finished for now.
Create a window and transfer the project to an indicator as required.

■ Caution

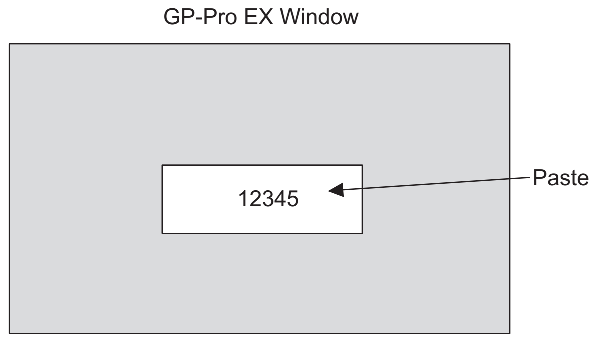
- Set up a unique IP address in the network.
  The CPU-03/CPU-04 side IP address is set to “192.168.1.1” in self-configuration.
  For the IP address, check with your network administrator.

Note: Set the GP3000 series IP address in the off-line mode of the indicator.
Contact Digital Electronics Corp. for more information.

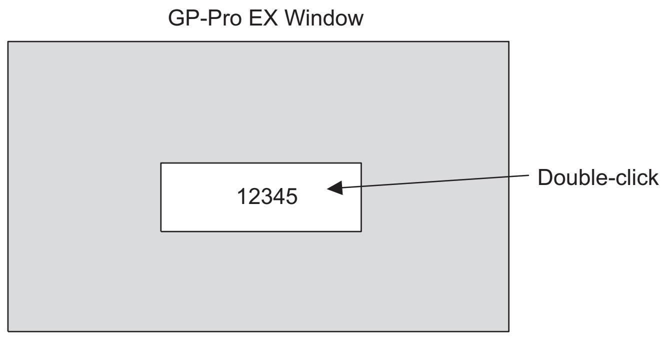
[b] Window Creation Example

1. Create a base window.

2. Select **Data Indicator** from the toolbar to paste it on the window.



3. Double-click the **Data Indicator** pasted on the window.



4. Set as follows in the detailed setting window of **Data Indicator** and click OK.

Table 6.21 Data Indicator Detailed Setting

Display Data	Numeric display
Monitor Word Address	GMW00100

■ Relationship between GP-Pro EX address display and CPU-03/CPU-04 register

Device	GP-Pro EX Address Display	CPU-03/CPU-04 Register
Coil (bit)	GMB□□□□□	MB□□□□□
Coil (word)	GMW□□□□□	MW□□□□□
Input Relay (bit)	GIB□□□□□	IB□□□□□
Input Relay (word)	GIW□□□□	IW□□□□

(3) How to Start Communication

1. The CPU-03/CPU-04 side starts to receive the messages.

When the automatic receive function is used, the message receive operation starts automatically, and you are not required to do anything.

2. Start up the touch panel (GP3000 series) to display the main window.

After the system startup of the touch panel, communication with the CPU-03/CPU-04 will start.

Note: Contact Digital Electronics Corp. for more information.

6.4 Communication with PLC Manufactured by Mitsubishi Electric Corporation (MELSEC protocol)

In Ethernet communication between the CPU-03/CPU-04 and MELSEC (Q, A series) general-purpose PLC manufactured by Mitsubishi Electric Corporation, the MELSEC protocol (A-compatible 1E frame) is used as a communication protocol.
Using the MELSEC protocol allows a master to read/write the slave register content.
This section explains communication when the CPU-03/CPU-04 acts as a slave and a master respectively.
For using the CPU-03/CPU-04 as a slave, we explain communication using the automatic receive function.
For using the CPU-03/CPU-04 as a master, we explain communication using the I/O message communication function.

6.4.1 When the CPU-03/CPU-04 Acts as Slave (automatic receive function is used)

This section explains how to carry out a fixed buffer communication with the BUFSND command (with procedure) of the MELSEC Q series using the CPU-03/CPU-04 automatic receive function.

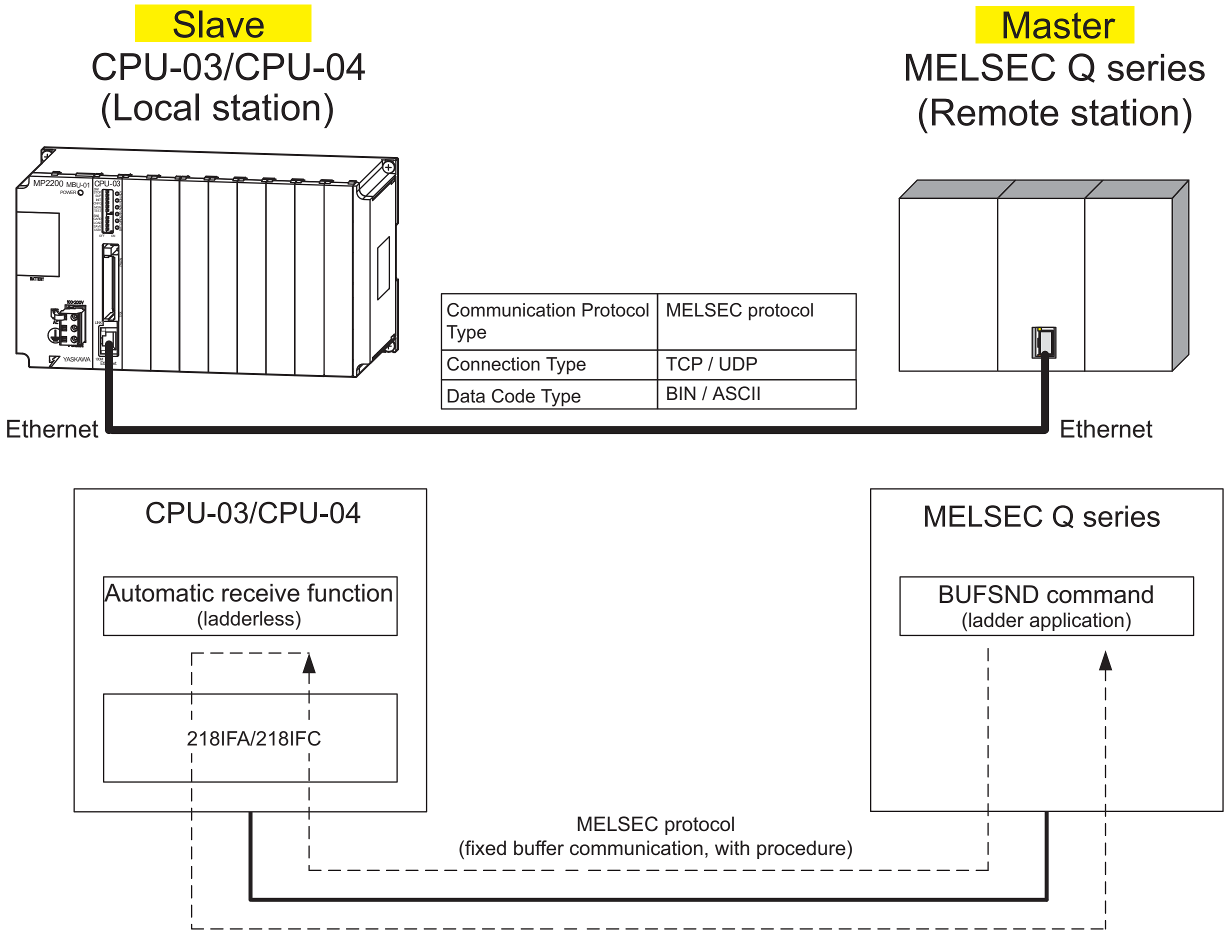


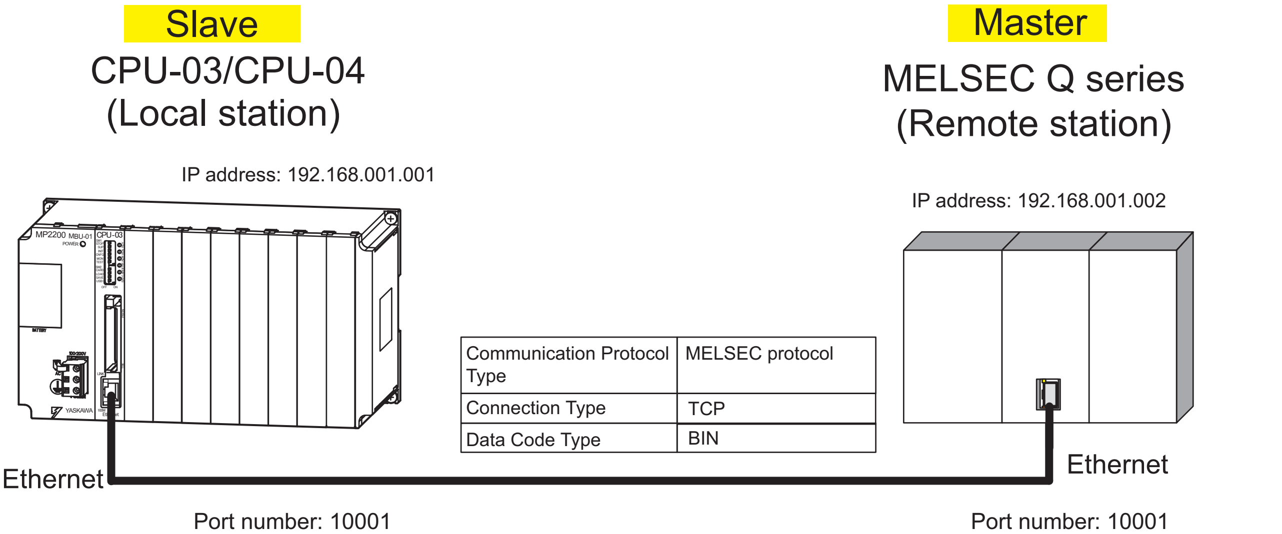
Fig. 6.12 Message Flow with the MELSEC Q Series when the Automatic Receive Function Is Used

- **Caution**

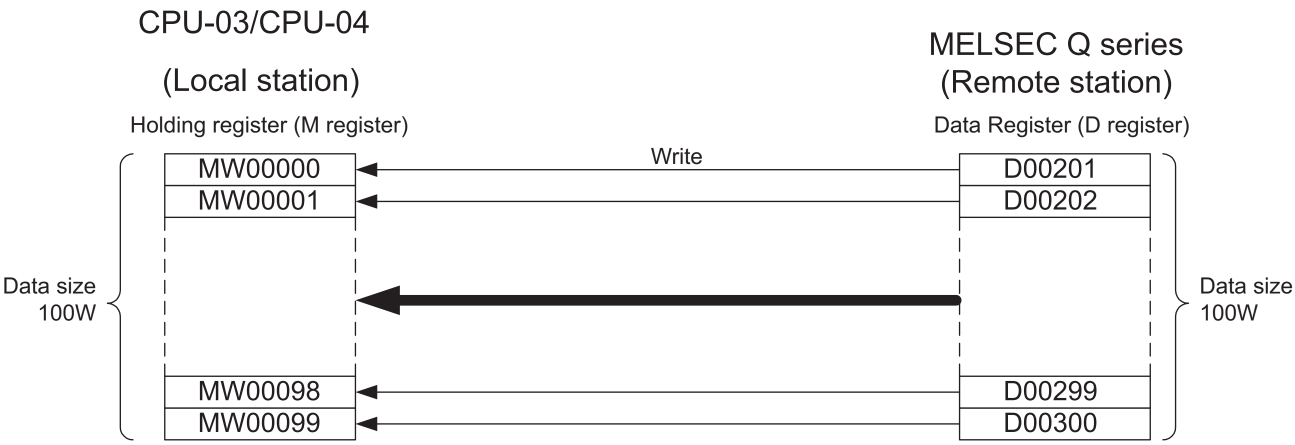
Communication using the automatic receive function is 1:1 communication.
Also, when "Communication Protocol Type: MELSEC" is used in communication with the MELSEC Q series, the MELSEC Q series (master) side can read/write the holding register of the CPU-03/CPU-04 (slave) using fixed buffer communication.
However, when the CPU-03/CPU-04 acts as a slave, you cannot use the inter-CPU or random access communication, because of the MELSEC specifications.
In addition, use the message receive function (MSG-RCV) when communicating with multiple remote devices.

■ Setting Example

The following figure illustrates one example of writing the contents of the data register (D00201 to D00300) of MELSEC Q series (master) into the CPU-03/CPU-04 (slave) holding register (MW00000 to MW00099).



Communication Protocol Type	MELSEC protocol
Connection Type	TCP
Data Code Type	BIN

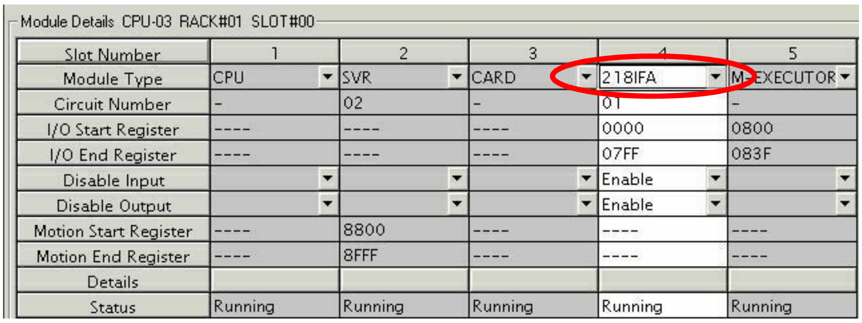


The setup procedure is explained in the following pages.

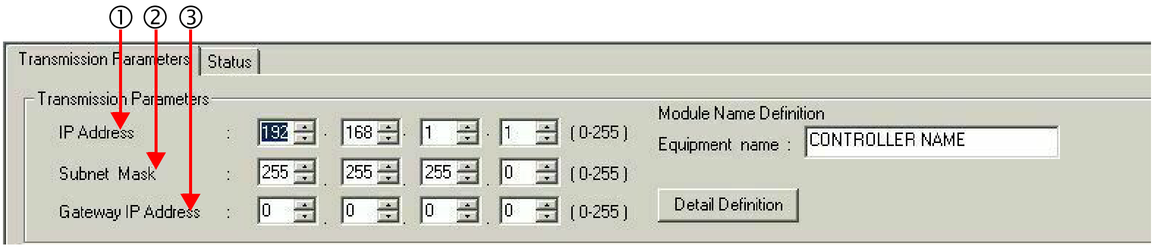
(1) How to Set up the CPU-03/CPU-04 Side

If the setting of transmission parameters (IP address, subnet mask) is already completed, start from step 3.

1. Double-click the **218IFA** Tab (**218IFC** Tab for the CPU-04) in the **Module Details** Area in the CPU-03 **Module Configuration** Window.



2. Set transmission parameters.



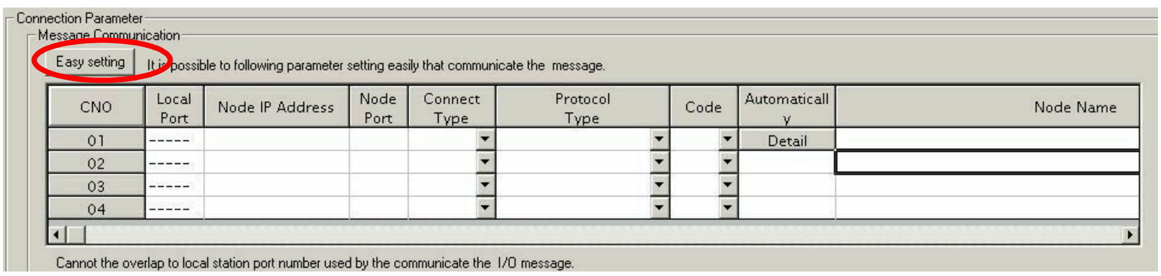
■ How to set up transmission parameters

① Set **IP Address** ("192.168.001.001," for example).
② Set **Subnet Mask** ("255.255.255.000," for example).
③ Set **Gateway IP Address** ("000.000.000.000," for example).

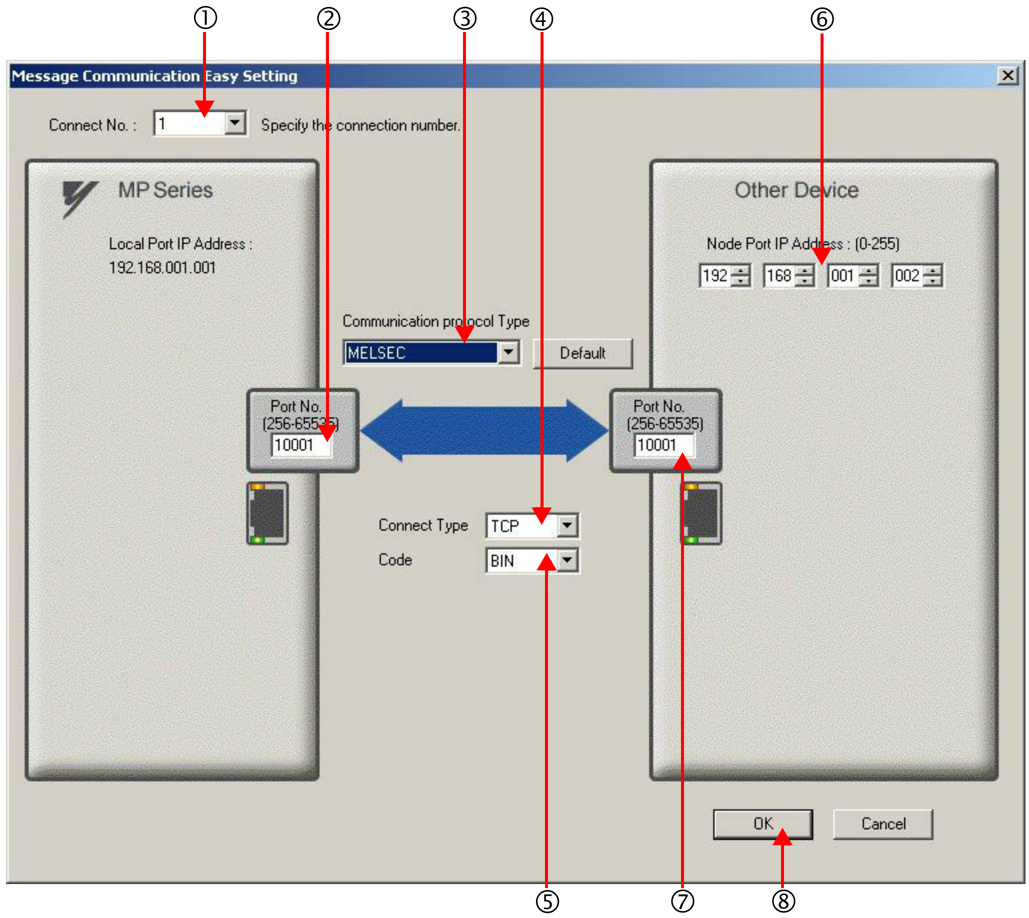
■ Caution

Set up a unique IP address in the network.
For the IP address, check with your network administrator.

3. Click the **Easy Setting** Button in the **Message Communication** Area of the connection parameter setting.



4. Set up the communication settings in the **Message Communication Easy Setting** Dialog Box.



■ How to set up in the **Message Communication Easy Setting** Dialog Box

① When automatic receive function is used, select "1" for the **Connect No.**
② Set **Port No.** of the CPU-03/CPU-04 side ("10001," for example).
③ Select **MELSEC** for **Communication protocol Type**, and click the **Default** Button.
④ Select **Connect Type** (TCP, for example).
⑤ Select **Code** (BIN, for example).
⑥ Set **Node port IP Address** of the other device (MELSEC Q series) to be connected ("192.168.001.002," for example).
⑦ Set **Port No.** of the other device (MELSEC Q series) to be connected ("10001," for example).
⑧ Click **OK**.

■ Caution

When message functions (MSG-SND, MSG-RCV) are used with the connection number 01, disable the automatic receive function. If message functions are used while the automatic receive function is enabled, the communications will not function properly.

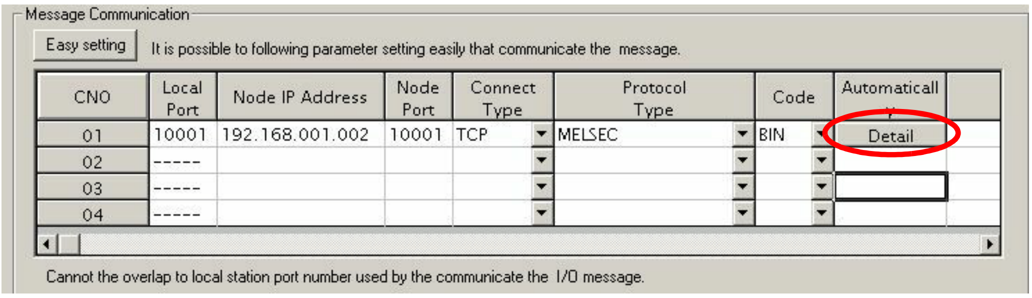
Note: The automatic receive function with a connection number 01 is set to "Enable" by default.

5. Click **Yes** in the parameter setting confirmation dialog box.

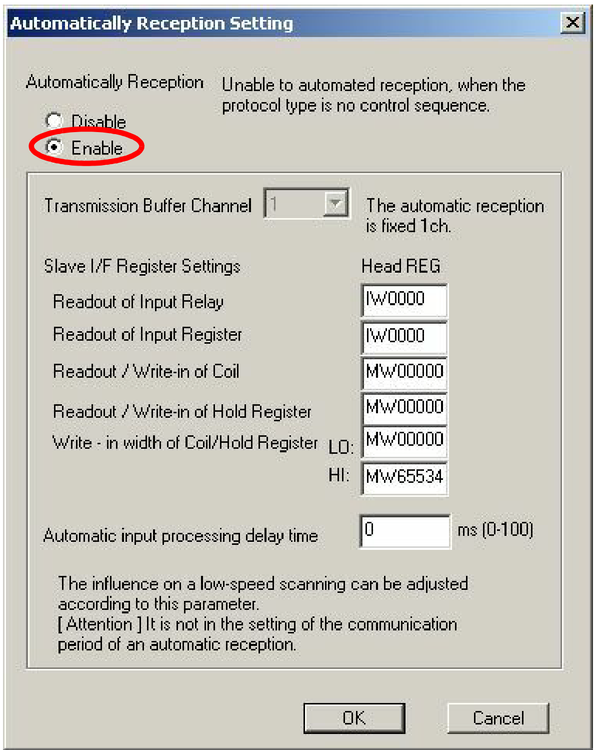
■ Caution

Note that when a parameter with the same connection number is already set and you click **Yes** in the parameter setting confirmation dialog, the setting will be overwritten by the parameter configured in the **Message Communication Easy Setting** Dialog Box.

6. Check the setting value and click the **Detail** Button in the **Automatically** Column.



7. Select the **Enable** Option for **Automatically Reception** in the **Automatically Reception Setting** Dialog Box and then click the **OK** Button.



Note: For more information on Slave Side I/F Register Setting and Automatic Receive Process Delay Time, refer to *2.5.6 (4) [b] ■ Automatically Reception Dialog Box for Message Communication*.

The automatic receive function is now set up, when the CPU-03/CPU-04 acts as a slave.

■ Caution

When any transmission or connection parameter is changed, the change will be reflected after it has been saved in flash memory and the power supply is turned ON again.

(2) How to Set up the Remote Device (MELSEC Q series) to Be Connected

This section explains the MELSEC Q series side procedure to set up for connecting the CPU-03/CPU-04 with the MELSEC Q series.

Note: MELSEC Q series are manufactured by Mitsubishi Electric Corporation.
Contact Mitsubishi Electric Corporation for more information.

1. Start up GX Developer.

2. Create a new project.

3. Set up network parameters (MELSECNET/Ethernet).

Table 6.22 Network Parameter Setting (example)

Setting Item	Setting Details
Network Type	Ethernet
Start I/O No.	Any
Network No.	Any
Group No.	Any
Exchange Number	Any
Mode	Online

4. Set up Ethernet operation.

Table 6.23 Ethernet Operation Setting (example)

Setting Item	Setting Details
Communication Data Code Setting	Binary code communication
Initial Timing Setting	Any
IP Address	192.168.1.2
Transmit Frame Setting	Ethernet (V2.0)
TCP Alive Check Setting	Any
Permit Writing during RUN	Permitted

5. Set the open setting.

Table 6.24 Open Setting (example)

Setting Item	Setting Details (connection number=1)
Protocol	TCP
Open System	Active
Fixed Buffer	Transmit
Procedure to Communicate with Fixed Buffer	With procedure
Pairing Open	Any
Check Alive	Any
Local Port Number	2711H (10001)
Remote IP Address for Communication	192.168.1.1
Remote Port Number for Update	2711H (10001)

■ Caution

Set up a unique IP address in the network.
For the IP address, check with your network administrator.

■ Complement

Set up an initial setting and a router relay parameter below, if needed:
- Initial setting
  Set a timer relevant configuration when TCP is selected as a protocol. In most cases, accept the default.
  Set up if changes such as a shortened a TCP retransmit timer are required.
- Router relay parameter
  Set up when you use a subnet mask pattern or default gateway.

6. Create a ladder program for communication.

■ Procedure overview to communicate using a ladder program

① Use an OPEN command to establish a connection with the remote device.
② Use a BUFSND command to write the register content configured by parameters below to the CPU-03/CPU-04 holding register (M register).
Setting example: When the BUFSND command is used to set the device start number for storing the transmit data to “D00200”
D00200 (transmit data length):100W
D00201 to D00300 (transmit data): Written into MW00000 to MW00099
③ If necessary, use a CLOSE command to close the operation.

Note: Contact Mitsubishi Electric Corporation for more information on the ladder program.

The setting is finished for now. If necessary, transfer the settings to the PLC after setting all parameters.

(3) How to Start Communication

1. The CPU-03/CPU-04 side starts to receive the messages.

When an automatic receive function is used, the message receive operation starts automatically, so you are not required to do anything.

2. Send an OPEN command from the MELSEC Q series side to establish a connection with the CPU-03/CPU-04, and use a BUFSND command to transmit messages.

When messages are transmitted from the MELSEC Q series, communication with the CPU-03/CPU-04 will start.

6.4.2 When the CPU-03/CPU-04 Acts as Master (I/O message communication function is used)

This section explains how to carry out the communications between CPU and the MELSEC Q series using the CPU-03/CPU-04 I/O message communication function.

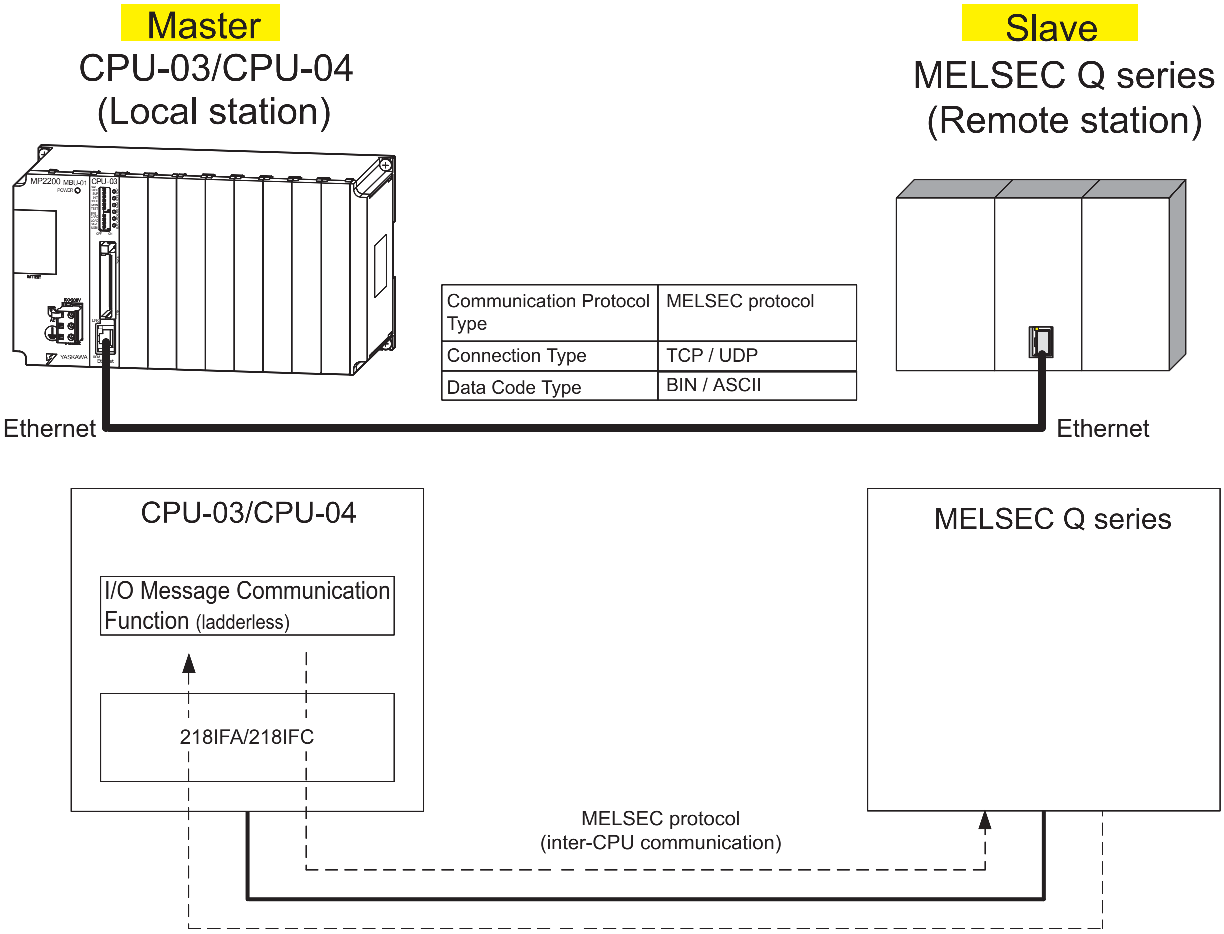


Fig. 6.13 Message Flow with MELSEC Q series when I/O Message Communication Function Is Used

■ Caution

I/O message communication is 1:1 communication.
In addition, you can read and write the registers below using inter-CPU communication when “Communication Protocol Type: MELSEC” is used in the communication with the MELSEC series.

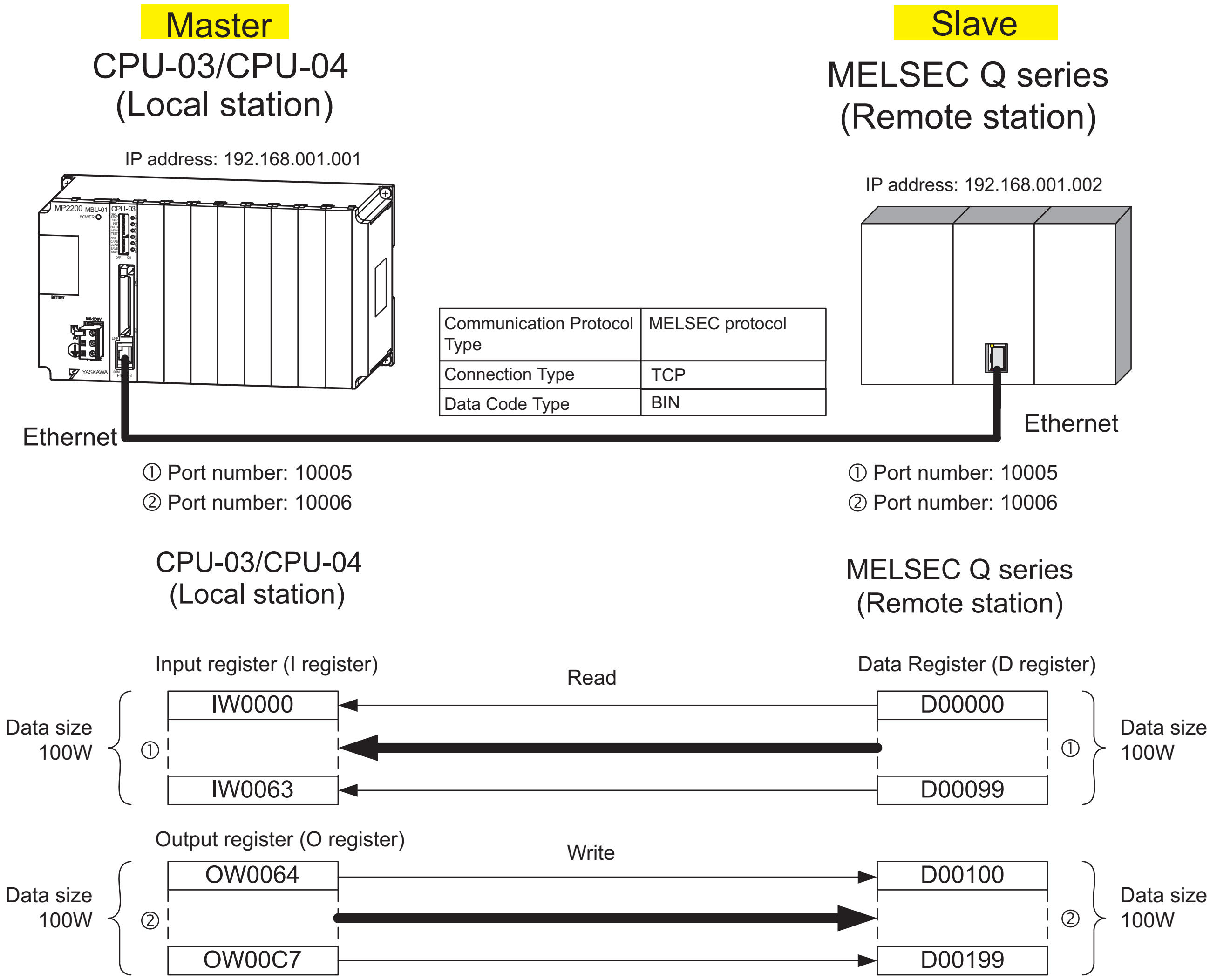
- Bit device register - - - - X, Y (read only), M, B
- Word device register - - - D, W, R

A bit device register reads or writes on a per-word (16 bit) basis.

When communicating with multiple remote devices.
To carry out a fixed buffer/random access buffer communication when reading/writing registers other than those mentioned above, use the message send function (MSG-SND).

■ Setting Example

The following figure illustrates one example of reading the content of the data register (D00000 to D00099) of the MELSEC Q series (slave) into an input register (IW0000 to IW0063) of the CPU-03/CPU-04 (master) and writing the content of an output register (OW0064 to OW00C7) of the CPU-03/CPU-04 (master) in a data register (D00100 to D00199) of the MELSEC Q series (slave).

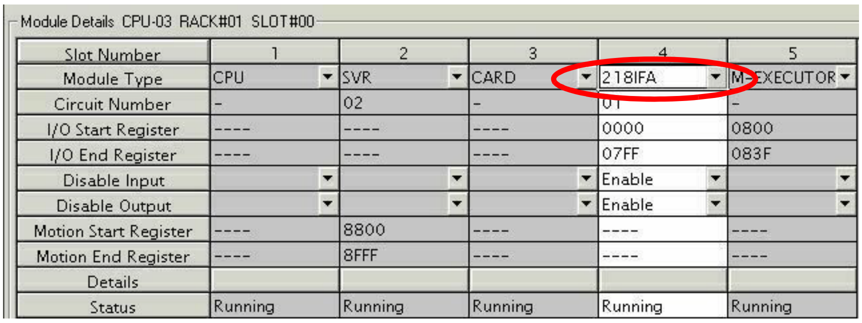


The setup procedure is explained in the following pages.

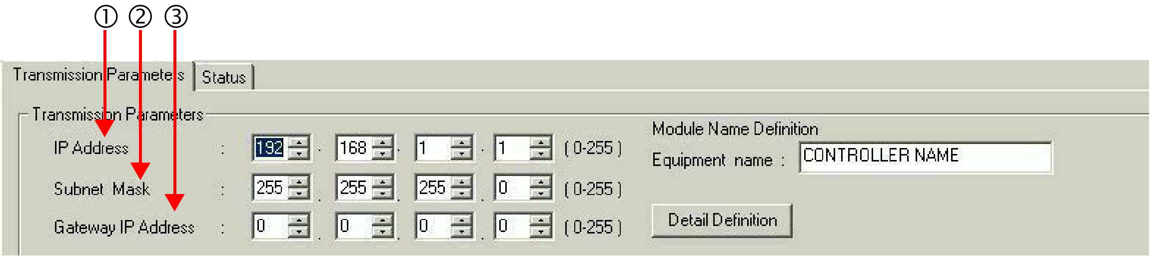
(1) How to Set up the CPU-03/CPU-04 Side

If the setting of transmission parameters (IP address, subnet mask) is already completed, start from step 3.

1. Double-click the **218IFA** Tab (**218IFC** Tab for the CPU-04) in the **Module Details** Area in the CPU-03 **Module Configuration** Window.



2. Set transmission parameters.



■ How to set up transmission parameters

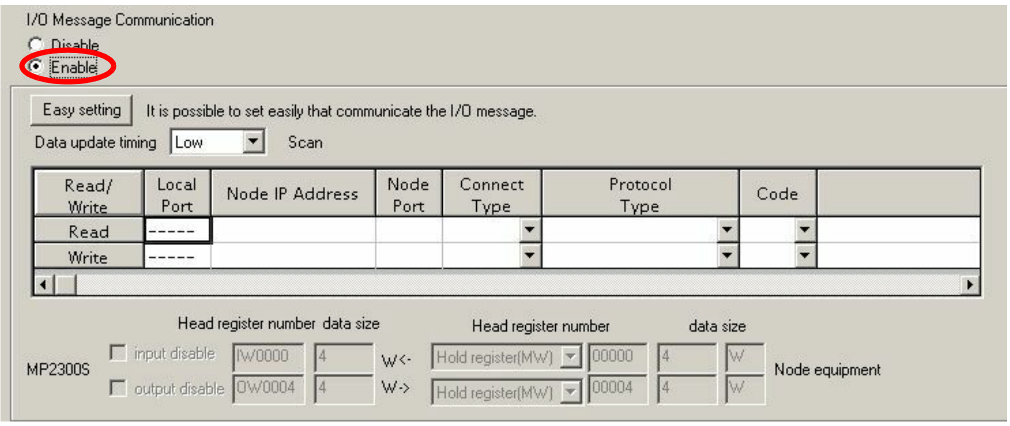
① Set **IP Address** (“192.168.001.001,” for example).
② Set **Subnet Mask** (“255.255.255.000,” for example).
③ Set **Gateway IP Address** (“000.000.000.000,” for example).

■ Caution

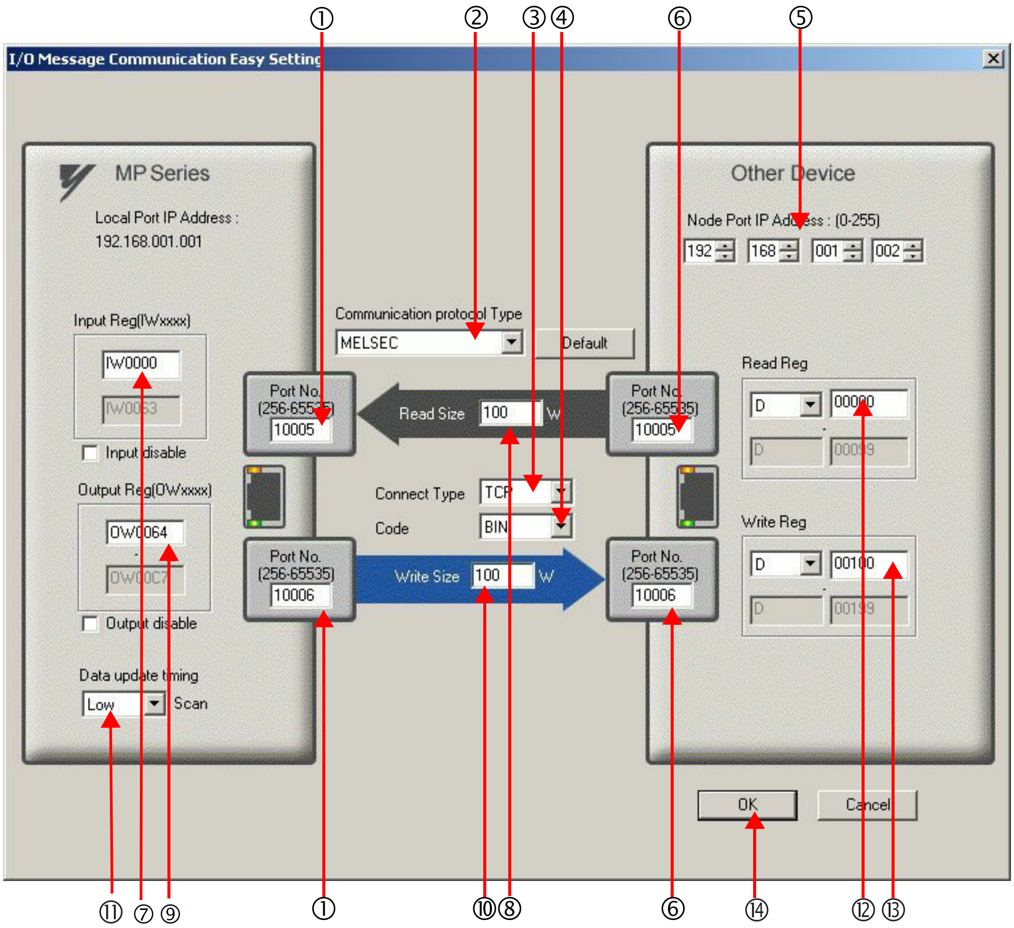
Set up a unique IP address in the network.
For the IP address, check with your network administrator.

---

3. Select the **Enable** Option in the **I/O Message Communication** of the connection parameter setting.



4. Set-up a communication settings in the **I/O Message Communication Easy Setting** Dialog Box.



■ How to set up in the **I/O Message Communication Easy Setting** Dialog Box

① Set **Port No.** of the CPU-03/CPU-04 side (“10005, 10006,” for example).
② Select **MELSEC** for **Communication protocol Type**, and click the **Default** Button.

■ Caution

When the communication protocol is MELSEC, the default register type for the read/write is “Word Device Register: D.”

---

③ Select **Connect Type** (TCP, for example).
④ Select **Code** (BIN, for example).
⑤ Set **Node Port IP Address** of the other device (MELSEC Q series) to be connected (“192.168.001.002,” for example).
⑥ Set **Port No.** of the other device (MELSEC Q series) to be connected (“10005, 10006,” for example).

■ Caution

In I/O message communications, since a message is transmitted from each port number for register read/write, a connected remote device needs the two receive connections for receiving messages.

---

⑦ Set a storage area (**Input Reg**) for data read by the CPU-03/CPU-04 (IW0000, for example).
⑧ Set **Read Size** of data to be read by the CPU-03/CPU-04 (“100” W, for example).
⑨ Set a storage area (**Output Reg**) for data written by CPU-03/CPU-04 (OW0064, for example).
⑩ Set **Write Size** of data written by the CPU-03/CPU-04 (“100” W, for example).
⑪ Set an I/O data update timing (**Data update timing**) for the CPU and built-in Ethernet (“Low” scan, for example).

■ Data Update Timing

Data update timing indicates when to give and receive data between the CPU and built-in Ethernet. Communication with the remote device is carried out asynchronously, so note that a message is not necessarily transmitted to the remote device at each data update timing.

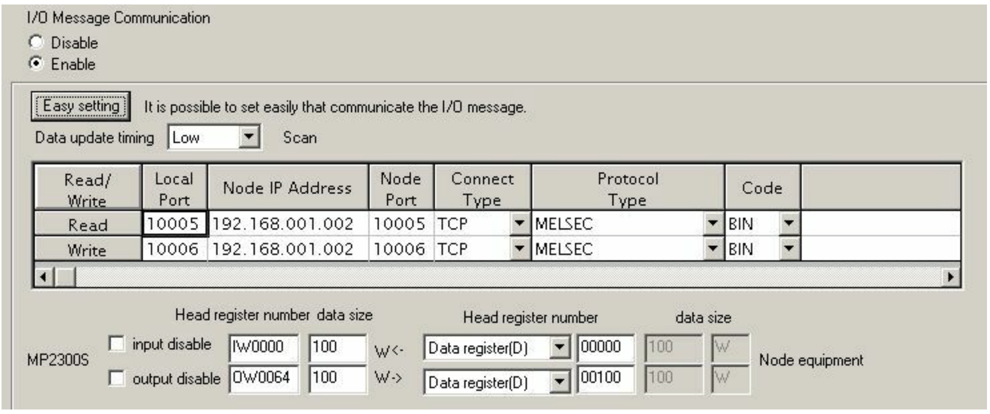
⑫ Set the register type and start address (**Read Reg**) of the remote device (MELSEC Q series) read by the CPU-03/CPU-04 (“D00000,” for example).
⑬ Set the register type and start address (**Write Reg**) of the remote device (MELSEC Q series) written by the CPU-03/CPU-04 (“D00100,” for example).
⑭ Click **OK**.

5. Click **Yes** in the parameter setting confirmation dialog box.

■ Caution

Note that when a parameter with the same connection number is already set and you click **Yes** in the parameter setting confirmation dialog, the setting will be overwritten by the parameter configured in the **Message Communication Easy Setting** Window.

6. Check the setting values.



The I/O message communication is now set up, when the CPU-03/CPU-04 acts as a master.

■ Caution

When any transmission or connection parameter is changed, the change will be reflected after FLASH has been saved and the power supply is turned ON again.

(2) How to Set up the Remote Device (MELSEC Q series) to Be Connected

This section explains the MELSEC Q series side procedure to set up for connecting the CPU-03/CPU-04 with the MELSEC Q series.

Note: MELSEC Q series are products manufactured by Mitsubishi Electric Corporation.
Contact Mitsubishi Electric Corporation for more information.

1. Start up GX Developer.

2. Create a new project.

3. Set up network parameters (MELSECNET/Ethernet).

Table 6.25 Network Parameter Setting (example)

Setting Item	Setting Details
Network Type	Ethernet
Start I/O No.	Any
Network No.	Any
Group No.	Any
Exchange Number	Any
Mode	Online

4. Set up Ethernet operation.

Table 6.26 Ethernet Operation Setting (example)

Setting Item	Setting Details
Communication Data Code Setting	Binary mode communication
Initial Timing Setting	Always waiting OPEN
IP Address	192.168.1.2
Transmit Frame Setting	Ethernet (V2.0)
TCP Alive Check Setting	Any
Permit Writing during RUN	Permitted

5. Set the open setting.

Table 6.27 Open Setting (example)

Setting Item	Setting Details (connection number=1)	Setting Details (connection number=2)
Protocol	TCP	TCP
Open System	Fullpassive	Fullpassive
Fixed Buffer	Any	Any
Procedure to Communicate with Fixed Buffer	Any	Any
Pairing Open	Any	Any
Check Alive	Any	Any
Local Port Number	2715H (10005)	2716H (10006)
Remote IP Address for Communication	192.168.1.1	192.168.1.1
Remote Port Number for Update	2715H (10005)	2716H (10006)

■ Caution

Set up a unique IP address in the network.
For the IP address, check with your network administrator.

The setting is finished for now. If necessary, transfer the settings to the PLC after setting all parameters.

■ Complement

Set up an initial setting and a router relay parameter below, if needed:

- Initial setting
  Set a timer relevant configuration when TCP is selected as a protocol. In most cases, accept the default.
  Set up if changes such as a shortened TCP retransmit timer are required.
- Router relay parameter
  Set up when you use a subnet mask pattern or default gateway.

(3) How to Start Communication

1. The MELSEC Q series starts to receive messages.

The message receive operation starts automatically, so you are not required to do anything.

2. The CPU-03/CPU-04 side transmits messages.

When an I/O message communication function is used, the message transmit operation starts automatically, so you are not required to do anything.

7

Easy Programming (Supported by the CPU-03 and CPU-04)

This chapter explains easy programming using the M-EXECUTOR Module (motion program executor) functions.

7.1 System Startup Overview

The start-up procedure for a model system is as follows. For detailed information of each step, refer to the cited references.

This chapter explains a procedure where you can easily run and check a program without external signals.

The motion program created here is simple and has only three lines to move two axes 150,000 pulses from the current position and stop them.

```
INC;        Specify an incremental mode
MOV         [A1]150000, [B1]150000;Position two-axes 150,000 pulses
END;
```

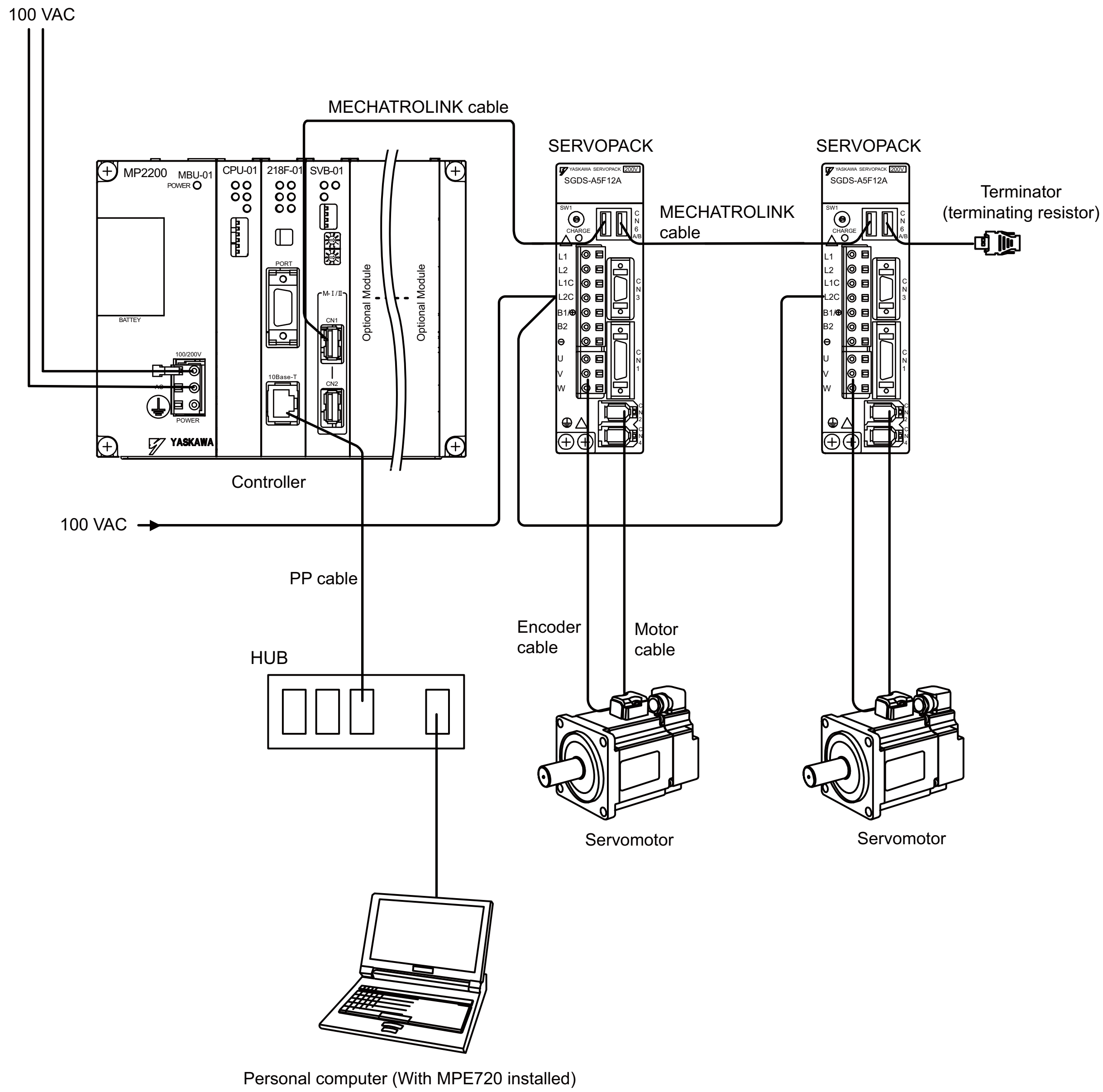
7.2 Preparation (step 1)

This section explains the steps of "wiring," "self-configuration," and "test operation" for starting up the model system.

7.2.1 Wiring

We use the following layout model to explain the startup of the model system. Prepare each device listed on the next page and connect them as shown in the figure below.

(1) System Layout Model



[a] Required Equipment

Product Name		Model	Q'ty
MP2200	Base Unit with AC power supply	JEPMC-BU2200	1
	CPU-03 Module (or CPU-04 Module)	JAPMC-CP2220-E (or JAPMC-CP2230-E)	1
	SVB-01 Module	JAPMC-MC2310	1
MECHATROLINK cable (0.5m)		JEPMC-W6002-A5	2
Terminator (terminating resistor)		JEPMC-W6022	2
Σ-III SERVOPACK		SGDS-A5F12A	2
Σ-III servomotor		SGMAS-A5A2A21	2
Motor cable (3m)		JZSP-CSM01-03	2
Encoder cable (3m)		JZSP-CSP05-03	2
HUB (commercialized product)		LSW-TX-8EP	1
MPE720 Ver.6		CPMC-MPE770	1
LAN cable (for Ethernet connection)		Commercialized straight cable	2
Personal computer		Commercialized product	1

■ Caution

- Install MPE720 Ver.6 in the personal computer before starting step 1.
  For information on its installation, refer to *Engineering Tool for MP2000 Series MPE720 Version 6 User's Manual* (manual no.: SIEP C880700 30).
- Set the PC Ethernet port in advance. For the procedure to set up communication, refer to the *Engineering Tool for MP2000 Series MPE720 Version 6 User's Manual* (manual no.: SIEP C880700 30).
- The SERVOPACK station number (SW1) is set to 1 and 2.
- In a 1:1 connection without HUB, use a cross cable as a LAN cable.

7.2.2 Self Configuration

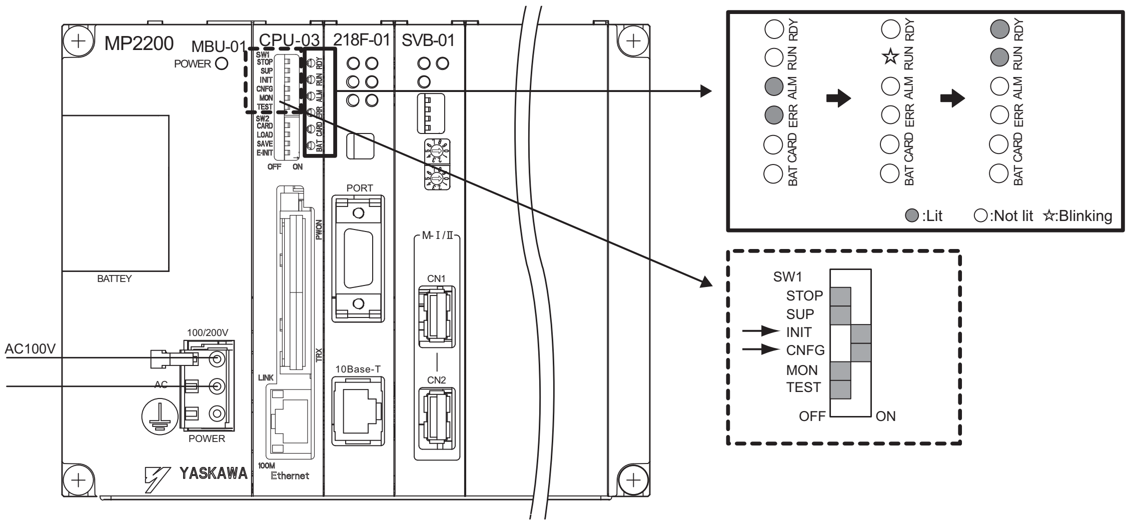
Run the self configuration to automatically recognize devices connected to the MECHATROLINK connector.
Steps for self configuration are as follows.

1. Check that the Σ-III SERVOPACK power supply is ON.

2. Turn OFF the 100-VAC power supply to the MP2200.

3. Set the INIT and CNFG pins on the DIP switch (SW1) to ON on the MP2200.

4. Turn ON the 100-VAC power supply to the MP2200, and confirm that the indicators change as follows:



5. Self configuration is complete, and MECHATROLINK slave device information has been written to a definition information file.

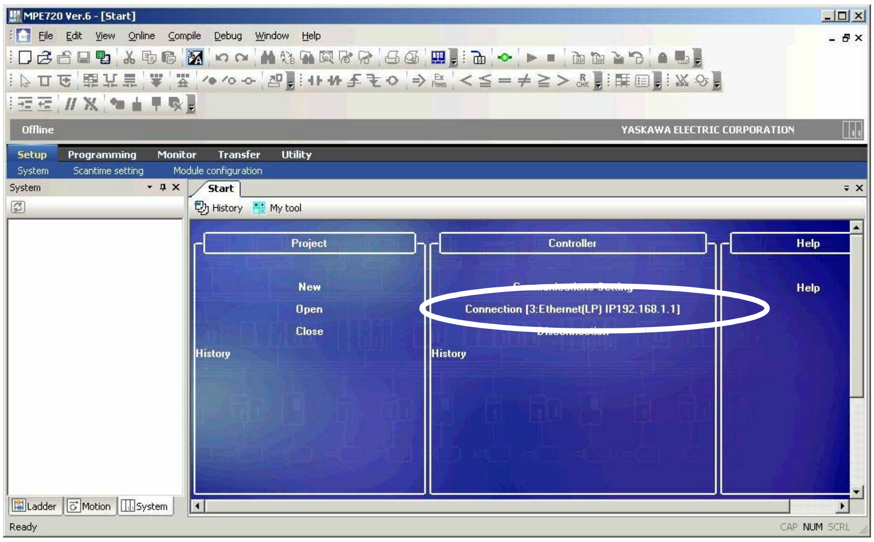
6. Set the INIT and CNFG pins on the DIP switch (SW1) to OFF on the MP2200.

7.2.3 Test Operation

Confirm that the machine controller can command axis servo ON/OFF and jog operation.

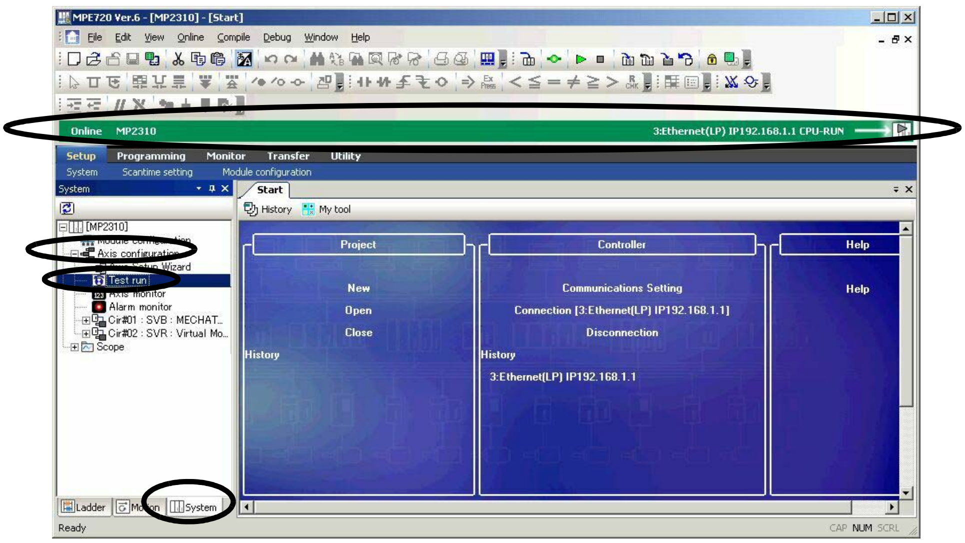
(1) Starting and Connecting MPE720 Ver.6

Launch MPE720Ver.6 and click “3:Ethernet(LP)192.168.1.1” to connect to the controller.

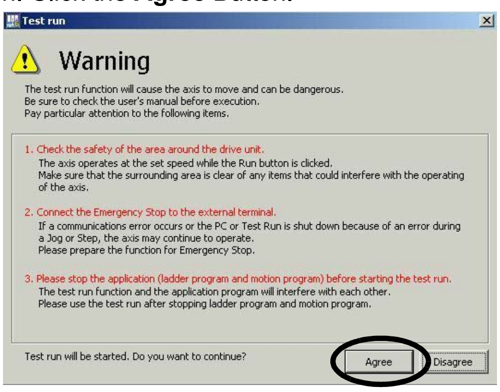


(2) Operating Manually in the Test Operation Window

When the connection is complete, the display will change from offline to online.

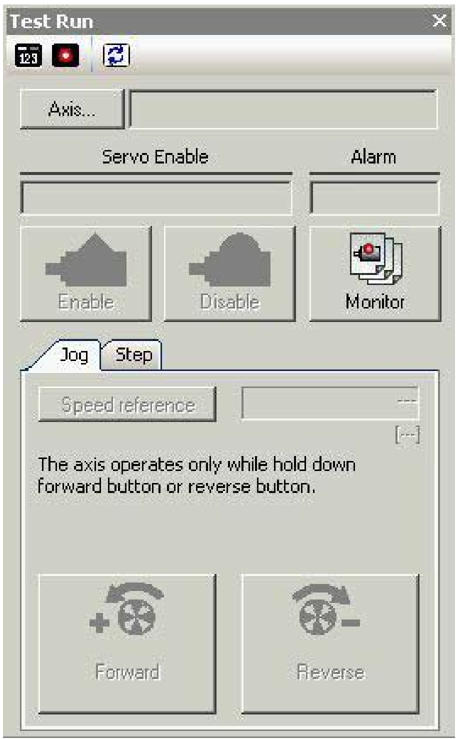


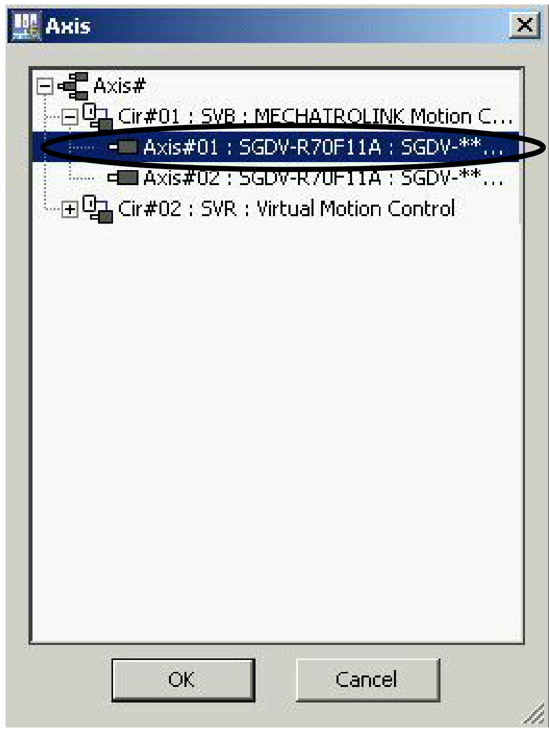
1. Click **System** in the subwindow and double-click ***Axis configuration - Test run*** to display a warning dialog a box for the test run. Click the **Agree** Button.

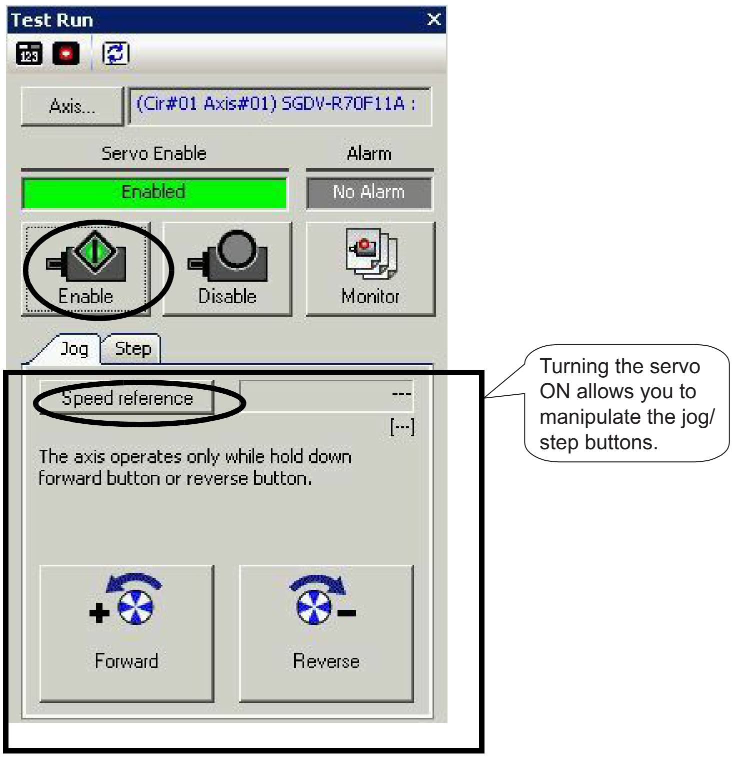


2. Axis Selection and Servo ON

Set an axis number in the **Axis** Window and click the **Enable** (Servo ON) Button in the **Test Run** Window.

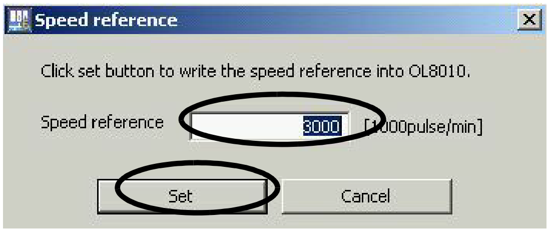


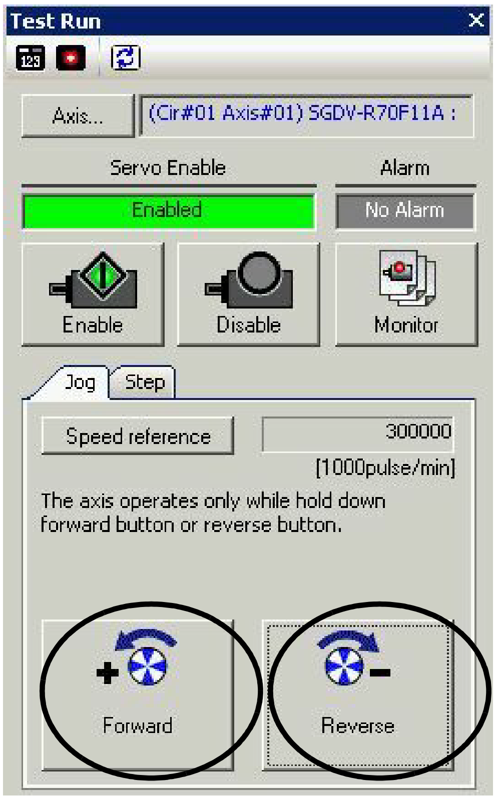




3. Jog Operation

Click the **Speed reference** Icon and set a speed reference value, and check that the axis rotates normally while the **Forward** Button or **Reverse** Button is pressed.





The operation check of the first axis is complete.

Press the **Axis ...** Button to change to “Axis #02” in the **Axis** Window, and perform the steps 1 to 3 above.

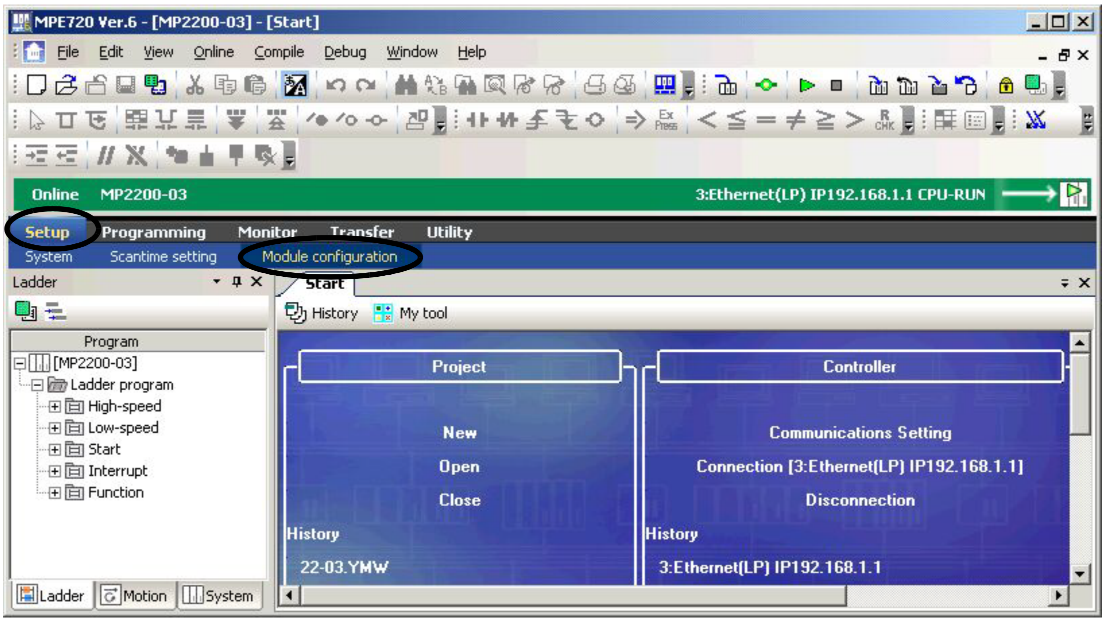
7.3 Programming (step 2)

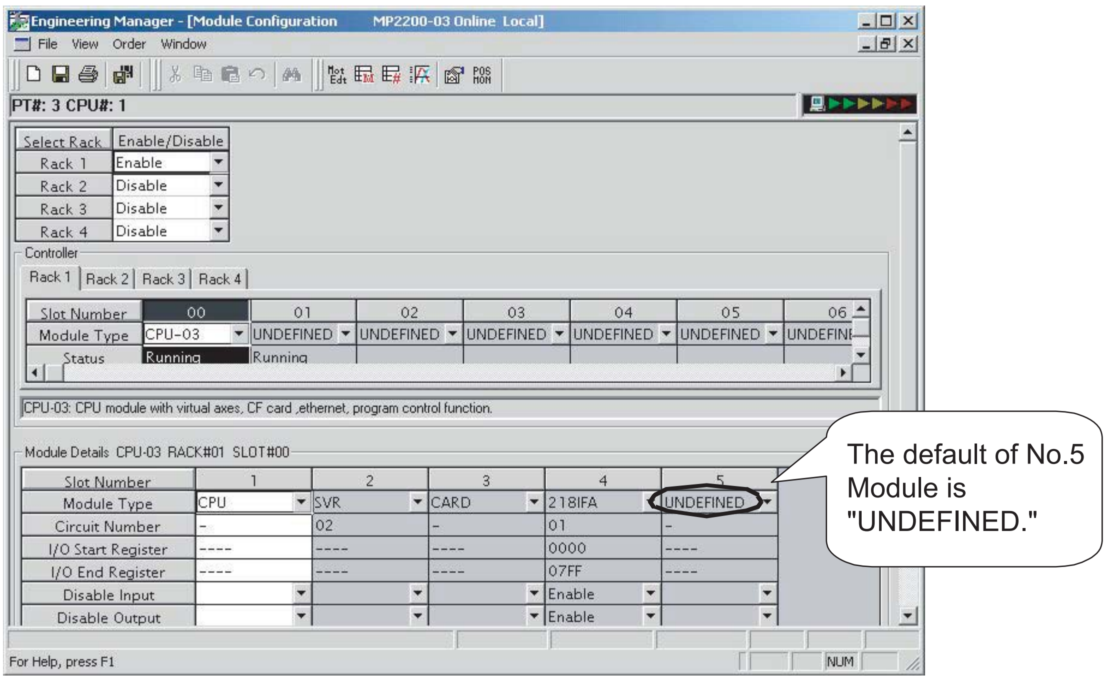
This section describes the procedure of initializing the M-EXECUTOR Module and the procedure from creating to saving a motion program.

7.3.1 Initializing the M-EXECUTOR Module

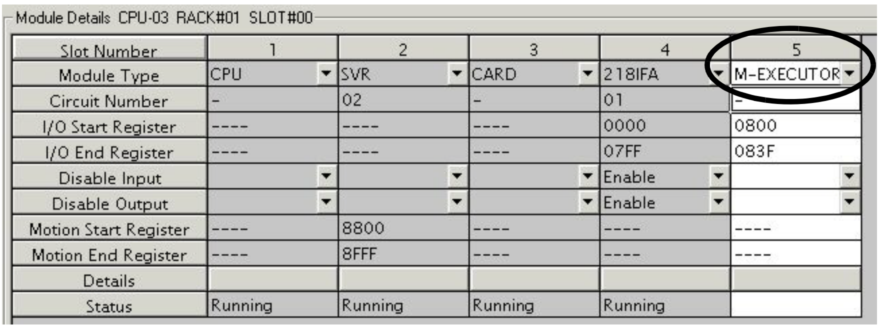
The M-EXECUTOR Module is disabled **(UNDEFINED)** with the default settings. Initialize the M-EXECUTOR Module to enable it by performing the following procedure.

1. Double-click **Module configuration** Tab from the **Setup** Menu. The **Module Configuration** Window will appear.

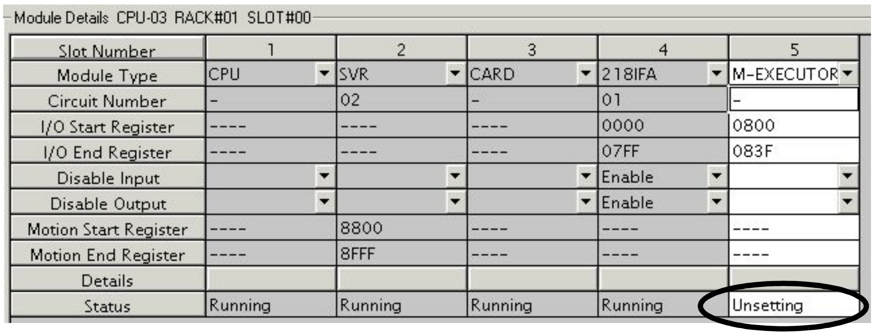




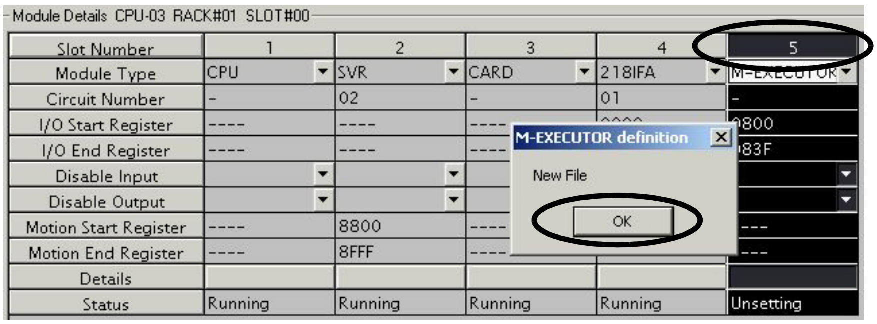
2. Allocate **M-EXECUTOR** to No.5 cell in the **Module Details** Area.



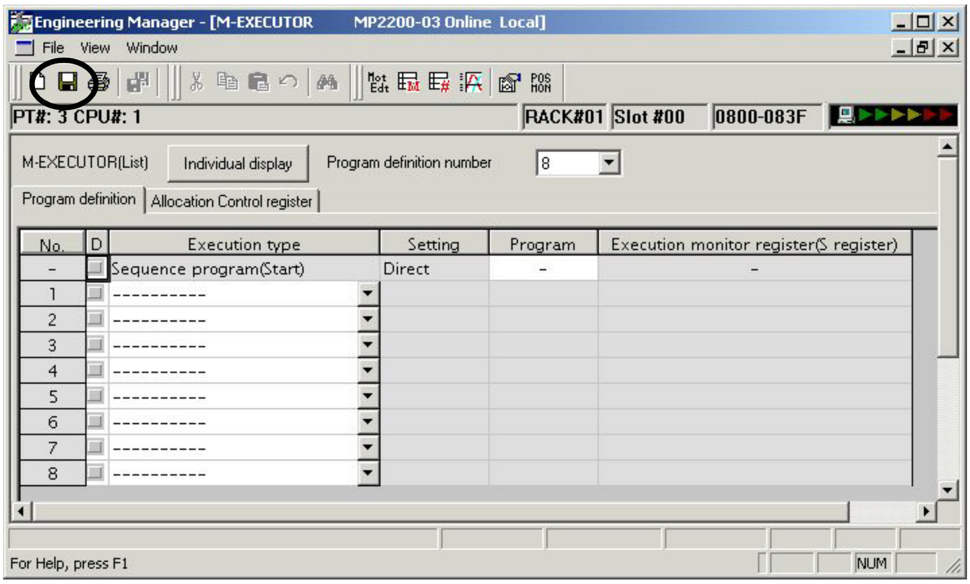
3. Click **Save** Icon to save the module configuration definition. Check the status of No.5 cell becomes **Unsetting**.



4. Double-click No.5 cell, then the **M-EXECUTOR definition** Dialog Box will appear. Click **OK** Button.

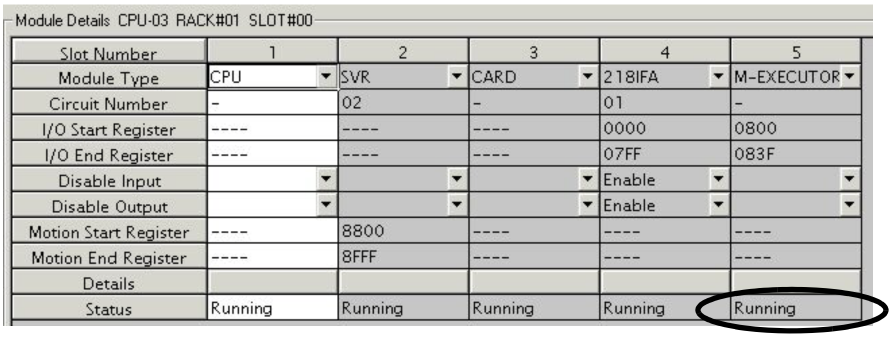


5. The following window will appear. Click **Save** Icon to save the **M-EXECUTOR (List)** Window.



This completes the initialization.

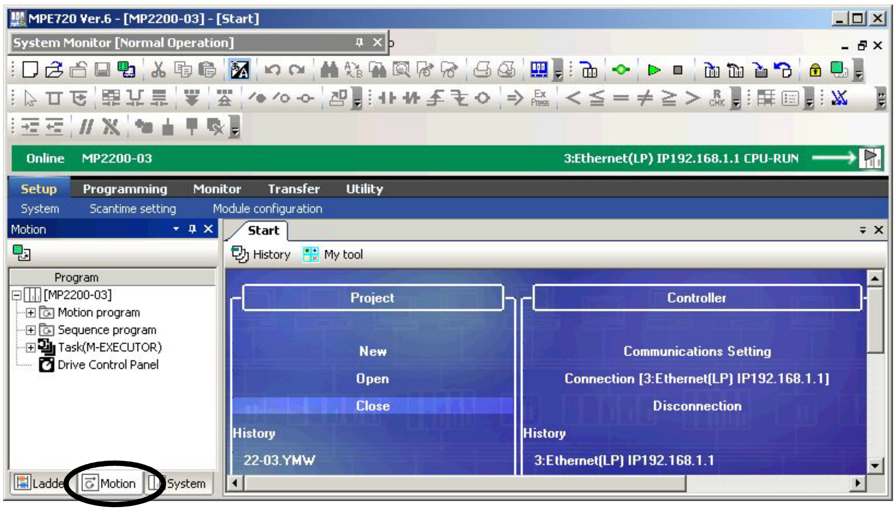
6. Return to Module Configuration Window. Check the status of No.5 cell changes from *Unsetting* to *Running*.



This enables the M-EXECUTOR function.

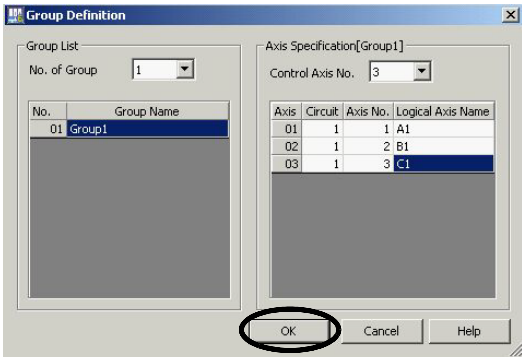
7.3.2 Programming Procedure

1. Click the **Motion** Tab in the subwindow.

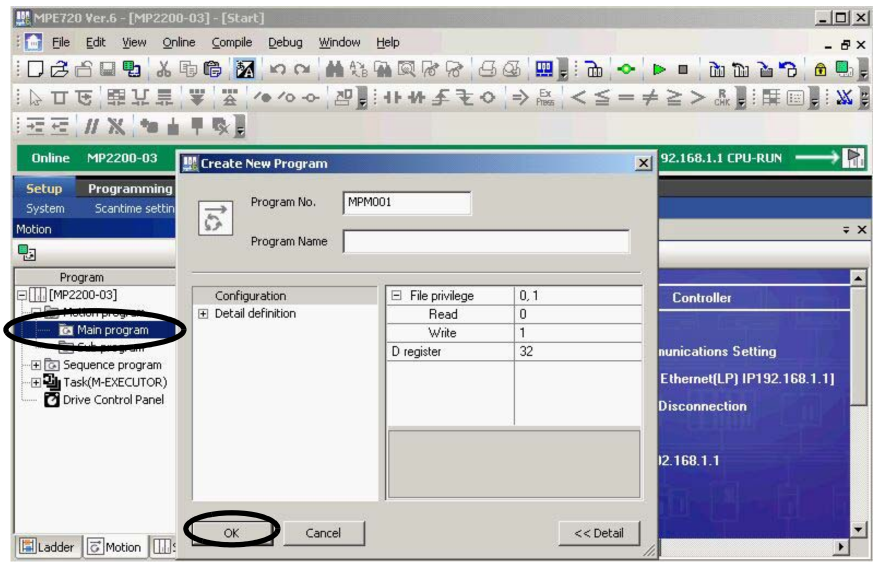


2. The motion program subwindow will appears.

When you double-click **Motion program** and there is not any group definition, the **Group Definition** Dialog Box will be shown. For this setting example, you do not need to change it, so accept the default setting and click the **OK** Button. Note that if a group definition already exists, the **Group Definition** Dialog Box screen will not be shown.



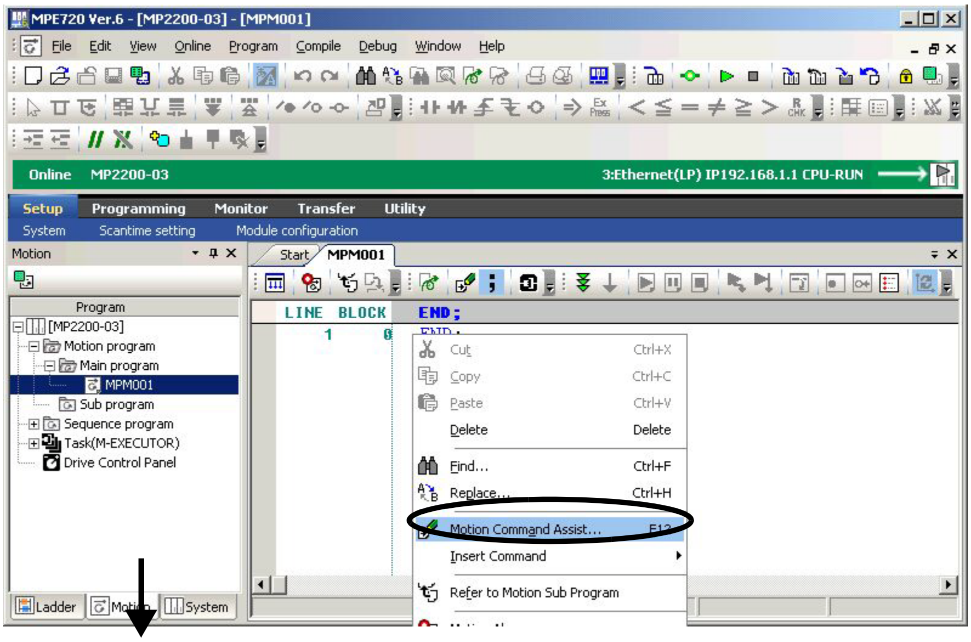
3. Right-click **Main program** and select **Create New** to display the **Create New Program** Dialog Box. Then click the **OK** Button.



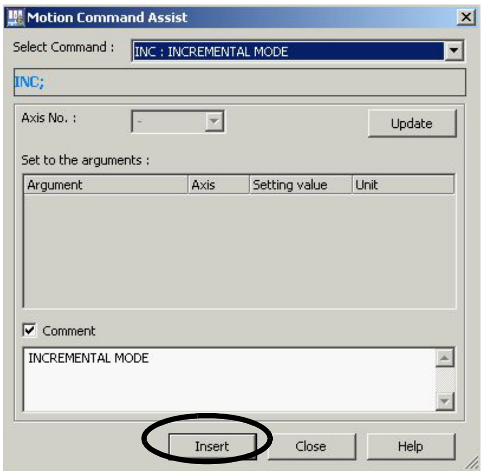
4. Editing Motion Program

Use the Motion Command Assist function to insert INC and MOV commands into the motion program.
The Motion Command Assist function is made accessible by right-clicking the mouse on the **Motion Editor** Window.

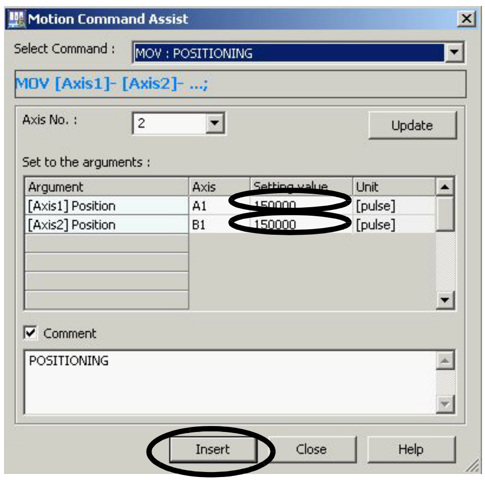
- Call the Motion Command Assist Function



- Insert an INC Command



- Insert a MOV Command

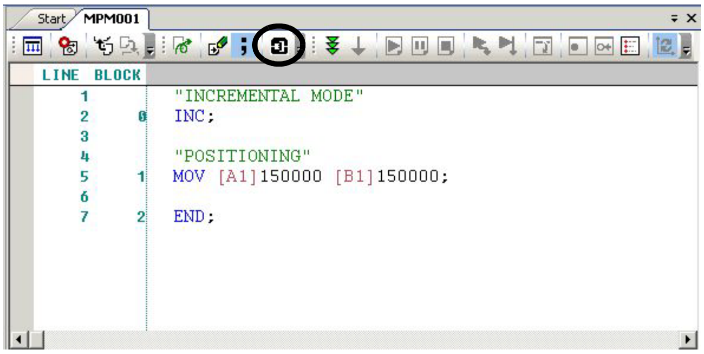


Click the save icon to save the motion program.

7.4 Executing Motion (step 3)

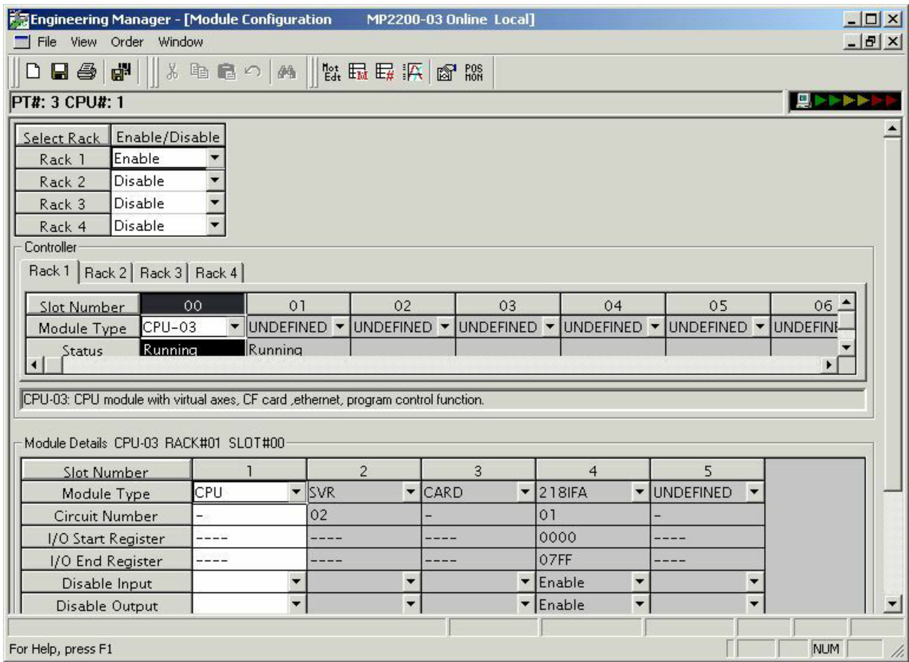
7.4.1 Registering Program Execution

1. Click the **Execution Registration** Icon.



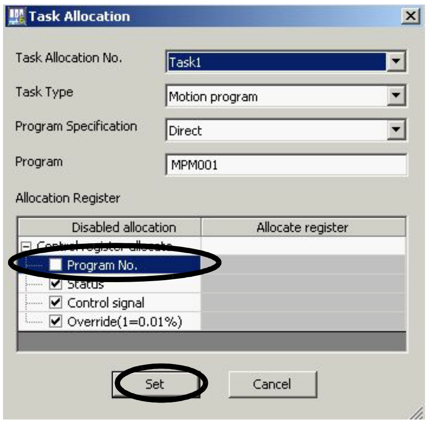
Note: 1. This program assumes that the servo has been turned ON using step 2 (Axis Selection and Servo ON) of *7.2.3 (2) Operating Manually in the Test Operation Window*.

2. If the following window appears after this operation, allocate the M-EXECUTOR. Refer to *7.3.1 Initializing the M-EXECUTOR Module*.



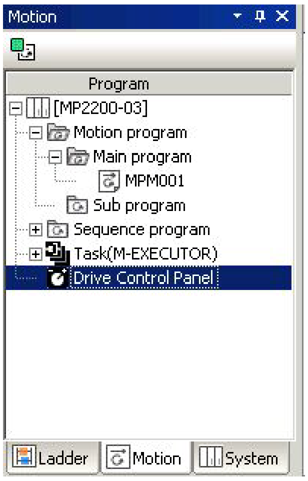
The **Program Execution Registry Screen** Dialog Box will appear.

2. Select **Program No.** and click the **Set** Button to save the registered contents.

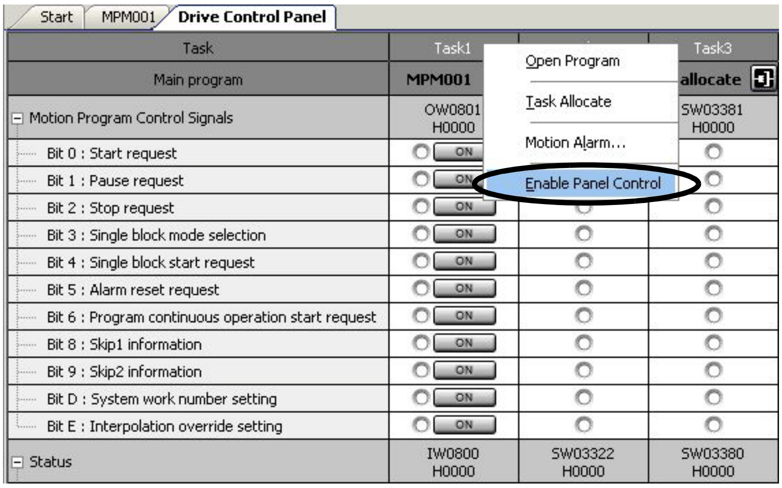


7.4.2 Starting a Motion Program Using the Drive Control Panel

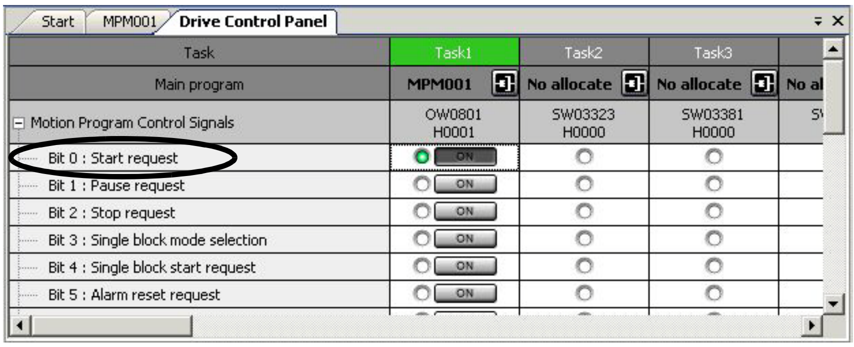
1. Double-click the **Drive Control Panel** Icon.



2. Right-click **Task1** in the dialog box that appears, and then select ***Enable Panel Control*** from the pop-up menu.



3. Click the **ON** Button for **Start request.** The MPM001 motion program will be executed.



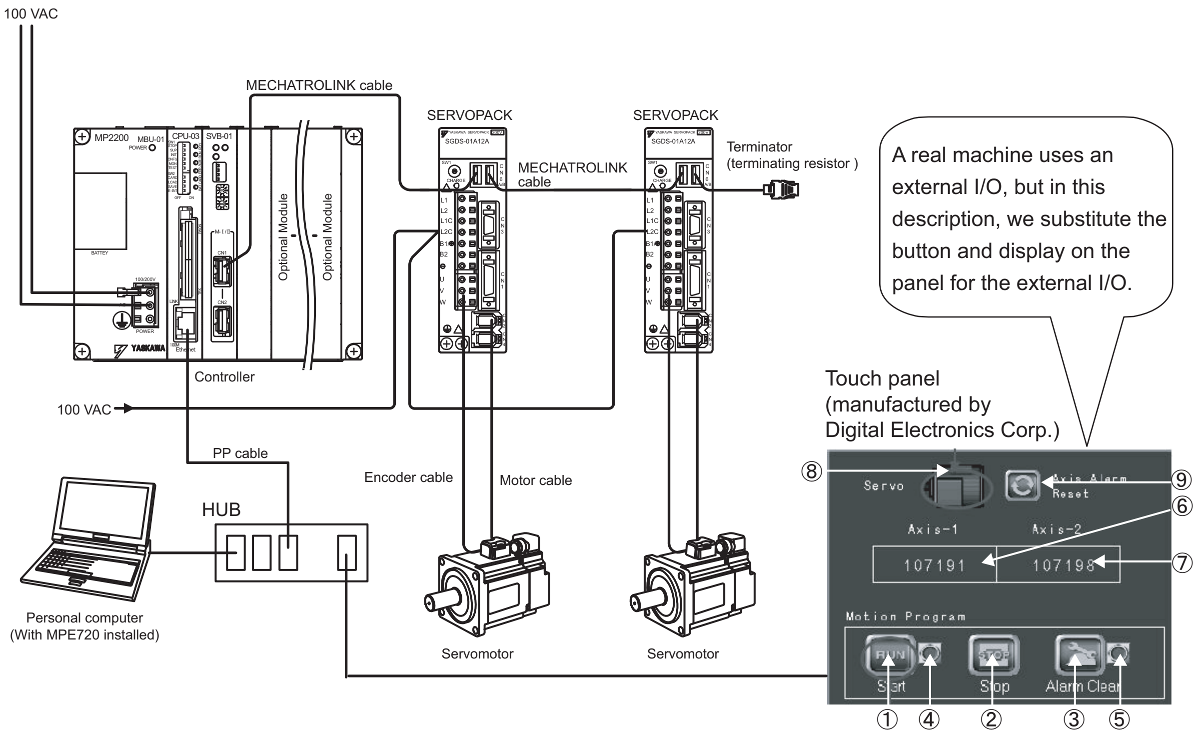
■ Caution

- This chapter explains the simple procedure where you can easily run and check a program without external signals. In practice, you need to connect to external signals and create a sequence.
- Registering a program execution enables the M-EXECUTOR definition. The MP2200 automatically controls the motion program, so be aware that changing registers registered in ladder and sequence programs may cause problems.

7.5 Starting Motion Program from an External Signal

7.5.1 Overview

This section explains how to start a motion program created in *7.3 Programming (step 2)* from external signals. Note that in this section, we show an example which substitutes a touch panel for the external signal.



7.5.2 Required Equipment

Product Name		Model	Q'ty
MP2200	Base Unit with AC power supply	JEPMC-BU2200	1
	CPU-03 Module (or CPU-04 Module)	JAPMC-CP2220-E (or JAPMC-CP2230-E)	1
	SVB-01 Module	JAPMC-MC2310	1
MECHATROLINK cable (0.5 m)		JEPMC-W6002-A5	2
Terminator (terminating resistor)		JEPMC-W6022	2
Σ-III SERVOPACK		SGDS-A5F12A	2
Σ-III servomotor		SGMAS-A5A2A21	2
Motor cable (3 m)		JZSP-CSM01-03	2
Encoder cable (3 m)		JZSP-CSP05-03	2
Touch panel (manufactured by Digital Electronics Corp.)		AGP3300-T1-D24	1
HUB (commercial product)		LSW-TX-8EP	1
MPE720 Ver.6		CPMC-MPE770	1
LAN cable (for Ethernet connection)		Commercial straight cable	3
Personal computer		Commercial product	1

Mapping of the panel manufactured by Digital Electronics Corp.

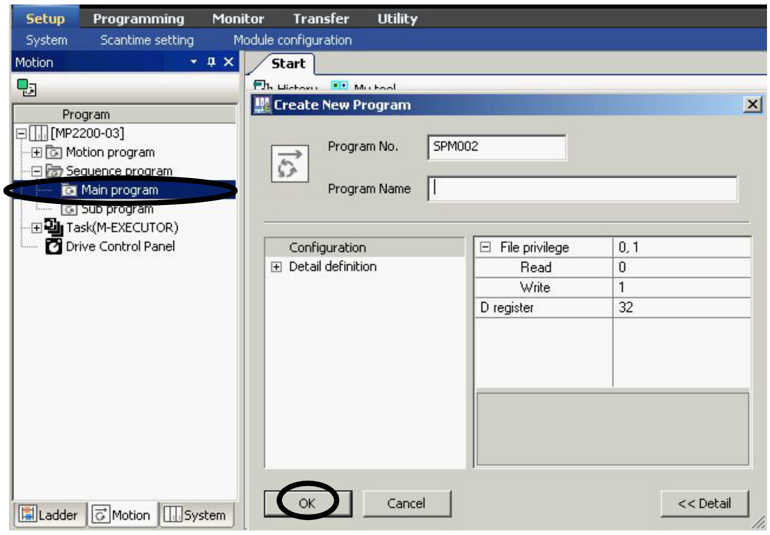
No.	Name	Mapping	Category	Description	MP2200 Operation
①	Start	MB5000	Control signal	Starts up a motion program	Sets in M-EXECUTOR
②	Stop	MB5002	Control signal	Displays the running of a motion program	
③	Clear Alarm	MB5005	Control signal	Stops a motion program	
④	Running Program	MB5010	Status	Clears an alarm of a motion program	
⑤	Alarm	MB5018	Status	Indicates an alarm is occurring in a motion program	
⑥	Axis 1 (current position)	IL8016	Monitor parameter	Displays current axis 1 position	Automatic receive function
⑦	Axis 2 (current position)	IL8096	Monitor parameter	Displays current axis 2 position	
⑧	Servo (ON/OFF)	MB5020	External signal	Axis 1, axis 2 servo ON signal	Sequence program is needed
⑨	Reset Axis Alarm	MB5021	External signal	Axis 1, axis 2 alarm reset signal	

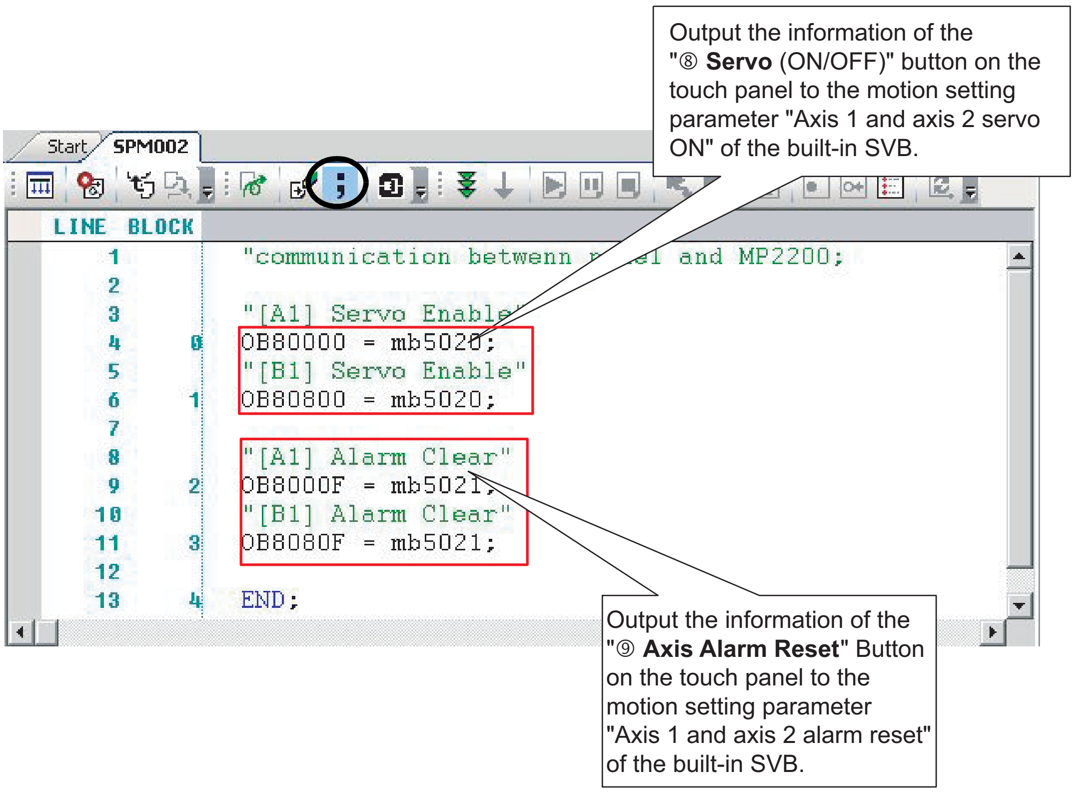
Note: 1. You do not need to create a program for signals and data in ① to ⑦.
2. You need to create a sequence program for outputting signals of ⑧ and ⑨ to the motion parameters.
3. For information on creating a program for the panel side, refer to *6.3.1 When CPU-03/CPU-04 Acts as Slave (automatic receive function is used)*.

7.5.3 Creation Procedure

1. Creating a Sequence Program

Now create a sequence program which copies the M register content mapped to "⑧ **Servo** (ON/OFF)" and "⑨ **Axis Alarm Reset**" Buttons on the touch panel to the relevant registers in the motion setting parameter of the built-in SVB.
Follow a procedure similar to creating a motion program from the motion program subwindow.

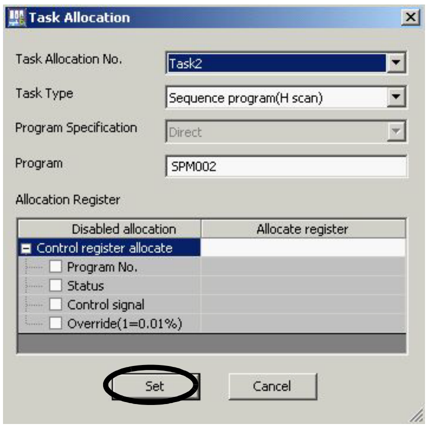




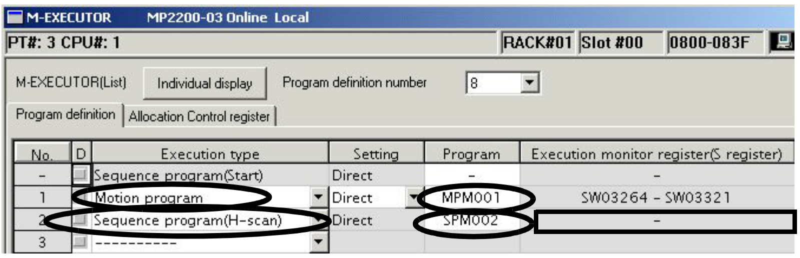
Click the **Execution Registration** Icon.

2. Registering Program Execution

- When the **Task Allocation** Dialog Box is displayed, click the **Set** Button to register execution of SPM002.

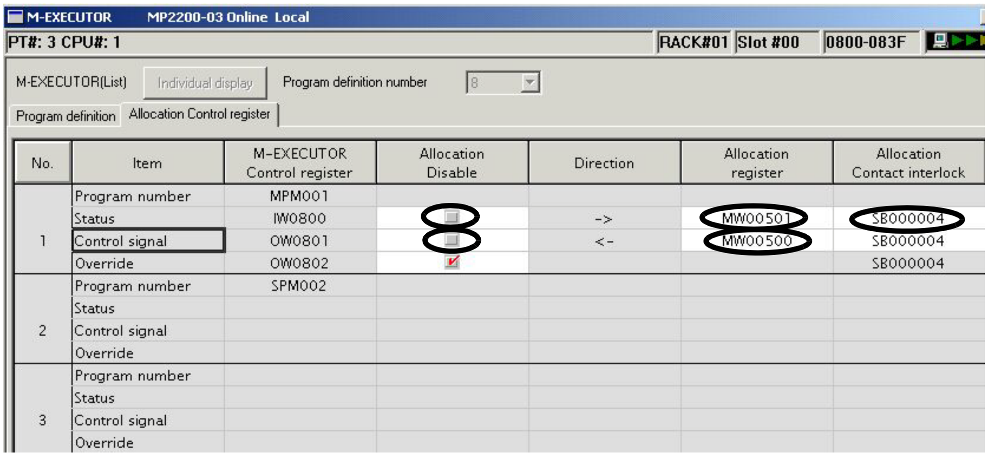


- Refer to *2.5.7 M-EXECUTOR Module (Motion Program Executor)*.

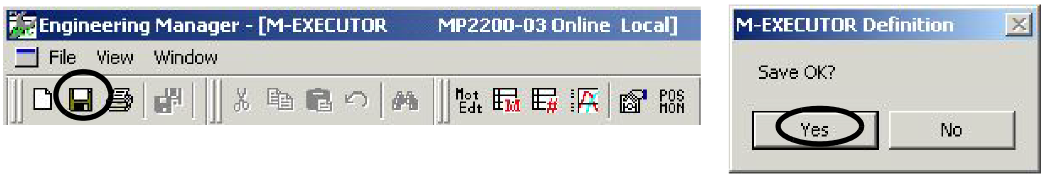


- In the **Allocation Control register** Tab Page, map the M registers allocated to control signals (① **Start** / ② **Stop** / ③ **Alarm Clear**) and status (④ **Running Program** / ⑤ **Alarm**) on the touch panel as an M-EXECUTOR allocation register for the motion program created in *7.3 Programming (step 2)*.

  Status=MW00501, control signal=MW00500, allocation contact interlock =SB00004



- Click the **Save** Icon to save the M-EXECUTOR definition.

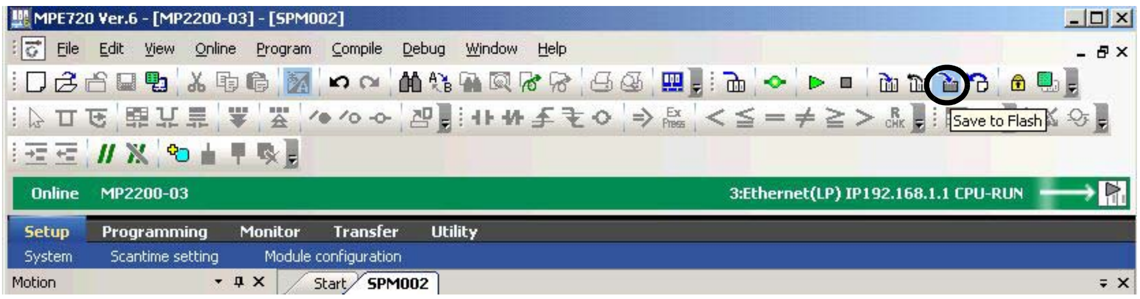


3. Communication Setting with Touch Panel

Refer to *6.3.1 When CPU-03/CPU-04 Acts as Slave (automatic receive function is used)* (automatic receive function is used) for the setting procedure for the touch panel.

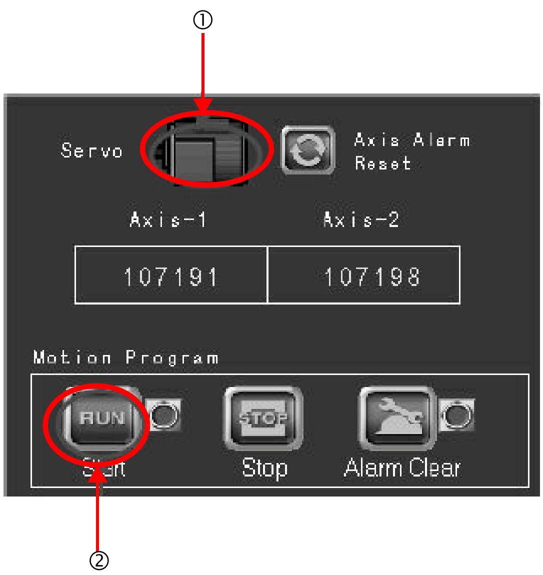
4. FLASH Save

When all settings are completed, click the **FLASH Save** Icon to save the data to the flash memory.



5. Operation Check

Turn ON MP2200 power again and press "① **Servo**" and "② **Start**" on the panel window.
Then check that the motion program starts and the motors of two axes begins to operate to change the current positions of two axes.



8

Maintenance, Inspection, and Troubleshooting

This chapter explains daily and regular inspection items to ensure that the MP2200 can always be used at its best conditions.

8.1 Inspection Items

This section summarizes daily and regular inspection items that must be performed by the customer.

8.1.1 Daily Inspections

The following table lists the daily inspection items.

No.	Inspection Item		Inspection Details	Criteria	Action
1	Installation conditions of Module, etc.		Check the mounting screws for looseness. Check whether the covers are all in place.	The screws and covers must be secured correctly.	Retighten the screws.
2	Connection conditions		Check the terminal screws for looseness.	The screws must be tight.	Retighten the screws.
			Check the connectors for looseness.	The connectors must be tight.	Retighten the connector set screws.
			Check the gap between crimp terminals.	There must be an appropriate gap between the terminals.	Correct as necessary.
3	LED Indicators	POWER*	Check whether the indicator is lit.	The indicator must be lit. (It is abnormal if the indicator is not lit.)	Refer to *8.2 Troubleshooting* .
		RDY	Check whether the indicator is lit.	The indicator must be lit. (It is abnormal if the indicator is not lit.)	
		RUN	Check whether the indicator is lit while the system is in RUN state.	The indicator must be lit. (It is abnormal if the indicator is not lit.)	
		ERR	Check whether the indicator is not lit.	The indicator must be not lit. (It is abnormal if the indicator is lit.)	
		ALM	Check whether the indicator is not lit.	The indicator must be not lit. (It is abnormal if the indicator is lit.)	
		BAT	Check whether the indicator is not lit.	The indicator must not be lit. (The battery voltage is too low if the indicator is lit.)	Replace the battery.

* Supported by the MBU-01 and MBU-02.

8.1.2 Regular Inspections

This section explains inspection items that must be performed once or twice every six months to one year.

Inspections must also be performed when the equipment is relocated or modified or when the wiring is changed.

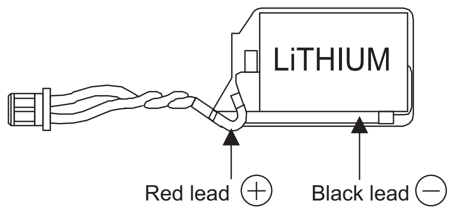
PROHIBITED

- Do not replace the built-in fuse.
  If the customer replaces the built-in fuse, the MP2200 may malfunction or break down.
  Contact your Yaskawa representative.

No.	Inspection Item		Inspection Details	Criteria	Action
1	Operating Environment	Ambient Temperature	Check the temperature and humidity with a thermometer and hygrometer, respectively. Check for corrosive gases.	0°C to 55°C	If the MP2200 is used inside a panel, treat the temperature inside the panel as the ambient temperature.
		Ambient Humidity		30% to 95% RH	
		Atmosphere		There must be no corrosive gases.	
2	Power Supply Voltage Check	MBU-01 Unit	Measure the voltage across 100/200-VAC terminals	85 to 132 VAC / 198 to 276 VAC	Change the power supply as necessary.
		MBU-02 and MBU-03 Units	Measure the voltage between 24-VDC terminals.	19.2 to 28.8 VDC	
3	Installation Conditions	Looseness and Excess Play	Attempt to move the Module.	The Module must be secured properly.	Retighten the screws.
		Dust and Other Foreign Matter	Visually check.	The Module must be free from dust and other foreign matter.	Clean.
4	Connection Conditions	Check the Terminal Screws for Looseness.	Check by retightening the screws.	The screws must be tight.	Retighten.
		Gap between Crimp Terminals	Visually check.	There must be an appropriate gap between the terminals	Correct.
		Looseness of Connectors	Visually check.	The screws must be tight.	Retighten the connector set screws.
5	Battery		Check the BAT indicator on the front panel of the Basic Unit.	The BAT indicator must be not lit.	If the BAT indicator is lit, replace the battery.

8.1.3 Replacing the Basic Unit Battery

The Basic Unit has one replaceable built-in battery. This battery is used to back up data to prevent the data stored in the memory from being lost when power is interrupted (e.g., when the power supply to the Basic Unit is turned OFF).
The built-in battery can retain the contents of the memory until the total time of power interruptions reaches one year. The warranty period of the battery is five years from the date of purchase. These values, however, differ according to the operating conditions, including the ambient temperature.
If the BAT indicator on the Basic Unit lights, replace the battery with a replacement battery (JZSP-BA01) within two weeks. Any delay in battery replacement will result in the data stored in the memory of the Basic Unit being lost.
The appearance of the battery is illustrated below.



• Battery Specifications

Type	Primary lithium battery
Battery Model	ER3VC
Manufacturer	Toshiba Home Appliances
Nominal Voltage	3.6 V
Nominal Capacity	1,000 mAh
Lithium Contents	0.3 g
Number Used	1

Fig. 8.1 JZSP-BA01 (Battery with Cable)

Note: This battery is not commercially available. Contact your Yaskawa representative.

(1) Procedure

⚠ CAUTION

- Do not allow the battery to be replaced by anyone other than qualified personnel who have received safety training.
  Improper battery replacement can result in electric shock. It can also cause equipment malfunction, injury to operators, or equipment damage.
- When replacing the battery, always do so with power supplied to the Basic Unit.
  If power to the Basic Unit is turned OFF when the battery is replaced, data stored in the memory in the CPU Module may be lost.
- Do not touch the battery terminals when replacing the battery.
  There is a risk of electrostatic discharge failure.
- A lithium battery is built into the Controller. After replacing the battery, dispose of the old battery separate from regular waste and in accordance with local regulations.

1. Save the data stored in the Basic Unit to a Compact Flash, hard disk on an external computer, or other media. This data is used to restore any data accidentally lost during battery replacement.

Note: For information on saving methods, refer to the *MP900/MP2000 Series MPE720 Software for Programming Device User’s Manual (Manual No.: SIEP C880700 05)*.

2. Check that the RDY indicator on the Basic Unit is lit.

3. Open the battery cover on the unit front surface.

4. Remove the connector on the end of lead of the built-in battery from the connector on the Basic Unit. Then, remove the built-in battery from the battery holder.

5. Insert securely the connector on the end of the lead of the replacement battery into the connector on the Basic Unit. Then, insert the replacement battery into the battery holder.

6. Check that the BAT indicator on the Basic Unit is unlit.

7. Close the battery cover. This completes replacing the battery.

8.2 Troubleshooting

This section describes the basic troubleshooting methods and provides a list of errors.

8.2.1 Basic Flow of Troubleshooting

When problems occur, it is important to quickly find the cause of the problems and get the system running again as soon as possible. The basic flow of troubleshooting is illustrated below.

Step 1	Visually confirm the following items.

- Machine movement (or status if stopped)
- Power supply
- I/O device status
- Wiring status
- Indicator status (LED indicators on each Module)
- Switch settings (e.g., DIP switches)
- Parameter settings and program contents

↓

Step 2	Monitor the system to see if the problem changes for the following operations.

- Switching the Controller to STOP status
- Resetting alarms
- Turning the power supply OFF and ON

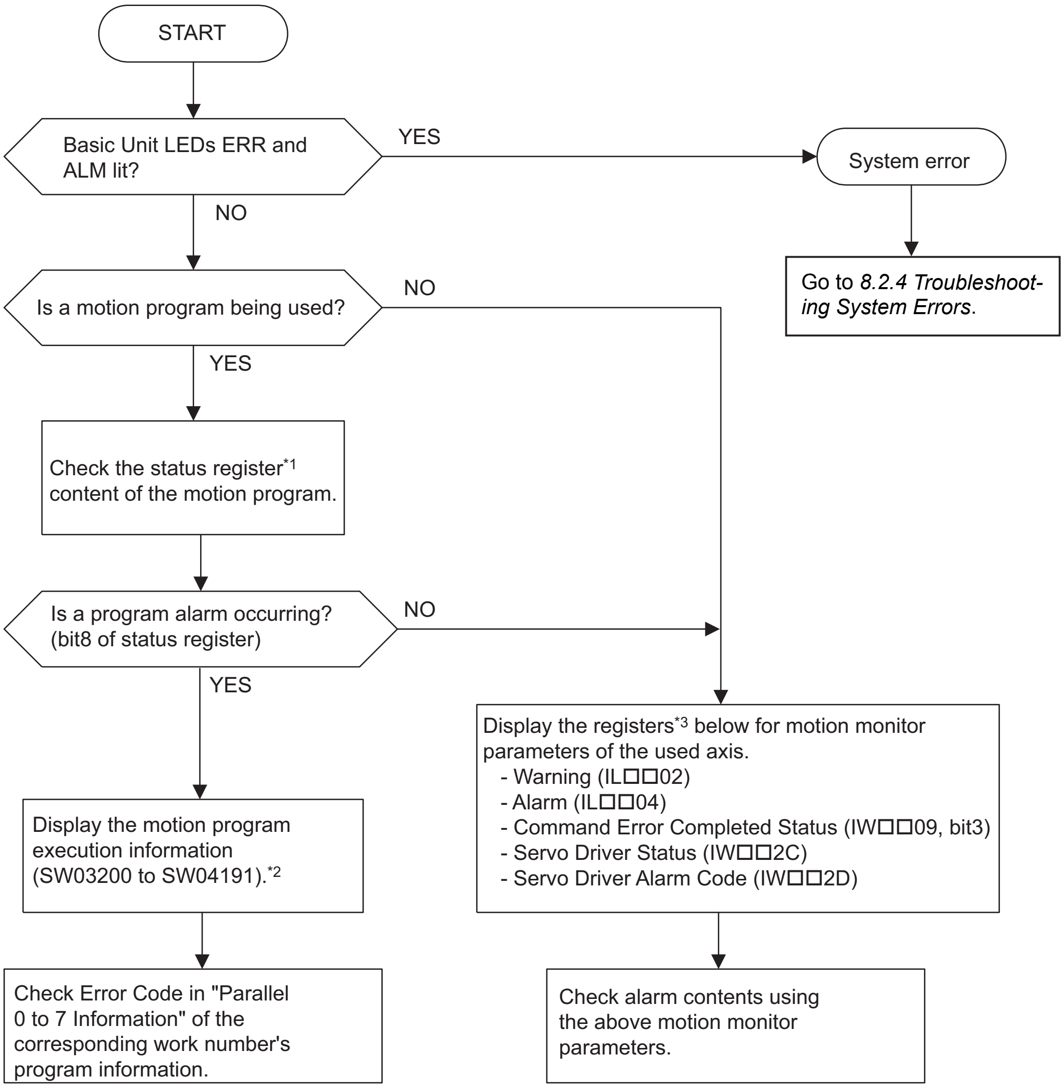
↓

Step 3	Determine the location of the cause from the results of steps 1 and 2.

- Controller or external?
- Sequence control or motion control?
- Software or hardware?

8.2.2 Error Check Flowchart

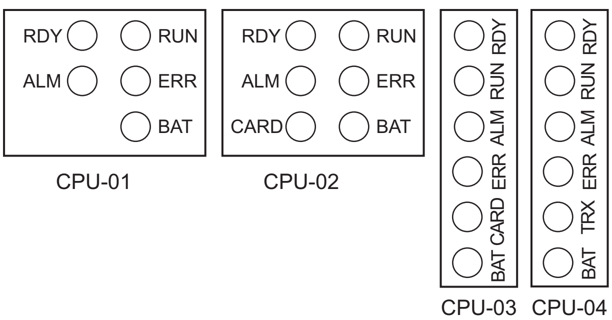
Find corrective actions for the problem using the following flowchart, if the cause of the problem is thought to be the MP2200 or SERVOPACK.



* 1. Refer to *5.2.3 (4) Work Register*.
* 2. Refer to *5.2.3 (6) Monitor the Motion Program Execution Information using S register*.
* 3. Refer to *8.2.6 List of Causes for Command Error Completed Status* and *8.2.7 Troubleshooting Motion Errors*.

8.2.3 LED Indicators

(1) LED Indicators



The status of the LED indicators on the front of the MP2200 Basic Unit can be used to determine the error status and meaning.
The locations in the program that need to be corrected can be determined by using the LED indicator status to determine the general nature of the error, using the contents of system (S) registers to check drawings and function numbers causing the error, and knowing the meaning of operation errors.

(2) LED Indicator Meanings

The following table shows how to use the LED indicators to determine the operating status of the MP2200, as well as relevant error information when the LED indicator status indicates an error.

<table>
<tr><th rowspan="2">Classification</th><th colspan="5">LED Indicator</th><th rowspan="2">Indicator Details</th><th rowspan="2">Countermeasures</th></tr>
<tr><th>RDY</th><th>RUN</th><th>ALM</th><th>ERR</th><th>BAT</th></tr>
<tr><td rowspan="5">Normal operation</td><td>Not lit</td><td>Not lit</td><td>Lit</td><td>Lit</td><td>Not lit</td><td>Hardware reset status</td><td rowspan="3">Usually the CPU will start within 10 seconds. If this status continues for more than 10 seconds, either a program error or hardware failure has occurred. Refer to 8.2.4 Troubleshooting System Errors and correct any system errors.</td></tr>
<tr><td>Not lit</td><td>Not lit</td><td>Not lit</td><td>Not lit</td><td>Not lit</td><td>Initialization</td></tr>
<tr><td>Not lit</td><td>Lit</td><td>Not lit</td><td>Not lit</td><td>Not lit</td><td>Drawing A (DWG.A) being executed.</td></tr>
<tr><td>Lit</td><td>Not lit</td><td>Not lit</td><td>Not lit</td><td>Not lit</td><td>User program stopped.
(Offline Stop Mode)</td><td>This status occurs
• When the stop operation is executed from the MPE720
• When the STOP switch is turned ON
This status does not indicate an error.</td></tr>
<tr><td>Lit</td><td>Lit</td><td>Not lit</td><td>Not lit</td><td>Not lit</td><td>User program being executed normally.</td><td>This is the normal status.</td></tr>
<tr><td rowspan="3">Errors</td><td>Not lit</td><td>Not lit</td><td>Not lit</td><td>Lit</td><td>Not lit</td><td>A serious error has occurred.</td><td>Refer to 8.2.4 (4) Correcting User Program Errors.</td></tr>
<tr><td>Not lit</td><td>Not lit</td><td>Not lit</td><td>Blinking</td><td>Not lit</td><td>Software Error
No. of blinks
3: Address error (read) exception
4: Address error (write) exception
5: FPU exception
6: Illegal general command error
7: Illegal slot command error
8: General FPU inhibited error
9: Slot FPU inhibited error
10: TLB duplicated bit error
11: TLB mistake error (read)
12: TLB mistake error (write)
13: TLB protection violation error (read)
14: TLB protection violation error (write)
15: Initial page write error</td><td rowspan="2">A hardware error has occurred.
Replace the Module.</td></tr>
<tr><td>Not lit</td><td>Not lit</td><td>Blinking</td><td>Blinking</td><td>Not lit</td><td>Hardware Error
Number of LED blinks indicates error type.
2: RAM diagnostic error
3: ROM diagnostic error
4: CPU function diagnostic error
5: FPU function diagnostic error</td></tr>
<tr><td rowspan="2">Warnings</td><td>–</td><td>–</td><td>–</td><td>–</td><td>Lit</td><td>Battery alarm</td><td>Refer to 8.1.3 Replacing the Basic Unit Battery and replace the Battery.</td></tr>
<tr><td>Lit</td><td>Lit</td><td>Lit</td><td>Not lit</td><td>Not lit</td><td>Operation error
I/O error</td><td>Refer to 8.2.4 (5) [c] Ladder Program User Operation Error Status and 8.2.4 (5) [e] System I/O Error Status.</td></tr>
</table>

8.2.4 Troubleshooting System Errors

The LED indicators on the front of the Basic Unit can be used to determine MP2200 operating status and error status. To obtain more detailed information on errors, the system (S) registers can be used. A detailed check of the contents of system registers can be used to determine the location of the error and take the corrective measures.
Details on system registers are provided below.

(1) System Register Allocations

The following table shows the overall structure of the system registers.

SW00000	System Service Register	
SW00030	System Status	→ *8.2.4 (5) [a] System Status*
SW00050	System Error Status	→ *8.2.4 (5) [b] System Error Status*
SW00080	User Operation Error Status	→ *8.2.4 (5) [c] Ladder Program User Operation Error Status*
SW00090	System Service Execution Status	→ *8.2.4 (5) [d] System Service Execution Status*
SW00110	User Operation Error Status Details	→ *8.2.4 (5) [c] Ladder Program User Operation Error Status*
SW00190	Alarm Counter and Alarm Clear	→ *8.2.4 (5) [e] System I/O Error Status*
SW00200	System I/O Error Status	
SW00500	Reserved by the system.	
SW00698	Interrupt Status	
SW00800	Module Information	→ *8.2.4 (5) [h] Module Information*
SW01312	Reserved by the system.	
SW02048	Reserved by the system.	
SW03200	Motion Program Information	→ *8.2.5 Motion Program Alarms*
SW05200 to SW08191	Reserved by the system.	

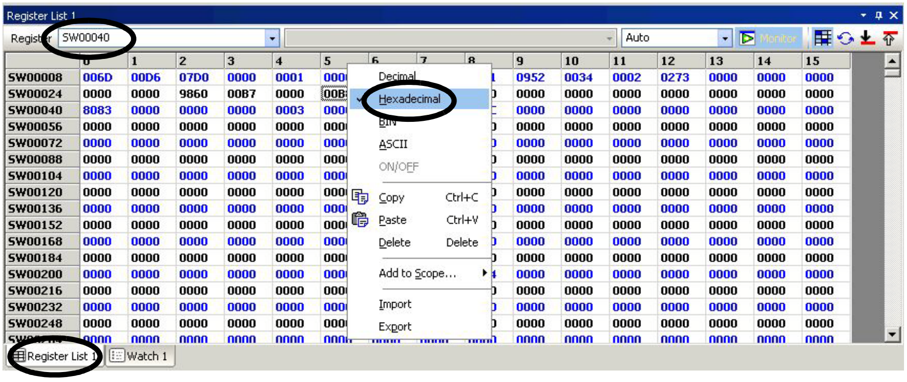
(2) Accessing System Registers

To access the contents of system registers, start the MPE720 Programming Tool and use the Register List.

[a] Register List Display Procedure

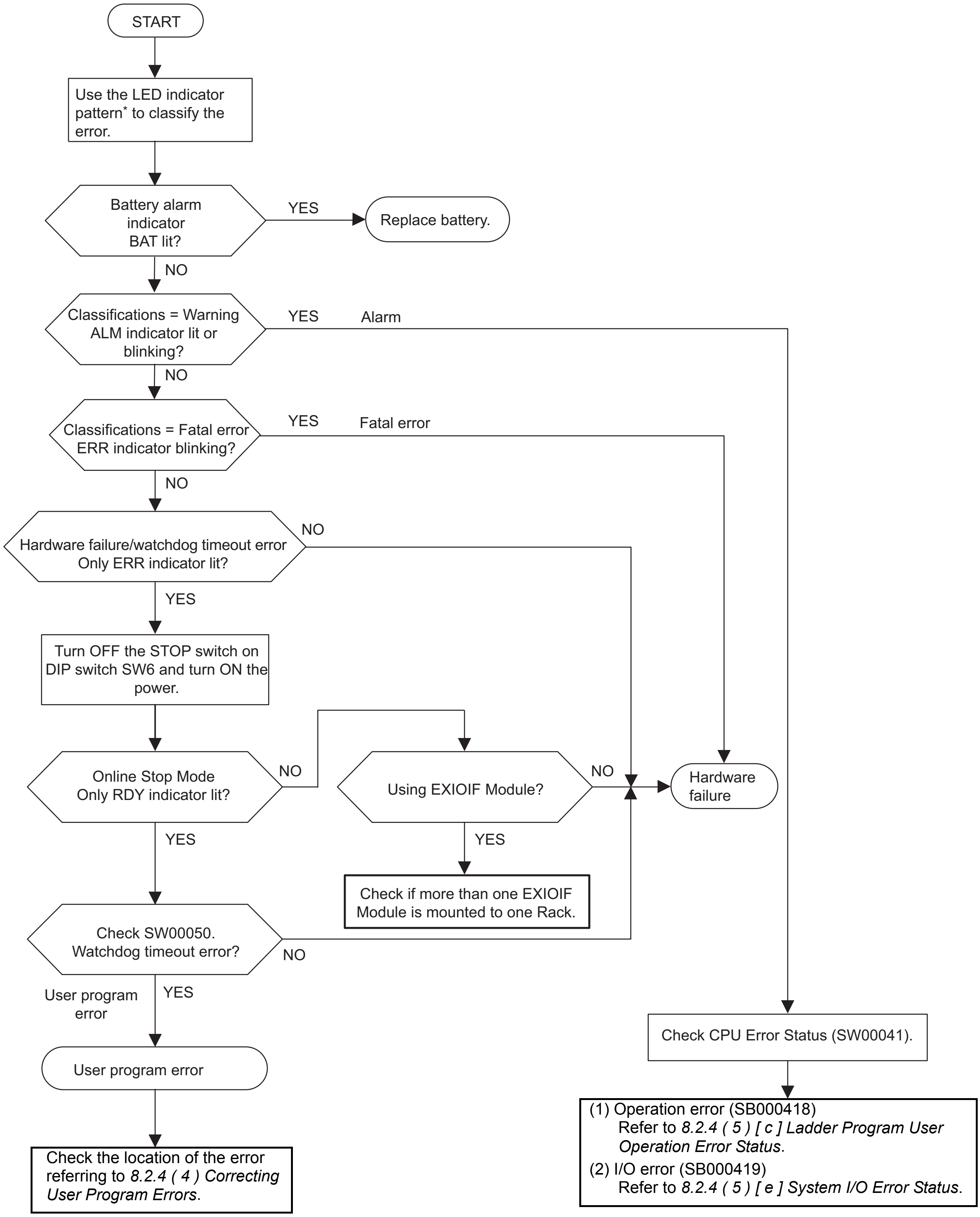
Use the following procedure to display the register list.

1. Click **Register List 1** to display the **Register List 1** Window.
2. Enter the number of the first system register (SWxxxx) to be referenced (here, SW00040).
3. Right-click any cell in the **Register List 1** Window that appears and change the display mode to hexadecimal.



(3) Troubleshooting Flowchart for System Errors

A troubleshooting flowchart for system errors is provided below.



* For LED indicator pattern, refer to *8.2.3 (2) LED Indicator Meanings*.

(4) Correcting User Program Errors

A serious error may have occurred if the ALM and ERR indicators on the front of the MP2200 Basic Module are lit red. Set the MP2200 in stop status (STOP switch on DIP switch 1-6: ON) and investigate the error.
Use the following procedure to investigate ladder program errors.

Indicator	Step	Details
ERR indicator lit	(1) Investigate type of serious error.	Check the contents of SW00050 (Error Code) to determine if the type of the serious error is a system error or a user program error.
ERR indicator lit	(2) Investigate type of program in which there is an error.	Check the contents of SW00055 (Program Type) to determine if the error is in a drawing or function.
ERR indicator lit	(3) Investigate the drawing with the error.	Check the contents of SW00054 (Program Error Task) and SW00056 (Program Error Drawing Number) to determine the drawing with the error.
ERR indicator lit	(4) Investigate the function with the error.	If SW00056 (Program Error Drawing Number) contains 0100H, the error is in a function. Check the contents of SW00057 (Calling Drawing Type) and SW00058 (Ladder Program Function Calling Drawing Number) to determine the drawing with the error. Check the contents of SW00059 (Ladder Program Function Calling Drawing Step Number) to determine the step number with the operation error.
ALM indicator lit	(5) Check to see whether an operation error has occurred.	Check the error count for each drawing in SW00080 to SW00089. If errors have been counted, an operation error has occurred. Go to (6).
ALM indicator lit	(6) Investigate the type of operation error and its location.	1. Check Error Details Check error codes for drawings where the error is counted. DWG.A: SW00111, DWG.H: SW00143 DWG.I: SW00127, DWG.L: SW00175 2. Check the Drawing Number Check the error drawing number for the drawing number where an error occurred. DWG.A: SW00122, DWG.H: SW00154 DWG.I: SW00138, DWG.L: SW00186 3. Errors in Functions Check the Function Referencing Drawing Number and Function Referencing STEP Number. DWG.A: SW00123, 4; DWG.H: SW00155, 6 DWG.I: SW00139, 40; DWG.L: SW00187, 8
	(7) Determine the error occurrence location.	After the investigation of an error drawing or error function is complete, set the corresponding drawing, function, or sequence program to Disable and turn on the power supply in the RUN state to check that no error occurs. Refer to *8.2.4 (4) [a] How to Disable a User Program*.

Go on to the next page.

Continued on from the previous page.

(8) Investigate the I/O state	When the error cause can be assumed to be a problem with the external input data or output data from the user program, disable the corresponding I/O process to investigate a cause based on the I/O data. Refer to *8.2.4 (4) (b) How to Disable the I/O Process*. Also, when investigating an output point, refer to *8.2.4 (4) [c] How to Forcibly Turn ON/OFF Coil*.

(9) Correct Programs.	Correct the program that causes error.

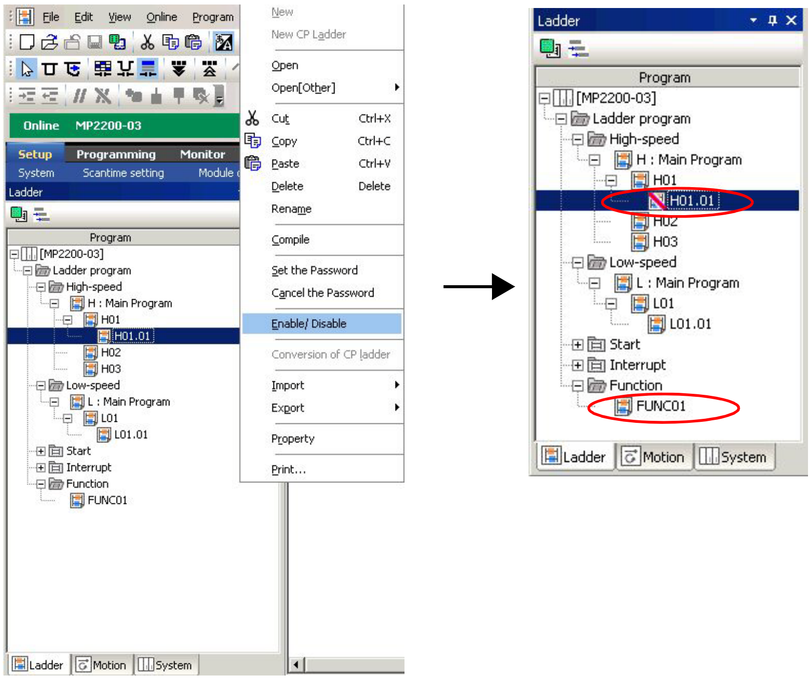
(10) Investigate the scan time	If no problem is found in the program, investigate the scan time. Check to see whether the maximum value for high-speed scan time is in excess of or equal to the setting value in the scan time window, or whether the maximum value for slow-speed scan time is in excess of the setting value. Refer to *8.2.4 (4) [d] Operation in Case of Scan Time Over*.

(11) Change the scan time	Change the scan time setting value. Refer to *5.5.3 Setting and Changing the Scan Time*.

[a] How to Disable a User Program

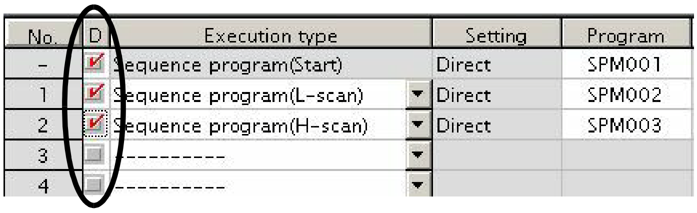
- Drawing or Function

  In the ladder subwindow of the MPE720 online mode, right-click the corresponding drawing and function and select Enable/Disable in the pop-up menu.



- Sequence Program (Supported by the CPU-03 and CPU-04)

  In the **Module Configuration** Window of the MPE720 online mode, open the M-EXECUTOR module detailed definition window and select **D** check box of the sequence program definition to disable the definition and save the change.

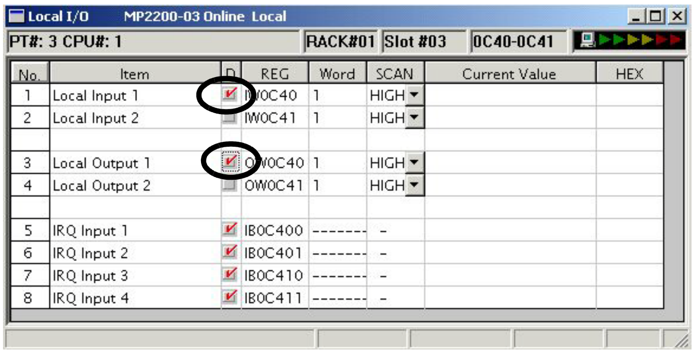


■ Caution

When a drawing, function, or sequence program (supported by the CPU-03/CPU-04) is disabled, the equipment may become unstable, causing personal injury or damage to the equipment. If carrying out an investigation, be aware of the behavior of the equipment when a drawing, function, or sequence program is disabled.
After the investigation, make sure to enable the drawing, function, or sequence program (supported by the CPU-03/CPU-04) again.

(b) How to Disable the I/O Process

In the **Module Configuration** Window of the MPE720 online mode, open the detailed definition window of the module for which you want to disable the I/O process. Click “D” and save it, to disable the I/O process for the clicked item. You can change the disabled input register to any value.



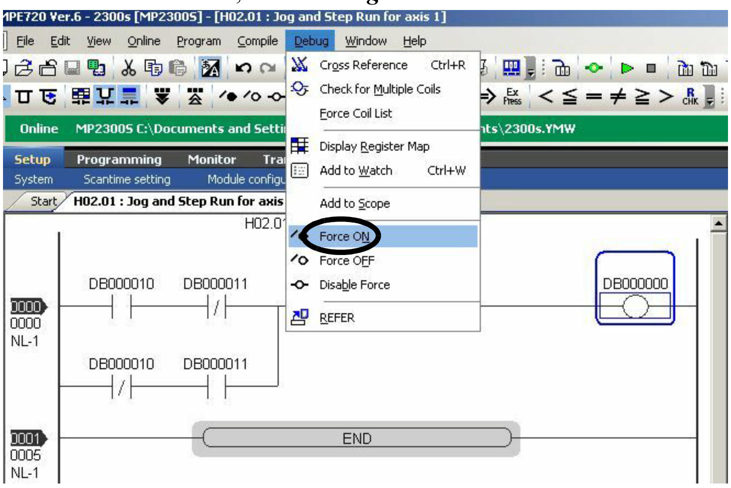
■ Caution

When an I/O process is disabled, the equipment may become unstable, causing personal injury or damage to the equipment.
If carrying out an investigation, be aware of the behavior of the equipment when an I/O process is disabled.
After the investigation, make sure you enable the I/O service again.

---

[c] How to Forcibly Turn ON/OFF Coil

In the main menu of the MPE720 online mode, select ***Debug - Force ON*** or ***Force OFF*** menus, in that order.



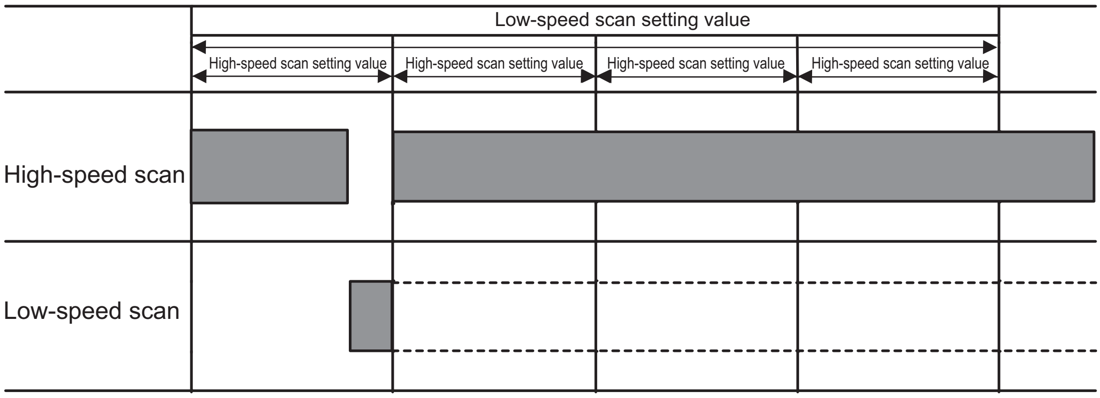
■ Caution

When a coil is set to forced ON or OFF, the equipment may become unstable, causing personal injury or damage to the equipment. If carrying out an investigation, be aware of the behavior of the equipment when a coil is forcibly turned ON/OFF.
After the investigation, make sure to forcibly cancel the setting.

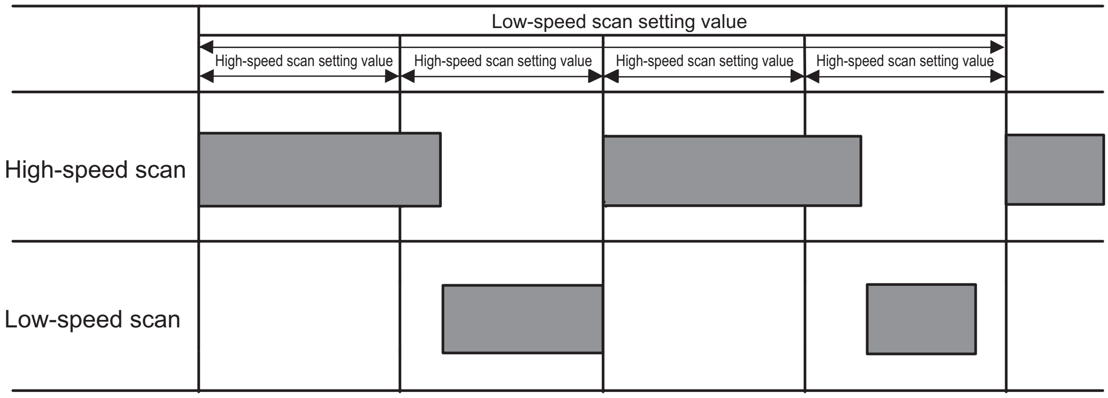
---

[d] Operation in Case of Scan Time Over

When the maximum value for high-speed scan time is equal to a setting value, a watchdog timeout error will occur because the time for performing a low-speed scan cannot be ensured.



When the maximum value for a scan time is in excess of a setting value, the scan cannot be performed at every setting value. SW00044 is added due to a high-speed scan over, SW00046 is added due to a low-speed scan over.



(5) System Register Configuration and Error Status

[a] System Status

System operating status and error status is stored in registers SW00040 to SW00048. Check system status details to determine whether hardware or software is the cause of an error.

Name	Register No.	Description		
Reserved by the system.	SW00030 to SW00039			
CPU Status	SW00040	SB000400	READY	0: Failure, 1: Normal
		SB000401	RUN	0: Stopped, 1: Running
		SB000402	ALARM	0: Normal, 1: Alarm
		SB000403	ERROR	0: Normal, 1: Error
		SB000404	Reserved by the system.	
		SB000405	Reserved by the system.	
		SB000406	FLASH	1: FLASH operation
		SB000407	WEN	0: Write-disabled, 1: Write-enabled
		SB000408 to SB00040D	Reserved by the system.	
		SB00040E	Operation Stop Request	0: RUN, 1: STOP
		SB00040F	Run Switch Status at Power ON	0: STOP 1: RUN
CPU Error Status	SW00041	SB000410	Serious Failure	1: WDGE, undefined command See SW00050 for details.
		SB000411	Reserved by the system.	
		SB000412	Reserved by the system.	
		SB000413	Exception Error	
		SB000414 to SB000417	Reserved by the system.	
		SB000418	User operation error	1: User operation error
		SB000419	I/O Error	1: I/O error
		SB00041A to SB00041F	Reserved by the system.	
H Scan Over Counter	SW00044			
L Scan Over Counter	SW00046			
Reserved by the system.	SW00047			

(cont'd)

Name	Register No.	Description		
Hardware Configuration Status	SW00048	SB000480	TEST	Store DIP switch. 0: ON, 1: OFF
		SB000481	MON	
		SB000482	CNFG	
		SB000483	INIT	
		SB000484	SUP	
		SB000485	STOP	
		SB000486	–	
		SB000487	Battery Alarm	
		SB000488	USB1/E-INIT (for CPU-02 and CPU-03)	Store DIP switch. 0: ON, 1: OFF
		SB000489	SAVE (for CPU-02 and CPU-03)	
		SB00048A	LOAD (for CPU-02 and CPU-03)	
		SB00048B	CARD (for CPU-02 and CPU-03)	
		SB00048C to SB00048F	Reserved by the system.	
Reserved by the system.	SW00049	SW000490 to SW00049F	Reserved by the system.	

[b] System Error Status

System error status is stored in registers SW00050 to SW00060.

Name	Register No.	Description		
32-bit Error Code	SW00050	0001H	Watchdog timeout error	
		0041H	ROM diagnosis error	
		0042H	RAM diagnosis error	
		0043H	CPU diagnosis error	
		0044H	FPU diagnosis error	
		0050H	EXIOIF error	
		0051H	Module synchronous error[*1]	
		00E0H	Address read exception error	
		0100H	Address write exception error	
		0120H	FPU exception error	
		0180H	Illegal general command error	
		01A0H	Illegal slot command error	
		01E0H	User break after command execution	
		0800H	General FPU prohibition exception error	
		0820H	Slot FPU prohibition exception error	
	SW00051	For system error analysis		
32-bit Addresses Generating Error	SW00052	For system error analysis		
	SW00053			
Program Error Task	SW00054	0000H: System 0001H: DWG.A	0002H: DWG.I 0003H: DWG.H	0005H: DWG.L
Program Type	SW00055	0000H: System 0001H: DWG.A	0002H: DWG.I 0003H: DWG.H	0005H: DWG.L 0008H: Function 000FH: Motion program/ sequence program[*2]

(cont'd)

Name	Register No.	Description
Program Error Drawing Number	SW00056	Ladder program parent drawing: FFFFH Ladder program function: 8000H Ladder program child drawing: xx00H (Hxx: Child drawing number) Ladder program grandchild drawing: xxyyH (Hyy: Grandchild drawing number) Motion program/sequence program*2: F0xxH (Hxx: program number)
Calling Drawing Type	SW00057	Type of drawing that calls the ladder program function in which an error occurred. 0001H: DWG.A 0002H: DWG.I 0003H: DWG.H 0005H: DWG.L 0008H: Ladder program function 000FH: Motion program/ sequence program*2 0010H: Reserved by system. 0011H: Reserved by system.
Ladder Program Function Calling Drawing Number	SW00058	Number of drawing that calls the ladder program function in which an error occurred. Parent drawing: FFFFH Function: 0100H Child drawing: xx00H (Hxx: Child drawing number) Grandchild drawing: xxyyH (Hyy: Grandchild drawing number)
Ladder Program Function Calling Drawing Step Number	SW00059	Step number of the drawing that calls the ladder program function in which an error occurred. 0 when there is an error in the drawing.
Error Data	SW00060 and SW00061	Reserved by the system.
	SW00062 to SW00065	Name of Task Generating Error
	SW00066 and SW00067	Reserved by the system.
	SW00068	Year Generated
	SW00069	Month Generated
	SW00070	Day of Week Generated
	SW00071	Day of Month Generated
	SW00072	Hour Generated
	SW00073	Minutes Generated
	SW00074	Seconds Generated
	SW00075	Milliseconds Generated (Not used.)
	SW00076	Number of slot with a module synchronous error *3 □□yyH □□: Rack number (01 to 04) yy: Slot number (01 to 09)
	SW00078 and SW00079	Reserved by the system.

* 1. The CPU system program version 2.75 or later: 0051H will be reported.
The CPU system program version is earlier than 2.75: 0001H (watchdog timer over error) will be reported.

* 2. Sequence program is supported by the CPU-03 and CPU-04.

* 3. The CPU system program version 2.75 or later: Number of slot with a module synchronous error will be reported.

[c] Ladder Program User Operation Error Status

Error information for user operation errors in ladder programs is stored in registers SW00080 to SW00089 (Error Status 1) and SW00110 to SW00189 (Error Status 2).

Table 8.1 Ladder Program User Operation Error Status 1

Name	Register No.	Description
DWG.A Error Count Error Code	SW00080	Operation error code: See *Ladder Program User Operation Error Codes 1*. Error code when an index error occurs: See *Ladder Program User Operation Error Codes 2*.
	SW00081	
DWG.I Error Count Error Code	SW00082	
	SW00083	
DWG.H Error Count Error Code	SW00084	
	SW00085	
Reserved by the system.	SW00086	
	SW00087	
DWG.L Error Count Error Code	SW00088	
	SW00089	

Table 8.2 Ladder Program User Operation Error Status 2

Name	Register No.				Remarks
	DWG.A	DWG.I	DWG.H	DWG.L	
Error Count	SW00110	SW00126	SW00142	SW00174	Error Drawing Number Parent drawing: FFFFH Child drawing: xx00H (Hxx: Child drawing number) Grandchild drawing: xxyyH (Hyy: Grandchild drawing number) Function: 8000H Motion program/sequence program*: F0xxH (Hxx: program number) • Function Calling Drawing Number Number of the drawing that calls the function in which an error occurred. • Function Calling DWG Step Number Step number of the drawing that calls the function in which an error occurred. 0 when there is an error in the parent drawing.
Error Code	SW00111	SW00127	SW00143	SW00175	
Error A Register	SW00112	SW00128	SW00144	SW00176	
	SW00113	SW00129	SW00145	SW00177	
Modification A Register	SW00114	SW00130	SW00146	SW00178	
	SW00115	SW00131	SW00147	SW00179	
Error F Register	SW00116	SW00132	SW00148	SW00180	
	SW00117	SW00133	SW00149	SW00181	
Modification F Register	SW00118	SW00134	SW00150	SW00182	
	SW00119	SW00135	SW00151	SW00183	
Error Generating Address	SW00120	SW00136	SW00152	SW00184	
	SW00121	SW00137	SW00153	SW00185	
Error Drawing Number	SW00122	SW00138	SW00154	SW00186	
Function Calling Drawing Number	SW00123	SW00139	SW00155	SW00187	
Function Calling DWG Step Number	SW00124	SW00140	SW00156	SW00188	
Reserved by the system.	SW00125	SW00141	SW00157	SW00189	

* Sequence programming is supported by the CPU-03 and CPU-04.

Table 8.3 Ladder Program User Operation Error Codes 1

<table>
<tr><th></th><th>Error Code</th><th colspan="2">Error Contents</th><th>User*</th><th>System Default Value</th></tr>
<tr><td rowspan="7">Integer Operations</td><td>0001H</td><td colspan="2">Integer operation - underflow</td><td>Yes</td><td>–32768 [–32768]</td></tr>
<tr><td>0002H</td><td colspan="2">Integer operation - overflow</td><td>Yes</td><td>32767 [32767]</td></tr>
<tr><td>0003H</td><td colspan="2">Integer operation - division error</td><td>Yes</td><td>The A register remains the same.</td></tr>
<tr><td>0009H</td><td colspan="2">Double-length integer operation - underflow</td><td>Yes</td><td>–2147483648 [–2147483648]</td></tr>
<tr><td>000AH</td><td colspan="2">Double-length integer operation - overflow</td><td>Yes</td><td>2147483647 [2147483647]</td></tr>
<tr><td>000BH</td><td colspan="2">Double-length integer operation - division error</td><td>Yes</td><td>The A register remains the same.</td></tr>
<tr><td>010xH</td><td colspan="2">Operation error drawing - integer operation error (□ = 1 to B)</td><td>No</td><td>Default indicated above.</td></tr>
<tr><td rowspan="21">Real Number Operation</td><td>0010H</td><td colspan="2">Integer storage - non-numeric error</td><td>Yes</td><td>Store not executed. [00000]</td></tr>
<tr><td>0011H</td><td colspan="2">Integer storage - underflow</td><td>Yes</td><td>Store not executed. [–32768]</td></tr>
<tr><td>0012H</td><td colspan="2">Integer storage - overflow</td><td>Yes</td><td>Store not executed. [+32767]</td></tr>
<tr><td>0021H</td><td colspan="2">Real number storage - underflow</td><td>Yes</td><td>Store not executed. [–1.OE+38]</td></tr>
<tr><td>0022H</td><td colspan="2">Real number storage - overflow</td><td>Yes</td><td>Store not executed. [1.OE+38]</td></tr>
<tr><td>0023H</td><td colspan="2">Real number operation - division-by-zero error</td><td>Yes</td><td>Operation not executed.
The F register remains the same.</td></tr>
<tr><td>0030H</td><td colspan="2">Real number operation - invalid operation (non-numeric)</td><td>No</td><td>Operation not executed.</td></tr>
<tr><td>0031H</td><td colspan="2">Real number operation - exponent underflow</td><td>No</td><td>0.0</td></tr>
<tr><td>0032H</td><td colspan="2">Real number operation - exponent overflow</td><td>No</td><td>Maximum value</td></tr>
<tr><td>0033H</td><td colspan="2">Real number operation - division error (non-numeric 0/0)</td><td>No</td><td>Operation not executed.</td></tr>
<tr><td>0034H</td><td colspan="2">Real number storage - exponent underflow</td><td>No</td><td>Stores 0.0.</td></tr>
<tr><td>0035H</td><td colspan="2">Real number operation - stack error</td><td></td><td></td></tr>
<tr><td rowspan="9">0040H
to
0059H</td><td colspan="2">Standard System Functions
Real number operation errors</td><td>No</td><td>Interrupt operation and output = 0.0</td></tr>
<tr><td>0040H: SQRT</td><td>0041H: SIN</td><td>0042H: COS</td><td>0043H: TAN</td></tr>
<tr><td>0044H: ASIN</td><td>0045H: ACOS</td><td>0046H: ATAN</td><td>0047H: EXP</td></tr>
<tr><td>0048H: LN</td><td>0049H: LOG</td><td>004AH: DZA</td><td>004BH: DZB</td></tr>
<tr><td>004CH: LIM</td><td>004DH: PI</td><td>004EH: PD</td><td>004FH: PID</td></tr>
<tr><td>0050H: LAG</td><td>0051H: LLAG</td><td>0052H: FGN</td><td>0053H: IFGN</td></tr>
<tr><td>0054H: LAU</td><td>0055H: SLAU</td><td>0056H: REM</td><td>0057H: RCHK</td></tr>
<tr><td>0058H: BSRCH</td><td>0059H: SORT</td><td></td><td></td></tr>
<tr><td colspan="4">1000H or 2000H is added for an index error.</td></tr>
</table>

Note: Yes: Can be set to value other than system default from the user program.
No: The system default cannot be changed from the user program.

Table 8.4 Ladder Program User Operation Error Codes 2

<table>
<tr><th></th><th>Error Code</th><th colspan="2">Error Contents</th><th>User</th><th>System Default</th></tr>
<tr><td rowspan="2">Integer - Real Number Operations</td><td>1000H</td><td colspan="2">Index error within drawing</td><td>No</td><td>Execute again with i, j = 0.
The i register remains the same.</td></tr>
<tr><td>2000H</td><td colspan="2">Index error within function</td><td>No</td><td>Execute again with i, j = 0.
The i register remains the same.</td></tr>
<tr><td rowspan="4">Integer Operation</td><td rowspan="4">x060H
to
x077H
(x = 1, 2)</td><td colspan="2">Integer system functions
Index error</td><td>No</td><td>Operation stopped and output = input.
The A register remains the same.</td></tr>
<tr><td>x06DH: PI</td><td>x06EH: PD</td><td>x06FH: PID</td><td>x070H: LAG</td></tr>
<tr><td>x071H: LLAG</td><td>x072H: FGN</td><td>x073H: IFGN</td><td>x074H: LAU</td></tr>
<tr><td>x075H: SLAU</td><td>x076H: FGN</td><td>x077H: IFGN</td><td></td></tr>
</table>

[d] System Service Execution Status

Table 8.5 Data Trace Execution Status

Name	Register No.	Remarks
Reserved by the system.	SW00090 to SW00097	
Existence Of Data Trace Definition	SW00098	Bit 0 to 3 = Group 1 to 4 Definition exists = 1, No definition = 0
Data Trace Execution Status	SW00099	Bit 0 to 3 = Group 1 to 4 Trace stopped = 1, Trace executing = 0

Table 8.6 Latest Data Trace Record Numbers

Name	Register No.	Remarks
Data Trace Group 1	SW00100	Latest record number
Data Trace Group 2	SW00101	Latest record number
Data Trace Group 3	SW00102	Latest record number
Data Trace Group 4	SW00103	Latest record number

[e] System I/O Error Status

Name	Register No.	Remarks
Current Alarm	SW00190	Cleared when power is turned ON.
Number of Alarm History Records	SW00191	The number of alarms in the alarm history.
Clear Alarms	SW00192	1: Alarm cleared 2: Current alarm and alarm history cleared
I/O Error Count	SW00200	Number of I/O errors
Input Error Count	SW00201	Number of input errors
Input Error Address	SW00202	Latest input error address (IW□□□□ register number)
Output Error Count	SW00203	Number of output errors
Output Error Address	SW00204	Latest output error address (OW□□□□ register number)
Reserved by the system.	SW00205	(Not used.)
	SW00206	
	SW00207	
I/O Error Status	SW00208 to SW00215	Rack 1, Slot 0, Error status
	SW00216 to SW00223	Reserved by the system.
	SW00224 to SW00231	Rack 1, Slot 1, Error status
	SW00232 to SW00239	Rack 1, Slot 2, Error status
	SW00240 to SW00247	Rack 1, Slot 3, Error status
	SW00248 to SW00255	Rack 1, Slot 4, Error status
	...	
	SW00496 to SW00503	Rack 4, Slot 9, Error status

[f] Actions to be Taken when an I/O Error Occurs

When an I/O error occurs during system I/O, the error status is reported in the system registers as shown in the following table.

Name		Register No.	Remarks
CPU	CPU-01/CPU-02	SW00208 to SW00217	Not used.
	CPU-03/CPU-04		Refer to the next page for details on the error status.
Reserved by system.		SW00218 to SW00223	Not used.
Rack 1, Slot 1 Information		SW00224 to SW00231	Depends on the mounted Module and error code.
Rack 1, Slot 2 Information		SW00232 to SW00239	Same as above.
Rack 1, Slot 3 Information		SW00240 to SW00247	Same as above.
Rack 1, Slot 4 Information		SW00248 to SW00255	Same as above.
Rack 1, Slot 5 Information		SW00256 to SW00263	Same as above.
Rack 1, Slot 6 Information		SW00264 to SW00271	Same as above.
Rack 1, Slot 7 Information		SW00272 to SW00279	Same as above.
Rack 1, Slot 8 Information		SW00280 to SW00287	Same as above.
Rack 2, Slot 1 Information		SW00288 to SW00295	Same as above.
Rack 2, Slot 2 Information		SW00296 to SW00303	Same as above.
Rack 2, Slot 3 Information		SW00304 to SW00311	Same as above.
Rack 2, Slot 4 Information		SW00312 to SW00319	Same as above.
Rack 2, Slot 5 Information		SW00320 to SW00327	Same as above.
Rack 2, Slot 6 Information		SW00328 to SW00335	Same as above.
Rack 2, Slot 7 Information		SW00336 to SW00343	Same as above.
Rack 2, Slot 8 Information		SW00344 to SW00351	Same as above.
Rack 2, Slot 9 Information		SW00352 to SW00359	Same as above.
Rack 3, Slot 1 Information		SW00360 to SW00367	Same as above.
Rack 3, Slot 2 Information		SW00368 to SW00375	Same as above.
Rack 3, Slot 3 Information		SW00376 to SW00383	Same as above.
Rack 3, Slot 4 Information		SW00384 to SW00391	Same as above.
Rack 3, Slot 5 Information		SW00392 to SW00399	Same as above.
Rack 3, Slot 6 Information		SW00400 to SW00407	Same as above.
Rack 3, Slot 7 Information		SW00408 to SW00415	Same as above.
Rack 3, Slot 8 Information		SW00416 to SW00423	Same as above.
Rack 3, Slot 9 Information		SW00424 to SW00431	Same as above.
Rack 4, Slot 1 Information		SW00432 to SW00439	Same as above.
Rack 4, Slot 2 Information		SW00440 to SW00447	Same as above.
Rack 4, Slot 3 Information		SW00448 to SW00455	Same as above.
Rack 4, Slot 4 Information		SW00456 to SW00463	Same as above.
Rack 4, Slot 5 Information		SW00464 to SW00471	Same as above.
Rack 4, Slot 6 Information		SW00472 to SW00479	Same as above.
Rack 4, Slot 7 Information		SW00480 to SW00487	Same as above.
Rack 4, Slot 8 Information		SW00488 to SW00495	Same as above.
Rack 4, Slot 9 Information		SW00496 to SW00503	Same as above.

■ CPU-03 and CPU-04 Error Status (Rack 1, Slot 0)

(Bit number)	F ------ 8	7 ------ 0
SW00208	Status	Subslot (function) number

	F ------ 1		0
SW00209	Unused	Write	Read

	F ------ C	B ------ 8	7 ------ 4	3 ------ 0
SW00210	Transmission station for writing	Reserved by the system.	Transmission station for reading	Reserved by the system.

SW00211	Unused
SW00212	Unused

[Details of 218IFA/218IFC Error Status] (SW00208 to SW00212)

Items	Code	Remarks
Subslot Number	4	4 = 218IFA/218IFC (Ethernet)
Status	0	Normal
	1	Station error
Read/ Write	0	Normal communications
	1	Communications error
Transmission Station for Reading/ Writing	0x0	No error
	0x4	Parameter format error
	0x5	Command sequence error
	0x6	Reset status
	0x7	Data receiving error
	0x8	Data sending error
	0xA	Connection error

■ LIO-01/02 Module Error Status (Example for Rack 1, Slot 1)

(Bit No.)	F--------8	7--------0
SW00224	Error Code (I/O error = 2)	Subslot No. (= 1)
SW00225	Error Code (I/O error = 2)	Subslot No. (= 2)

SW00226	Not used	----------------	Not used
SW00227	Not used	----------------	Not used
SW00228	Not used	----------------	Not used
SW00229	Not used	----------------	Not used
SW00230	Not used	----------------	Not used
SW00231	Not used	----------------	Not used

■ 260IF-01 Module Error Status (Example for Rack 1, Slot 3)

(Bit No.)	F--------8	7--------0
SW00240	Error Code (Station error = 1)	Subslot No. (= 2)

SW00241	ST#15	----------------	ST#0
SW00242	ST#31	----------------	ST#16
SW00243	ST#47	----------------	ST#22
SW00244	ST#63	----------------	ST#48

[Error status details]

Item	Code	Remarks
ST#n	0	Normal communication
	1	Communication error at station n (when slave, n becomes a local station number)

[g] Interrupt Status

Name	Register No.	Remarks
Interrupt Detection Counter	SW00698	
Module Generating Interrupt	SW00699	Number of Interrupt Modules for one time
Interrupt Module	SW00700	Interrupt Module 1
	SW00701	
	SW00702	
	SW00703	Interrupt Module 2
	SW00704	
	SW00705	
	:	
	:	
	SW00787	Interrupt Module 30
	SW00788	
	SW00789	

■ Interrupt Module Details

	F 8	7 0	(Bit No.)
SW00□□□ + 0	Rack	Slot	mmssH
SW00□□□ + 1	Interrupt Type		
SW00□□□ + 2	Hardware Interrupt Factor Register Values		

1. Rack
   mm = 01 to 04
   The rack number where the Module that was the interrupt factor is mounted.
2. Slot
   ss = 01 to 09
   The slot number where the Module that was the interrupt factor is mounted.
3. Interrupt Type
   1: Reserved by the system.
   2: LIO-01/LIO-02/LIO-04 DI interrupts
   3: LIO-01/LIO-02 counter interrupts
4. Hardware Interrupt Factor Register Values
   - Interrupt Type = 2 (LIO-01, LIO-02, and LIO-04 DI Interrupts)

Bit	Meaning
0 to 4	Reserved by the system.
5	LIO-01 and LIO-02 interrupt inputs: 1 = Interrupt input, 0 = No interrupt input
6 to 8	Reserved by the system.
9	LIO-04 interrupt input 1: 1 = Interrupt input, 0 = No interrupt input
10	LIO-04 interrupt input 2: 1 = Interrupt input, 0 = No interrupt input
11	LIO-04 interrupt input 3: 1 = Interrupt input, 0 = No interrupt input
12	LIO-04 interrupt input 4: 1 = Interrupt input, 0 = No interrupt input
13 to 15	Reserved by the system.

   - Interrupt Type = 3 (LIO-01 and LIO-02 Counter Interrupts)

Bit	Meaning
0 to 3	Reserved by the system.
4	Counter Agreement Status: 1 = Counter agreement, 0 = No counter agreement
5 to 15	Reserved by the system.

[h] Module Information

Name	Register No.	Remarks		
		CPU-01/CPU-02	CPU-03	CPU-04
CPU Information	SW00800	Basic Unit ID		
	SW00801	CPU Hardware Version (BCD)		
	SW00802	CPU Software Version (BCD)		
	SW00803	Number of Subslots		
	SW00804	CPU Function Module ID		
	SW00805	CPU Function Module Status		
	SW00806	SVR Function Module ID		
	SW00807	SVR Function Module Status		
	SW00808	Reserved by the system.	CARD Function Module ID	Reserved by system.
	SW00809		CARD Function Module ID Status	
	SW00810		218IFA function Module ID	218IFC Function Module ID
	SW00811		218IFA Function Module Status	218IFC Function Module Status
	SW00812		M-EXECUTOR Function Module ID	
	SW00813		M-EXECUTOR Function Module Status	
	SW00814 to SW00815		Reserved by system.	
Rack 1, Slot 1 Information	SW00816	Module ID		
	SW00817	Hardware Version (BCD)		
	SW00818	Software Version (BCD)		
	SW00819	Number of Subslots		
	SW00820	Subslot 1 Function Module ID		
	SW00821	Subslot 1 Function Module Status		
	SW00822	Subslot 2 Function Module ID		
	SW00823	Subslot 2 Function Module Status		
Rack 1, Slot 2 Information	SW00824 to SW00831	Same as above.		
Rack 1, Slot 3 Information	SW00832 to SW00839	Same as above.		
Rack 1, Slot 4 Information	SW00840 to SW00847	Same as above.		
Rack 1, Slot 5 Information	SW00848 to SW00855	Same as above.		
Rack 1, Slot 6 Information	SW00856 to SW00863	Same as above.		
Rack 1, Slot 7 Information	SW00864 to SW00871	Same as above.		
Rack 1, Slot 8 Information	SW00872 to SW00879	Same as above.		
Rack 2, Slot 1 Information	SW00880	Module ID		
	SW00881	Hardware Version (BCD)		
	SW00882	Software Version (BCD)		
	SW00883	Number of Subslots		
	SW00884	Subslot 1 Function Module ID		
	SW00885	Subslot 1 Function Module Status		
	SW00886	Subslot 2 Function Module ID		
	SW00887	Subslot 2 Function Module Status		
Rack 2, Slot 2 Information	SW00888 to SW00895	Same as above.		
Rack 2, Slot 3 Information	SW00896 to SW00903	Same as above.		
Rack 2, Slot 4 Information	SW00904 to SW00911	Same as above.		
Rack 2, Slot 5 Information	SW00912 to SW00919	Same as above.		
Rack 2, Slot 6 Information	SW00920 to SW00927	Same as above.		
Rack 2, Slot 7 Information	SW00928 to SW00935	Same as above.		
Rack 2, Slot 8 Information	SW00936 to SW00943	Same as above.		
Rack 2, Slot 9 Information	SW00944 to SW00951	Same as above.		

(cont'd)

Name	Register No.	Remarks		
		CPU-01/CPU-02	CPU-03	CPU-04
Rack 3, Slot 1 Information	SW00952	Module ID		
	SW00953	Hardware Version (BCD)		
	SW00954	Software Version (BCD)		
	SW00955	Number of Subslots		
	SW00956	Subslot 1 Function Module ID		
	SW00957	Subslot 1 Function Module Status		
	SW00958	Subslot 2 Function Module ID		
	SW00959	Subslot 2 Function Module Status		
Rack 3, Slot 2 Information	SW00960 to SW00967	Same as above.		
Rack 3, Slot 3 Information	SW00968 to SW00975	Same as above.		
Rack 3, Slot 4 Information	SW00976 to SW00983	Same as above.		
Rack 3, Slot 5 Information	SW00984 to SW00991	Same as above.		
Rack 3, Slot 6 Information	SW00992 to SW00999	Same as above.		
Rack 3, Slot 7 Information	SW01000 to SW01007	Same as above.		
Rack 3, Slot 8 Information	SW01008 to SW01015	Same as above.		
Rack 3, Slot 9 Information	SW01016 to SW01023	Same as above.		
Rack 4, Slot 1 Information	SW01024	Module ID		
	SW01025	Hardware Version (BCD)		
	SW01026	Software Version (BCD)		
	SW01027	Number of Subslots		
	SW01028	Subslot 1 Function Module ID		
	SW01029	Subslot 1 Function Module Status		
	SW01030	Subslot 2 Function Module ID		
	SW01031	Subslot 2 Function Module Status		
Rack 4, Slot 2 Information	SW01032 to SW01039	Same as above.		
Rack 4, Slot 3 Information	SW01040 to SW01047	Same as above.		
Rack 4, Slot 4 Information	SW01048 to SW01055	Same as above.		
Rack 4, Slot 5 Information	SW01056 to SW01063	Same as above.		
Rack 4, Slot 6 Information	SW01064 to SW01071	Same as above.		
Rack 4, Slot 7 Information	SW01072 to SW01079	Same as above.		
Rack 4, Slot 8 Information	SW01080 to SW01087	Same as above.		
Rack 4, Slot 9 Information	SW01088 to SW01095	Same as above.		

8.2.5 Motion Program Alarms

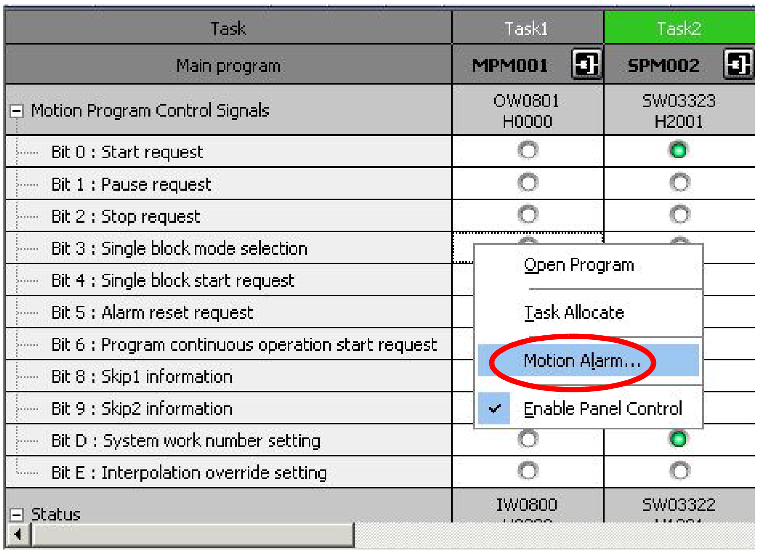
If a motion program alarm occurs, find the cause of alarm indicated by the alarm code.
The alarm code, alarm name, and its corrective actions in a motion program can be checked on the **Motion Alarm** Window.
This section explains the **Motion Alarm** Window and motion alarm codes:

(1) Motion Alarm Window

The following two options are available for displaying the **Motion Alarm** Window.

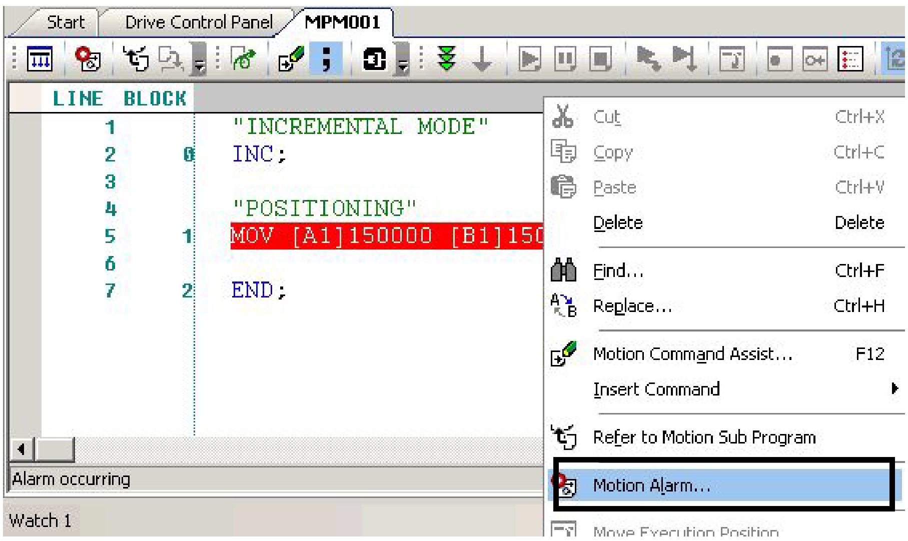
[a] Open from Drive Control Panel Tab

Right-click any cell in the **Drive Control Panel** Tab Page. Click the **Motion Alarm** Button in the pop-up menu that will appear.

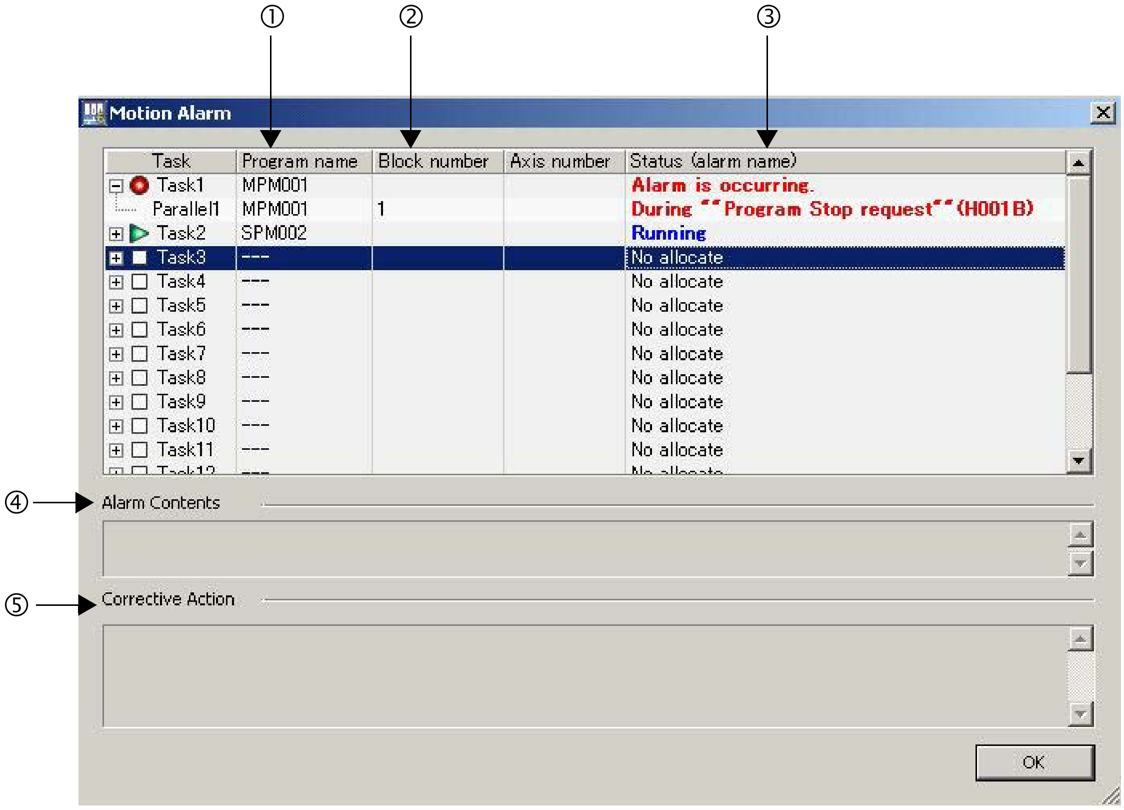


[b] Open from Right-click Menu on the Motion Editor

Select ***Motion Alarm*** from the menu displayed by right-clicking on the motion editor.



This section explains the **Motion Alarm** Window.

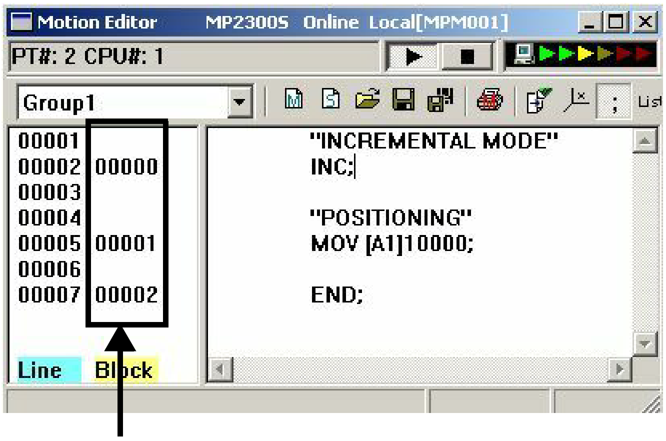


① Program number

The name of the program where an error occurred is shown.

② Block number

The number of the block where an error occurred is shown.
Double-clicking the number will bring you to the corresponding program where the error occurred.
The block number is shown in the motion editor.



③ Status (alarm name)

The alarm name is shown.

④ Alarm Contents

The alarm content are shown.

⑤ Corrective Action

Corrective actions for the alarm are shown.

(2) Motion Program Alarm Codes

(a) Configuration of Motion Program Alarms

The following diagram shows the configuration of alarms.

Bit15	Bit12 Bit8	Bit7 Bit0
	Alarm occurrence axis information (1 to 16)	Alarm code (When Bit 7 is ON: Axis alarm)

(b) Alarm Code List for Motion Program

The following table shows the alarm codes of motion programs.

Alarm Code	Name	Description	Corrective Actions
02h	Division error	Data divided by 0	Review the motion program.
10h	A circle instead of radius was specified	Turn number was specified instead of radius in the circular arc or helical interpolation command.	• Designate a center coordinate instead of a radius to perform the circular arc or helical interpolation command. • Never specify the turn number.
11h	Interpolation feeding speed over limit	Interpolation feeding speed exceeded the valid range of the FMX command.	Modify the interpolation feeding speed of the interpolation command
12h	No interpolation feeding speed specified	No interpolation feeding speed was specified. (once specified, this can be omitted as in the motion program)	Specify the interpolation feeding speed in the interpolation command.
13h	Range exceeded after converting acceleration parameter	Indirect acceleration parameter exceeded the valid range.	Change the indirect register value.
14h	Circular arc length exceeded LONG_MAX	Circular arc length exceeded the valid range in the circular arc or helical interpolation command.	Review the circular arc length in the circular arc or helical interpolation command.
15h	Vertical axis not specified for circular arc plane	Vertical axis was not specified in the circular arc or helical interpolation command.	Use PLN command to specify the axis.
16h	Horizontal axis not specified for circular arc plane	Horizontal axis was not specified in the circular arc or helical interpolation command.	Use PLN command to specify the axis.
17h	Specified axis over limit	Too many axes were configured in the circular arc (two axes) or helical (three axes) interpolation command.	Modify the axis in the circular arc or helical interpolation command.
18h	Turn number over limit	Turn number exceeded the valid range in the circular arc or helical interpolation command.	Modify the turn number in the circular arc or helical interpolation command.
19h	Radius exceeded LONG_MAX	Radius exceeded the valid range in the circular arc or helical interpolation command.	Review the radius in the circular arc or helical interpolation command.
1Ah	Center point error	Improper center point was specified in the circular arc or helical interpolation command.	Specify the center point properly in the circular arc or helical interpolation command.
1Bh	Running emergency stop command	Axis move command stopped due to a program stop request.	Turn OFF the program stop request for the motion program control signal, and turn ON the alarm reset request.
1Ch	Linear interpolation moving amount exceeded LONG_MAX	Moving amount exceeded the valid range in the linear interpolation command.	Review the moving amount in the linear interpolation command.
1Dh	FMX undefined	FMX command not executed in the motion program containing an interpolation command.	Execute an FMX command. The FMX command is required in each program containing an interpolation command.
1Eh	Address T out of range	Designation exceeded the valid range in the IAC/IDC/FMX commands.	Review the setting in the IAC/IDC/FMX command.
1Fh	Address P out of range	Designation exceeded the valid range in the IFP command.	Review the setting in the IFP command.

(cont'd)

Alarm Code	Name	Description	Corrective Actions
21h	PFORK execution error	A motion command was instructed simultaneously at the second line in the PFORK of both a source motion program and a subprogram.	Review the source motion program or subprogram.
22h	Indirect register range error	Specified register address exceeds the register size range.	Review the motion program.
23h	Moving amount out of range	Axis moving amount with decimal point for an axis move command exceeded the possible range.	Review the axis moving amount.
80h	Use of logical axis prohibited	Multiple motion commands instructed against the same axis at the same time.	Review the motion program.
81h	Designation exceeded POSMAX in the infinite length axis	Moving distance designation exceeded POSMAX in the infinite length axis.	• Modify a fixed parameter "Maximum infinite length axis counter" • Review the motion program.
82h	Axis moving distance exceeded LONG_MAX	Axis moving distance designation exceeded the valid range.	Review the motion program.
84h	Duplicated motion command	Multiple commands were executed against a single axis.	Check whether another program gave a command to the same axis at the same time. If so, review the program.
85h	Motion command response error	A motion command response different from that instructed by the motion command is reported from a motion module.	• Remove the alarm cause from the destination axis. • If the servo is not turned ON, turn ON the servo. • Check whether another program gave a command to the same axis at the same time. If so, review the program.
87h	VEL setting data out of range	An instruction in the VEL command exceeded the valid range.	Review the VEL command.
88h	INP setting data out of range	An instruction in the INP command exceeded the valid range.	Review the INP command.
89h	ACC/SCC/DCC setting data out of range	An instruction in the ACC/SCC/DCC command exceeded the valid range.	Review the ACC/SCC/DCC command.
8Ah	No time specified in the MVT command	T designation in the MVT command was zero.	Review the MVT command.
8Bh	Command execution disabled	A motion command which cannot be executed by the destination motion module was instructed.	Review the motion program.
8Ch	Distribution incompleted	A motion command was executed when a motion module was not in the Distribution Completed state.	Review the motion program so that a motion command is executed in the Distribution Completed state.
8Dh	Motion command abnormally aborted	Motion module fell into the "Motion command abnormally aborted" state.	• Release the destination axis error. • Review the motion program.

8.2.6 List of Causes for Command Error Completed Status

The Command Error Completed Status (IW□□09, bit 3) turns ON when the set motion command cannot be executed for some reasons or the execution of motion command ended with error. The cause for which this bit turns ON differ depending on motion command.
The following table shows the causes of Command Error Completed Status by motion command.

Motion Command Code		Cause of Command Error Occurrence	Warning (W:) and Alarm (A:) That Occur at Command Error Occurrence
1	Positioning (POSING)	The positioning moving amount exceeds the allowable range.	A: Excessive Positioning Moving Amount
		The axis is a ABS infinite-length, and the zero point return setting is not completed	A: Zero Point Not Set
		In servo OFF status	A: Servo OFF
		Alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
2	External Positioning (EX_POSING)	The positioning moving amount exceeds the allowable range.	A: Excessive Positioning Moving Amount
		The axis is a ABS infinite-length, and the zero point return setting is not completed	A: Zero Point Not Set
		In servo OFF status	A: Servo OFF
		Alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
		SERVOPACK parameter writing was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Warning A.94 or A.95 occurred in the SERVOPACK.	W: Servo Driver Error
		The selected external signal is out of the setting range.	W: Setting Parameter Error
3	Zero Point Return (ZRET)	In machine lock status	–
		In servo OFF status	A: Servo OFF
		An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
		SERVOPACK parameter reading or writing was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Warning A.94 or A.95 is occurring in the SERVOPACK.	W: Servo Driver Error
		The selected zero point return method is out of the setting range.	W: Setting Parameter Error
		POT method is selected for zero point return, but the approach speed is a negative value.	W: Setting Parameter Error
		NOT method is selected for zero point return, but the approach speed is a positive value.	W: Setting Parameter Error
		During zero point return using DEC1 + Phase-C, ZERO signal, or Phase-C method, the OT signal in zero point return direction was ON.	OT Alarm or OT Warning in Zero Point Return Direction

(cont'd)

Motion Command Code		Cause of Command Error Occurrence	Warning (W:) and Alarm (A:) That Occur at Command Error Occurrence
4 ・ 5	Interpolation (INTERPOLATE) Interpolation last segment (ENDOF_INTERPOLATE)	The commanded moving distance for one scan exceeds the segment that can be commanded to the MECHATROLINK SERVOPACK, or the speed feed forward value exceeds the allowable maximum speed.	A: Excessive Speed
		The axis is ABS infinite length, and the zero point return (setting) is not completed.	A: Zero Point Not Set
		In servo OFF status	A: Servo OFF
		An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
6	Latch (LATCH)	The commanded moving amount for one scan exceeds the segment that can be commanded to the MECHATROLINK SERVOPACK, or the speed feed forward value exceeds the allowable maximum speed.	A: Excessive Speed
		The axis is ABS infinite length, and the zero point return (setting) is not completed.	A: Zero Point Not Set
		In servo OFF status	A: Servo OFF
		An alarm is occurring.	–
		The selected latch signal is out of the setting range.	W: Setting Parameter Error
7	JOG Operation (FEED)	In machine lock status	–
		In servo OFF status	A: Servo OFF
		An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
8	STEP operation (STEP)	Positioning moving amount exceeds the allowable value.	A: Excessive Positioning Moving Amount
		In servo OFF status	A: Servo OFF
		An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
9	Zero Point setting (ZSET)	An alarm is occurring.	–
		Asynchronized communication status	A: Servo Driver Synchronization Communication Error
10 ・ 11	Change Acceleration Time (ACC) Change Deceleration Time (DCC)	An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
		Executed while the distribution has not been completed (DEN = OFF)	–
		SERVOPACK parameter writing was not completed within the specified time.	A: Servo Command Timeout Error
		Warning A.94 or A.95 occurred in the SERVOPACK.	W: Servo Driver Error
12	Change Filter Time Constant (SCC)	An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
		Executed while the distribution has not been completed (DEN = OFF)	A: Filter Time Constant Change Error
		SERVOPACK parameter writing was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Warning A.94 or A.95 occurred in the SERVOPACK.	W: Servo Driver Error

(cont'd)

Motion Command Code		Cause of Command Error Occurrence	Warning (W:) and Alarm (A:) That Occur at Command Error Occurrence
13	Change Filter Type (CHG_FILTER)	An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
		Executed while the distribution has not been completed (DEN = OFF).	A: Filter Time Constant Change Error
		The selected filter type is out of the setting range.	W: Setting Parameter Error
14 • 15 • 16	Change Speed Loop Gain (KVS) Change Position Loop Gain (KPS) Change Speed Feed Forward (KFS)	An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
		SERVOPACK parameter writing was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Warning A.94 or A.95 occurred in the SERVOPACK.	W: Servo Driver Error
17 • 18	Read SERVOPACK Parameter (PRM_RD) Write SERVOPACK Parameter (PRM_WR)	An alarm is occurring.	–
		Asynchronized communication status	A: Servo Driver Synchronization Communication Error
		SERVOPACK parameter reading was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Warning A.94 or A.95 occurred in the SERVOPACK.	W: Servo Driver Error
		SERVOPACK parameter number or size is out of the setting range.	W: Setting Parameter Error
19 • 20	Monitor SERVOPACK Alarms (ALM_MON) Monitor SERVOPACK Alarm History (ALM_HIST)	The command to the SERVOPACK was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Servo driver alarm monitor number is out of setting range.	W: Setting Parameter Error
21	Clear SERVOPACK Alarm History (ALMHIST_CLR)	The command to the SERVOPACK was not completed within the specified time.	A: Servo Driver Command Timeout Error
22	Reset Absolute Encoder (ABS_RST)	This command was used for Σ-I SERVOPACK.	–
		Executed while servo is ON.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
		The command to the SERVOPACK was not completed within the specified time.	A: Servo Driver Command Timeout Error
23	Speed Reference (VELO)	Commanded when having been connected to MECHATROLINK-I	–
		An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
24	Torque Reference (TRQ)	Commanded when having been connected to MECHATROLINK-I	–
		An alarm is occurring	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
25	Phase Reference (PHASE)	The axis is ABS infinite length, and the zero point return (setting) is not completed.	A: Zero Point Not Set
		In servo OFF status	A: Servo OFF
		An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error

(cont'd)

Motion Command Code		Cause of Command Error Occurrence	Warning (W:) and Alarm (A:) That Occur at Command Error Occurrence
26	Change Position Loop Integration Time Constant (KIS)	An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
		SERVOPACK parameter writing was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Warning A.94 or A.95 occurred in the SERVOPACK.	W: Servo Driver Error
	Others Parameter Automatic Updating when Execution of Move Command Starts*	An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communication Error
		SERVOPACK parameter writing was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Warning A.94 or A.95 occurred in the SERVOPACK.	W: Servo Driver Error
		The distribution was not completed (DEN = OFF).	–

* When the fixed parameter Automatic Updating of Parameter was enabled, and the setting of Filter Time Constant, Acceleration Rate/Time, or Deceleration Rate/Time was changed at the time a move command was set

8.2.7 Troubleshooting Motion Errors

Motion errors in the MP2000-series Machine Controller include axis alarms detected for individual SERVOPACKs. The failure location can be determined and appropriate corrections can be taken simply by checking the contents of the Warning (IL□□02) and Alarm (IL□□04) monitoring parameters.

Refer to *12.5 Troubleshooting Motion Errors* in the *Machine Controller MP2000 Series Built-in SVB/SVB-01 Motion Module User's Manual* (manual no.: SIEP C880700 33) for details.

Appendices

Appendix A System Registers Lists

A.1 System Service Registers

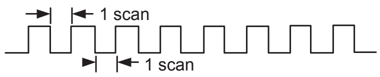
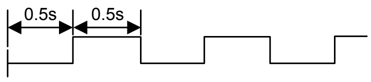
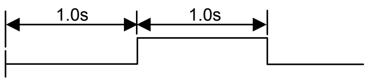
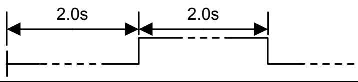
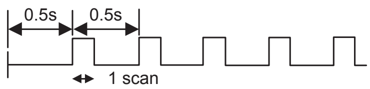
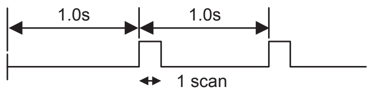
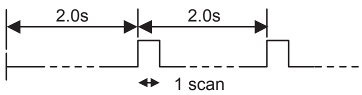
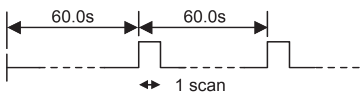
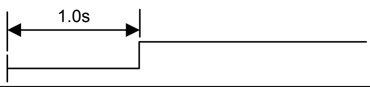
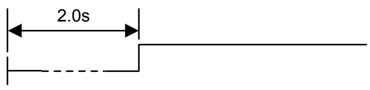
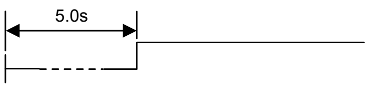
(1) Shared by All Drawings

Name	Register No.	Remarks
Reserved by the system	SB000000	(Not used)
High-speed Scan	SB000001	ON for only one scan after high-speed scan is started after turning ON the power supply.
Reserved by the system	SB000002	(Not used)
Low-speed Scan	SB000003	ON for only one scan after low-speed scan is started after turning ON the power supply.
Always ON	SB000004	Always ON (= 1)
High-speed scan 2	SB000005	ON for only one scan after the start of a high-speed scan that is begun after a CPU Module operation starts.*
Low-speed scan 2	SB000006	ON for only one scan after the start of a low-speed scan that is begun after a CPU Module operation starts.*
High-speed Scan Flag	SB000007	ON during execution of the high-speed scan.
Reserved by the system	SB000008 to SB00000F	(Not used)

* Does not include when MPE720 version 2.75 or earlier is used to execute a batch load or to clear the memory before starting the CPU Module.

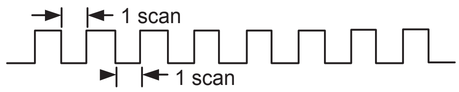
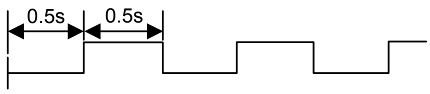
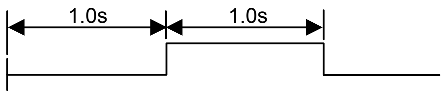
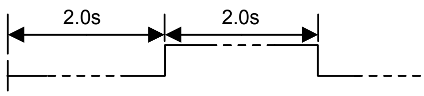
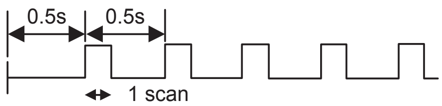
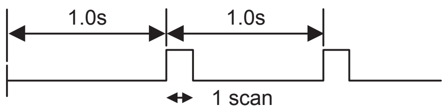
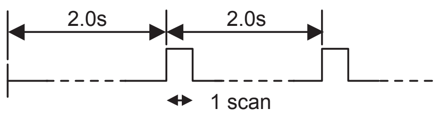
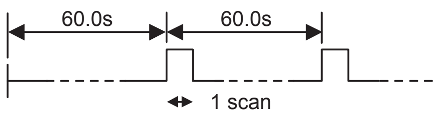
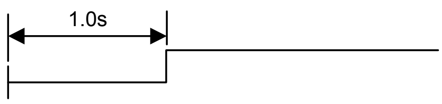
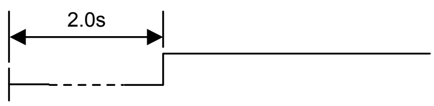
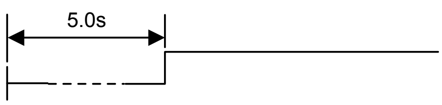
(2) DWG.H Only

The following relays are reset at the start of the high-speed scan.

Name	Register No.	Remarks
1-scan Flicker Relay	SB000010	
0.5-s Flicker Relay	SB000011	
1.0-s Flicker Relay	SB000012	
2.0-s Flicker Relay	SB000013	
0.5-s Sampling Relay	SB000014	
1.0-s Sampling Relay	SB000015	
2.0-s Sampling Relay	SB000016	
60.0-s Sampling Relay	SB000017	
1.0 s After Start of Scan Relay	SB000018	
2.0 s After Start of Scan Relay	SB000019	
5.0 s After Start of Scan Relay	SB00001A	

■ DWG.L Only

The following relays are reset at the start of the low-speed scan.

Name	Register No.	Remarks
One-scan Flicker Relay	SB000030	
0.5-s Flicker Relay	SB000031	
1.0-s Flicker Relay	SB000032	
2.0-s Flicker Relay	SB000033	
0.5-s Sampling Relay	SB000034	
1.0-s Sampling Relay	SB000035	
2.0-s Sampling Relay	SB000036	
60.0-s Sampling Relay	SB000037	
1.0 s After Start of Scan Relay	SB000038	
2.0 s After Start of Scan Relay	SB000039	
5.0 s After Start of Scan Relay	SB00003A	

Appendices

A

A.2 Scan Execution Status and Calendar

Name	Register No.	Remarks
High-speed Scan Set Value	SW00004	High-speed Scan Set Value (0.1 ms)
High-speed Scan Current Value	SW00005	High-speed Scan Current Value (0.1 ms)
High-speed Scan Maximum Value	SW00006	High-speed Scan Maximum Value (0.1 ms)
Reserved by the system.	SW00007 to SW00009	(Not used)
Low-speed Scan Set Value	SW00010	Low-speed Scan Set Value (0.1 ms)
Low-speed Scan Current Value	SW00011	Low-speed Scan Current Value (0.1 ms)
Low-speed Scan Maximum Value	SW00012	Low-speed Scan Maximum Value (0.1 ms)
Reserved by the system.	SW00013	(Not used)
Executing Scan Current Value	SW00014	Executing Scan Current Value (0.1 ms)
Calendar: Year	SW00015	1999: 0099 (BCD) (Last two digits only)
Calendar: Month Day	SW00016	December 31: 1231 (BCD)
Calendar: Hours Minutes	SW00017	23 hours 59 minutes: 2359 (BCD)
Calendar: Seconds	SW00018	59 s: 59 (BCD)
Calendar: Day of Week	SW00019	0 to 6: Sun., Mon. to Sat.

A.3 Program Software Numbers and Remaining Program Memory Capacity

Name	Register No.	Remarks
System Program Software Number	SW00020	S□□□□ (□□□□ is stored as BCD)
Reserved by the system.	SW00021 to SW00025	(Not used)
Remaining Program Memory Capacity	SL00026	Bytes
Total Memory Capacity	SL00028	Bytes

Appendix B SVR Motion Parameter Details

B.1 Fixed Parameter List

The following table lists the SVR motion fixed parameters. (Use this list as reference for the SVB.)

Slot Number	Name	Contents	SVB	SVR
0	Selection of Operation Modes	0: Normal Operation Mode	Yes	Yes
		1: Axis Unused	Yes	Yes
		2: Simulation Mode	Yes	
		3: Servo Driver Transmission Reference Mode	Yes	
		4 and 5: Reserved	–	–
1	Function Selection Flag 1	Bit 0: Axis Selection (0: Finite length axis/1: Infinite length axis) • Set to 0 for linear type.	Yes	Yes
		Bit 1: Soft Limit (Positive Direction) Enable/Disable	Yes	
		Bit 2: Soft Limit (Negative Direction) Enable/Disable	Yes	
		Bit 3: Overtravel Positive Direction Enable/Disable	Yes	
		Bit 4: Overtravel Negative Direction Enable/Disable	Yes	
		Bits 5 to 7: Reserved	–	–
		Bit 8: Interpolation Segment Distribution Processing	Yes	
		Bit 9: Simple ABS Rotary Pos. Mode (Simple Absolute Infinite Axis Position Control) (0: Disabled/1: Enabled) • Set to 0 for linear type.	Yes	
		Bit A: User Constants Self-writing Function	Yes	
		Bits B to F: Reserved		
2	Function Selection Flag 2	Bit 0: Communication Abnormality Detection Mask	Yes	
		Bit 1: WDT Abnormality Detection Mask	Yes	
		Bits 2 to F: Reserved for system use.	–	–
3	–	Reserved	–	–
4	Reference Unit Selection	0: pulse, 1: mm, 2: deg, 3: inch, 4:µm • For linear type, only valid for 0: pulse, 1: mm, 4: µm. When 2: deg, 3: inch is set, converted into 1: mm.	Yes	Yes
5	Number of Digits below Decimal Places	1 = 1 digit	Yes	Yes
6	Travel Distance per Machine Rotation (Rotary Motor)	1 = 1 reference unit	Yes	Yes
	Linear Scale Pitch (Linear Type)	1 = 1 reference unit	Yes	Yes
8	Servo Motor Gear Ratio	1 = 1 rotation • Invalid for linear type.	Yes	Yes
9	Machine Gear Ratio	1 = 1 rotation • Invalid for linear type.	Yes	Yes
10	Infinite Length Axis Reset Position (POSMAX)	1 = 1 reference units • Invalid for linear type.	Yes	Yes
12	Positive Software Limit Value	1 = 1 reference unit	Yes	
14	Negative Software Limit Value	1 = 1 reference unit	Yes	
16	Backlash Compensation Amount	1 = 1 reference unit	Yes	
18 to 29	–	Reserved	–	–
30	Encoder Selection	0: Incremental Encoder 1: Absolute Encoder 2: Absolute Encoder (Incremental encoder is used.) 3: Reserved	Yes	
31 to 33	–	Reserved	–	–

(cont'd)

Slot Number	Name	Contents	SVB	SVR
34	Rated Motor Speed (Rotary Motor)	1 = 1 min^{-1}	Yes	Yes
	Rated Speed (Linear Type)	1 = 0.1 m/s, 0.1 mm/s	Yes	Yes
36	Number of Pulses per Motor Rotation (Rotary Motor)	1 = 1 pulse/rev Set the value after multiplication.	Yes	Yes
	Number of Pulses per Linear Scale Pitch (Linear Type)	1 = 1 pulse/scale pitch	Yes	Yes
38	Maximum Number of Absolute Encoder Turns Rotation	1 = 1 rotation • Set to 0 when a direct drive motor is being used. • Invalid for linear type.	Yes	
40 to 41	–	Reserved	–	–
42	Feedback Speed Movement Averaging Time Constant	1 = 1 ms	Yes	Yes

B.2 Setting Parameter List

The following table lists the SVR motion setting parameters. (Use this list as reference for the SVB.)

<table>
<tr><th>Register No.</th><th>Name</th><th>Contents</th><th>SVB</th><th>SVR</th></tr>
<tr><td rowspan="14">OW□□00</td><td rowspan="14">RUN Command Setting</td><td>Bit 0: Servo ON (0: OFF/1: ON)</td><td>Yes</td><td>Yes</td></tr>
<tr><td>Bit 1: Machine Lock (0: Normal operation/1: Machine locked)</td><td>Yes</td><td></td></tr>
<tr><td>Bits 2 to 3: Reserved</td><td></td><td></td></tr>
<tr><td>Bit 4: Latch Detection Demand (0: OFF/1: ON)</td><td>Yes</td><td></td></tr>
<tr><td>Bit 5: Reserved for system use.</td><td></td><td></td></tr>
<tr><td>Bit 6: POSMAX Turn Number Presetting Demand
(0: OFF/1: ON)
• Set to 0 for linear type.</td><td>Yes</td><td>Yes</td></tr>
<tr><td>Bit 7: Request ABS Rotary Pos. Load (Absolute System Infinite Length Position Information LOAD) (0: OFF/1: ON)
• Set to 0 for linear type.</td><td>Yes</td><td></td></tr>
<tr><td>Bit 8: Forward Outside Limiting Torque/Thrust Input
(Forward External Limiting Torque/Thrust Input) (0: OFF/1: ON)</td><td>Yes</td><td></td></tr>
<tr><td>Bit 9: Reverse Outside Limiting Torque/Thrust Input
(Reverse External Limiting Torque/Thrust Input) (0: OFF/1: ON)</td><td>Yes</td><td></td></tr>
<tr><td>Bit A: Reserved</td><td></td><td></td></tr>
<tr><td>Bit B: Integration Reset (0: OFF/1: ON)</td><td>Yes</td><td></td></tr>
<tr><td>Bits C to D: Reserved</td><td></td><td></td></tr>
<tr><td>Bit E: Communication Reset (0: OFF/1: ON)</td><td>Yes</td><td></td></tr>
<tr><td>Bit F: Alarm Clear (0: OFF/1: ON)</td><td>Yes</td><td>Yes</td></tr>
<tr><td rowspan="6">OW□□01</td><td rowspan="6">Mode Setting 1</td><td>Bit 0: Excessive Deviation Error Level Setting
(0: Alarm/1: Warning)</td><td>Yes</td><td></td></tr>
<tr><td>Bits 1 to 2: Reserved</td><td></td><td></td></tr>
<tr><td>Bit 3: Speed Loop P/PI Switch</td><td>Yes</td><td></td></tr>
<tr><td>Bit 4: Gain Switch</td><td>Yes</td><td></td></tr>
<tr><td>Bit 5: Gain Switch 2</td><td>Yes</td><td></td></tr>
<tr><td>Bits 6 to F: Reserved</td><td></td><td></td></tr>
<tr><td rowspan="5">OW□□02</td><td rowspan="5">Mode Setting 2</td><td>Bit 0: Monitor 2 Enabled (0: Disabled/1: Enabled)</td><td>Yes</td><td></td></tr>
<tr><td>Bits 1 to 3: Reserved</td><td></td><td></td></tr>
<tr><td>Bits 4: Reserved</td><td></td><td></td></tr>
<tr><td>Bits 5 to 7: Reserved</td><td></td><td></td></tr>
<tr><td>Bits 8 to 15: Stop Mode Selection</td><td>Yes</td><td></td></tr>
<tr><td rowspan="4">OW□□03</td><td rowspan="4">Function Setting 1</td><td>Bits 0 to 3: Speed Unit Selection
0: Reference unit/s
1: 10^n reference unit/min
2: Percentage of rated speed (1 = 0.01%)
3: Percentage of rated speed (1 = 0.0001%)</td><td>Yes</td><td>Yes</td></tr>
<tr><td>Bits 4 to 7: Acceleration/Deceleration Degree Unit Selection
0: Reference units/s^2
1: ms</td><td>Yes</td><td>Yes</td></tr>
<tr><td>Bits 8 to B: Filter Type Selection
0: No filter
1: Exponential acceleration/deceleration filter
2: Moving average filter</td><td>Yes</td><td>Yes</td></tr>
<tr><td>Bits C to F: Torque Unit Selection
0: Percentage of rated toque (1 = 0.01%)
1: Percentage of rated toque (1 = 0.0001%)</td><td>Yes</td><td>Yes</td></tr>
</table>

(cont'd)

Register No.	Name	Contents	SVB	SVR
OW□□04	Function Setting 2	Bits 0 to 3: Latch Detection Signal Selection		
		0: -		
		1: -		
		2: Phase-C Pulse Input Signal	Yes	
		3: /EXT1	Yes	
		4: /EXT2	Yes	
		5: /EXT3	Yes	
		Bits 4 to 7: External Positioning Signal Setting		
		0: –		
		1: –		
		2: Phase-C Pulse Input Signal	Yes	
		3: /EXT1	Yes	
		4: /EXT2	Yes	
		5: /EXT3	Yes	
		Bits 8 to B: Reserved		
		Bits C to F: Bank Selector	Yes	
OW□□05	Function Setting 3	Bit 1: Phase Reference Creation Calculation Disable (0: Enabled/1: Disabled)	Yes	
		Bits 2 to A: Reserved		
		Bit B: Zero Point Return Input Signal (0: OFF/1: ON)	Yes	
		Bits C to F: Reserved		
OW□□06 to OW□□07	–	Reserved	–	–
OW□□08	Motion Command	0: NOP (No Command) 1: POSING (Position Mode)(Positioning) 2: EX_POSING (Latch Target Positioning)(External Positioning) 3: ZRET (Zero Point Return) 4: INTERPOLATE (Interpolation) 5: ENDOF_ INTERPOLATE (Last Interpolation Segment) 6: LATCH (Interpolation Mode with Latch Input) 7: FEED (Jog Mode) 8: STEP (Relative Position Mode)(Step Mode) 9: ZSET (Set Zero Point) 10: ACC (Change Acceleration Time) 11: DCC (Change Deceleration Time) 12: SCC (Change Filter Time Constant) 13: CHG FILTER (Change Filter Type) 14 : KVS (Change Speed Loop Gain) 15 : KPS (Change Position Loop Gain) 16: KFS (Change Feed-Forward) 17: PRM_RD (Read User Constant)(Read SERVOPACK Parameter) 18: PRM_WR (Write User Constant)(Write SERVOPACK Parameter) 19: ALM_MON (Alarm Monitor) 20: ALM_HIST (Alarm History Monitor) 21: ALMHIST_CLR (Clear Alarm History) 22: ABS_RST (Absolute Encoder Reset) 23: VELO (Speed Reference) 24: TRQ (Torque/Thrust Reference) 25: PHASE (Phase Reference) 26: KIS (Change Position Loop Integral Time Constant) 27: PPRM_WR (Stored Parameter Write)	Yes	Yes

(cont'd)

<table>
<tr><th>Register No.</th><th>Name</th><th>Contents</th><th>SVB</th><th>SVR</th></tr>
<tr><td rowspan="8">OW□□09</td><td rowspan="8">Motion Command Control Flag</td><td>Bit 0: Holds a Command. (0: OFF/1: ON)</td><td>Yes</td><td>Yes</td></tr>
<tr><td>Bit 1: Interrupt a Command. (0: OFF/1: ON)</td><td>Yes</td><td>Yes</td></tr>
<tr><td>Bit 2: Moving Direction (JOG/ STEP)
(0: Forward rotation/1: Reverse rotation)</td><td>Yes</td><td>Yes</td></tr>
<tr><td>Bit 3: Zero Point Return Direction Selection
(0: Reverse rotation/1: Forward rotation)</td><td>Yes</td><td></td></tr>
<tr><td>Bit 4: Latch Zone Effective Selection (0: Disabled/1: Enabled)</td><td>Yes</td><td></td></tr>
<tr><td>Bit 5: Position Reference Type
(0: Incremental Addition Mode/1: Absolute Mode)</td><td>Yes</td><td>Yes</td></tr>
<tr><td>Bit 6: Phase Compensation Type
(0: Incremental Addition Mode/1: Absolute Mode)</td><td>Yes</td><td></td></tr>
<tr><td>Bits 7 to F: Reserved</td><td></td><td></td></tr>
<tr><td rowspan="3">OW□□0A</td><td rowspan="3">Motion Subcommand</td><td>0: NOP (No Command)</td><td>Yes</td><td>Yes</td></tr>
<tr><td>1: PRM_RD (Read User Constant)(Read SERVOPACK Parameter)
2: PRM_WR (Write User Constant)(Write SERVOPACK Parameter)
3: Reserved
4: SMON (Status Monitor)</td><td>Yes</td><td></td></tr>
<tr><td>5: FIXPRM_RD (Read Fixed Parameters)</td><td>Yes</td><td>Yes</td></tr>
<tr><td>OW□□0B</td><td>–</td><td>Reserved</td><td></td><td></td></tr>
<tr><td>OL□□0C</td><td>Torque/Thrust Reference Setting</td><td>Unit is according to OW□□03, bits 12 to 15 (Torque Unit).</td><td>Yes</td><td>Yes</td></tr>
<tr><td>OW□□0E</td><td>Speed Limit Setting at the Torque/Thrust Reference</td><td>1 = 0.01% (percentage of rated speed)</td><td>Yes</td><td></td></tr>
<tr><td>OW□□0F</td><td>–</td><td>Reserved</td><td></td><td></td></tr>
<tr><td>OL□□10</td><td>Speed Reference Setting</td><td>Unit is according to OW□□03, bits 0 to 3 (Speed Unit).</td><td>Yes</td><td>Yes</td></tr>
<tr><td>OW□□12
to
OW□□13</td><td>–</td><td>Reserved</td><td>–</td><td>–</td></tr>
<tr><td>OL□□14</td><td>Positive Side Limiting Torque/Thrust Setting at the Speed Reference</td><td>Unit is according to OW□□03, bits C to F (Torque Unit).</td><td>Yes</td><td></td></tr>
<tr><td>OL□□16</td><td>Secondly Speed Compensation</td><td>Unit is according to OW□□03, bits 0 to 3 (Speed Unit).</td><td>Yes</td><td>Yes</td></tr>
<tr><td>OW□□18</td><td>Override</td><td>1 = 0.01%</td><td>Yes</td><td></td></tr>
<tr><td>OW□□19
to
OW□□1B</td><td>–</td><td>Reserved</td><td>–</td><td>–</td></tr>
<tr><td>OL□□1C</td><td>Position Reference Setting</td><td>1 = 1 reference unit</td><td>Yes</td><td>Yes</td></tr>
<tr><td>OL□□1E</td><td>Width of Positioning Completion</td><td>1 = 1 reference unit</td><td>Yes</td><td></td></tr>
<tr><td>OL□□20</td><td>NEAR Signal Output Width</td><td>1 = 1 reference unit</td><td>Yes</td><td></td></tr>
<tr><td>OL□□22</td><td>Error Count Alarm Detection</td><td>1 = 1 reference unit</td><td>Yes</td><td></td></tr>
<tr><td>OL□□24</td><td>–</td><td>Reserved for system use.</td><td>–</td><td>–</td></tr>
<tr><td>OW□□26</td><td>Position Complete Cheek Time</td><td>1 = 1 ms</td><td>Yes</td><td></td></tr>
<tr><td>OW□□27</td><td>–</td><td>Reserved for system use.</td><td></td><td></td></tr>
<tr><td>OL□□28</td><td>Phase Correction Setting</td><td>1 = 1 reference unit</td><td>Yes</td><td></td></tr>
<tr><td>OL□□2A</td><td>Latch Zone Lower Limit Setting</td><td>1 = 1 reference unit</td><td>Yes</td><td></td></tr>
</table>

(cont'd)

Register No.	Name	Contents	SVB	SVR
OL□□2C	Latch Zone Upper Limit Setting	1 = 1 reference unit	Yes	
OW□□2E	Position Loop Gain	1 = 0.1/s	Yes	
OW□□2F	Speed Loop Gain	1 = 1 Hz	Yes	
OW□□30	Speed Feedforward Amends	1 = 0.01% (percentage of distribution segment)	Yes	
OW□□31	Speed Compensation	1 = 0.01% (percentage of rated speed)	Yes	Yes
OW□□32	Position Integration Time Constant	1 = 1 ms	Yes	
OW□□33	–	Reserved	–	–
OW□□34	Speed Integration Time Constant	1 = 0.01 ms	Yes	
OW□□35	–	Reserved	–	–
OL□□36	Straight Line Acceleration/ Acceleration Time Constant	Unit is according to OW□□03, bits 4 to 7 (Speed Unit).	Yes	Yes
OL□□38	Straight Line Deceleration/ Deceleration Time Constant	Unit is according to OW□□03, bits 4 to 7 (Speed Unit).	Yes	Yes
OW□□3A	Filter Time Constant	1 = 0.1 ms	Yes	Yes
OW□□3B	Bias Speed for Index Deceleration/Acceleration Filter	Unit is according to OW□□03, bits 0 to 3 (Speed Unit).		Yes
OW□□3C	Zero Point Return Method	0: DEC1 + C (DEC 1 and C-Phase) 1: ZERO (Zero signal) 2: DEC1 + ZERO (DEC 1 and zero signal) 3: C (C-pulse)	Yes	
		4 to 10: Reserved	–	–
		11: C Pulse 12: POT & C Pulse 13: POT Only 14: HOME LS & C Pulse 15: HOME Only	Yes	
		16: NOT & C Pulse 17: NOT Only 18: INPUT & C Pulse 19: INPUT Only	Yes	
OW□□3D	Width of Starting Point Position Output	1 = 1 reference unit	Yes	Yes
OL□□3E	Approach Speed	Unit is according to OW□□03, bits 0 to 3 (Speed Unit).	Yes	
OL□□40	Creep Rate	Unit is according to OW□□03, bits 0 to 3 (Speed Unit).	Yes	
OL□□42	Zero Point Return Travel Distance	1 = 1 reference unit	Yes	
OL□□44	STEP Travel Distance	1 = 1 reference unit	Yes	Yes
OL□□46	External Positioning Final Travel Distance	1 = 1 reference unit	Yes	
OL□□48	Zero Point Position in Machine Coordinate System Offset	1 = 1 reference unit	Yes	Yes
OL□□4A	Work Coordinate System Offset	1 = 1 reference unit	Yes	Yes
OL□□4C	Number of POSMAX Turns Presetting Data	1 = 1 reference unit • Invalid for linear type.	Yes	Yes

(cont'd)

Register No.	Name	Contents	SVB	SVR
OW☐☐4E	Servo User Monitor Setting	Bits 0 to 3: Monitor 1 (Setting impossible) Bits 4 to 7: Monitor 2 Bits 8 to B: Monitor 3 (Setting impossible) Bits C to F: Monitor 4	Yes	
OW☐☐4F	Servo Driver Alarm Monitor No.	Set the number of the alarm to monitor.	Yes	
OW☐☐50	Servo Driver User Constant No.	Set the number of the SERVOPACK parameter.	Yes	
OW☐☐51	Servo Driver User Constant Number Size	Set the number of words in the SERVOPACK parameter.	Yes	
OL☐☐52	Servo Driver User Constant Set Point	Set the setting for the SERVOPACK parameter.	Yes	
OW☐☐54	Servo Driver for Assistance User Constant No.	Set the number of the SERVOPACK parameter number.	Yes	
OW☐☐55	Servo Driver for Assistance User Constant Size	Set the number of words in the SERVOPACK parameter.	Yes	
OL☐☐56	Servo Driver for Assistance User Constant Set Point	Set the setting for the SERVOPACK parameter.	Yes	
OW☐☐58 to OW☐☐5B	–	Reserved	–	–
OW☐☐5C	Fixed Parameter Number	Set the number of the fixed parameter to read with the FIXPRM_RD motion subcommand.	Yes	Yes
OW☐☐5D	–	Reserved	–	–
OL☐☐5E	Encoder Position When Power is OFF (Lower 2 Words)	1 = 1 pulse • Do not set in the linear type.	Yes	
OL☐☐60	Encoder Position When Power is OFF (Upper 2 Words)	1 = 1 pulse • Do not set in the linear type.	Yes	
OL☐☐62	Pulse Position When Power is OFF (Lower 2 Words)	1 = 1 pulse • Do not set in the linear type.	Yes	
OL☐☐64	Pulse Position When Power is OFF (Upper 2 Words)	1 = 1 pulse • Do not set in the linear type.	Yes	
OL☐☐66 to OL☐☐6E	–	Reserved	–	–
OW☐☐70 to OW☐☐7F	Command Buffer for Transparent Command Mode	This area is used for command data when MECHATROLINK servo commands are specified directly.	Yes	

B.3 Monitoring Parameter List

The following table lists the SVR motion monitor parameters. (Use this list as reference for the SVB.)

Register No.	Name	Contents	SVB	SVR
IW□□00	RUN Status	Bit 0 Motion Controller Operation Ready	Yes	Yes
		Bit 1: Running (At Servo ON)	Yes	Yes
		Bit 2: System Busy	Yes	
		Bit 3: Servo Ready	Yes	
		Bit 4: Latch Mode	Yes	
		Bits 5 to F: Reserved	–	–
IW□□01	Parameter Number When Range Over is Generated	Setting parameters: 0 or higher Fixed Parameters: 1000 or higher	Yes	Yes
IL□□02	Warning	Bit 0: Excessive Deviation	Yes	
		Bit 1: Set Parameter Error (Setting Parameter Error)	Yes	Yes
		Bit 2: Fixed Parameter Error	Yes	Yes
		Bit 3: Servo Driver Error	Yes	
		Bit 4: Motion Command Set Error	Yes	Yes
		Bit 5: Reserved (AD Conversion Error)	–	–
		Bit 6: Positive Direction Overtravel	Yes	
		Bit 7: Negative Direction Overtravel	Yes	
		Bit 8: Servo ON Incomplete	Yes	
		Bit 9: Servo Driver Communication Warning	Yes	
		Bits A to 1F: Reserved		
IL□□04	Alarm	Bit 0: Servo Driver Error	Yes	
		Bit 1: Positive Direction Overtravel	Yes	
		Bit 2: Negative Direction Overtravel	Yes	
		Bit 3: Positive Direction Software Limit	Yes	
		Bit 4: Negative Direction Software Limit	Yes	
		Bit 5: Servo OFF	Yes	Yes
		Bit 6: Positioning Time Over	Yes	
		Bit 7: Excessive Positioning Moving Amount	Yes	
		Bit 8: Excessive Speed	Yes	
		Bit 9: Excessive Deviation	Yes	
		Bit A: Filter Type Change Error	Yes	
		Bit B: Filter Time Constant Change Error	Yes	
		Bit C: Reserved	–	–
		Bit D: Zero Point Unsetting • Invalid for linear type.	Yes	
		Bit E: Reserved	–	–
		Bit F: Reserved	–	–
		Bit 10: Servo Driver Synchronization Communications Error	Yes	
		Bit 11: Servo Driver Communication Error	Yes	
		Bit 12: Servo Driver Command Time-out Error	Yes	
		Bit 13: Excessive ABS Encoder Rotations • Invalid for linear type.	Yes	
		Bits 14 to 1D: Reserved	–	–
		Bit1E: Motor Type Set Error	Yes	–
		Bit1F: Connected Encoder Type Error	Yes	–
IL□□06	–	Reserved	–	–
IW□□08	Motion Command Response Code	Same as OW□□08 (Motion Command).	Yes	Yes

(cont'd)

Register No.	Name	Contents	SVB	SVR
IW☐☐09	Motion Command Status	Bit 0: Command Execution Flag	Yes	Yes
		Bit 1: Command Hold Completed	Yes	Yes
		Bit 2: Reserved	–	–
		Bit 3: Command Error Completed Status (Command Error Occurrence)	Yes	Yes
		Bits 4 to 6: Reserved	–	–
		Bit 7: Reset Absolute Encoder Completed	Yes	
		Bit 8: Command Execution Completed	Yes	Yes
		Bits 9 to F: Reserved	–	–
IW☐☐0A	Subcommand Response Code	Same as OWxx0A (Motion Subcommand).	Yes	Yes
IW☐☐0B	Subcommand Status	Bit 0: Command Execution Flag	Yes	Yes
		Bits 1 to 2: Reserved	–	–
		Bit 3: Command Error Completed Status (Command Error Occurrence)	Yes	Yes
		Bits 4 to 7: Reserved	–	–
		Bit 8: Command Execution Completed	Yes	Yes
		Bits 9 to F: Reserved	–	–
IW☐☐0C	Position Management Status	Bit 0: Distribution Completed	Yes	Yes
		Bit 1: Positioning Completed	Yes	Yes
		Bit 2: Latch Completed	Yes	
		Bit 3: NEAR Position	Yes	Yes
		Bit 4: Zero Point Position	Yes	Yes
		Bit 5: Zero Point Return (Setting) Completed	Yes	Yes
		Bit 6: During Machine Lock	Yes	
		Bit 7: Reserved	–	–
		Bit 8: ABS Rotary Pos. Load Complete (ABS System Infinite Length Position Control Information Load Completed) • Invalid for linear type.	Yes	
		Bit 9: POSMAX Turn Preset Complete (TPRSE) • Invalid for linear type.	Yes	Yes
		Bits A to F: Reserved		
IW☐☐0D	–	Reserved	–	–
IL☐☐0E	Target Position in Machine Coordinate System (TPOS)	1 = 1 reference unit	Yes	Yes
IL☐☐10	Calculated Position in Machine Coordinate system (CPOS)	1 = 1 reference unit	Yes	Yes
IL☐☐12	Machine Coordinate System Reference Position (MPOS)	1 = 1 reference unit	Yes	Yes
IL☐☐14	CPOS for 32 bit	1 = 1 reference unit	Yes	Yes
IL☐☐16	Machine Coordinate System Feedback Position (APOS)	1 = 1 reference unit	Yes	Yes
IL☐☐18	Machine Coordinate System Latch Position (LPOS)	1 = 1 reference unit	Yes	
IL☐☐1A	Position Error (PERR)	1 = 1 reference unit	Yes	
IL☐☐1C	Target Position Difference Monitor	1 = 1 reference unit		Yes
IL☐☐1E	Number of POSMAX Turns	1 = 1 turn • Invalid for linear type.	Yes	Yes

(cont'd)

Register No.	Name	Contents	SVB	SVR
IL□□20	Speed Reference Output Monitor	pulse/s	Yes	
IL□□22 to IL□□2A	–	Reserved	–	–
IW□□2C	Servo Driver Status	Bit 0: Alarm (ALM) Bit 1: Warning (WARNG) Bit 2: Command Ready (CMDRDY) Bit 3: Servo ON (SVON) Bit 4: Main Power Supply ON (PON) Bit 5: Machine Lock (MLOCK) Bit 6: Zero Position (ZPOINT) Bit 7: Locating Completed (Positioning Completed) (PSET) Velocity Complete (V-CMP) Bit 8: Command Profile Complete (Distribution Completed) (DEN) Bit 9: Torque Restriction (T_LIM) Bit A: Latch Complete (L_CMP) Bit B: Locating Neighborhood (NEAR Position) (NEAR) / Velocity Limit (V-LIM) Bit C: Positive Software Limit (P_SOT) Bit D: Negative Software Limit (N_SOT)	Yes	
		Bits E to F: Reserved	–	–
IW□□2D	Servo Driver Alarm Code	Stores the alarm code from the SERVOPACK.	Yes	
IW□□2E	Servo Driver I/O Monitor	Bit 0: Forward Side Limit Switch Input (P_OT) Bit 1: Reverse Side Limit Switch Input (N_OT) Bit 2: Deceleration Dog Switch Input (DEC) Bit 3: Encoder Phase-A Signal Input (PA) Bit 4: Encoder Phase-B Signal Input (PB) Bit 5: Encoder Phase-C Signal Input (PC) Bit 6: EXT1 Signal Input Bit 7: EXT2 Signal Input Bit 8: EXT3 Signal Input Bit 9: Brake State Output (BRK) Bit A: Reserved Bit B: Reserved Bit C: CN1 Input Signal (IO12) Bit D: CN1 Input Signal (IO13) Bit E: CN1 Input Signal (IO14) Bit F: CN1 Input Signal (IO15)	Yes	
IW□□2F	Servo Driver User Monitor Information	Bits 0 to 3: Monitor 1 Bits 4 to 7: Monitor 2 Bits 8 to B: Monitor 3 Bits C to F: Monitor 4	Yes	
IL□□30	Servo Driver User Monitor 2	Stores the result of the selected monitor.	Yes	
IL□□32	Servo Driver User Monitor 3	Reserved		
IL□□34	Servo Driver User Monitor 4	Stores the result of the selected monitor.	Yes	
IW□□36	Servo Driver User Constant No. (SERVOPACK Parameter No. for MECHATROLINK Command Area)	Stores the number of the parameter being processed.	Yes	
IW□□37	Supplementary Servo Driver User Constant No. (SERVOPACK Parameter No. for MECHATROLINK Subcommand Area)	Stores the number of the parameter being processed.	Yes	

(cont'd)

Register No.	Name	Contents	SVB	SVR
IL□□38	Servo Driver User Constant Reading Data (SERVOPACK Parameter Reading Data for MECHATROLINK Command Area)	Stores the data of the parameter being read.	Yes	
IL□□3A	Supplementary Servo Driver User Constant Reading Data (SERVOPACK Parameter Reading Data for MECHATROLINK Subcommand Area)	Stores the data of the parameter being read.	Yes	
IW□□3F	Motor Type	Stores the type of motor actually connected. 0: Rotation type motor 1: Linear motor	Yes	
IL□□40	Feedback Speed	Unit is according to OW□□03, bits 0 to 3 (Speed Unit).	Yes	Yes
IL□□42	Feedback Torque/Thrust	Unit is according to OW□□03, bits 12 to 15 (Torque Unit).	Yes	Yes
IW□□44 to IW□□55	–	Reserved	–	–
IL□□56	Fixed Parameter Monitor	Stores the data of the fixed parameter when FIXPRM-RD has been specified in the Motion Subcommand.	Yes	Yes
IW□□58 to IW□□5C	–	Reserved	–	–
IL□□5E	Encoder Position When the Power is OFF (Lower 2 Words)	1 = 1 pulse	Yes	
IL□□60	Encoder Position When the Power is OFF (Upper 2 Words)	1 = 1 pulse	Yes	
IL□□62	Pulse Position When the Power is OFF (Lower 2 Words)	1 = 1 pulse	Yes	
IL□□64	Pulse Position when the Power is OFF (Upper 2 Words)	1 = 1 pulse	Yes	
IW□□66 to IW□□6F	–	Reserved	–	–
IW□□70 to IW□□7F	Response Buffer for Transparent Command Mode	Stores the response data when MECHATROLINK Servo commands are specified directly.	Yes	

Appendix C Simple Connection Function of the Engineering Tool (Supported by the CPU-03 and CPU-04)

This section explains how to set up a communication process for connecting the MPE720 and MP2200.
In MPE720 Ver.6, set the communication process on the MPE720 window.
Prepare the following equipment to carry out this procedure:

C.1 Preparation

(1) Controller

Product Name		Model	Q'ty
MP2200	Base Unit with AC power supply	JEPMC-BU2200	1
	CPU-03 Module (or CPU-04 Module)	JAPMC-CP2220-E (or JAPMC-CP2230-E)	1

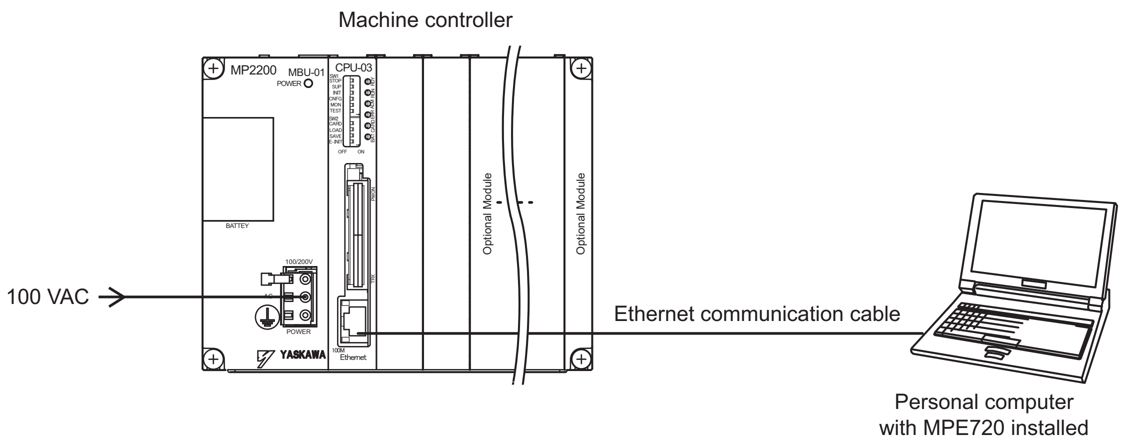
(2) Personal Computer

Product Name	Model	Q'ty
MPE720	CPMC-MPE770 (Ver.6.04 or later)	1
Ethernet Communication Cable	Any commercialized product Ethernet cross cable (category 5 or more)	1
Personal Computer	Any commercialized product	1

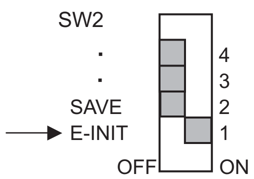
C.2 Procedure

1. Turn OFF the 100-VAC power supply to the MP2200.

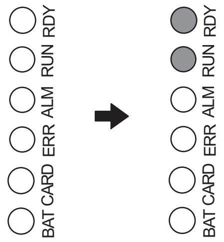
2. Connect the MPE720 installed personal computer and MP2200.



3. Set the E-INIT on the DIP switch (SW2) to ON in the CPU-03/CPU-04 Module.



4. Turn ON the 100-VAC power supply to the MP2200, and confirm that the RDY and RUN are lit on the CPU-03/CPU-04 Module.

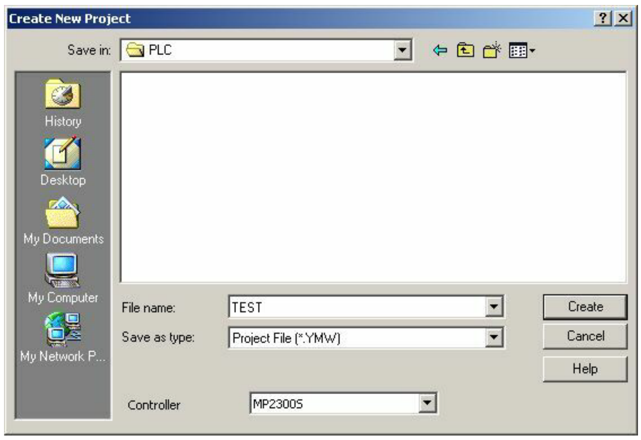


● :Lit   ○ :Unlit

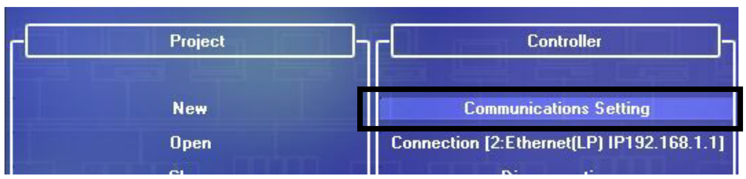
Note: For the CPU-04, the TRX indicator functions as a CARD indicator.

5. Double-click the icon on the personal computer desktop to start up MPE720 Ver6.

6. Create a new PLC folder.



7. Click **Communications Setting**.



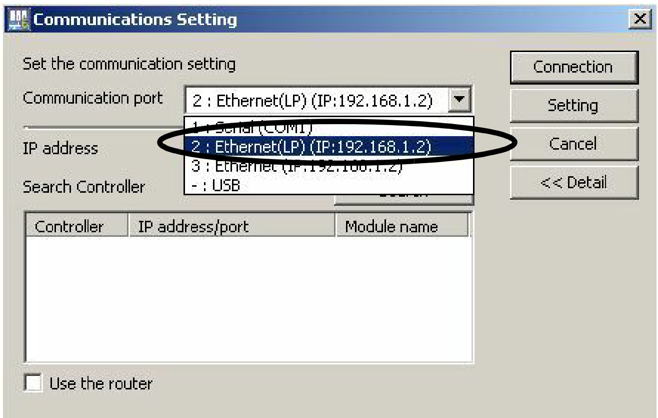
Appendices

A

8. Select **Ethernet (LP) (IP:192.168.1.2)** as the communication port.

Personal computer IP address

Note: You can check the personal computer IP address in advance in Control Panel.



■ Difference between Ethernet (LP) and Ethernet

The LP of Ethernet (LP) is short for "Long packet." Compared with Ethernet, Ethernet (LP) transmits and receives larger packets at one time, resulting in high-speed data transfer. Available communication ports may differ depending on the module of the connected controller. Select the communication port according to the table below.

Module of the Connected Controller Side	Name	Communication Port to Be Selected in MPE720
218IF-01	218IF	Ethernet
218IF-02	218IFB	Ethernet (LP)
CPU-03 Built-in Ethernet	218IFA	
CPU-04 Built-in Ethernet	218IFC	

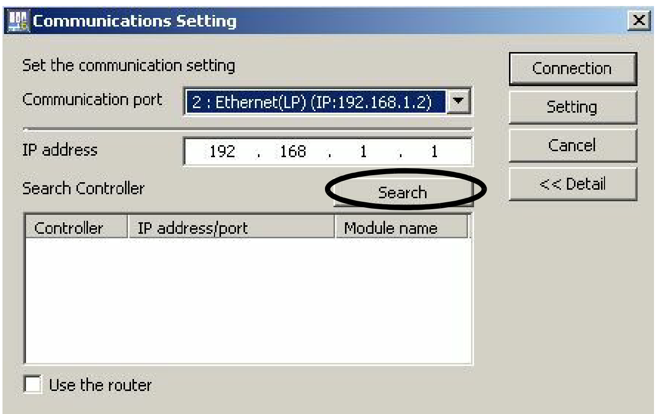
■ When there are multiple LAN ports on the personal computer

If there are multiple LAN ports on the personal computer, multiple IP addresses will be shown in the communication port. Select the IP address of the LAN port to which the cable is connected.

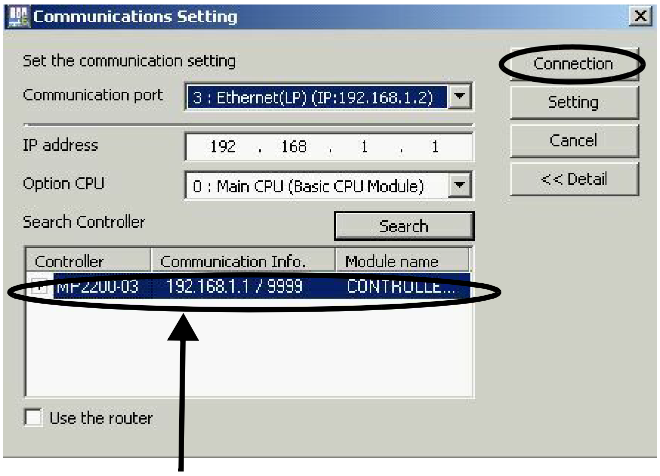
■ Controller search function

When Ethernet is selected in the communication port, the controller search function will be unavailable.

9. Click the **Search** Button.

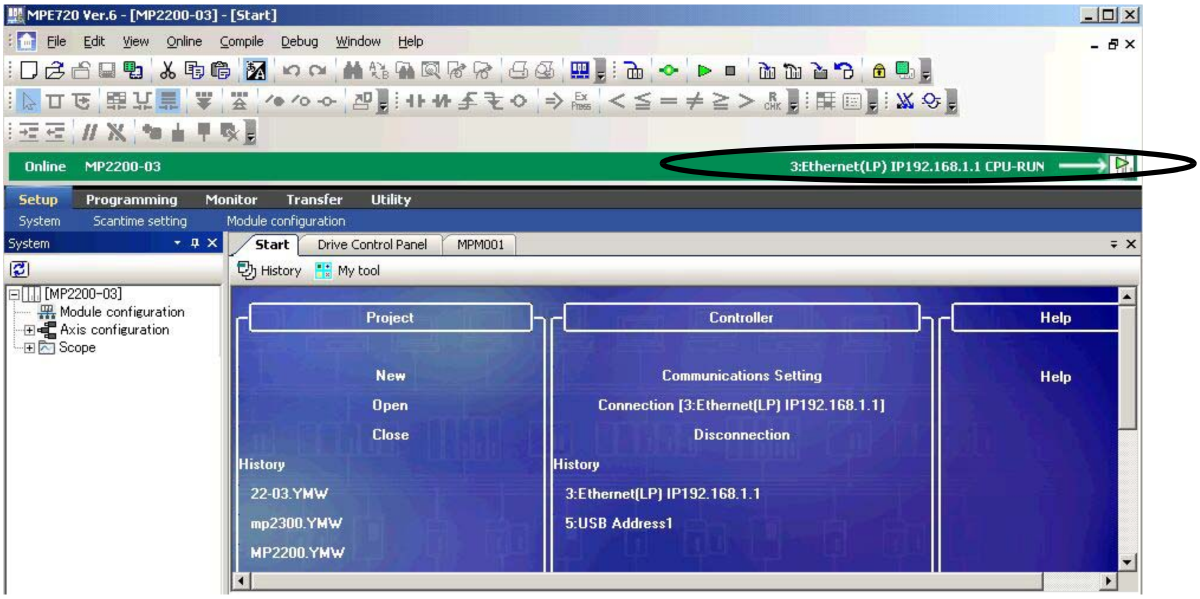


10. A controller search list will appear. Select the found controller and click the **Connection** Button.



Personal computer IP address

11. MPE720 connects to the controller.

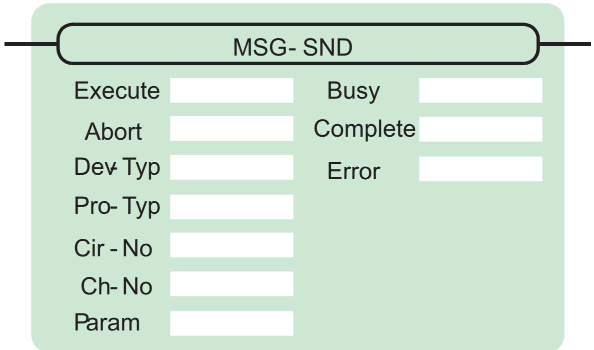


Appendix D MSG-SND/ MSG-RCV Functions (Ethernet)

D.1 Message Send Function (MSG-SND)

This section explains the message send function (MSG-SND) used in a ladder program when transmitting messages.

D.1.1 Specification Overview of the Message Send Function

Function Name	MSG-SND			
Function	Transmits messages to a remote station on the circuit specified by the communication device type. Supports multiple protocols. Keep the execution command (Execute) until Complete or Error turns ON.			
Function Definition				
I/O Definition	No.	Name	I/O Option [*1]	Contents
Input Item	1	Execute	B-VAL	Executes a transmission
	2	Abort	B-VAL	Forcibly ends a transmission
	3	Dev-Typ	I-REG	Communication device type Ethernet (218IF) = 6, Ethernet (218IFA/218IFB/218IFC) = 16
	4	Pro-Typ	I-REG	Communication protocol MEMOBUS [*2] = 1, non-procedure 1 [*3] = 2, non-procedure 2 [*3] = 3
	5	Cir-No	I-REG	Circuit number Ethernet (218IF/218IFA/218IFC) = 1 to 8
	6	Ch-No	I-REG	Communication buffer channel number Ethernet (218IF/218IFC) = 1 to 10, Ethernet (218IFA) = 1 to 4
	7	Param	Address input	Parameter list start address (MA, DA)
Output Item	1	Busy	B-VAL	In process
	2	Complete	B-VAL	Process completed
	3	Error	B-VAL	Error has occurred

* 1. The meanings of I/O options are as follows:
B-VAL: Specify I/O by bit type data.
I-REG: Specify I/O by integer type data. When specifying, set an integer type register number.
As for the input only, it can be a constant (literal).
Address input: The address of the specified register (any integer register) is passed to the function.

* 2. When transmitting in MEMOBUS, Extended MEMOBUS, MELSEC, OMRON, or MODBUS/TCP protocol, set the communication protocol (Pro-Typ) to MEMOBUS(=1). The communication device automatically converts the protocol.

* 3. Non-procedure 1: In non-procedural communication, data is transmitted on a per-word basis.
Non-procedure 2: In non-procedural communication, data is transmitted on a per-byte basis.

D.1.2 I/O Item Details of the Message Send Function

(1) Input Item

The following table shows registers available for each input item.

Input Item	I/O Option	Available Register
Execute Abort	B-VAL	Every bit type register (except #, C registers), Same as above with subscript
Dev-typ Pro-Typ Cir-No Ch-No	I-REG	Every integer type register, Same as above with subscript, Constant
Param	Address input	Register address (except #, C registers), Same as above with subscript

1. Execute (executes a transmission)

Specifies a bit to command execution of a message transmission.
When the Execute bit turns ON, message transmission is implemented. To execute the process, a ladder program or the like needs to be used to switch it ON/OFF.

Note: Keep Execute (executes a transmission) ON until Complete (process completed) or Error (error occurred) is turned ON. When the command turns ON, the message transmission is implemented. To continuously command the transmit execution, make sure to turn Execute (executes a transmission) OFF for one scan or more.

2. Abort (forcibly ends a transmission)

Specify a bit to command a forced termination of a message transmission.
When the Abort bit turns ON, the message transmission is forcibly terminated. Abort takes precedence over Execute.
In order to execute the forced abort, a ladder program or the like needs to be used to switch it ON/OFF.

3. Dev-Typ (communication device type)

Specify the type code of the communication device.

Communication Device	Type Code
Ethernet (218IF)	6
Ethernet (218IFA/218IFC)	16

4. Pro-Typ (communication protocol)

Specify the type code of the communication device.

Type Code	Communication Protocol	Remarks
1	MEMOBUS	Set the type code to “1” when also transmitting using Extended MEMOBUS, MELSEC, or MODBUS/TCP protocol. The communication device automatically converts the protocol.
2	Non-procedure 1 (per word)	Data is transmitted on a per-word basis in non-procedural communication. No response is received from the remote.
3	Non-procedure 2 (per byte)	Data is transmitted on a per-byte basis in the non-procedural communication. No response is received from the remote.

5. Cir-No (circuit number)

Specify a circuit number for the communication device.
Specify it in accordance with the circuit number displayed in the MPE720 **Module Configuration** Window.

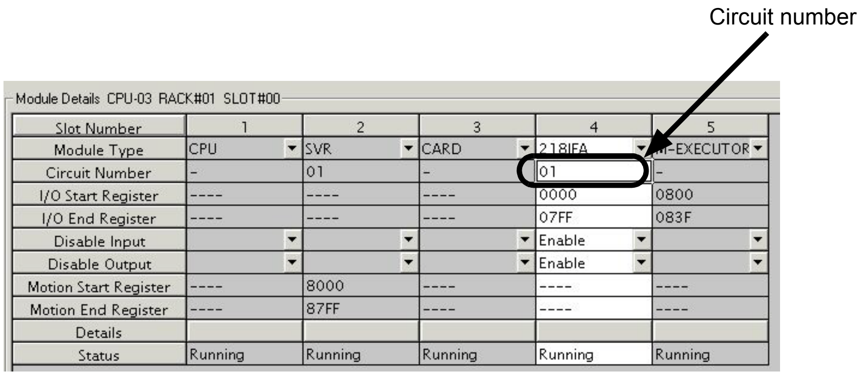


Fig. D.1 MPE720 Module Configuration Window

The following table shows the scope of circuit numbers.

Communication Device	Valid Circuit Number
Ethernet (218IF, 218IFA or 218IFC)	1 to 8

6. Ch-No (communication buffer channel number)

Specify the channel of the communication buffer.
It can be any channel in the scope. However, when starting up multiple functions at the same time, set a unique channel for each function. If you do not start up multiple functions at the same time, the channel numbers can duplicate each other.
The following table shows the scope of channel numbers.

Communication Device	Valid Channel Number
Ethernet (218IF/218IFC)	1 to 10
Ethernet (218IFA)	1 to 4

When the communication device is Ethernet (218IFA), because the communication buffer common to the transmission and reception has four channels, four transmissions (or receptions) are available at the same time by using channels 1 to 4.
In the same way, when the communication device is Ethernet (218IFC), because the communication buffers common to the transmission and reception has ten channels, ten transmissions (or receptions) are available at the same time by using channels 1 to 10.

Note: 1. As many MSG-SND (or MSG-RCV) functions as lines used at the same time are required.
2. For information on communication buffer channels, refer to *D.3 Communication Buffer Channel* .

7. Param (parameter list start address)

Specify the start address of the parameter list. For the “parameter list,” 17 words are automatically assigned from the configured address. In the parameter list, enter the function code and its relevant parameter data. Also, process result and status are output.

Note: For more information about the parameter list, refer to the parameter details for each protocol from *D.1.4 Function Setting and Parameter Details for MEMOBUS and Extended MEMOBUS Protocols* to *D.1.7 Function Setting and Parameter Details for Non-procedural Protocol.*

Example: When “DA00000” is specified as a parameter list start address.

Register	Parameter list (F ··· ···· ···· ··· 0)
DW00000	PARAM00
DW00001	PARAM01
DW00002	PARAM02
DW00003	PARAM03
DW00004	PARAM04
DW00005	PARAM05
DW00006	PARAM06
DW00007	PARAM07
DW00008	PARAM08
DW00009	PARAM09
DW00010	PARAM10
DW00011	PARAM11
DW00012	PARAM12
DW00013	PARAM13
DW00014	PARAM14
DW00015	PARAM15
DW00016	PARAM16

(2) Output Item

The following table shows the registers available for each output item.

Input Item	I/O Option	Available Register
Busy Complete Error	B-VAL	Every bit type register (except #, C registers), Same as above with subscript

1. Busy (in process)

Specify a bit that reports a message is transmitting.
The Busy bit is ON while executing a message transmission or forced abort process.
Keep Execute or Abort ON while Busy is ON.

2. Complete (process completed)

Specify a bit that reports that message transmission has ended.
When a message transmission or forced abort process is completed properly, the Complete bit will turn ON only for one scan.

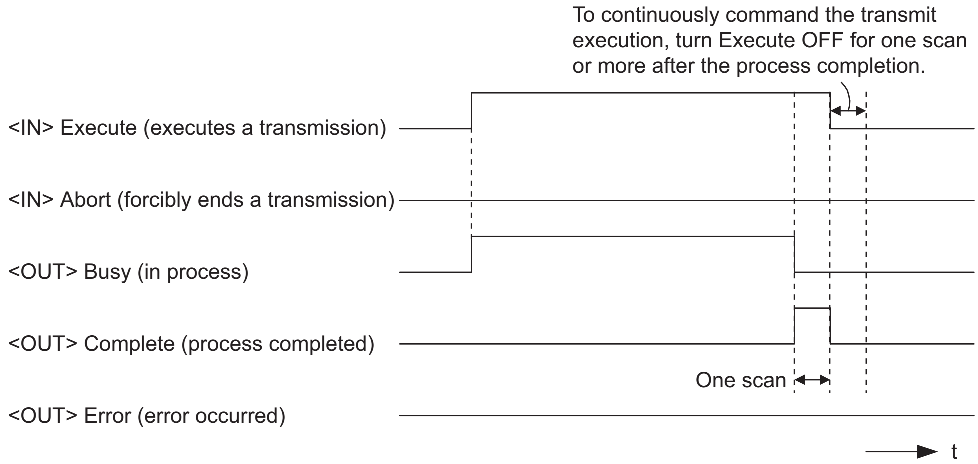
3. Error (error occurred)

Specify a bit that reports that an error has occurred in the message transmission.
When an error occurs, the Error bit will turn ON only for one scan.

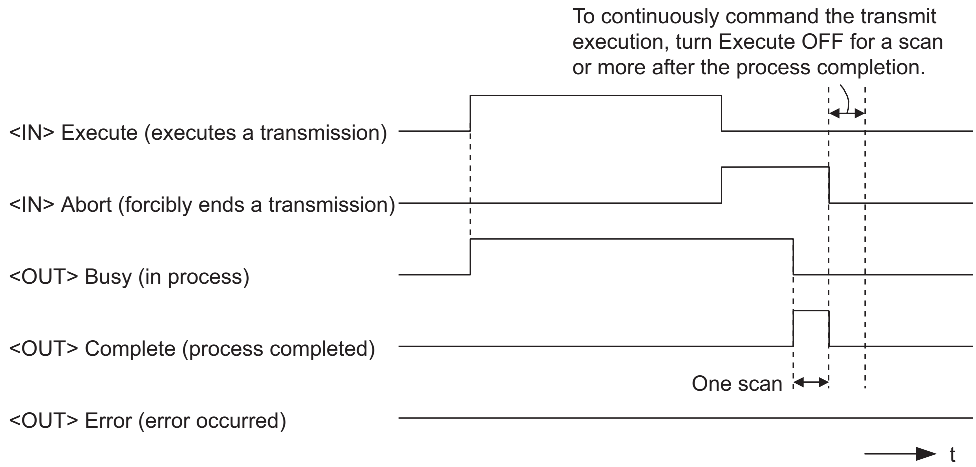
Note: For more information about the error cause, refer to *D.1.4 (2) Process Result (PARAM00)* and *D.1.4 (3) Status (PARAM01)*.

A timing chart of bit type I/O items in the MSG-SND function is as follows:

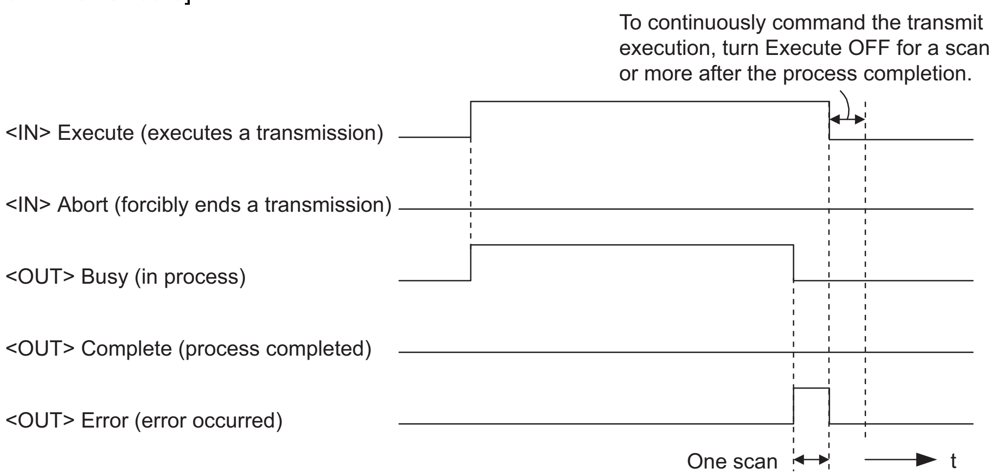
[In Normal Condition]



[When Forcibly Aborted]



[When Error Occurs]



D.1.3 Message Send Function Parameter List (Param) Overview

The param of the MSG-SND function has a parameter list structure composed of 17 words. (The value of the Param itself is the start address (MA, DA) of the parameter list.)
In the parameter list, enter a connection number, function code and its relevant parameter data. Process results and status are also output.
When MEMOBUS or non-procedure is used as a communication protocol, the parameter list is as follows:

Note: Parameter details are explained in the parameter details for each protocol type. Refer to the following items:

- *D.1.4 Function Setting and Parameter Details for MEMOBUS and Extended MEMOBUS Protocols*
- *D.1.5 Function Setting and Parameter Details for MELSEC Protocol*
- *D.1.6 Function Setting and Parameter Details for MODBUS/TCP Protocol*
- *D.1.7 Function Setting and Parameter Details for Non-procedural Protocol*

(1) MEMOBUS Parameter List

Param No.	IN/OUT	Contents	Description
00	OUT	Process result	Process result is output.
01	OUT	Status	The status of the communication device is output.
02	IN	Connection number	Specifies the remote destination.
03	IN	Option	Sets a unique setting for each communication device.
04	IN	Function code	Sets a function code to transmit.
05	IN	Data address	Specifies the start address of the data.
06	IN	Data size	Sets the data size for a read/write request.
07	IN	Remote CPU number	Sets a remote CPU number.
08	IN	Coil offset	Sets the coil's offset word address.
09	IN	Input relay offset	Sets the offset word address of an input relay.
10	IN	Input register offset	Sets the offset word address of an input register.
11	IN	Holding register offset	Sets the offset word address of a holding register.
12	SYS	Reserved 1	
13 to 16	SYS	Reserved 2	

Note: IN: Input, OUT: Output, SYS: For system use

(2) Non-procedual Parameter List

Param No.	IN/OUT	Contents	Description
00	OUT	Process result	Process result is output.
01	OUT	Status	The status of the communication device is output.
02	IN	Connection number	Specifies the remote destination.
03	IN	(unused)	
04	IN	(unused)	
05	IN	Data address	Specifies the start address of the data.
06	IN	Data size	Sets the data size for a write request.
07	IN	(unused)	
08	IN	(unused)	
09	IN	(unused)	
10	IN	(unused)	
11	IN	Register offset	Sets the offset word address of the register.
12	SYS	Reserved 1	
13 to 16	SYS	Reserved 2	

Note: IN: Input, OUT: Output, SYS: For system use

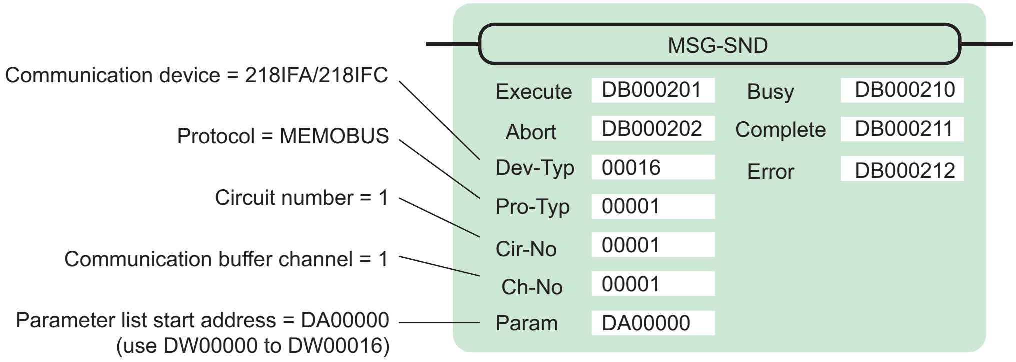
D.1.4 Function Setting and Parameter Details for MEMOBUS and Extended MEMOBUS Protocols

This section explains the MSG-SND function setting and its parameter list details when MEMOBUS or Extended MEMOBUS is used as a protocol type.

(1) Message Send Function Setting

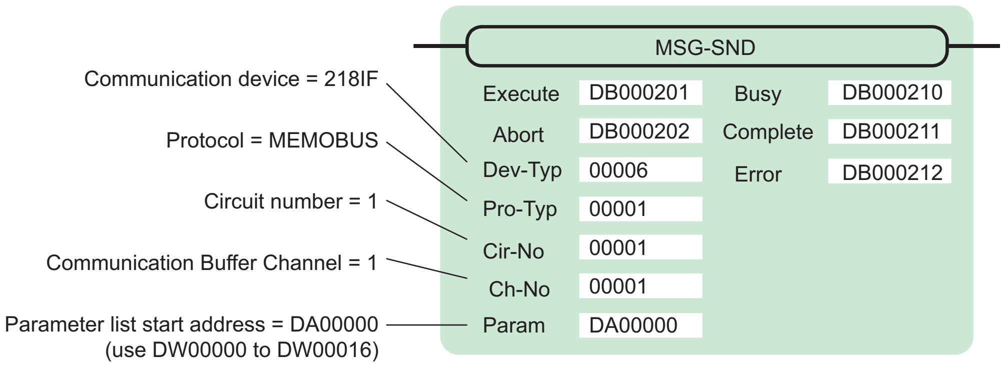
(a) 218IFA/218IFC Setting Example

An example of a function setting when 218IFA/218IFC is used as a communication device follows:
Set the protocol type to MEMOBUS even when used in Extended MEMOBUS protocol.
Set the circuit number in accordance with the circuit number allocated to the target 218IFA/218IFC.
Set a unique communication buffer channel for the same circuit.
For information on the register number, refer to *D.1.2 (1) Input Item* and *D.1.2 (2) Output Item*.



(b) 218IF Setting Example

An example of a function setting when 218IF is used as a communication device follows:
Set the protocol type to MEMOBUS even when used in Extended MEMOBUS protocol.
Set the circuit number in accordance with the circuit number allocated to the target 218IF.
Set a unique communication buffer channel for the same circuit.
For information on the register number, refer to *D.1.2 (1) Input Item* and *D.1.2 (2) Output Item*.



(2) Process Result (PARAM00)

Process result is output to the upper byte. Lower byte is used for system analysis.

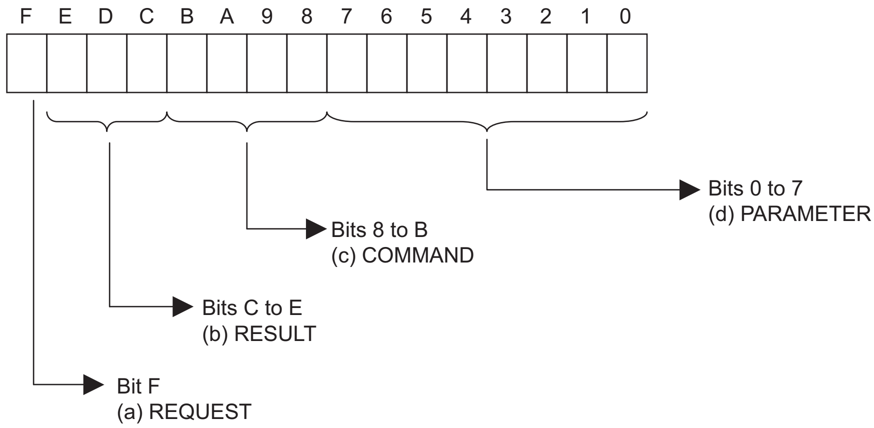
Value of Process Result	Meaning
00xxH	In process (Busy)
10xxH	Process completed (Complete)
8yxxH	Error occurred (Error)

When an error occurs, take corrective action by referring to the following error contents:

Value of Process Result	Error Contents	Description
80xxH	–	Reserved
81xxH	Function code error	Unused function code was transmitted or received. Check PARAM04 (function code).
82xxH	Error in setting address	The following setting is out of the setting range. Check the setting. PARAM05 (data address) PARAM08 (coil offset) PARAM09 (input relay offset) PARAM10 (input register offset) PARAM11 (holding register offset)
83xxH	Data size error	The transmit or receive data size is out of the setting range. Check PARAM04 (data size).
84xxH	Error in setting circuit number	The circuit number is out of the setting range. Check Cir-No (circuit number) of the MSG-SND function.
85xxH	Error in setting channel number	The communication buffer channel number is out of the setting range. Check Ch-No (communication buffer channel number) of the MSG-SND function.
86xxH	Connection number error	The connection number is out of the setting range. Check PARAM02 (connection number).
87xxH	–	Reserved
88xxH	Communication section error	An error response was returned from the communication section (communication device). Check the equipment connection. In addition, confirm communication with the remote device.
89xxH	Device select error	Unavailable device is set. Check Dev-Typ (communication device type) of the MSG-SND function.

(3) Status (PARAM01)

Outputs status of the communication section (communication device).
The following figure shows the bit assignment and the bit assignment details is listed in the table (a) and after.



(a) REQUEST (request)

Outputs whether the MSG-SND function is requesting processing.

Bit State	Description
1	Requesting processing.
0	The acceptance of the process request is complete.

(b) RESULT (result)

Outputs the execution result of the MSG-SND function

Code	Abbreviation	Meaning
0	CONN_NG	In Ethernet communication, transmit error or connection error is complete
1	SEND_OK	Normal transmission complete
2	REC_OK	Normal reception complete
3	ABORT_OK	Forced abort complete
4	FMT_NG	Parameter format error
5	SEQ_NG	Command sequence error
6	RESET_NG	Reset state
7	REC_NG	Data reception error (error detected in the lower layer program)

(c) COMMAND (command)

Outputs a process command for the MSG-SND function. The executed process contents may differ depending on the command.

Code	Abbreviation	Meaning
1	U_SEND	General-purpose message transmission (for non-procedural protocol)
2	U_REC	General-purpose message reception (for non-procedural protocol)
3	ABORT	Forced termination
8	M_SEND	MEMOBUS command transmission: Completed when response is received
9	M_REC	MEMOBUS command reception: Accompanies a response transmission
C	MR_SEND	MEMOBUS response transmission

(d) PARAMETER (parameter)

When RESULT(process result) = 4 (FMT_NG: parameter format error), an error code in the table below is output. Otherwise, the connection number is output.

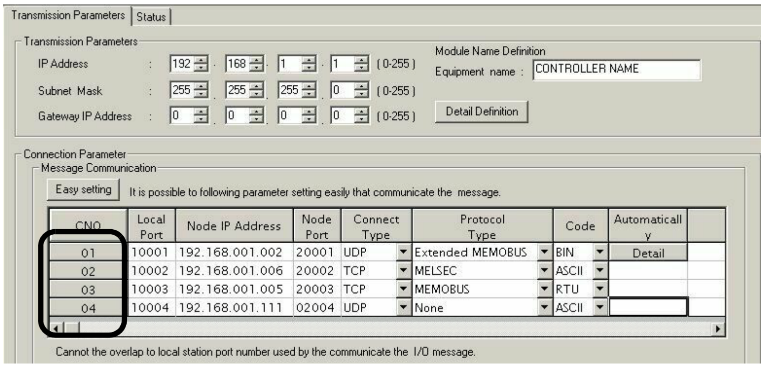
RESULT (process result)	Code (Hex)	Meaning
When RESULT (process result) = 4 (FMT_NG: parameter format error)	00	No error
	01	Connection number is out of range
	02	Time error for monitoring to receive MEMOBUS response
	03	Error in setting retransmit count
	04	Error in setting cyclic area
	05	CPU number error
	06	Data address error
	07	Data size error
	08	Function code error
Others	xx	Connection number

(4) Connection Number (PARAM02)

Specify the remote destination.
When the communication device is Ethernet (218IF/218IFA/218IFC), set the connection number.
The following table shows the range of settings.

Communication Device	Connection Number	Remarks
Ethernet (218IF/218IFC)	1 to 20	Transmits to the remote station set for the specified connection number.
Ethernet (218IFA)	1 to 4	Same as above

Note: When the communication device is Ethernet (218IF/218IFA/218IFC), set the connection number according to the connection number in the 218IF/218IFA/218IFC **Transmission Parameters** Tab Page for the MPE720 **Module Configuration** Window.



Note: For the 218IFC, the setting range for the connection number is larger, i.e., 01 to 20.

Fig. D.2 218IFA/218IFC Transmission Parameters Tab Page for the MPE720 Module Configuration Window

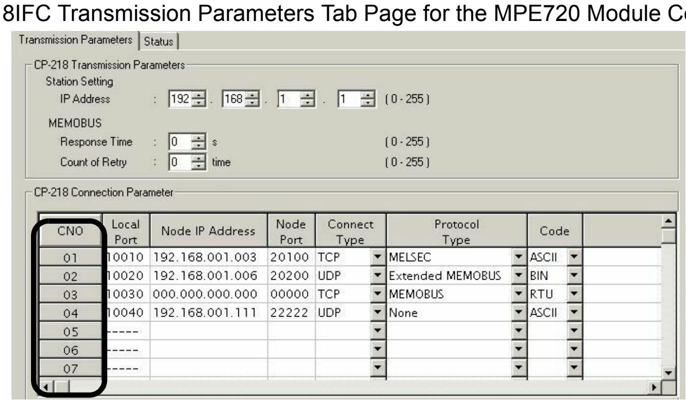


Fig. D.3 218IF Transmission Parameters Tab Page for the MPE720 Module Configuration Window

(5) Option (PARAM03)

Choose a unique setting for each communication device.
When the protocol is MEMOBUS or Extended MEMOBUS, this is not used, and no setting is necessary.

(6) Function Code (PARAM04)

Set a function code to transmit.
The functions (read coil or input relay state, write to holding register, etc.) registered in the function codes are made available by specifying the code.
The following table shows the function codes available when using a MEMOBUS or Extended MEMOBUS protocol.

Table D.1 Function Code List (MEMOBUS, Extended MEMOBUS)

Function Code	Target Data Type	Function	Protocol Type	
			Extended MEMOBUS	MEMOBUS
00H	–	Unused	–	–
01H	B	Reads coil state	√	√
02H	B	Reads input relay state	√	√
03H	W	Reads holding register content	√	√
04H	W	Reads input register content	√	√
05H	B	Changes single coil state	√	√
06H	W	Writes to single holding register	√	√
07H	–	Unused	–	–
08H	–	Loopback test	√	√
09H	W	Reads holding register content (extended)	√	–
0AH	W	Reads input register content (extended)	√	–
0BH	W	Writes to holding register (extended)	√	–
0CH	–	Unused	–	–
0DH	W	Discontinuously reads holding register (extended)	√	–
0EH	W	Discontinuously writes to holding register (extended)	√	–
0FH	B	Changes multiple coil states	√	√
10H	W	Writes to multiple holding registers	√	√

Note: 1. B: Bit type, W: Integer type
2. √: Available, -: Not available
3. Transmit and receive registers in the master operation mode are MW (MB) only.
4. In slave operation mode, the coil, holding register, input relay, and input register are MB, MW, IB, IW respectively.

(7) Data Address (PARAM05)

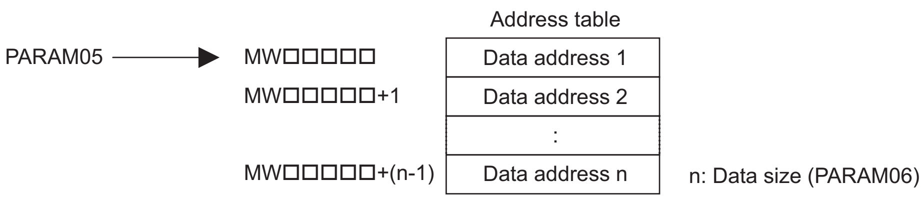
Specify the start address of the data.
The address must be input in decimal or hexadecimal numbers.
Example: When the start address is MW01000, specify “1000 (decimal)” or “3E8H (hexadecimal).”
The range of data addresses may differ, depending on the function code.
The following table shows the valid ranges of data addresses when using a MEMOBUS or Extended MEMOBUS protocol.

Table D.2 Valid Range of Data Addresses (MEMOBUS, Extended MEMOBUS)

<table>
<tr><th rowspan="2">Function Code</th><th rowspan="2">Target Data Type</th><th rowspan="2">Function</th><th colspan="2">Valid Range of Data Addresses</th></tr>
<tr><th>Ethernet (218IF)</th><th>Ethernet (218IFA/ 218IFC)</th></tr>
<tr><td>00H</td><td>–</td><td>Unused</td><td colspan="2">Disable</td></tr>
<tr><td>01H</td><td>B</td><td>Reads coil state*1</td><td colspan="2">0 to 65535 (0 to FFFFH)</td></tr>
<tr><td>02H</td><td>B</td><td>Reads input relay state*1</td><td colspan="2">0 to 65535 (0 to FFFFH)</td></tr>
<tr><td>03H</td><td>W</td><td>Reads holding register content*2</td><td colspan="2">0 to 65534 (0 to FFFEH)</td></tr>
<tr><td>04H</td><td>W</td><td>Reads input register content*2</td><td>0 to 32767 (0 to 7FFFH)</td><td>0 to 65535 (0 to FFFFH)</td></tr>
<tr><td>05H</td><td>B</td><td>Changes single coil state*1</td><td colspan="2">0 to 65535 (0 to FFFFH)</td></tr>
<tr><td>06H</td><td>W</td><td>Writes to single holding register*2</td><td colspan="2">0 to 65534 (0 to FFFEH)</td></tr>
<tr><td>07H</td><td>–</td><td>Unused</td><td colspan="2">Disable</td></tr>
<tr><td>08H</td><td>–</td><td>Loopback test</td><td colspan="2">Disable</td></tr>
<tr><td>09H</td><td>W</td><td>Reads holding register content (extended)*2</td><td colspan="2">0 to 65534 (0 to FFFEH)</td></tr>
<tr><td>0AH</td><td>W</td><td>Reads input register content (extended)*2</td><td>0 to 32767 (0 to 7FFFH)</td><td>0 to 65535 (0 to FFFFH)</td></tr>
<tr><td>0BH</td><td>W</td><td>Writes to holding register (extended)*2</td><td colspan="2">0 to 65534 (0 to FFFEH)</td></tr>
<tr><td>0CH</td><td>–</td><td>Unused</td><td colspan="2">Disable</td></tr>
<tr><td>0DH</td><td>W</td><td>Discontinuously reads holding register (extended)*3</td><td colspan="2">0 to 65534 (0 to FFFEH)</td></tr>
<tr><td>0EH</td><td>W</td><td>Discontinuously writes to holding register (extended)*3</td><td colspan="2">0 to 65534 (0 to FFFEH)</td></tr>
<tr><td>0FH</td><td>B</td><td>Changes multiple coil states*1</td><td colspan="2">0 to 65535 (0 to FFFFH)</td></tr>
<tr><td>10H</td><td>W</td><td>Writes to multiple holding registers*2</td><td colspan="2">0 to 65534 (0 to FFFEH)</td></tr>
</table>

* 1. Request for reading/writing coil or input relay: Specifies the start bit address of data
* 2. Request for continuously reading/writing register: Specifies the start word address of data
* 3. Request for discontinuously reading/writing register: Specifies the start M register number of the address table

Note: Address Table
An address table is used for specifying addresses indirectly in order to indicate discontinuous data. The PARAM06 (data size) sizes of addresses at the beginning of the M register set by PARAM05 (data address) are used as an address table.
When reading, specify the remote station's address to read for the data addresses 1-n. Read values are stored locally according to the data addresses 1-n.
When writing, data stored in the local data addresses 1-n is picked up and written into the remote station’s data addresses 1-n.
The address table used when discontinuously reading/writing registers is as follows:



(8) Data Size (PARAM06)

Set the data size (number of bits or words) for the read/write request.
Be sure that the last data address determined based on the offset, data address, and data size does not go beyond the scope of the data addresses.
The range of data addresses may differ, depending on the function code and communication device.
The following table shows the valid ranges of data sizes when using a MEMOBUS or Extended MEMOBUS protocol.

Table D.3 Valid Range of Data Sizes (MEMOBUS, Extended MEMOBUS)

<table>
<tr><th rowspan="2">Function Code</th><th rowspan="2">Target Data Type</th><th rowspan="2">Function</th><th colspan="2">Valid Range of Data Sizes</th></tr>
<tr><th>Ethernet(218IF)</th><th>Ethernet(218IFA/ 218IFC)</th></tr>
<tr><td>00H</td><td>–</td><td>Unused</td><td colspan="2">Disable</td></tr>
<tr><td>01H</td><td>B</td><td>Reads coil state*1</td><td colspan="2">1 to 2000</td></tr>
<tr><td>02H</td><td>B</td><td>Reads input relay state*1</td><td colspan="2">1 to 2000</td></tr>
<tr><td>03H</td><td>W</td><td>Reads holding register content*2</td><td colspan="2">1 to125</td></tr>
<tr><td>04H</td><td>W</td><td>Reads input register content*2</td><td colspan="2">1 to 125</td></tr>
<tr><td>05H</td><td>B</td><td>Changes single coil state</td><td colspan="2">Disable</td></tr>
<tr><td>06H</td><td>W</td><td>Writes to single holding register</td><td colspan="2">Disable</td></tr>
<tr><td>07H</td><td>–</td><td>Unused</td><td colspan="2">Disable</td></tr>
<tr><td>08H</td><td>–</td><td>Loopback test</td><td colspan="2">Disable</td></tr>
<tr><td>09H</td><td>W</td><td>Reads holding register content (extended)*2</td><td>1 to 508</td><td>1 to 2044 (BIN)
1 to 1020 (ASCII)</td></tr>
<tr><td>0AH</td><td>W</td><td>Reads input register content (extended)*2</td><td>1 to 508</td><td>1 to 2044 (BIN)
1 to 1020 (ASCII)</td></tr>
<tr><td>0BH</td><td>W</td><td>Writes to holding register (extended)*2</td><td>1 to 507</td><td>1 to 2043 (BIN)
1 to 1019 (ASCII)</td></tr>
<tr><td>0CH</td><td>–</td><td>Unused</td><td colspan="2">Disable</td></tr>
<tr><td>0DH</td><td>W</td><td>Discontinuously reads holding register (extended)*2</td><td>1 to 508</td><td>1 to 2044 (BIN)
1 to 1020 (ASCII)</td></tr>
<tr><td>0EH</td><td>W</td><td>Discontinuously writes to holding register (extended)*2</td><td>1 to 254</td><td>1 to 1022 (BIN)
1 to 510 (ASCII)</td></tr>
<tr><td>0FH</td><td>B</td><td>Changes multiple coil states*1</td><td colspan="2">1 to 800</td></tr>
<tr><td>10H</td><td>W</td><td>Writes to multiple holding registers*2</td><td colspan="2">1 to 100</td></tr>
</table>

* 1. Specifies the number of bits
* 2. Specifies the number of words
Note: Data size in the table is represented as a decimal number.

(9) Remote CPU Number (PARAM07)

Set a remote CPU number.
When the remote device is MP2□00 series, specify “1”.
When the remote device is a controller manufactured by YASKAWA Electric Corporation, but other than the MP2□00 series, and comprises multiple CPU modules, specify the destination CPU number.
Otherwise, specify “0”.

(10) Offset (PARAM08, PARAM09, PARAM10, PARAM11)

Specify the offset addresses of read data storage areas and write data source of the transmission side.
The address for the transmission side will be displaced by the number of words designated by the offset.

Note: 1. For more information, refer to *D.1.8 Relationship between Data Address, Data Size, and Offset for MSG-SND Function*.
2. The offset cannot be a negative value.

The offset parameter is prepared for each target data type.
The following table lists the offset parameters.

Table D.4 Offset Parameter List

Parameter	Content	Description
PARAM08	Coil offset	Sets the coil's offset word address.
PARAM09	Input relay offset	Sets the offset word address of an input relay.
PARAM10	Input register offset	Sets the offset word address of an input register.
PARAM11	Holding register offset	Sets the offset word address of a holding register.

The valid offset parameter may differ, depending on the function code.
The following table provides the valid parameters for each function code.

Table D.5 Valid Parameter List for Offset of Each Function Code

Function Code	Function	Valid Offset Parameter	Protocol Type	
			Extended MEMOBUS	MEMOBUS
01H	Reads coil state	PARAM08	√	√
02H	Reads input relay state	PARAM09	√	√
03H	Reads holding register content	PARAM11	√	√
04H	Reads input register content	PARAM10	√	√
05H	Changes single coil state	PARAM08	√	√
06H	Writes to single holding register	PARAM11	√	√
09H	Reads holding register content (extended)	PARAM11	√	–
0AH	Reads input register content (extended)	PARAM10	√	–
0BH	Writes to holding register (extended)	PARAM11	√	–
0DH	Discontinuously reads holding register (extended)	PARAM11	√	–
0EH	Discontinuously writes to holding register (extended)	PARAM11	√	–
0FH	Changes multiple coil states	PARAM08	√	√
10H	Writes to multiple holding registers	PARAM11	√	√

Note: √: Available, -: Not available

(11) Reserved by System 1 (PARAM12)

Used by system (the channel number of the communication buffer in use is stored).

Note: Before the first scan during power up, make sure to set it to “0” by using a user program.
After that, the register is used by the system, so never change the value with the user program.

(12) Reserved by System 2 (PARAM13 to PARAM16)

Used by the system. Never change the value with the user program, etc.

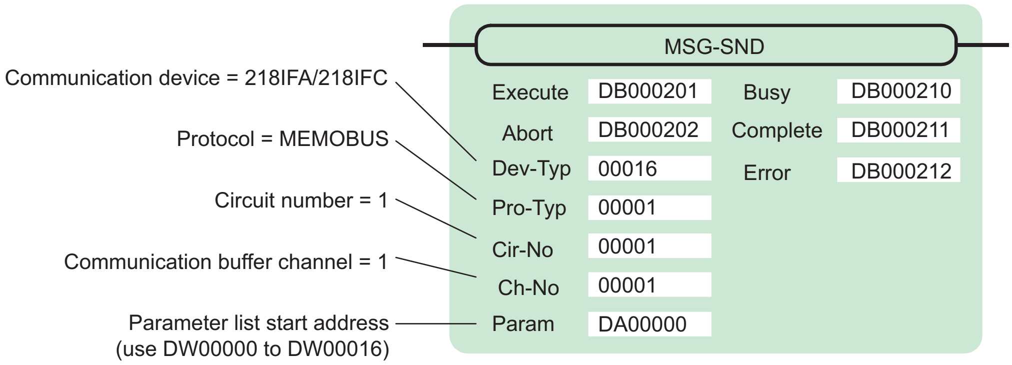
D.1.5 Function Setting and Parameter Details for MELSEC Protocol

This section explains MSG-SND function setting and its parameter list details when MELSEC is used as a protocol.

(1) Message Send Function Setting

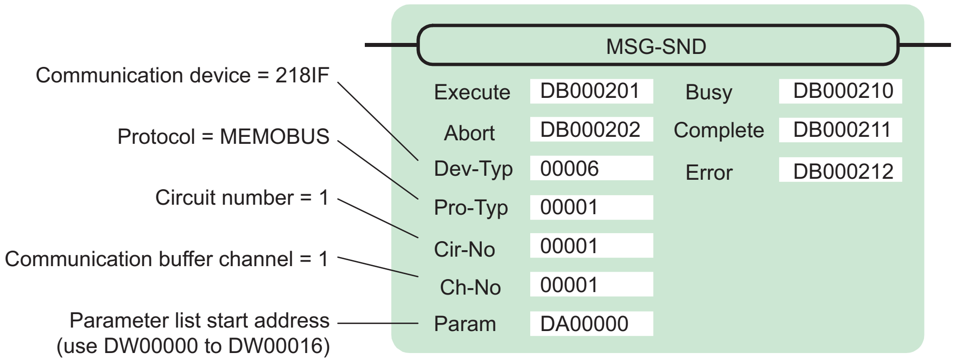
(a) 218IFA/218IFC Setting Example

An example of a function setting when 218IFA/218IFC is used as a communication device follows:
Set the protocol type to MEMOBUS when used in MELSEC protocol.
Set the circuit number in accordance with the circuit number allocated to the target 218IFA/218IFC.
Set a unique communication buffer channel for the same circuit.
For information on the register number, refer to *D.1.2 (1) Input Item* and *D.1.2 (2) Output Item*.



(b) 218IF Setting Example

An example of a function setting when 218IF is used as a communication device follows:
Set the protocol type to MEMOBUS when used in MELSEC protocol.
Set the circuit number in accordance with the circuit number allocated to the target 218IF.
Set a unique communication buffer channel for the same circuit.
For information on the register number, refer to *D.1.2 (1) Input Item* and *D.1.2 (2) Output Item*.



(2) Process Result (PARAM00)

Refer to *D.1.4 (2) Process Result (PARAM00)*.

(3) Status (PARAM01)

Refer to *D.1.4 (3) Status (PARAM01)*.

(4) Connection Number (PARAM02)

Refer to *D.1.4 (4) Connection Number (PARAM02)*.

(5) Option (PARAM03)

Choose a unique setting for each communication device.
This is not used by the MELSEC protocol, and does not require setting when MELSEC is used.

(6) Function Code (PARAM04)

Set a function code to transmit.
The functions (read bit/word device, write to word device, etc.) registered in the function codes are made available by specifying the code.
The following table lists function codes used with the MELSEC protocol.

Table D.6 Function Code List

Function Code	MELSEC ACPU Common Command	Target Data Type	Function
01H/02H	00H	B	Reads bit device in units
03H/04H/09H/0AH	01H	W	Reads word device in units
05H/0FH	02H	B	Writes to bit device in units
06H/0BH/10H	03H	W	Writes to word device in units
08H	16H	–	Loopback test
0EH	05H	B	Specifies a device number for each word device at random and sets/resets each device
31H	60H	W	Writes to the fixed buffer in words
32H	61H	W	Reads from the random access buffer in words
33H	62H	W	Writes to the random access buffer in words

Note: 1. B: Bit type, W: Integer type
2. AnCPU dedicated commands are not supported. When accessing AnCPU, also use the ACPU common commands. You cannot access the AnCPU extended file register.

(7) Data Address (PARAM05)

Specify the start address of the data.
The address must be input in decimal or hexadecimal numbers.
Example: When the start address is MW01000, specify “1000 (decimal)” or “3E8H (hexadecimal).”
The valid range of usable function codes and data addresses may differ, depending on the device type and device range of the MELSEC side.
The following table gives the valid ranges of data addresses when using MELSEC as a protocol.

Table D.7 Valid Range of Data Addresses (MELSEC bit device)

Device	Device Range of ACPU Common Commands	Decimal/ Hexa-decimal	Function Code	Valid Range of Data Addresses	Corresponding Register Number
X	X0000 to X07FF	Hexadecimal	02H: Input relay	0 to 2047	MB000000 to MB00127F
Y	Y0000 to Y07FF	Hexadecimal	01H/0FH: Coil	0 to 2047	MB000000 to MB00127F
M	M0000 to M2047	Decimal	01H/05H/0FH: Coil	2048 to 4095	MB001280 to MB00255F
M	M9000 to M9255	Decimal	01H/05H/0FH: Coil	4096 to 4351	MB002560 to MB00271F
B	B0000 to B03FF	Hexadecimal	01H/05H/0FH: Coil	4352 to 5375	MB002720 to MB00335F
F	F0000 to F0255	Decimal	01H/05H/0FH: Coil	5376 to 631	MB003360 to MB00351F
TS	TS000 to TS255	Decimal	02H: Input relay	2048 to 2303	MB001280 to MB00143F
TC	TC000 to TC255	Decimal	02H: Input relay	2304 to 2559	MB001440 to MB00159F
CS	CS000 to CS255	Decimal	02H: Input relay	2560 to 2815	MB001660 to MB00175F
CC	CC000 to CC255	Decimal	02H: Input relay	2816 to 3071	MB001760 to MB00191F
M	M2048 to M8191	Decimal	01H/05H/0FH: Coil	8192 to 14335	MB005120 to MB00895F

Table D.8 Valid Range of Data Addresses (MELSEC word device)

Device	Device Range of ACPU Common Commands	Decimal/ Hexa-decimal	Function Code	Valid Range of Data Addresses	Corresponding Register Number
TN	TN000 to TN255	Decimal	04H/0AH: Input register	0 to 255	MW00000 to MW0255
CN	CN000 to CN255	Decimal	04H/0AH: Input register	256 to 511	MW00256 to MW00511
D	D0000 to D1023	Decimal	03H/06H/09H/0BH/ 0EH/10H: Holding register	0 to 1023	MW00000 to MW01023
D (special)	D9000 to D9255	Decimal	03H/06H/09H/0BH/ 0EH/10H: Holding register	1024 to 1279	MW01024 to MW01279
W	W0000 to W03FF	Hexadecimal	03H/06H/09H/0BH/ 0EH/10H: Holding register	1280 to 2303	MW01280 to MW02303
R	R0000 to R8191	Decimal	03H/06H/09H/0BH/ 0EH/10H: Holding register	2304 to 10495	MW02304 to MW10495
D	D1024 to D6143	Decimal	03H/06H/09H/0BH/ 0EH/10H: Holding register	10496 to 15615	MW10496 to MW15615

Note: 1. The device range may differ according to the MELSEC sequencer even within the device range. For more information, refer to the MELSEC manual.
2. The register number corresponding to MP2☐00 can be adjusted by the offset setting of the MSG-SND function.

(8) Data Size (PARAM06)

Set the data size (number of bits or words) for the read/write request.
Be sure that the last data address determined based on the offset, data address, and data size does not go beyond the scope of the data addresses.
The range of data addresses may differ, depending on the function code and communication device.
The following table gives the valid ranges of data sizes when using MELSEC as a protocol.

Table D.9 Valid Range of Data Sizes

Function Code	MELSEC ACPU Common Command	Function	Valid Range of Data Sizes	
			Ethernet(218IF)	Ethernet(218IFA/ 218IFC)
01H/02H	00H	Reads bit device in units	1 to 256 units	
03H/04H/ 09H/0AH	01H	Reads word device in units	1 to 256 units	
05H/0FH	02H	Writes to bit device in units	1 to 256 units	
06H/0BH/ 10H	03H	Writes to word device in units	1 to 256 units	
08H	16H	Loopback test	–	
0EH	05H	Specifies a device number for each word device at random and sets/resets each word device	1 to 40 units	
31H	60H	Writes to the fixed buffer in words	See the table below.	
32H	61H	Reads from the random access buffer in words		
33H	62H	Writes to the random access buffer in words		

Function	Connection Type	Code	Valid Range of Data Sizes	
			Ethernet(218IF)	Ethernet(218IFA/ 218IFC)
Writes to the fixed buffer in words	TCP	BIN	1 to 507 words	1 to 727 words
		ASCII	1 to 362 words	1 to 362 words
	UDP	BIN	1 to 507 words	1 to 1017 words
		ASCII	1 to 507 words	1 to 508 words
Reads from the random access buffer in words	TCP	BIN	1 to 508 words	1 to 728 words
		ASCII	1 to 363 words	1 to 363 words
	UDP	BIN	1 to 508 words	1 to 1017 words
		ASCII	1 to 508 words	1 to 508 words
Writes to the random access buffer in words	TCP	BIN	1 to 507 words	1 to 726 words
		ASCII	1 to 361 words	1 to 361 words
	UDP	BIN	1 to 508 words	1 to 1017 words
		ASCII	1 to 508 words	1 to 508 words

Note: The restricted data size when using TCP is the maximum size transmitted using one segment.
The segment size is determined by MTU (maximum transfer unit) as a TCP data transfer unit.
The valid range of data sizes mentioned above is for MTU = 1500 bytes.

(9) Remote CPU Number (PARAM07)

Refer to *D.1.4 (9) Remote CPU Number (PARAM07)*.

(10) Offset (PARAM08, PARAM09, PARAM10, PARAM11)

Specify the offset addresses of the read data storage area and the write data source of the transmission side.
The address for the transmission side will be displaced by the number of words designated by the offset.

Note: 1. For more information, refer to *D.1.8 Relationship between Data Address, Data Size, and Offset for MSG-SND Function*.
2. The offset cannot be a negative value.

The offset parameter is prepared for each target data type.
The following table provides the offset parameters.

Table D.10 Offset Parameter List

Parameter	Contents	Description
PARAM08	Coil offset	Sets the coil's offset word address.
PARAM09	Input relay offset	Sets the offset word address of an input relay.
PARAM10	Input register offset	Sets the offset word address of an input register.
PARAM11	Holding register offset	Sets the offset word address of a holding register.

The valid offset parameter may differ according to function code.
The following table provides the valid parameters for each function code.

Table D.11 Valid Parameter List for Offset of Each Function Code

Function Code	Function	Valid Offset Parameter
01H	Reads coil state	PARAM08
02H	Reads input relay state	PARAM09
03H	Reads holding register content	PARAM11
04H	Reads input register content	PARAM10
05H	Changes single coil state	PARAM08
06H	Writes to single holding register	PARAM11
09H	Reads holding register content (extended)	PARAM11
0AH	Reads input register content (extended)	PARAM10
0BH	Writes to holding register (extended)	PARAM11
0EH	Discontinuously writes to holding register (extended)	PARAM11
0FH	Changes multiple coil states	PARAM08
10H	Writes to multiple holding registers	PARAM11
31H	Write to fixed buffer	PARAM11
32H	Reads the random access buffer	PARAM11
33H	Writes to the random access buffer	PARAM11

(11) Reserved by System 1 (PARAM12)

Used by system (the channel number of the communication buffer in use is stored).

Note: At the first scan during power up, make sure to set it to “0” by using the user program. After that, the register is used by system. Do not change the value thereafter with the user program.

(12) Reserved by System 2 (PARAM13-PARAM16)

Used by system. Never change this value with the user program, etc.

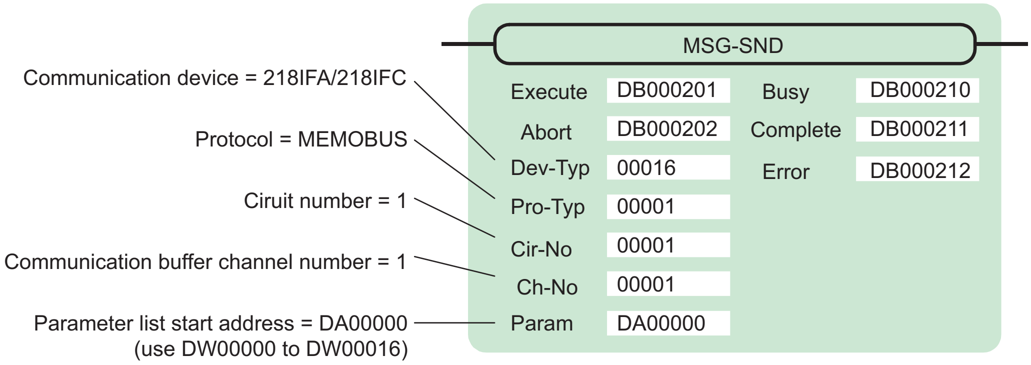
D.1.6 Function Setting and Parameter Details for MODBUS/TCP Protocol

This section explains the MSG-SND function setting and its parameter list details when MODBUS/TCP is used as a protocol type.

(1) Message Send Function Setting

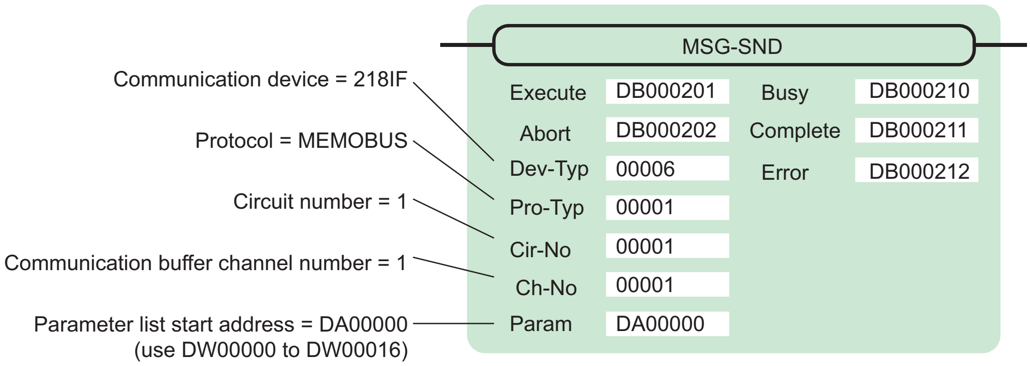
(a) 218IFA/218IFC Setting Example

An example of a function setting when 218IFA/218IFC is used as a communication device follows:
Set the protocol type to MEMOBUS when used in MODBUS/TCP protocol.
Set the circuit number in accordance with the circuit number allocated to the target 218IFA/218IFC.
Set a unique communication buffer channel number for the same circuit.
For information on the register number, refer to *D.1.2 (1) Input Item* and *D.1.2 (2) Output Item*.



(b) 218IF Setting Example

An example of a function setting when 218IF is used as a communication device follows:
Set the protocol type to MEMOBUS when used in MODBUS/TCP protocol.
Set the circuit number in accordance with the circuit number allocated to the target 218IF.
Set a unique communication buffer channel number for the same circuit.
For information on the register number, refer to *D.1.2 (1) Input Item* and *D.1.2 (2) Output Item*.



(2) Process Result (PARAM00)

Refer to *D.1.4 (2) Process Result (PARAM00)*.

(3) Status (PARAM01)

Refer to *D.1.4 (3) Status (PARAM01)*.

(4) Connection Number (PARAM02)

Refer to *D.1.4 (4) Connection Number (PARAM02)*.

(5) Option (PARAM03)

Choose a unique setting for each communication device.
The following table provides the scope of the setting.

Communication Device	Valid Range	Remarks
Ethernet(218IF, 218IFA or 218IFC)	0 to 247, 255	Set a remote unit ID when MODBUS/TCP protocol is used. When the transmission target is MP2□00 series, specify "0."

(6) Function Code (PARAM04)

Set a function code to transmit.
The functions (read coil and input relay state, write to holding register, etc.) registered in the function codes are made available by specifying the code.
The following table shows the available function codes when using MODBUS/TCP as a protocol.

Table D.12 Function Code List (MODBUS/TCP)

Function Code	Target Data Type	Function	Communication device	
			Ethernet(218IF)	Ethernet(218IFA/ 218IFC)
00H	–	Unused	–	–
01H	B	Reads coil state	√	√
02H	B	Reads input relay state	√	√
03H	W	Reads holding register content	√	√
04H	W	Reads input register content	√	√
05H	B	Changes single coil state	√	√
06H	W	Writes to single holding register	√	√
07H : 0EH	–	Unused	–	–
0FH	B	Changes multiple coil states	√	√
10H	W	Writes to multiple holding registers	√	√
11H : 15H	–	Unused	–	–
16H	W	Mask writes to holding register	–	√
17H	W	Reads/Writes multiple holding registers	–	√

Note: 1. B: Bit type, W: Integer type
2. √: Available, -: Not available
3. Transmit and receive registers in the master operation mode are MW (MB) only.
4. In the slave operation mode, coil, holding register, input relay, and input register are MB, MW, IB, IW respectively.

(7) Data Address (PARAM05)

Specify the start address of the data.
The address must be a decimal or hexadecimal number.
Example: When the start address is MW01000, specify “1000 (decimal)” or “3E8H (hexadecimal).”
The range of data addresses may differ according to the function code.
The following table gives the valid ranges of data addresses when using MODBUS/TCP as a protocol.

Table D.13 Valid Range of Data Addresses (MODBUS/TCP)

Function Code	Target Data Type	Function	Valid Range of Data Addresses
00H	–	Unused	Disable
01H	B	Reads coil state	0 to 65535 (0 to FFFFH)
02H	B	Reads input relay state	0 to 65535 (0 to FFFFH)
03H	W	Reads holding register content	0 to 65534 (0 to FFFEH)
04H	W	Reads input register content	0 to 32767 (0 to 7FFFH)
05H	B	Changes single coil state	0 to 65535 (0 to FFFFH)
06H	W	Writes to single holding register	0 to 65534 (0 to FFFEH)
07H : 0EH	–	Unused	Disable
0FH	B	Changes multiple coil states	0 to 65535 (0 to FFFFH)
10H	W	Writes to multiple holding registers	0 to 65534 (0 to FFFEH)
11H : 15H	–	Unused	Disable
16H	W	Mask writes to holding register*1, *2	0 to 65534 (0 to FFFEH)
17H	W	Reads/Writes multiple holding register*1, *3	0 to 65534 (0 to FFFEH)

* 1. Invalid when Ethernet (218IF) is used.
* 2. Mask write request to holding register: Specify the start M register number of the remote address-cum-local data table
* 3. Request for reading/writing multiple holding registers: Specify the start M register number of an address table

Note: 1. Data table
A data table used for a mask write request to the holding register is used for storing mask data. The two words of addresses at the beginning of the M register specified by PARAM05 (data address) are used as an address table.
Specify AND and OR mask data for the data table.
Note that PARAM05 (data address) used for the mask write request to the holding register doubles as the start M register number in the local station, which is also used for specifying the start data address and as the data table in the remote station for mask writing.
The contents of the data table used when mask writing to the holding register is as follows:

Data table for mask writing to holding register

PARAM05 →	MW□□□□□	AND mask data
	MW□□□□□+1	OR mask data

2. Address table
   An address table used for the read/write request to the multiple holding registers is used for specifying addresses indirectly in order to indicate read/write data. The four words of addresses at the beginning of the M register set by PARAM05 (data address) are used as an address table.
   For the address table, specify a data address and data size for reading and a data address and data size for writing.
   Read behaves same as “Reads holding register content (function code: 03H)”. Write behaves same as “Writes to multiple holding registers (function code: 10H)”.
   The address table used when reading/writing multiple holding registers is as follows:

Address table for reading/
writing multiple holding registers

		Address table for reading/writing multiple holding registers	
PARAM05 →	MW□□□□□	Data address	For reading
	MW□□□□□+1	Data size	
	MW□□□□□+2	Data address	For writing
	MW□□□□□+3	Data size	

(8) Data Size (PARAM06)

Set the data size (number of bits or words) for the read/write request.
Be sure that the last data address determined based on the offset, data address, and data size does not go beyond the scope of the data addresses.
The range of data sizes may differ, depending on the function code.
The following table gives the valid ranges of data sizes when using MODBUS/TCP as a protocol.

Table D.14 Valid Range of Data Sizes (MODBUS/TCP)

Function Code	Target Data Type	Function	Valid Range of Data Addresses
00H	–	Unused	Disable
01H	B	Reads coil state*1	1 to 2000
02H	B	Reads input relay state*1	1 to 2000
03H	W	Reads holding register content*2	1 to 125
04H	W	Reads input register content*2	1 to 125
05H	B	Changes single coil state*1	Disable
06H	W	Writes to single holding register*2	Disable
07H : 0EH	–	Unused	Disable
0FH	B	Changes multiple coil states*1	1 to 800
10H	W	Writes to multiple holding registers*2	1 to 100
11H : 15H	–	Unused	Disable
16H	W	Mask writes to holding register*2, *3	Disable
17H	W	Reads/Writes multiple holding registers*2, *3	Read: 1 to 125 Write: 1 to 100

* 1. Specify the number of bits
* 2. Specify the number of words
* 3. Invalid when Ethernet (218IF) is used.
Note: The data size in the table is represented in decimal number.

(9) Remote CPU Number (PARAM07)

Refer to *D.1.4 (9) Remote CPU Number (PARAM07)*.

(10) Offset (PARAM08, PARAM09, PARAM10, PARAM11)

Specify the offset addresses of the read data storage area and the write data source of the transmission side.
The address for the transmission side will be displaced by the number of words designated by the offset.

Note: 1. For more information, refer to *D.1.8 Relationship between Data Address, Data Size, and Offset for MSG-SND Function*.
2. The offset cannot be a negative value.

The offset parameter is prepared for each target data type.
The following table lists the offset parameters.

Table D.15 Offset Parameter List

Parameter	Contents	Description
PARAM08	Coil offset	Sets the coil's offset word address.
PARAM09	Input relay offset	Sets the offset word address of an input relay.
PARAM10	Input register offset	Sets the offset word address of an input register.
PARAM11	Holding register offset	Sets the offset word address of a holding register.

The valid offset parameter may differ, depending on the function code.
The following table lists the valid parameters for each function code.

Table D.16 Valid Parameter List for Offset of Each Function Code

Function Code	Function	Valid Offset Parameter
01H	Reads coil state	PARAM08
02H	Reads input relay state	PARAM09
03H	Reads holding register content	PARAM11
04H	Reads input register content	PARAM10
05H	Changes single coil state	PARAM08
06H	Writes to single holding register	PARAM11
0FH	Changes multiple coil states	PARAM08
10H	Writes to multiple holding registers	PARAM11
16H	Mask writes to holding register	PARAM11
17H	Reads/Writes multiple holding registers	PARAM11

(11) Reserved by System 1 (PARAM12)

Used by system (the channel number of communication buffer in use is stored).

Note: At the first scan during power up, make sure to set it to “0” by using the user program. After that, the register is used by system. Do not change the value thereafter with the user program.

(12) Reserved by System 2 (PARAM13 to PARAM16)

Used by system. Never change the value with the user program, etc.

D.1.7 Function Setting and Parameter Details for Non-procedural Protocol

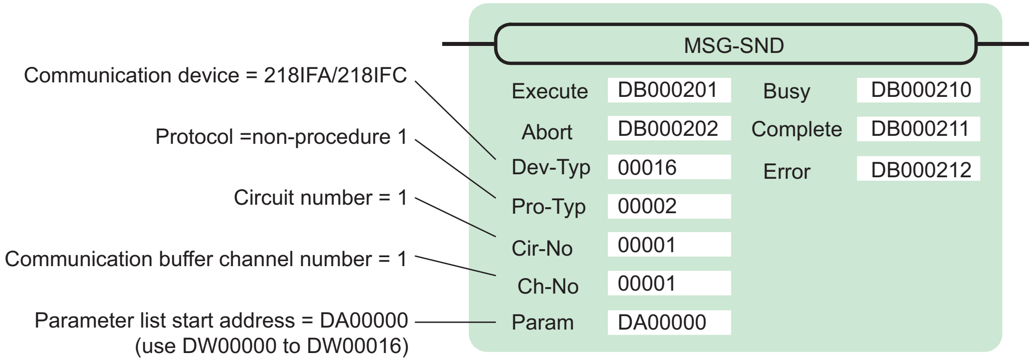
This section explains the MSG-SND function setting and its parameter list details when non-procedure is used as a protocol.

Note: Non-procedure communication protocol transmits the M register content intact without a protocol conversion. You can create any protocol in accordance with the remote device.

(1) Message Send Function Setting

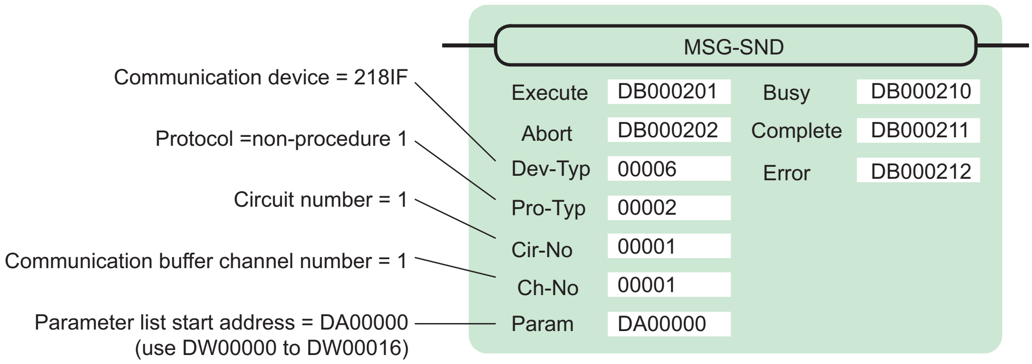
(a) 218IFA/218IFC Setting Example

An example of a function setting when 218IFA/218IFC is used as a communication device is as follows:
When non-procedure 2 (per byte) is used, set the Pro-Typ field to “00003” (non-procedure 1 (per word)).
Set the circuit number in accordance with the circuit number allocated to the target 218IFA/218IFC.
Set a unique communication buffer channel number for the same circuit.
For information on the register number, refer to *D.1.2 (1) Input Item* and *D.1.2 (2) Output Item*.



(b) 218IF Setting Example

An example of a function setting when 218IF is used as a communication device follows:
When non-procedure 2 (per byte) is used, set the Pro-Typ field to “00003” (non-procedure 1 (per word)) .
Set the circuit number in accordance with the circuit number allocated to the target 218IF.
Set a unique communication buffer channel number for the same circuit.
For information on the register number, refer to *D.1.2 (1) Input Item* and *D.1.2 (2) Output Item*.



(2) Process Result (PARAM00)

The process result is output to the upper byte. The lower byte is used for system analysis.

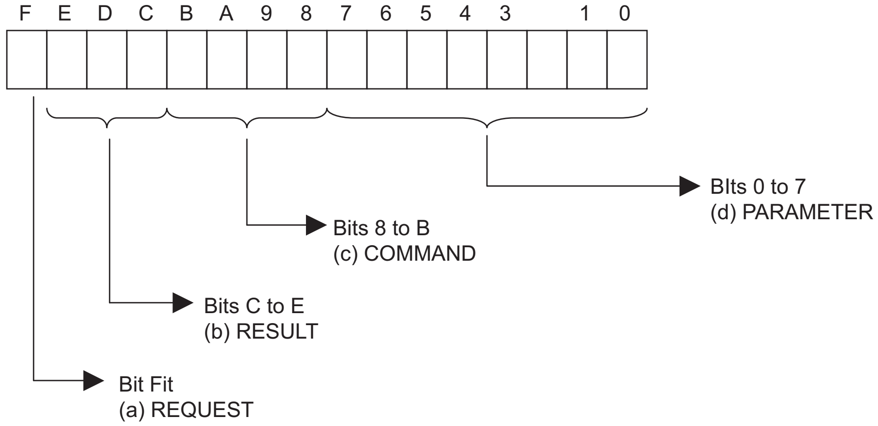
Value of Process Result	Meaning
00xxH	In process (Busy)
10xxH	Process completed (Complete)
8yxxH	Error occurred (Error)

When an error occurs, take corrective action by referring to the following error contents:

Value of Process Result	Error Contents	Description
80xxH	–	Reserved
81xxH	–	–
82xxH	Error in setting address	The following setting is out of range. Check the setting. PARAM05 (data address) PARAM11 (register offset)
83xxH	Data size error	The transmit or receive data amount is out of range. Check PARAM06 (data size).
84xxH	Error in setting circuit number	The circuit number is out of range. Check Cir-No (circuit number) of the MSG-SND function.
85xxH	Error in setting channel number	The communication buffer channel number is out of range. Check Ch-No (communication buffer channel number) of the MSG-SND function.
86xxH	Connection number error	The connection number is out of range. Check PARAM02 (connection number).
87xxH	–	Reserved
88xxH	Communication section error	An error response was returned from the communication section (communication device). Check the equipment connection. In addition, confirm that the remote device is open to communication.
89xxH	Device select error	Unavailable device is set. Check Dev-Typ (communication device type) of the MSG-SND function.

(3) Status (PARAM01)

Outputs status of the communication section (communication device).
The following figure illustrates bit assignment. Bit assignment details are listed in the table below.



(a) REQUEST (request)

Outputs whether MSG-SND function is requesting a process.

Bit State	Contents
1	Requesting processing
0	The acceptance of process request is completed

(b) RESULT (result)

Outputs an execution result of the MSG-SND function

Code	Abbreviation	Meaning
0	CONN_NG	In Ethernet communications, transmit error or connection error is complete
1	SEND_OK	Normal transmission complete
2	REC_OK	Normal reception complete
3	ABORT_OK	Forced abort complete
4	FMT_NG	Parameter format error
5	SEQ_NG	Command sequence error
6	RESET_NG	Reset state
7	REC_NG	Data reception error (error detected in the lower layer program)

(c) COMMAND (command)

Outputs a process command for the MSG-SND function. The executed process contents can be found according to the command.

Code (Hex)	Abbreviation	Meaning
1	U_SEND	General-purpose message transmission
2	U_REC	General-purpose message reception
3	ABORT	Forced abort
8	M_SEND	MEMOBUS command transmission: Complete when response is received (for MEMOBUS protocol)
9	M_REC	MEMOBUS command reception: Accompanies a response transmission (for MEMOBUS protocol)
C	MR_SEND	MEMOBUS response transmission (for MEMOBUS protocol)

(d) PARAMETER (parameter)

When RESULT (process result) = 4 (FMT_NG: parameter format error), an error code in the table below is output. Otherwise, the connection number is output.

RESULT (process result)	Code (Hex)	Meaning
When RESULT (process result) = 4 (FMT_NG: parameter format error)	00	No error
	01	Connection number is out of range
	02	Time error while monitoring to receive MEMOBUS response (for MEMOBUS protocol)
	03	Error in setting retransmit count
	04	Error in setting cyclic area
	05	CPU number error
	06	Data address error
	07	Data size error
	08	Function code error (for MEMOBUS protocol)
Others	xx	Connection number

(4) Connection Number (PARAM02)

Refer to *D.1.4 (4) Connection Number (PARAM02)*.

(5) Data Address (PARAM05)

Specify the start address of the data.
The address must be input in a decimal or hexadecimal number.
Example: When the start address is MW01000, specify “1000 (decimal)” or “3E8H (hexadecimal).”
The following table provides the valid ranges of the data addresses.

Table D.17 Valid Range of Data Addresses (non-procedure)

Non-procedural Type	Target Data Type	Function	Valid Range of Data Addresses
			Ethernet(218IF/218IFA/218IFC)
Non-procedure 1	W	Transmits data in words	0 to 65534 (0 to FFFEH)
Non-procedure 2	B	Transmits data in bytes*1	0 to 65534 (0 to FFFEH)

* 1. The unit for address setting is word address.

(6) Data Size (PARAM06)

Set the data size (number of bits or words) for the write request.
Be sure that the last data address determined based on the offset, data address, and data size does not go beyond the scope of the data addresses.
The valid range of data sizes may differ according to the communication device.

Table D.18 Valid Range of Data Sizes (non-procedure)

Non-procedural Type	Target Data Type	Function	Valid Range of Data Sizes	
			Ethernet (218IF)	Ethernet (218IFA/ 218IFC)
Non-procedure 1	W	Transmits data in words*1	1 to 510	1 to 2046 (BIN) 1 to 1023 (ASCII)
Non-procedure 2	B	Transmits data in bytes*2	1 to 1020	1 to 4092 (BIN) 1 to 2046 (ASCII)

* 1. Specify the number of words
* 2. Specify the number of bytes

Note: The data size in the table is represented in decimal numbers.

(7) Register Offset (PARAM11)

Specify the offset address of write data source in the transmission side.
The address for the transmission side will be displaced by the number of words designated by the offset.

Note: 1. For more information, refer to *D.1.8 Relationship between Data Address, Data Size, and Offset for MSG-SND Function*.
2. The offset cannot be a negative value.
Example: When specifying 1000 words of offset for the register address:
PARAM11=1000

(8) Reserved by System 1 (PARAM12)

Used by system (the channel number of the communication buffer in use is stored).

Note: At the first scan during power up, make sure to set this to “0” by the user program. After that, the register is used by system, so do not change the value with the user program.

(9) Reserved by System 2 (PARAM13-PARAM16)

Used by system. Do not change the value with the user program, etc.

D.1.8 Relationship between Data Address, Data Size, and Offset for MSG-SND Function

The relationship between data address, data size, and offset is as follows, when transmitted with offset:

(1) When Reading

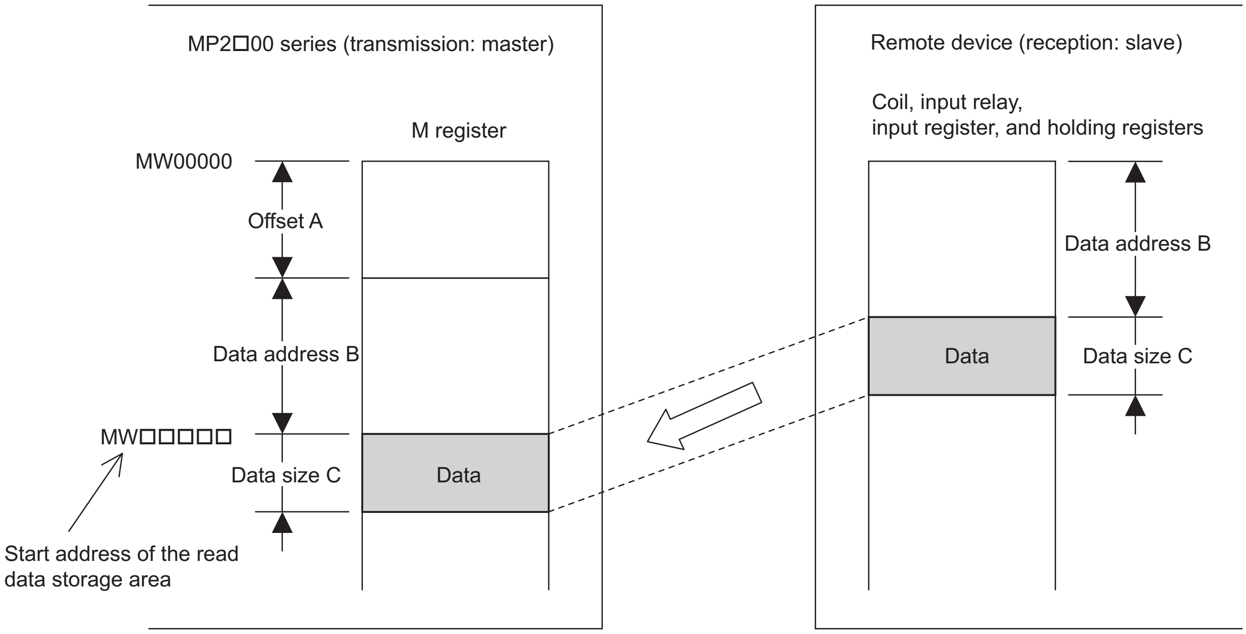


Fig. D.4 Data Flow when Reading

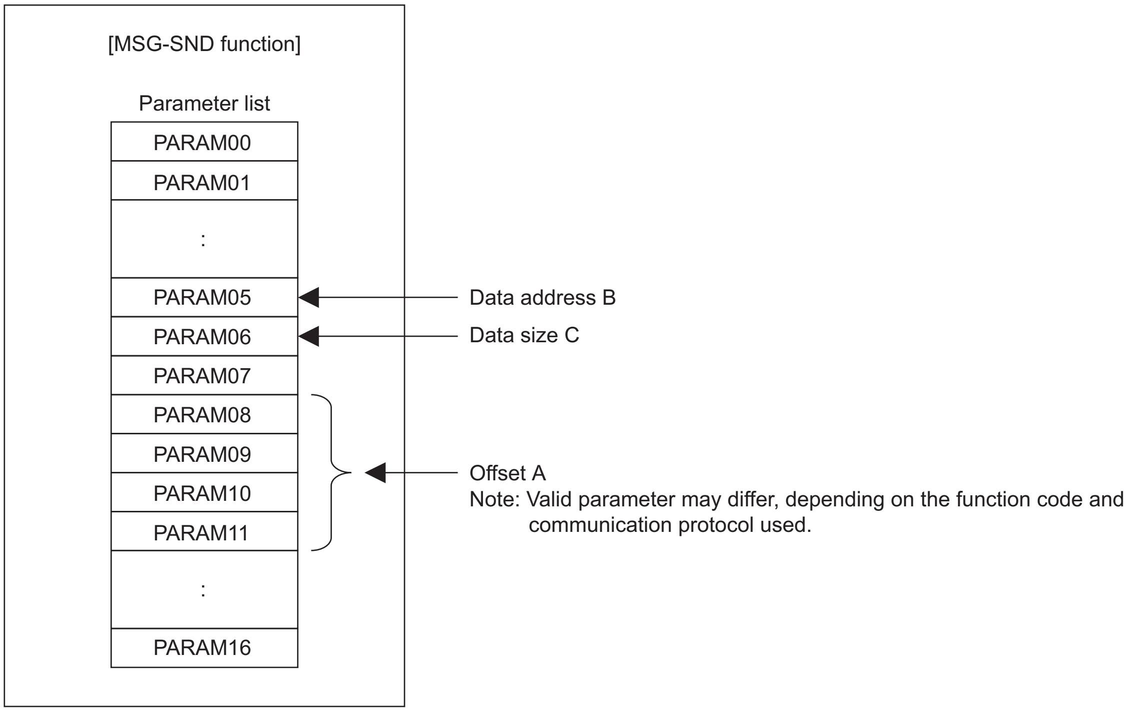


Fig. D.5 Parameter Setting

(2) When Writing

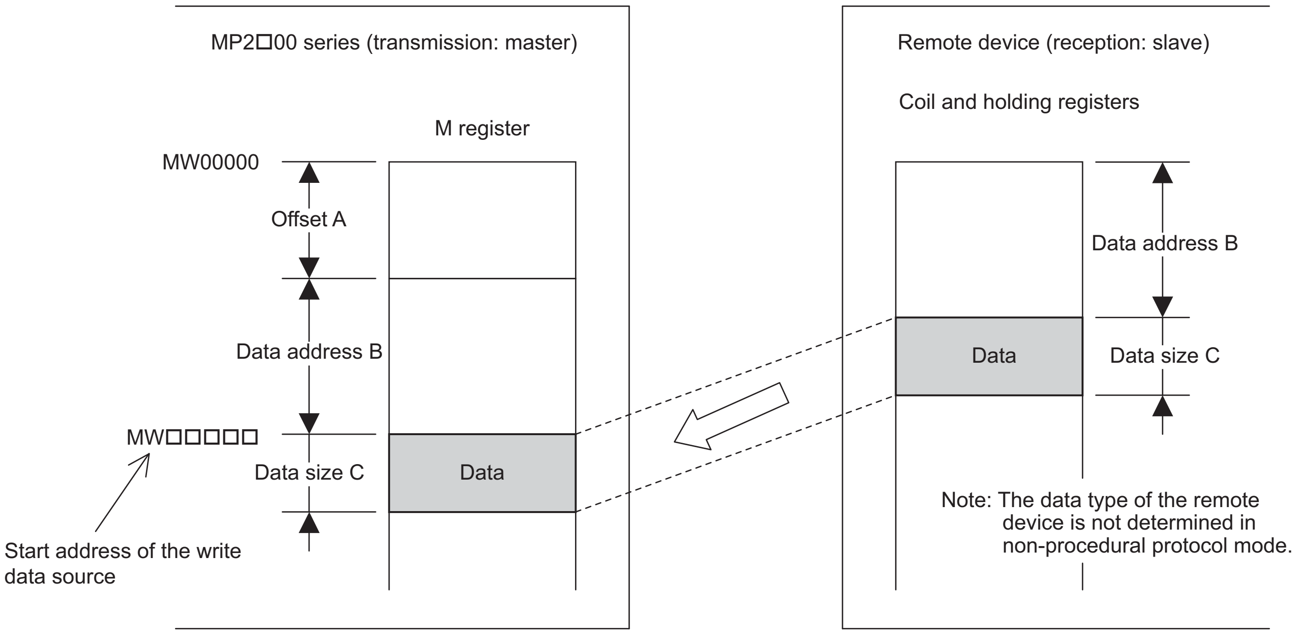


Fig. D.6 Data Flow when Writing

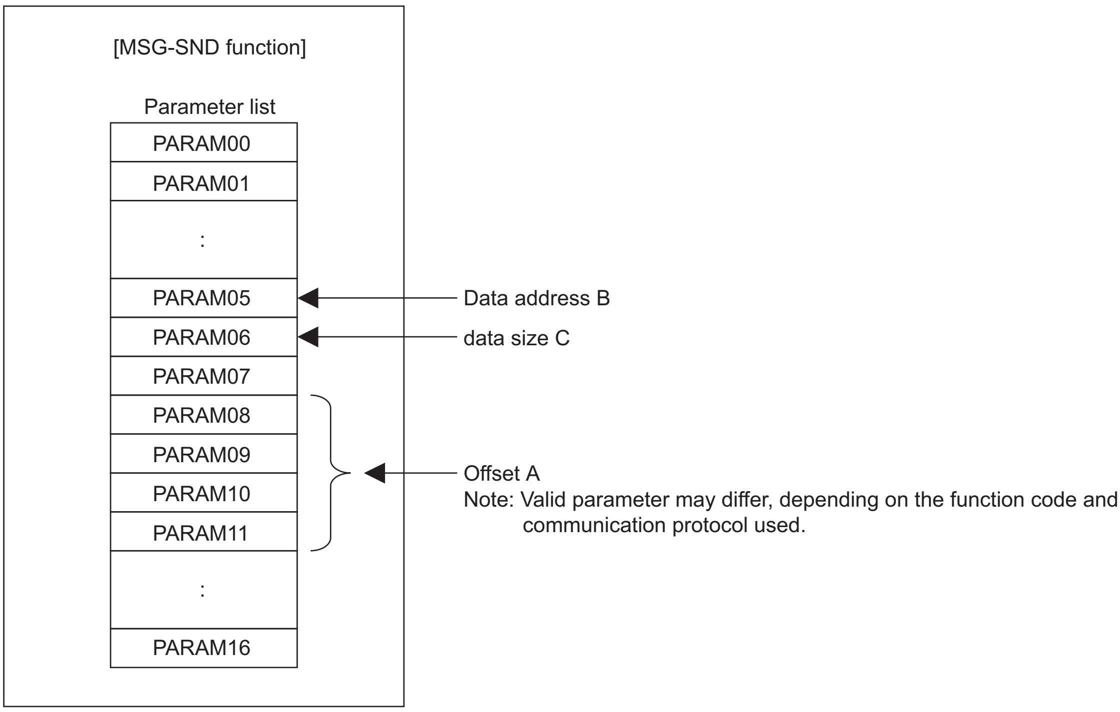
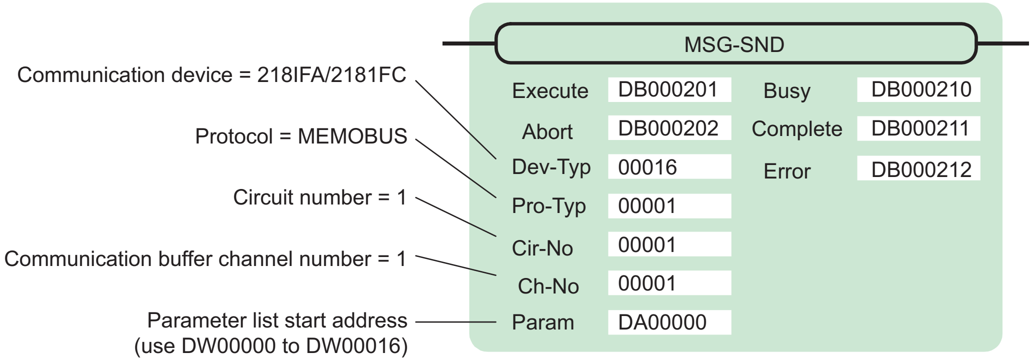


Fig. D.7 Parameter Setting

Example: When reading coil state with offset:

The various setting values and their relationships with the data of the remote device are as follows, when transmitting "reading coil state" with offset in MEMOBUS protocol:

- Description of the MSG-SND function in ladder program



- Parameter list setting of the MSG-SND function

Table D.19 Parameter List Setting

Register Number	Setting Value	Parameter Number	IN/OUT	Remarks
DW00000	–	PARAM00	OUT	Process result
DW00001	–	PARAM01	OUT	Status
DW00002	00001	PARAM02	IN	Connection number=1
DW00003	–	PARAM03	IN	Option (setting unnecessary)
DW00004	00001	PARAM04	IN	Function code=1 (reads coil state)
DW00005	08192	PARAM05	IN	Data address=8192 bits (512 words)
DW00006	00100	PARAM06	IN	Data size=100
DW00007	00001	PARAM07	IN	Remote CPU number=1
DW00008	01000	PARAM08	IN	Coil offset=1000 words
DW00009	00000	PARAM09	IN	Input relay offset=0 word
DW00010	00000	PARAM10	IN	Input register offset=0 word
DW00011	00000	PARAM11	IN	Holding register offset=0 word
DW00012	–	PARAM12	SYS	Reserved (zero clear at startup)
DW00013	–	PARAM13	SYS	Reserved
DW00014	–	PARAM14	SYS	Reserved
DW00015	–	PARAM15	SYS	Reserved
DW00016	–	PARAM16	SYS	Reserved

Note: IN: Input, OUT: Output, SYS: For system use

- Relationship with the Data of the Remote Device
  The following figure shows the data flow when transmitting “reads coil state” with offset:
  When transmission and reception are carried out normally in the figure below, the coil state in the device is stored in MW01512 and after of MP2200.

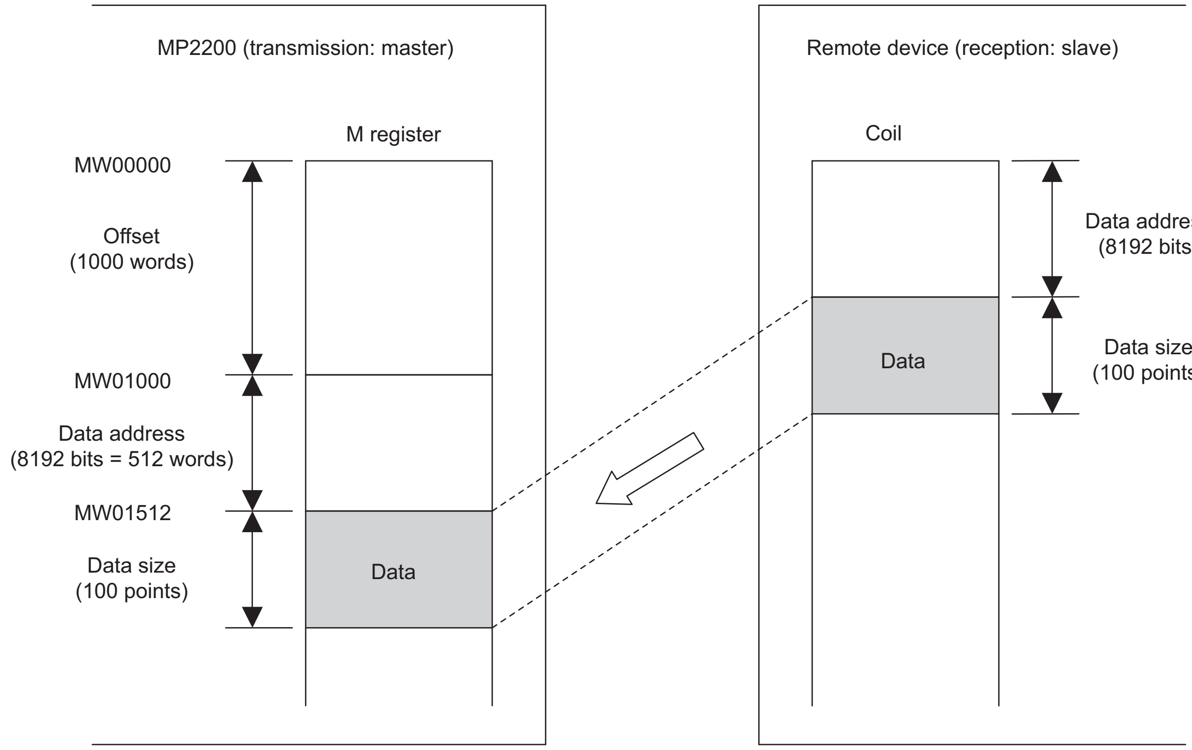
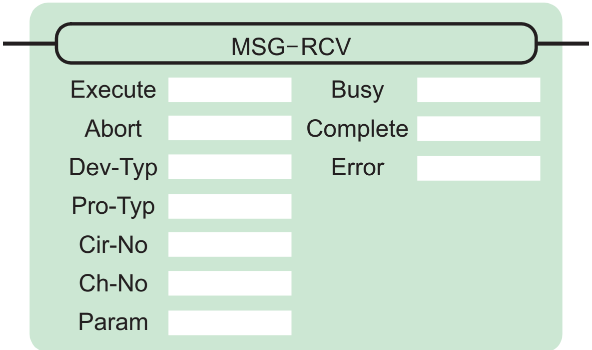


Fig. D.8 Reading Example of Coil State

D.2 Message Receive Function (MSG-RCV)

This section explains how the message receive function (MSG-RCV) is used in a ladder program when receiving messages.

D.2.1 Specification Overview of the Message Receive Function

Function Name	MSG-RCV			
Function	Receives messages from the remote station on the circuit specified by the communication device type. Supports multiple protocol types. Keep the execution command (Execute) until Complete or Error turns ON.			
Function Definition				
I/O Definition	No.	Name	I/O Option (*1)	Contents
Input Item	1	Execute	B-VAL	Executes a reception
	2	Abort	B-VAL	Forcibly ends a reception
	3	Dev-Typ	I-REG	Communication device type Ethernet (218IF) = 6, Ethernet (218IFA/218IFB/218IFC) = 16
	4	Pro-Typ	I-REG	Communication protocol MEMOBUS*2 = 1, non-procedure 1*3 = 2, non-procedure 2*3 = 3
	5	Cir-No	I-REG	Line number Ethernet (218IF/218IFA/218IFC) = 1 to 8
	6	Ch-No	I-REG	Communication buffer channel number Ethernet (218IF/218IFC) = 1 to 10, Ethernet (218IFA) = 1 to 4
	7	Param	Address input	Parameter list start address (MA, DA)
Output Item	1	Busy	B-VAL	In process
	2	Complete	B-VAL	Process completed
	3	Error	B-VAL	Error occurred

* 1. The meanings of I/O options are as follows:
  B-VAL: Specify I/O by bit type data.
  I-REG: Specify I/O by integer type data. When specifying, set an integer type register number.
  As for the input only, it can be a constant (literal).
  Address input: The address of the specified register (any integer register) is passed to the function.
* 2. When transmitting in MEMOBUS, Extended MEMOBUS, MELSEC, OMRON, or MODBUS/TCP protocol, set the communication protocol (Pro-Typ) to MEMOBUS(=1). The communication device automatically converts the protocol.
* 3. Non-procedure 1: In non-procedural communication, data is received on a per-word basis.
  Non-procedure 2: In non-procedural communication, data is received on a per-byte basis.

D.2.2 I/O Item Details of the Message Receive Function

(1) Input Item

The following table indicates the registers available for each input item.

Input Item	I/O Option	Available Register
Execute Abort	B-VAL	Every bit type register (except #, C registers), Same as above with subscript
Dev-typ Pro-Typ Cir-No Ch-No	I-REG	Every integer type register, Same as above with subscript, Constant
Param	Address input	Register address (except #, C registers), Same as above with subscript

1. Execute (executes a reception)

Specify a bit to command execution of a message reception.
When Execute turns ON, the message reception is carried out. In order to execute the process, a ladder program or the like needs to be used to switch the bit ON/OFF.

Note: Keep Execute (executes a reception) until Complete (process completed) or Error (error occurred) is turned ON. When the command turns ON, the message reception is carried out.
To continuously command the receive execution, make sure to turn Execute (executes a reception) OFF for one scan or more.

2. Abort (forcibly ends a reception)

Specify a bit to command a forced abort of a message reception.
When Abort turns ON, the message reception is forcibly terminated. Abort takes precedence over Execute.
In order to execute the forced abort, a ladder program or the like needs to be used to switch the bit ON/OFF.

3. Dev-Typ (communication device type)

Specify the type of communication device.

Device	Type Code
Ethernet (218IF)	6
Ethernet (218IFA/218IFC)	16

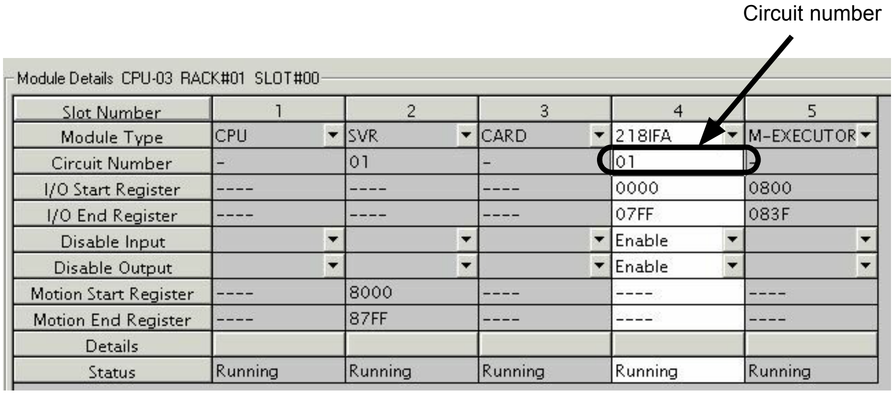
4. Pro-Typ (communication protocol)

Specify the communication protocol.

Type Code	Communication Protocol	Remarks
1	MEMOBUS	Set the type code to “1” when also transmitting using Extended MEMOBUS, MELSEC, or MODBUS/TCP protocol. The communication device will automatically convert the protocol.
2	Non-procedure 1 (per word)	Data is received on a per-word basis in non-procedural communication. No response is transmitted to the remote.
3	Non-procedure 2 (per byte)	Data is received on a per-byte basis in non-procedural communication. No response is transmitted to the remote.

5. Cir-No (circuit number)

Specify a circuit number for the communication device.
Specify it in accordance with the circuit number displayed in the MPE720 **Module Configuration** Window.



Slot Number	1	2	3	4	5
Module Type	CPU	SVR	CARD	218IFA	M-EXECUTOR
Circuit Number	-	01	-	01	-
I/O Start Register	----	----	----	0000	0800
I/O End Register	----	----	----	07FF	083F
Disable Input				Enable	
Disable Output				Enable	
Motion Start Register	----	8000	----	----	----
Motion End Register	----	87FF	----	----	----
Details					
Status	Running	Running	Running	Running	Running

Fig. D.9 MPE720 Module Configuration Window

The following table indicates the range of valid circuit numbers.

Communication device	Valid Circuit Number
Ethernet (218IF/218IFA/218IFC)	1 to 8

6. Ch-No (communication buffer channel number)

Specify the channel number of a communication buffer.
It can be any channel number in the range. However, when starting multiple functions at the same time, set a unique channel for each function. (If you do not start up multiple functions at the same time, the channel numbers can duplicate each other.)
The following table indicates the range of valid channel numbers.

Communication device	Valid Channel Number
Ethernet (218IF/218IFC)	1 to 10
Ethernet (218IFA)	1 to 4

When the communication device is Ethernet (218IFA), because the communication buffer common to the transmission and reception have four channels, four receptions (or transmissions) are available at the same time by using channel numbers 1 to 4.
In the same way, when the communication device is Ethernet (218IFC), because the communication buffers common to the transmission and reception has ten channels, ten transmissions (or receptions) are available at the same time by using channels 1 to 10.

Note: 1. As many MSG-RCV (or MSG-SND) functions as lines used at the same time are required.
2. For information on communication buffer channel, refer to *D.3 Communication Buffer Channel* .

7. Param (parameter list start address)

Specify the start address of the parameter list. For the “parameter list,” 17 words are automatically assigned from the configured address. In the parameter list, enter the function code and its relevant parameter data. Additionally, process result and status are output.

Note: For more information about the parameter list, refer to the parameter details for each protocol from *D.2.4 Function Setting and Parameter Details for MEMOBUS and Extended MEMOBUS Protocols* to *D.2.7 Function Setting and Parameter Details for Non-procedural Protocol*.

Example: When “DA00000” is specified as a parameter list start address:

Register	Parameter list F ··· ···· ···· ··· 0
DW00000	PARAM00
DW00001	PARAM01
DW00002	PARAM02
DW00003	PARAM03
DW00004	PARAM04
DW00005	PARAM05
DW00006	PARAM06
DW00007	PARAM07
DW00008	PARAM08
DW00009	PARAM09
DW00010	PARAM10
DW00011	PARAM11
DW00012	PARAM12
DW00013	PARAM13
DW00014	PARAM14
DW00015	PARAM15
DW00016	PARAM16

(2) Output Item

The following table lists the registers available for each output item.

Input Item	I/O Option	Available Register
Busy Complete Error	B-VAL	Every bit type register (except #, C registers), Same as above for subscript

1. Busy (in process)

Specify a bit to report message reception.
The Busy bit is turned ON while executing message reception or a forced abort process.
Keep Execute or Abort ON while Busy is ON.

2. Complete (process completed)

Specify a bit to report message reception ended.
When message reception or a forced abort process is completed properly, the Complete bit will turn ON only for one scan.

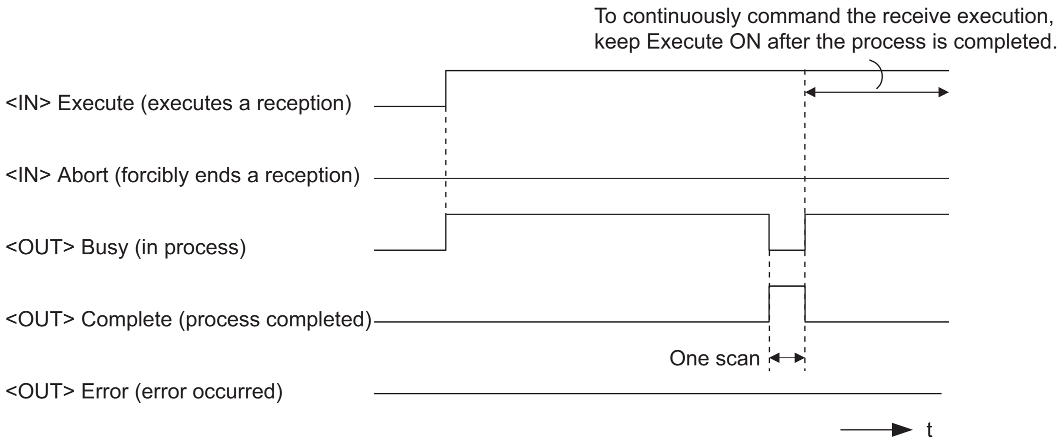
3. Error (error occurred)

Specify a bit to report when an error occurs in the message reception.
When an error occurs, the Error bit will turn ON only for one scan.

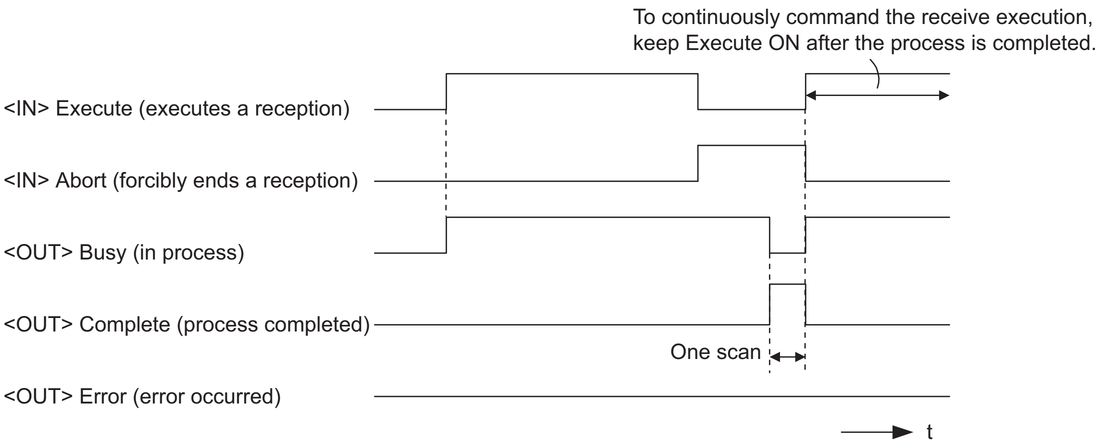
Note: For more information about the error cause, refer to *D.2.4 (2) Process Result (PARAM00)* and *D.2.4 (3) Status (PARAM01)*.

A timing chart of bit type I/O items in the MSG-RCV function follows:

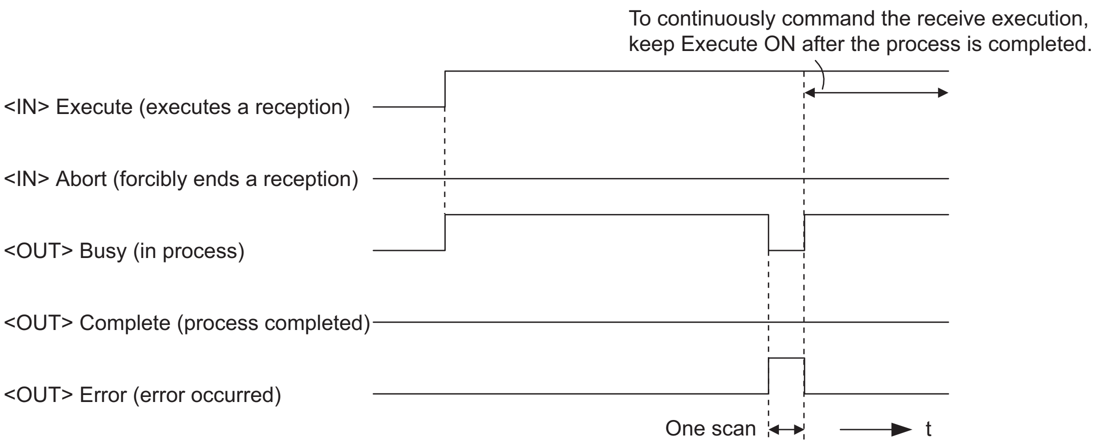
[In Normal Condition]



[When Forcibly Aborted]



[When Error Occurred]



D.2.3 Message Receive Function Parameter List (Param) Overview

Param of the MSG-RCV function has a parameter list structure composed of 17 words. (The value of Param itself is the start address (MA, DA) of the parameter list.)
In the parameter list, enter the function code and its relevant parameter data. Process result and status are also output.
When MEMOBUS and non-procedure are used as a transmission protocol, the parameter list is as follows:

Note: Parameter details are explained in the parameter details for each protocol type. Refer to the following items:

- *D.2.4 Function Setting and Parameter Details for MEMOBUS and Extended MEMOBUS Protocols*
- *D.2.5 Function Setting and Parameter Details for MELSEC Protocol*
- *D.2.6 Function Setting and Parameter Details for MODBUS/TCP Protocol*
- *D.2.7 Function Setting and Parameter Details for Non-procedural Protocol*

(1) MEMOBUS Parameter List

Param No.	IN/OUT	Contents	Description
00	OUT	Process result	Outputs process results.
01	OUT	Status	Outputs the status of the communication device.
02	IN	Connection number	Specifies the remote source.
03	OUT	Option	Outputs a unique value for each communication device.
04	OUT	Function code	Outputs a function code requested from the transmission side.
05	OUT*	Data address	Outputs the start address of data requested from the transmission side.
06	OUT	Data size	Outputs the read/write data size requested from the transmission side.
07	OUT	Remote CPU number	Outputs the remote CPU number.
08	IN	Coil offset	Sets the coil’s offset word address.
09	IN	Input relay offset	Sets the offset word address of an input relay.
10	IN	Input register offset	Sets the offset word address of an input register.
11	IN	Holding register offset	Sets the offset word address of a holding register.
12	IN	Write range LO	Sets the start address for a write range.
13	IN	Write range HI	Sets the last address for a write range.
14	SYS	Reserved 1	
15, 16	SYS	Reserved 2	

* It is IN/OUT for MODBUS/TCP.

Note: IN: Input, OUT: Output, SYS: For system use

(2) Non-procedural Parameter List

Param No.	IN/OUT	Contents	Description
00	OUT	Process result	Outputs the process results.
01	OUT	Status	Outputs the status of the communication device.
02	IN	Connection number	Specifies the remote source.
03	OUT	Not used	
04	OUT	Not used	
05	OUT	Not used	
06	OUT	Data size	Outputs the write data size requested from the transmission side.
07	OUT	Not used	
08 to 11	IN	Not used	
12	IN	Register offset	Sets the register's offset word address.
13	IN	Write range HI	Sets the last address for a write range.
14	SYS	Reserved 1	
15, 16	SYS	Reserved 2	

Note: IN: Input, OUT: Output, SYS: For system use

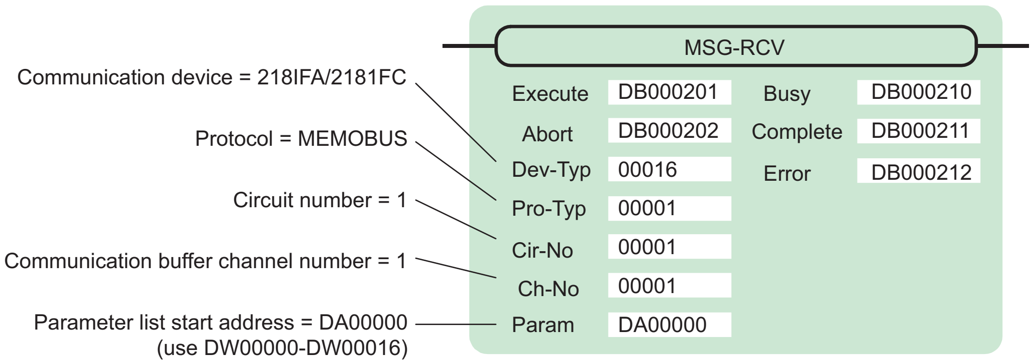
D.2.4 Function Setting and Parameter Details for MEMOBUS and Extended MEMOBUS Protocols

This section explains the MSG-RCV function setting and its parameter list details when MEMOBUS or Extended MEMOBUS is used as a protocol.

(1) Message Receive Function Setting

[a] 218IFA/218IFC Setting Example

An example of a function setting when 218IFA/218IFC is used as a communication device follows:
Set the protocol type to MEMOBUS even when used in Extended MEMOBUS protocol.
Set the circuit number in accordance with the circuit number allocated to the target 218IFA/218IFC.
Set a unique communication buffer channel number for the same circuit.
For information on the register number, refer to *D.2.2 (1) Input Item* and *D.2.2 (2) Output Item*.

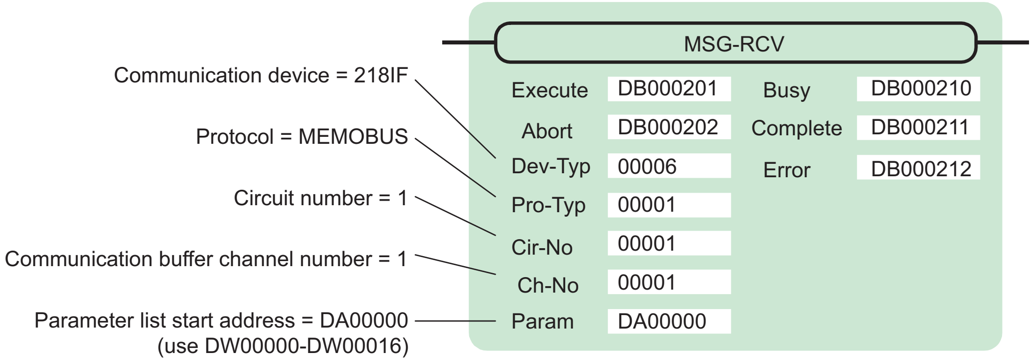


Appendices

A

[b] 218IF Setting Example

An example of a function setting when 218IF is used as a communication device follows:
Set the protocol type to MEMOBUS even when used in Extended MEMOBUS protocol.
Set the circuit number in accordance with the circuit number allocated to the target 218IF.
Set a unique communication buffer channel number for the same circuit.
For information on the register number, refer to *D.2.2 (1) Input Item* and *D.2.2 (2) Output Item*.



(2) Process Result (PARAM00)

The process result is output to the upper byte. The lower byte is used for system analysis.

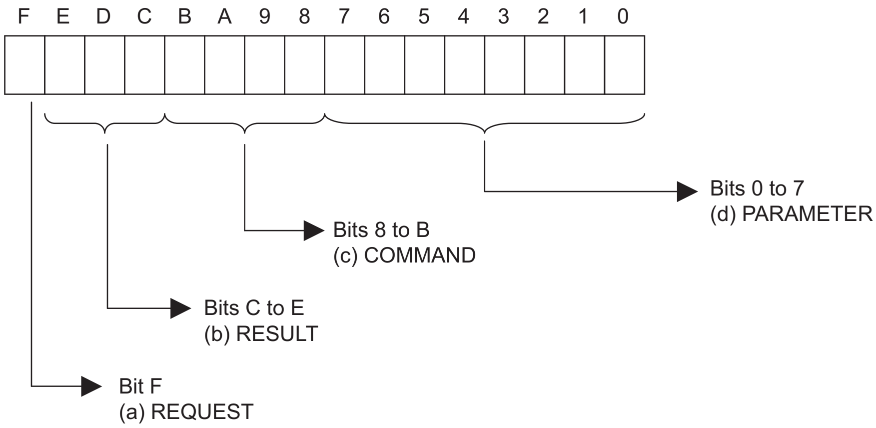
Value of Process Result	Meaning
00xxH	In process (Busy)
10xxH	Process completed (Complete)
8yxxH	Error occurred (Error)

When an error occurs, take corrective action by referring to the following error contents:

Value of Process Result	Error Contents	Description
80xxH	–	Reserved
81xxH	Function code error	Unused function code was received. Check the source function code. The following setting is out of range. Check the setting.
82xxH	Error in setting address	Data address (request from transmission side) PARAM08 (coil offset) PARAM09 (input relay offset) PARAM10 (input register offset) PARAM11 (holding register offset)
83xxH	Data size error	The received data size is out of range. Check the source data size.
84xxH	Error in setting circuit number	The circuit number is out of range. Check Cir-No (circuit number) of the MSG-RCV function.
85xxH	Error in setting channel number	The communication buffer channel number is out of range. Check Ch-No (communication buffer channel number) of the MSG-RCV function.
86xxH	Connection number error	The connection number is out of range. Check PARAM02 (connection number).
87xxH	–	Reserved
88xxH	Communication section error	An error response was returned from the communication section (communication device). Check the equipment connection. In addition, check that the remote device is open to communication.
89xxH	Device select error	Unavailable device is set. Check Dev-Typ (communication device type) of the MSG-RCV function.

(3) Status (PARAM01)

Outputs status of the communication section (communication device).
The following figure shows the bit assignment. Bit assignment details are listed in the tables following.



[a] REQUEST (request)

Outputs whether MSG-RCV function is requesting a process.

Bit State	Contents
1	Requesting processing
0	The acceptance of process request is completed

[b] RESULT (result)

Outputs the execution result of the MSG-RCV function.

Code	Abbreviation	Meaning
0	CONN_NG	In Ethernet communication, transmission or connection error is completed
1	SEND_OK	Normal transmission complete
2	REC_OK	Normal reception complete
3	ABORT_OK	Forced abort complete
4	FMT_NG	Parameter format error
5	SEQ_NG	Command sequence error
6	RESET_NG	Reset state
7	REC_NG	Data reception error (error detected in the lower layer program)

[c] COMMAND (command)

Outputs a process command for the MSG-RCV function. The executed process contents can be found according to the command.

Code (Hex)	Abbreviation	Meaning
1	U_SEND	General-purpose message transmission (for non-procedural protocol)
2	U_REC	General-purpose message reception (for non-procedural protocol)
3	ABORT	Forced abort
8	M_SEND	MEMOBUS command transmission: Completed when response is received
9	M_REC	MEMOBUS command reception: Accompanies a response transmission
C	MR_SEND	MEMOBUS response transmission

[d] PARAMETER (parameter)

When RESULT(process result) = 4 (FMT_NG: parameter format error), an error code in the table below is output. Otherwise, the connection number is output.

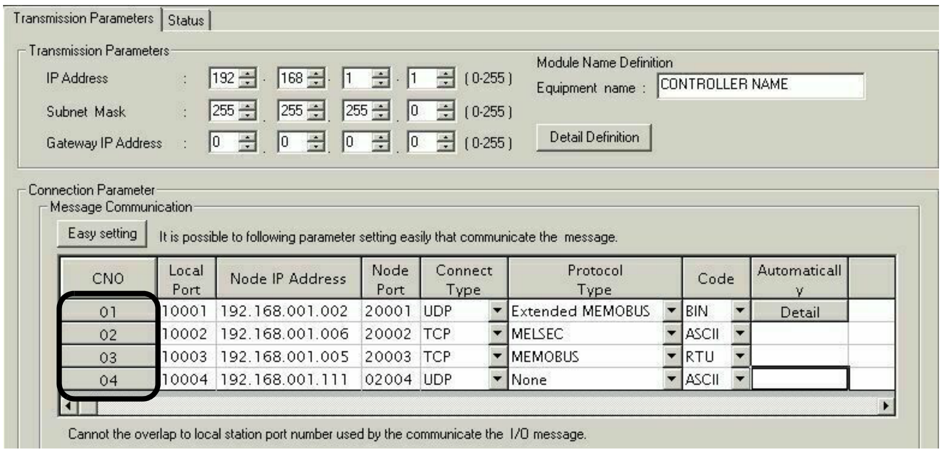
RESULT (process result)	Code (Hex)	Meaning
When RESULT (process result) =4 (FMT_NG: parameter format error)	00	No error
	01	Connection number is out of range
	02	Time error for monitoring to receive MEMOBUS response
	03	Error in setting retransmit count
	04	Error in setting cyclic area
	05	CPU number error
	06	Data address error
	07	Data size error
	08	Function code error
Others	xx	Connection number

(4) Connection Number (PARAM02)

Specify the remote source.
When the communication device is Ethernet (218IF, 218IFA or 218IFC), set the connection number.
The following table shows the setting range.

Communication Device	Connection Number	Remarks
Ethernet (218IF/218IFC)	1 to 20	Receives from the remote station set for the specified connection number.
Ethernet (218IFA)	1 to 4	Same as above

Note: When the communication device is Ethernet (218IF/218IFA/218IFC), set the connection number in accordance with the connection number in the 218IF/218IFA/218IFC **Transmission Parameters** Tab Page for the MPE720 **Module Configuration** Window.



Note: For the 218IFC, the setting range for the connection number is larger, i.e., 01 to 20.

Fig. D.10 218IFA/218IFC Transmission Parameters Tab Page for the MPE720 Module Configuration Window

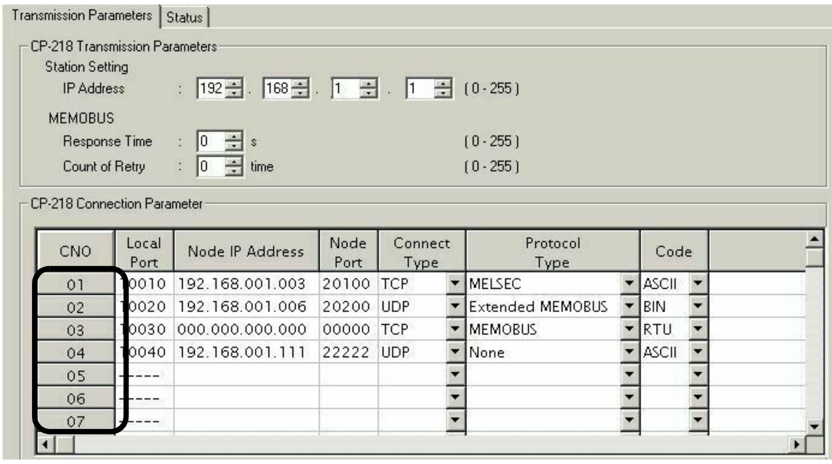


Fig. D.11 218IF Transmission Parameters Tab Page for the MPE720 Module Configuration Window

(5) Option (PARAM03)

A unique value is output for each communication device.
Not used for the MEMOBUS or Extended MEMOBUS protocols.

(6) Function Code (PARAM04)

Received function code is output.
The following table lists function codes available when using MEMOBUS or Extended MEMOBUS as a protocol.

Table D.20 Function Code List (MEMOBUS, Extended MEMOBUS)

Function Code	Target Data Type	Function	Protocol	
			Extended MEMOBUS	MEMOBUS
00H	–	Unused	–	–
01H	B	Reads coil state	√	√
02H	B	Reads input relay state	√	√
03H	W	Reads holding register content	√	√
04H	W	Reads input register content	√	√
05H	B	Changes single coil state	√	√
06H	W	Writes to single holding register	√	√
07H	–	Unused	–	–
08H	–	Loopback test	√	√
09H	W	Reads holding register content (extended)	√	–
0AH	W	Reads input register content (extended)	√	–
0BH	W	Writes to holding register (extended)	√	–
0CH	–	Unused	–	–
0DH	W	Discontinuously reads holding register (extended)	√	–
0EH	W	Discontinuously writes to holding register (extended)	√	–
0FH	B	Changes multiple coil states	√	√
10H	W	Writes to multiple holding registers	√	√

Note: 1. B: Bit type, W: Integer type
2. √: Available, -: Not available
3. Transmit and receive registers in the master operation mode are MW (MB) only.
4. In the slave operation mode, coil, holding register, input relay, and input register are MB, MW, IB, IW respectively.

(7) Data Address (PARAM05)

A data address requested from the transmission side is output.

(8) Data Size (PARAM06)

The read/write data size (number of bits or words) requested from the transmission side is output.

(9) Remote CPU Number (PARAM07)

When the remote device is MP2☐00 series, “1” is output.
When the remote device is a controller manufactured by YASKAWA Electric Corporation other than MP2☐00 series and is comprised of multiple CPU modules, the remote CPU number is output.
Otherwise, “0” is output.

(10) Offset (PARAM08, PARAM09, PARAM10, PARAM11)

Specify an offset address for the reception side data address.
The address for the reception side will be displaced by the number of words designated by the offset.

Note: 1. For more information, refer to *D.2.8 Relationship between Data Address, Data Size, and Offset for MSG-RCV Function*.
2. The offset cannot be a negative value.

The offset parameter is prepared for each target data type.
The following table lists the offset parameters.

Table D.21 Offset Parameter List

Parameter	Contents	Description
PARAM08	Coil offset	Sets the coil's offset word address.
PARAM09	Input relay offset	Sets the offset word address of an input relay.
PARAM10	Input register offset	Sets the offset word address of an input register.
PARAM11	Holding register offset	Sets the offset word address of a holding register.

The valid offset parameter may differ according to the function code.
The following table lists the valid parameters for each function code.

Table D.22 Valid Parameter List for Offset of Each Function Code

Function Code	Function	Valid Offset Parameter	Protocol Type	
			Extended MEMOBUS	MEMOBUS
01H	Reads coil state	PARAM08	√	√
02H	Reads input relay state	PARAM09	√	√
03H	Reads holding register content	PARAM11	√	√
04H	Reads input register content	PARAM10	√	√
05H	Changes single coil state	PARAM08	√	√
06H	Writes to single holding register	PARAM11	√	√
09H	Reads holding register content (extended)	PARAM11	√	–
0AH	Reads input register content (extended)	PARAM10	√	–
0BH	Writes to holding register (extended)	PARAM11	√	–
0DH	Discontinuously reads holding register (extended)	PARAM11	√	–
0EH	Discontinuously writes to holding register (extended)	PARAM11	√	–
0FH	Changes multiple coil states	PARAM08	√	√
10H	Writes to multiple holding registers	PARAM11	√	√

Note: √: Available, -: Not available

(11) Write Range (PARAM12, PARAM13)

Sets an available address range for the write request from the transmission side. A write request which is out of this available address range will cause an error.
Specify the address range (PARAM12, PARAM13) as a word address.

Note: 1. In MP9□0/ MP2□00 series, the data storage area for the write request from the transmission side is M register.
2. The write range parameter enables you to specify the range of M register which permits writing messages.

The following table indicates the write range parameters.

Table D.23 Write Range Parameter List

Parameter	Contents	Description
PARAM12	Write range LO	Start address of the write range
PARAM13	Write range HI	Last address of the write range

Specify the write range so that the expression below is met:

$$0 \le \text{Write range LO} \le \text{Write range HI} \le \text{Maximum address of M register}$$

This write range is effective for the following function codes:

05H (changes single coil state)
06H (writes to single holding register)
0BH (writes to holding register (extended))
0EH (discontinuously writes to holding register (extended))
0FH (changes multiple coil states)
10H (writes to multiple holding registers)

Example: When setting the address of M register which permits writing to 1000 to 1999:
PARAM12=1000
PARAM13=1999
The reception side will return an error against a write request to an address other than MW01000 to MW01999, and will not write it.

(12) Reserved by System 1 (PARAM14)

Used by system (the channel number of the communication buffer in use is stored).

Note: At the first scan during power up, make sure to set this to “0” with the user program. After that, the register is used by system, so do not change the value with the user program.

(13) Reserved by System 2 (PARAM15, PARAM16)

Used by system. Do not change the value with the user program, etc.

Appendices

A

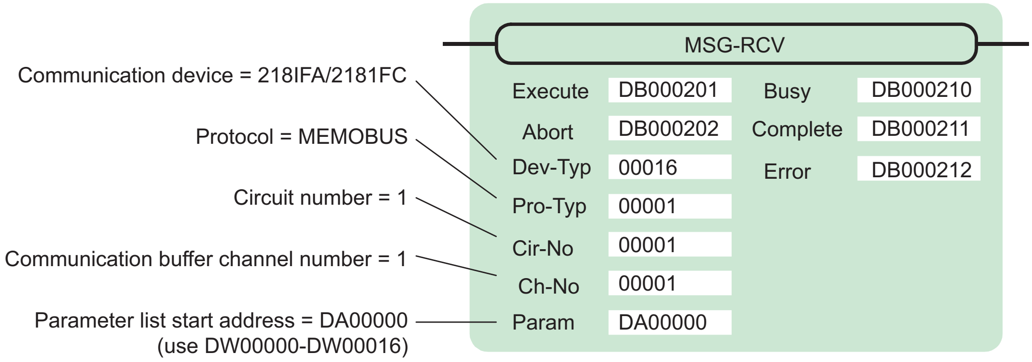
D.2.5 Function Setting and Parameter Details for MELSEC Protocol

This section explains the MSG-RCV function setting and its parameter list details when MELSEC is used as a protocol.

(1) Message Receive Function Setting

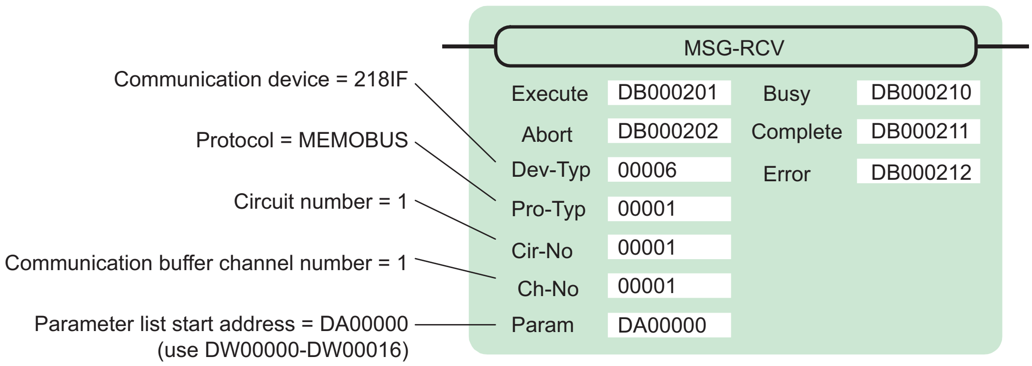
[a] 218IFA/218IFC Setting Example

An example of a function setting when 218IFA/218IFC is used as a transmission device follows:
Set the protocol type to MEMOBUS when used in MELSEC protocol.
Set the circuit number in accordance with the circuit number allocated to the target 218IFA/218IFC.
Set a unique communication buffer channel number for the same circuit.
For information on the register number, refer to *D.2.2 (1) Input Item* and *D.2.2 (2) Output Item*.



[b] 218IF Setting Example

An example of a function setting when 218IF is used as a communication device follows:
Set the protocol type to MEMOBUS when used in MELSEC protocol.
Set the circuit number in accordance with the circuit number allocated to the target 218IF.
Set a unique communication buffer channel number for the same circuit.
For information on the register number, refer to *D.2.2 (1) Input Item* and *D.2.2 (2) Output Item*.



(2) Process Result (PARAM00)

Refer to *D.2.4 (2) Process Result (PARAM00)*.

(3) Status (PARAM01)

Refer to *D.2.4 (3) Status (PARAM01)*.

(4) Connection Number (PARAM02)

Refer to *D.2.4 (4) Connection Number (PARAM02)*.

(5) Option (PARAM03)

A unique value is output for each communication device.
Not used for the MELSEC protocol.

(6) Function Code (PARAM04)

Received function code is output.
The following table lists the function codes available when using the MELSEC protocol.

Table D.24 Function Code List (MELSEC)

Function Code	MELSEC ACPU Common Command	Target Data Type	Function
01H/02H	00H	B	Reads bit device in units
03H/04H/09H/0AH	01H	W	Reads word device in units
05H/0FH	02H	B	Writes to bit device in units
06H/0BH/10H	03H	W	Writes to word device in units
08H	16H	–	Loopback test
0EH	05H	B	Specifies a device number for each word device at random and sets/resets each word device
31H	60H	W	Writes to the fixed buffer in words
32H	61H	W	Reads from the random access buffer in words (unable to receive for MP2□00 series)
33H	62H	W	Writes to the random access buffer in words

Note: 1. B: Bit type, W: Integer type
2. AnCPU dedicated commands are not supported. Commands for extended file register are not supported.

(7) Data Address (PARAM05)

A data address requested from the transmission side is output.

(8) Data Size (PARAM06)

The read/write data size (number of bits or words) requested from the transmission side is output.

(9) Remote CPU Number (PARAM07)

Refer to *D.2.4 (9) Remote CPU Number (PARAM07)*.

(10) Offset (PARAM08, PARAM09, PARAM10, PARAM11)

Specify an offset address for the reception side data address.
The address for the reception side will be displaced by the number of words designated by the offset.

Note: 1. For more information, refer to *D.2.8 Relationship between Data Address, Data Size, and Offset for MSG-RCV Function*.
2. The offset cannot be a negative value.

The offset parameter is prepared for each target data type.
The following table lists the offset parameters.

Table D.25 Offset Parameter List

Parameter	Contents	Description
PARAM08	Coil offset	Sets the coil's offset word address.
PARAM09	Input relay offset	Sets the offset word address of an input relay.
PARAM10	Input register offset	Sets the offset word address of an input register.
PARAM11	Holding register offset	Sets the offset word address of a holding register.

The valid offset parameter may differ according to the function code.
The following table lists the valid parameters for the function codes.

Table D.26 Valid Parameter List for Offset of Each Function Code

Function Code	Function	Valid Offset Parameter
01H	Reads coil state	PARAM08
02H	Reads input relay state	PARAM09
03H	Reads holding register content	PARAM11
04H	Reads input register content	PARAM10
05H	Changes single coil state	PARAM08
06H	Writes to single holding register	PARAM11
09H	Reads holding register content (extended)	PARAM11
0AH	Reads input register content (extended)	PARAM10
0BH	Writes to holding register (extended)	PARAM11
0EH	Discontinuously writes to holding register (extended)	PARAM11
0FH	Changes multiple coil states	PARAM08
10H	Writes to multiple holding registers	PARAM11
31H	Writes to fixed buffer	PARAM11
32H	Reads the random access buffer	Invalid
33H	Writes to the random access buffer	PARAM11

(11) Write Range (PARAM12, PARAM13)

Sets an available address range for a write request from the transmission side. A write request which is out of this available address range will cause an error.
Specify the address range (PARAM12, PARAM13) as a word address.

Note: 1. In MP9□0/MP2□00 series, the data storage area for the write request from the transmission side is the M register.
2. The write range parameter enables you to specify the range of M register which permits writing messages.

The following table shows the write range parameters.

Table D.27 Write Range Parameter List

Parameter	Contents	Description
PARAM12	Write range LO	Start address of the write range
PARAM13	Write range HI	Last address of the write range

Specify the write range so that the expression below is met:

$$0 \leq \text{Write range LO} \leq \text{Write range HI} \leq \text{Maximum address of M register}$$

This write range is effective for the following function codes:

05H (changes single coil state)
06H (writes to single holding register)
0BH (writes to holding register (extended))
0EH (discontinuously writes to holding register (extended))
0FH (changes multiple coil states)
10H (writes to multiple holding registers)
31H (writes to fixed buffer)
33H (writes to the random access buffer)

Example: When setting the address of the M register which permits writing to 1000 to 1999:
PARAM12=1000
PARAM13=1999
The reception side will return an error against a write request to an address other than MW01000 to MW01999, and will not write it.

(12) Reserved by System 1 (PARAM14)

Used by system (the channel number of the communication buffer in use is stored).

Note: At the first scan during power up, make sure to set this to “0” by user program. After that, the register is used by system, so do not change the value with the user program.

(13) Reserved by System 2 (PARAM15, PARAM16)

Used by system. Do not change the value with the user program, etc.

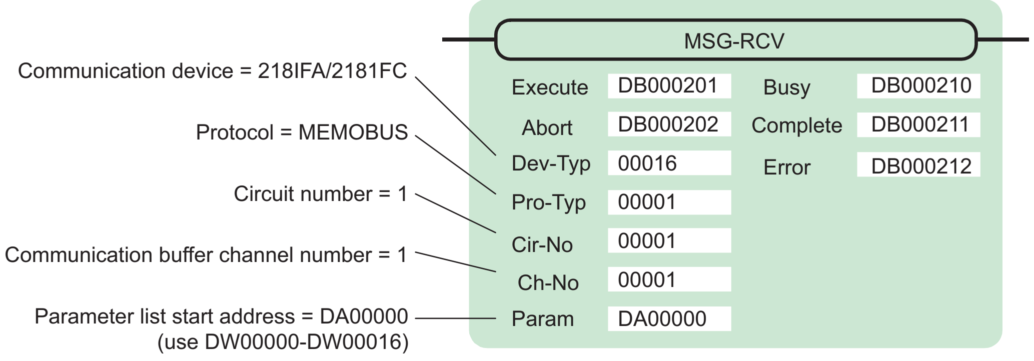
D.2.6 Function Setting and Parameter Details for MODBUS/TCP Protocol

This section explains the MSG-RCV function setting and its parameter list details when MODBUS/TCP is used as a protocol.

(1) Message Receive Function Setting

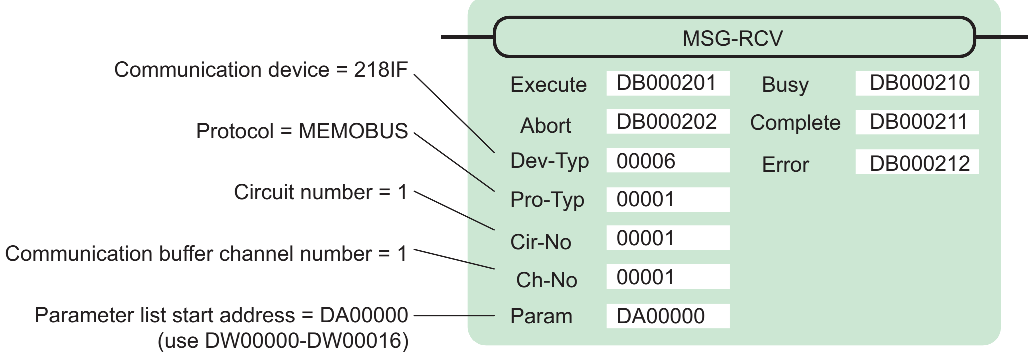
[a] 218IFA/218IFC Setting Example

An example of a function setting when 218IFA/218IFC is used as a transmission device follows:
Set the protocol type to MEMOBUS when used in MODBUS/TCP protocol.
Set the circuit number in accordance with the circuit number allocated to the target 218IFA/218IFC.
Set a unique communication buffer channel number for the same circuit.
For information on the register number, refer to in *D.2.2 (1) Input Item* and *D.2.2 (2) Output Item*.



[b] 218IF Setting Example

An example of a function setting when 218IF is used as a communication device follows:
Set the protocol type to MEMOBUS when used in MODBUS/TCP protocol.
Set the circuit number in accordance with the circuit number allocated to the target 218IF.
Set a unique communication buffer channel number for the same circuit.
For information on the register number, refer to *D.2.2 (1) Input Item* and *D.2.2 (2) Output Item*.



(2) Process Result (PARAM00)

Refer to *D.2.4 (2) Process Result (PARAM00)*.

(3) Status (PARAM01)

Refer to *D.2.4 (3) Status (PARAM01)*.

(4) Connection Number (PARAM02)

Refer to *D.2.4 (4) Connection Number (PARAM02)*.

(5) Option (PARAM03)

A unique value is output for each communication device. The output details are as follows:

Communication device	Output Details
Ethernet (218IF/218IFA/218IFC)	A remote unit ID is output when MODBUS/TCP protocol is used.

(6) Function Code (PARAM04)

Received function code is output.
The following table lists the function codes available when using the MODBUS/TCP protocol.

Table D.28 Function Code List (MODBUS/TCP)

Function Code	Target Data Type	Function	Protocol Type	
			Ethernet (218IF)	Ethernet (218IFA/ 218IFC)
00H	–	Not used	–	–
01H	B	Reads coil state	√	√
02H	B	Reads input relay state	√	√
03H	W	Reads holding register content	√	√
04H	W	Reads input register content	√	√
05H	B	Changes single coil state	√	√
06H	W	Writes to single holding register	√	√
07H : 0EH	–	Not used	–	–
0FH	B	Changes multiple coil states	√	√
10H	W	Writes to multiple holding registers	√	√
11H : 15H	–	Not used	–	–
16H	W	Mask writes to holding register	–	√
17H	W	Reads/Writes multiple holding registers	–	√

Note: 1. B: Bit type, W: Integer type
2. √: Available, -: Not available
3. Transmit and receive registers in the master operation mode are MW (MB) only.
4. In the slave operation mode, coil, holding register, input relay, and input register are MB, MW, IB, IW respectively.

(7) Data Address (PARAM05)

A data address requested from the transmission side is output.
However, when reading/writing multiple holding registers (function code: 17H), enter a start M register number for the address table. The four words of addresses at the beginning of the M register set by PARAM05 (data address) are used as an address table. In this address table, read address, read size, write address, and write size requested from the transmission side are output.
For more information about the address table, refer to *D.1.6 (7) Data Address (PARAM05)*.

(8) Data Size (PARAM06)

The read/write data size (number of bits or words) requested from the transmission side is output.

(9) Remote CPU Number (PARAM07)

Refer to *D.2.8 Relationship between Data Address, Data Size, and Offset for MSG-RCV Function*.

(10) Offset (PARAM08, PARAM09, PARAM10, PARAM11)

Specify an offset address for the reception side data address.
The address for the reception side will be displaced by the number of words designated by the offset.

Note: 1. For more information, refer to *D.2.8 Relationship between Data Address, Data Size, and Offset for MSG-RCV Function*.
2. The offset cannot be a negative value.

The offset parameter is prepared for each target data type.
The following table lists the offset parameters.

Table D.29 Offset Parameter List

Parameter	Contents	Description
PARAM08	Coil offset	Sets the coil's offset word address.
PARAM09	Input relay offset	Sets the offset word address of an input relay.
PARAM10	Input register offset	Sets the offset word address of an input register.
PARAM11	Holding register offset	Sets the offset word address of a holding register.

The valid offset parameter may differ according to the function code.
The following table lists the valid parameters for each function code.

Table D.30 Valid Parameter List for Offset of Each Function Code

Function Code	Function	Valid Offset Parameter
01H	Reads coil state	PARAM08
02H	Reads input relay state	PARAM09
03H	Reads holding register content	PARAM11
04H	Reads input register content	PARAM10
05H	Changes single coil state	PARAM08
06H	Writes to single holding register	PARAM11
0FH	Changes multiple coil states	PARAM08
10H	Writes to multiple holding registers	PARAM11
16H	Mask writes to holding register	PARAM11
17H	Reads/Writes multiple holding registers	PARAM11

(11) Write Range (PARAM12, PARAM13)

Sets an available address range for the write request from the transmission side. A write request which is out of this available address range will cause an error.
Specify the address range (PARAM12, PARAM13) as a word address.

Note: 1. In MP9□0/MP2□00 series, the data storage area for the write request from the transmission side is the M register.
2. The write range parameter enables you to specify the range of the M register which permits writing messages.

The following table indicates the write range parameters.

Table D.31 Write Range Parameter List

Parameter	Contents	Description
PARAM12	Write range LO	Start address of the write range
PARAM13	Write range HI	Last address of the write range

Specify the write range so that the expression below is met:

$0 \leq$ Write range LO $\leq$ Write range HI $\leq$ Maximum address of M register

This write range is effective for the following function codes:

05H (changes single coil state)
06H (writes to single holding register)
0BH (writes to holding register (extended))
0FH (changes multiple coil states)
10H (writes to multiple holding registers)
16H (mask writes to holding register)
17H (reads/writes multiple holding registers)

Example: When setting the address of the M register which permits writing to 1000 to 1999:
PARAM12=1000
PARAM13=1999
The reception side will return an error against a write request to an address other than MW01000 to MW01999, and will not write it.

(12) Reserved by System 1 (PARAM14)

Used by system (the channel number of the communication buffer in use is stored).

Note: At the first scan during power up, make sure to set this to “0” by user program. After that, the register is used by system, so do not change the value with the user program.

(13) Reserved by System 2 (PARAM15, PARAM16)

Used by system. Do not change the value with the user program, etc.

D.2.7 Function Setting and Parameter Details for Non-procedural Protocol

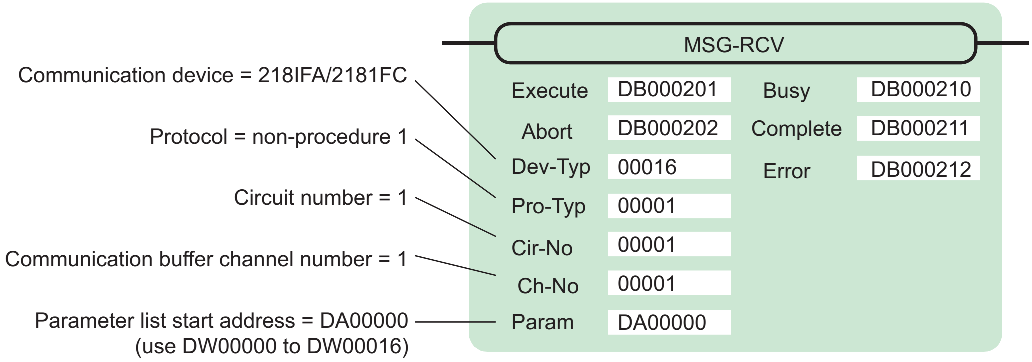
This section explains the MSG-RCV function setting and its parameter list details when non-procedure is used as a protocol.

Note: Non-procedure communication protocol stores the received data in the M register intact without a protocol conversion.
You can receive any protocol in accordance with the remote device.

(1) Message Receive Function Setting

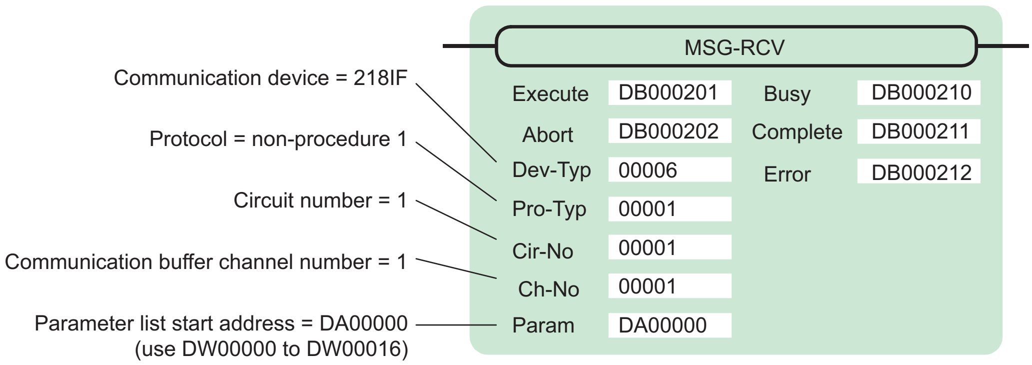
[a] 218IFA/218IFC Setting Example

An example of a function setting when 218IFA/218IFC is used as a transmission device follows:
When non-procedure 2 (per byte) is used, set the Pro-Typ field to “00003”. (non-procedure 1 (per word))
Set the circuit number in accordance with the circuit number allocated to the target 218IFA/218IFC.
Set a unique communication buffer channel number for the same circuit.
For information on the register number, refer to *D.2.2 (1) Input Item* and *D.2.2 (2) Output Item*.



[b] 218IF Setting Example

An example of a function setting when 218IF is used as a communication device follows:
When non-procedure 2 (per byte) is used, set the Pro-Typ field to “00003”. (non-procedure 1 (per word))
Set the circuit number in accordance with the circuit number allocated to the target 218IF.
Set a unique communication buffer channel number for the same circuit.
For information on the register number, refer to *D.2.2 (1) Input Item* and *D.2.2 (2) Output Item*.



(2) Process Result (PARAM00)

The process result is output to the upper byte. The lower byte is used for system analysis.

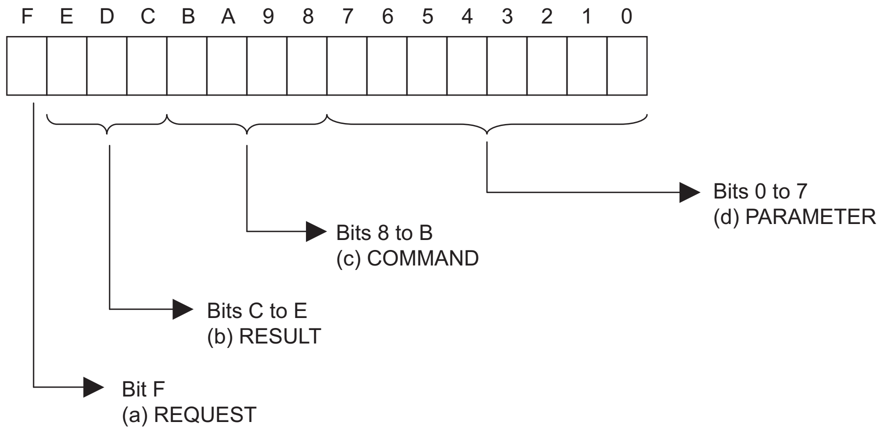
Value of Process Result	Meaning
00xxH	In process (Busy)
10xxH	Process completed (Complete)
8yxxH	Error occurred (Error)

When an error occurs, investigate it by referring to the following error contents:

Value of Process Result	Error Contents	Description
80xxH	–	Reserved
81xxH	–	–
82xxH	Error in setting address	The following setting is out of range. Check the setting. PARAM11 (holding register offset)
83xxH	Data size error	The received data size is out of range. Check the source data size.
84xxH	Error in setting circuit number	The circuit number is out of range. Check Cir-No (circuit number) of the MSG-RCV function.
85xxH	Error in setting channel number	The communication buffer channel number is out of range. Check Ch-No (communication buffer channel number) for the MSG-RCV function.
86xxH	Connection number error	The connection number is out of range. Check PARAM02 (connection number).
87xxH	–	Reserved
88xxH	Communication section error	An error response was returned from the communication section (communication device). Check the equipment connection. In addition, check that the remote device is open to communication.
89xxH	Device select error	Unavailable device is set. Check Dev-Typ (communication device type) of the MSG-RCV function.

(3) Status (PARAM01)

Outputs status of the communication section (communication device). The following figure shows the bit assignment. The bit assignment details are listed in the following tables.



[a] REQUEST (request)

Outputs whether MSG-RCV function is requesting a process.

Bit State	Contents
1	Requesting to process
0	The acceptance of process request is completed

[b] RESULT (result)

Outputs the execution result of the MSG-RCV function

Code	Abbreviation	Meaning
0	CONN_NG	In Ethernet communications, transmit error or connection error is completed
1	SEND_OK	Normal transmission completed
2	REC_OK	Normal reception completed
3	ABORT_OK	Forced abort completed
4	FMT_NG	Parameter format error
5	SEQ_NG	Command sequence error
6	RESET_NG	Reset state
7	REC_NG	Data reception error (error detected in the lower layer program)

[c] COMMAND (command)

Outputs a process command for the MSG-RCV function The executed process content can be found according to the command.

Code (Hex)	Abbreviation	Meaning
1	U_SEND	General-purpose message transmission
2	U_REC	General-purpose message reception
3	ABORT	Forced abort
8	M_SEND	Command transmission: Completed when response is received (for MEMOBUS protocol)
9	M_REC	Command reception: Accompanies a response transmission (for MEMOBUS protocol)
C	MR_SEND	Response transmission (for MEMOBUS protocol)

[d] PARAMETER (parameter)

When RESULT (process result) = 4 (FMT_NG: parameter format error), an error code in the following table is output. Otherwise, the connection number is output.

RESULT (process result)	Code (Hex)	Meaning
When RESULT (process result) = 4 (FMT_NG: parameter format error)	00	No error
	01	Connection number is out of range
	02	Time error while monitoring to receive MEMOBUS response (for MEMOBUS protocol)
	03	Error in setting retransmit count
	04	Error in setting cyclic area
	05	CPU number error
	06	Data address error
	07	Data size error
	08	Function code error (for MEMOBUS protocol)
Others	xx	Connection number

(4) Connection Number (PARAM02)

Refer to *D.2.4 (4) Connection Number (PARAM02)*.

(5) Data Size (PARAM06)

A data size requested from the transmission side is output.
In case of non-procedure 1, the number of words is output.
In case of non-procedure 2, the number of bytes is output.

(6) Register Offset (PARAM12)

Specify an offset address for the reception side data address.
The address for the reception side will be displaced by the number of words designated by the offset.

Note: 1. For more information, refer to *D.2.8 Relationship between Data Address, Data Size, and Offset for MSG-RCV Function*.
2. The offset cannot be a negative value.

In case of non-procedure, received continuous data is stored in the M register. Setting the register offset enables you to specify a start address of the M register as a storage area.

Example: When specifying MW01000 for the start address of the M register for storing received data:
PARAM12=1000

(7) Write Range HI (PARAM13)

Sets an available address range for the write request from the transmission side. A write request which is out of this available address range will cause an error.
Specify the write range (PARAM13) as a word address.

Note: The write range parameter enables you to specify the range of the M register which permits writing messages.

Specify the write range so that the expression below is met:

$$0 \leq \text{Write range HI} \leq \text{Maximum address of M register}$$

Example: When setting the last address of the M register which permits writing to 1999:
PARAM13=1999
The reception side will return an error against a write request to an address other than MW00000 to MW01999, and will not write it.

(8) Reserved by System 1 (PARAM14)

Used by system (the channel number of the communication buffer in use is stored).

Note: At the first scan during power up, make sure to set this to “0” by user program.
After that, the register is used by system, so do not change the value with the user program.

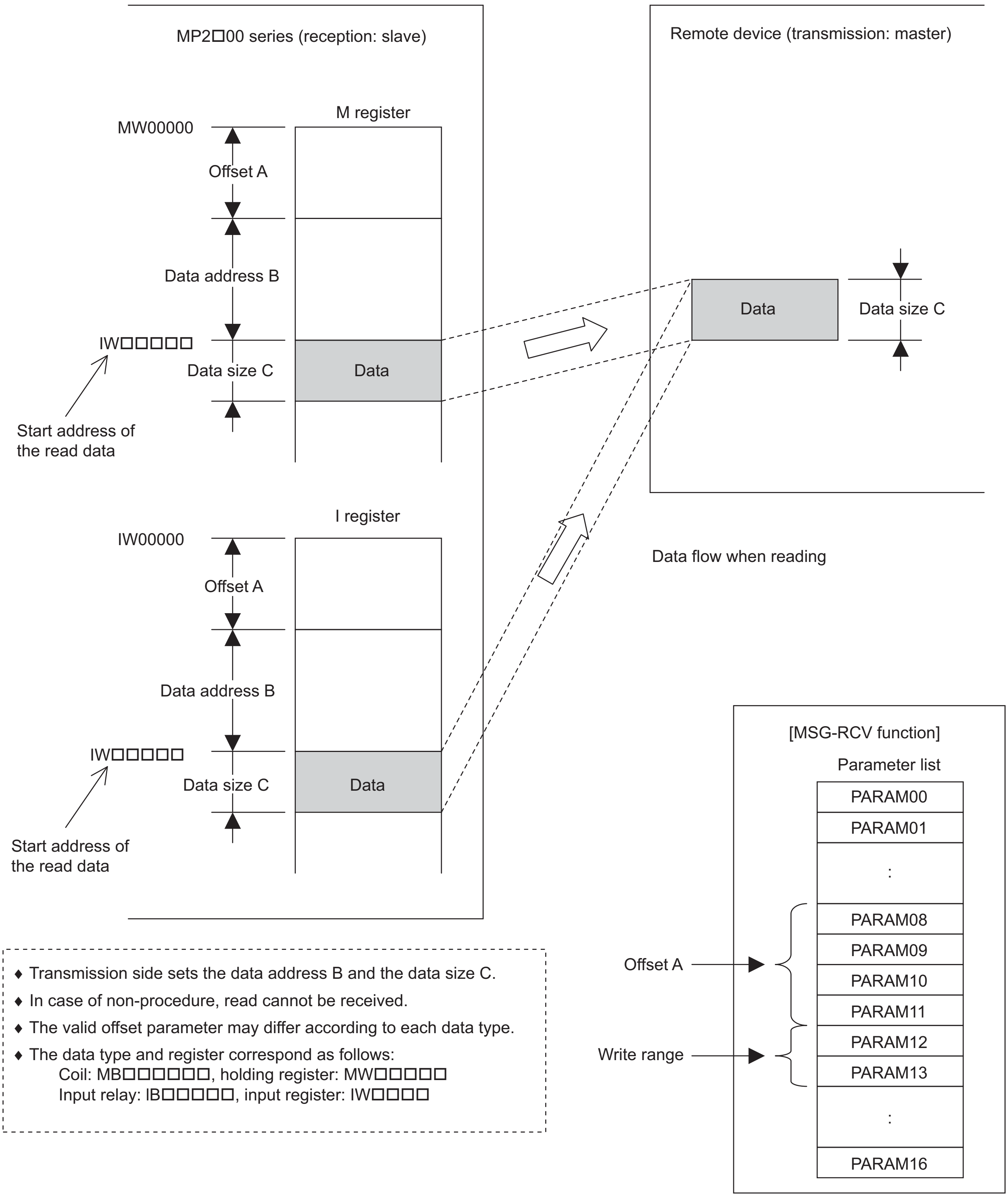
(9) Reserved by System 2 (PARAM15, PARAM16)

Used by system. Do not change the value with the user program, etc.

D.2.8 Relationship between Data Address, Data Size, and Offset for MSG-RCV Function

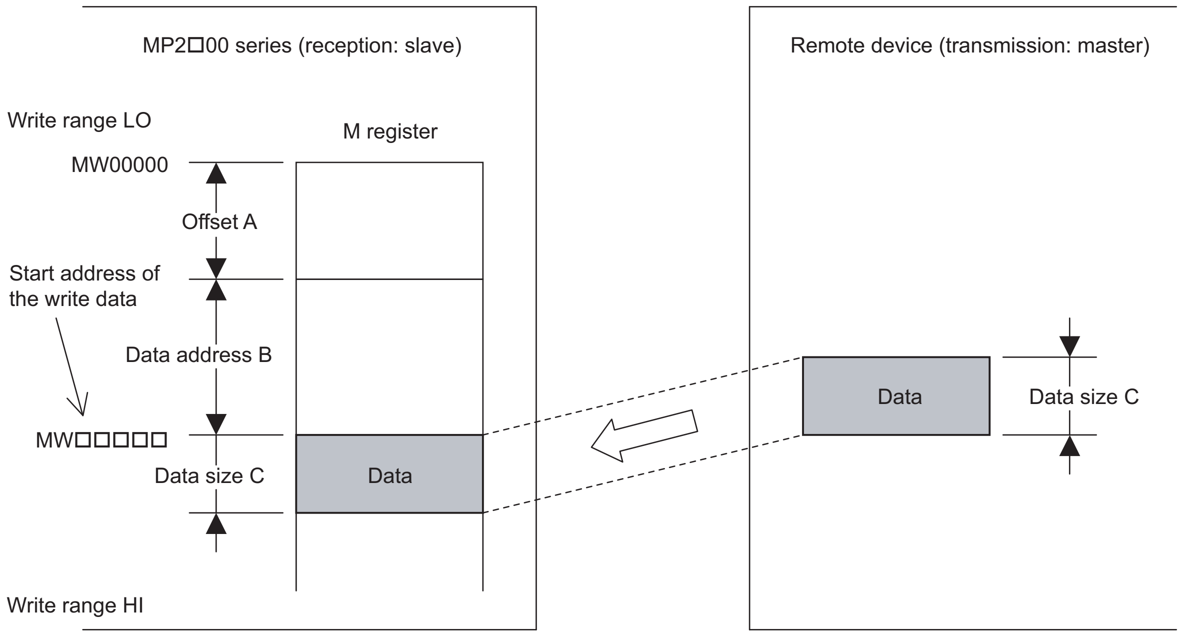
The relationships between data address, data size, and offset when received with offset are as follows:

(1) When Reading



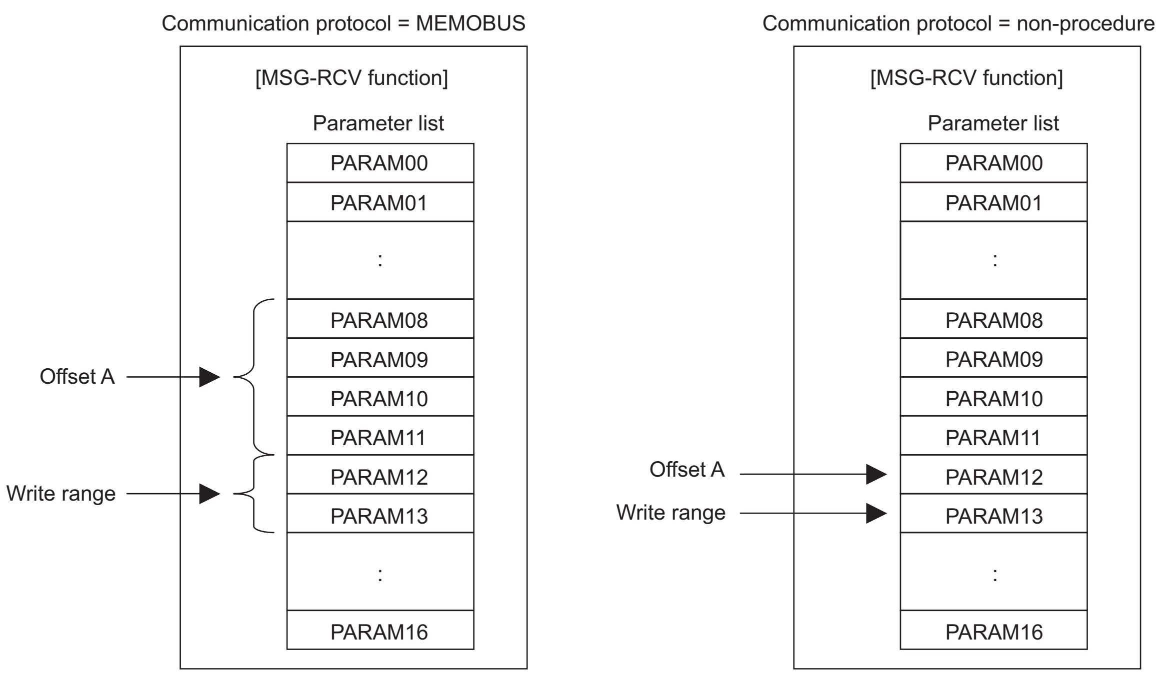
- Transmission side sets the data address B and the data size C.
- In case of non-procedure, read cannot be received.
- The valid offset parameter may differ according to each data type.
- The data type and register correspond as follows:
  Coil: MB□□□□□□□, holding register: MW□□□□□
  Input relay: IB□□□□□, input register: IW□□□□

(2) When Writing



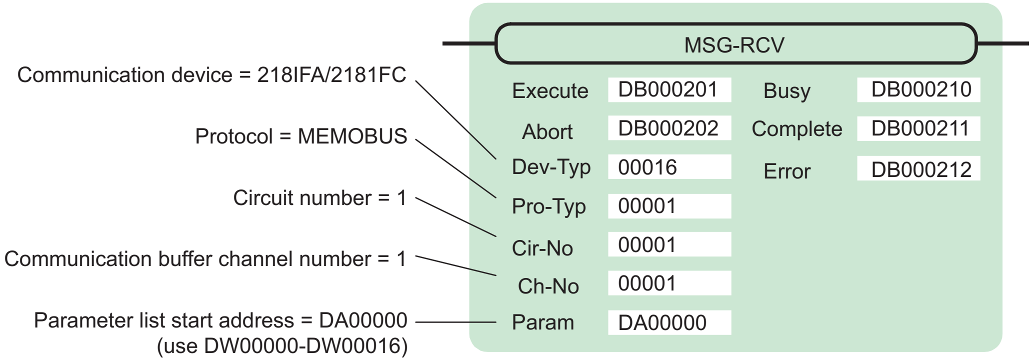
Data flow when writing

- ♦ Transmission side sets the data address B and the data size C.
- ♦ In case of non-procedure, read cannot be received.
- ♦ The valid offset parameter may differ according to each data type.
- ♦ The data type and register correspond as follows:
  Coil: MB□□□□□□, holding register: MW□□□□□



Example: "Writes to multiple holding registers" with offset is received:
When "writes to multiple holding registers" with offset is received in MEMOBUS protocol, various setting values and the relationship with the data of remote device are as follows:

- Description of the MSG-RCV function in ladder program



- Parameter list setting of the MSG-RCV function

Table D.32 Parameter List Setting

Register Number	Setting Value	Parameter Number	IN/OUT	Remarks
DW00000	-	PARAM00	OUT	Process result
DW00001	-	PARAM01	OUT	Status
DW00002	00001	PARAM02	IN	Connection number = 1
DW00003	-	PARAM03	OUT	Option (setting unnecessary)
DW00004	-	PARAM04	OUT	Function Code
DW00005	-	PARAM05	OUT	Data address
DW00006	-	PARAM06	OUT	Data size
DW00007	-	PARAM07	OUT	Remote CPU number
DW00008	00000	PARAM08	IN	Coil offset = 0 word
DW00009	00000	PARAM09	IN	Input relay offset = 0 word
DW00010	00000	PARAM10	IN	Input register offset = 0 word
DW00011	01000	PARAM11	IN	Holding register offset = 1000 words
DW00012	00000	PARAM12	IN	Write range LO = 0
DW00013	65534	PARAM13	IN	Write range HI = 65534
DW00014	-	PARAM14	SYS	Reserved (zero clear at startup)
DW00015	-	PARAM15	SYS	Reserved
DW00016	-	PARAM16	SYS	Reserved

Note: IN: Input, OUT: Output, SYS: For system use

- Relationship with the Remote Device Data
  The following figure shows the data flow when receiving “writes to multiple holding registers” with offset: When transmission and reception are carried out normally in the figure below, the data in the remote device is stored in MW03000 and after of MP2200.

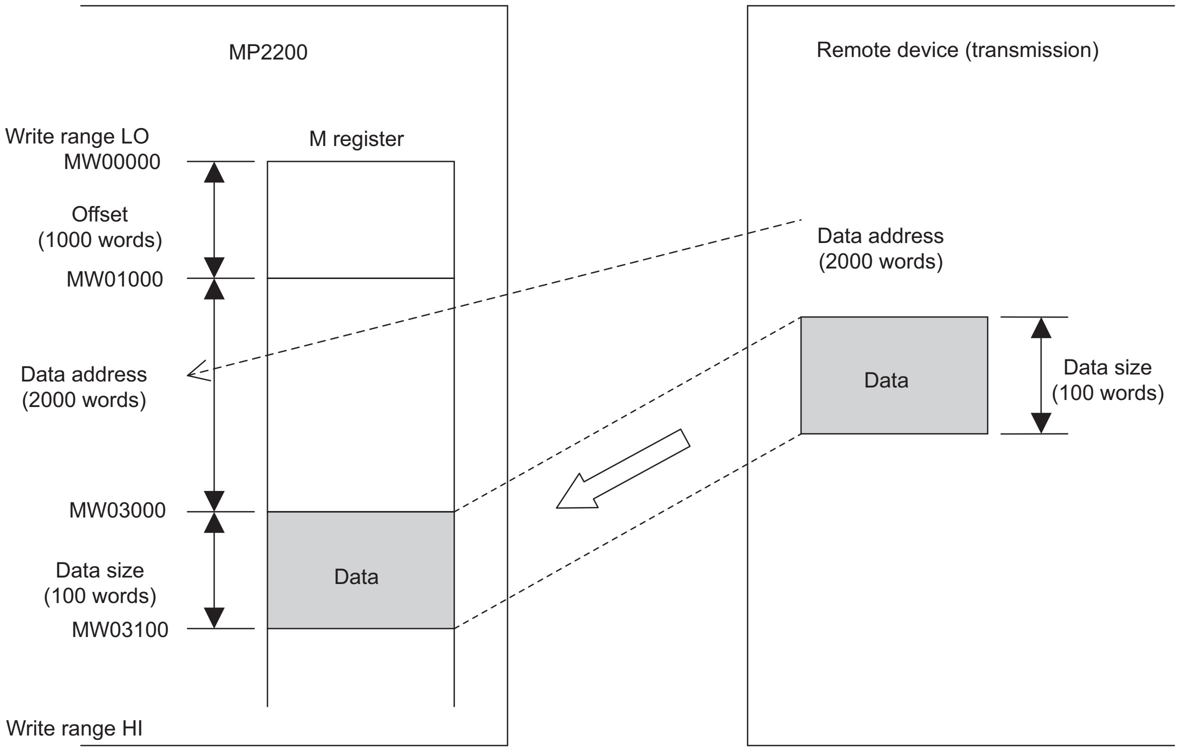
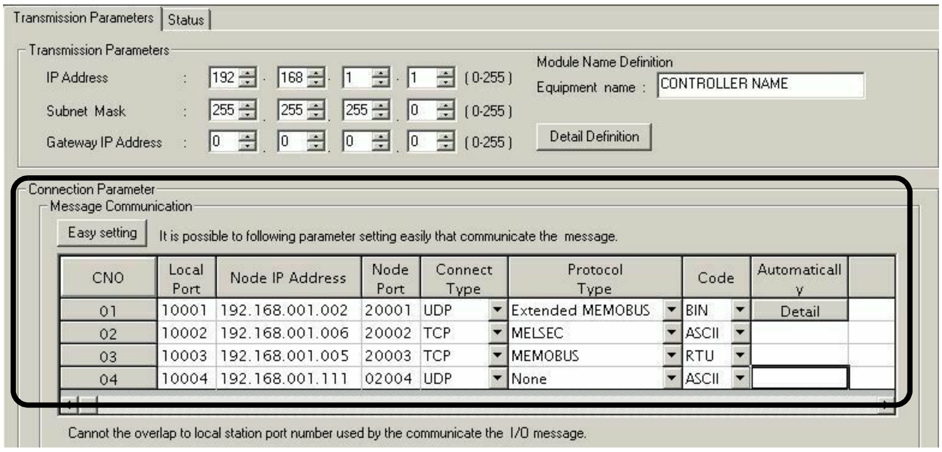


Fig. D.12 Writing Example to Multiple Holding Registers

D.3 Communication Buffer Channel

A communication buffer channel is used for giving and receiving data between the MSG-SND/ MSG-RCV function and communication device. This data buffer is composed of single or multiple channels, and each channel is distinguished by a communication buffer channel number.
The communication buffer channel is associated with a connection by setting the input item **Ch-No** (communication buffer channel number) of the MSG-SND/ MSG-RCV function and PARAM02 (connection number) in Param (parameter list).

A connection is configuration information for communication between local and remote stations, and is set in the 218IF/218IFA/218IFC **Transmission Parameters** Tab Page of the MPE720 **Module Configuration** Window.



Note: For the 218IFC, the setting range for the connection number is larger, i.e., 01 to 20.

Fig. D.13 218IFA/218IFC Transmission Parameters Tab Page for the MPE720 Module Configuration Window

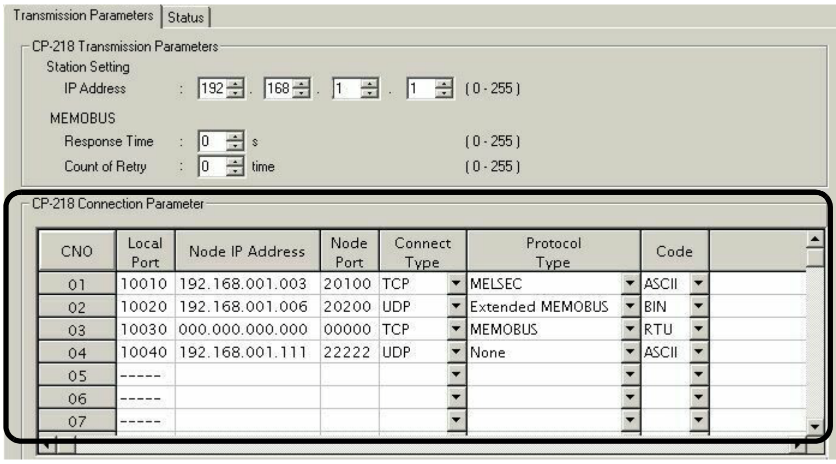


Fig. D.14 218IF Transmission Parameters Tab Page for the MPE720 Module Configuration Window

A schematic diagram of the communication buffer channel is shown in the next page.

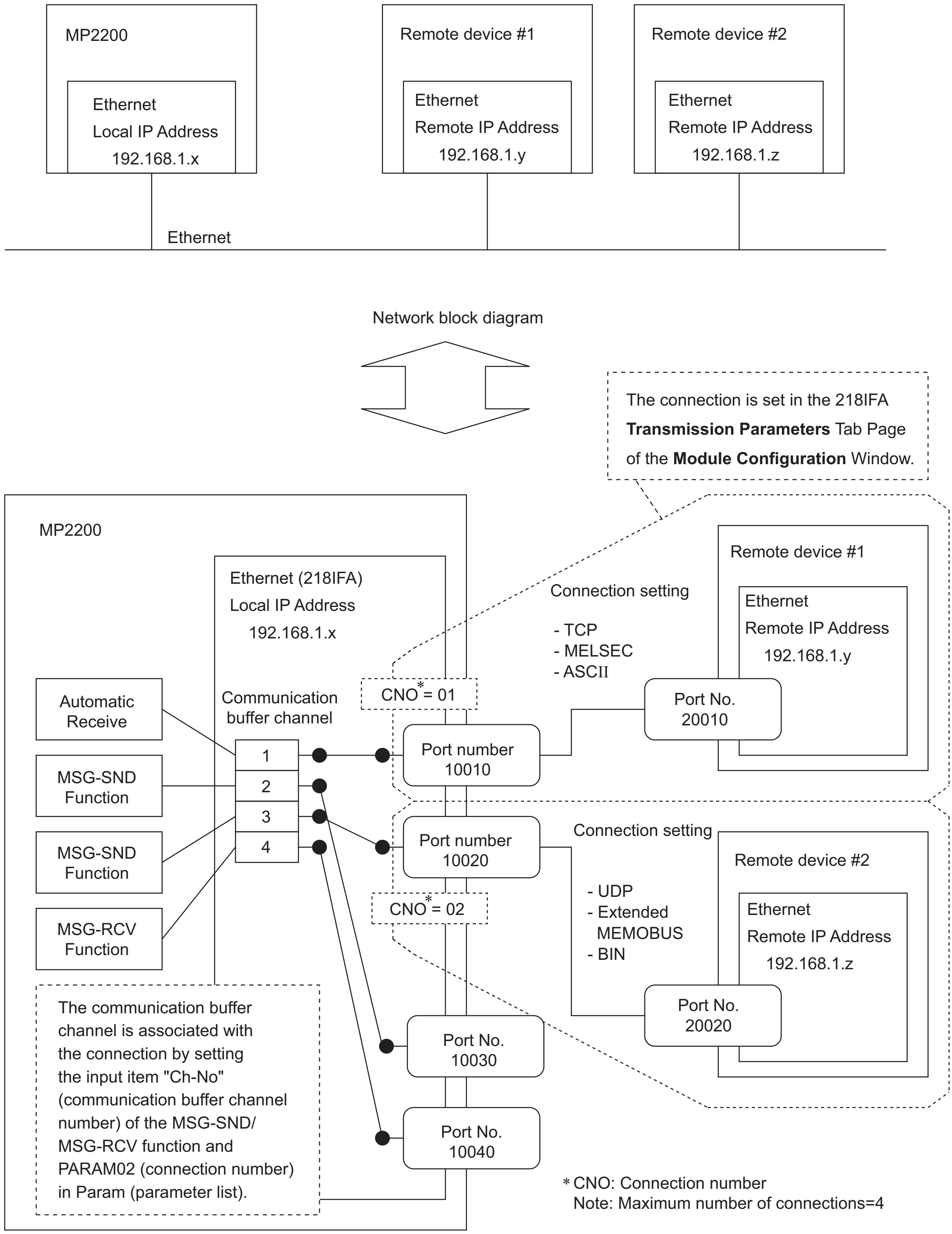


Fig. D.15 Schematic Diagram for Communication Buffer Channel

Appendix E Optional Functions

E.1 Clearing D Registers at Startup

(1) Overview of Functions

D registers are internal registers that are unique for each user program. Normally, the D register data is not defined at startup, but it is also possible to start with the data cleared. Specifically, the data cleared by this function is the D registers of DWG (drawings), user functions, and motion programs. Make the settings in the **System Setting** of the **Environment Setting** Dialog Box. With the default settings, the D registers will not be cleared at startup.

(2) Supported Versions

The following versions support function to clear D registers at startup.

- Controller: Version 2.32 or later
- MPE720: Version 5.21 or later
- MPE720 Version 6: All versions

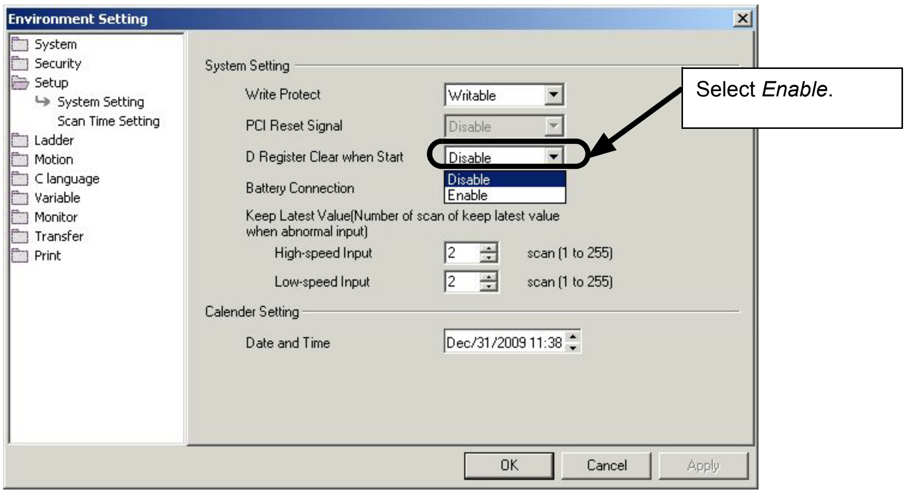
(3) Setting Procedure for Clearing D Registers at Startup

Use the following procedure to make the setting.

1. Connect the MPE720 online.

Select ***File (F) - Environment Setting - Setup***.

2. Select **Enable** under the **D Register Clear when Start**.



3. Save data to flash memory before turning OFF the power supply.

The system will start with the D registers cleared to zeroes the next time that the flash memory starts.

(4) Application Precautions

- Due to the nature of this function, it will only be enabled when starting with flash memory after having set this function to “enabled” and saving to flash memory.
- It is not possible to clear the D registers for only a specific user program.
- No check is performed to see if the CPU Module supports the function to clear D registers at startup when the MPE720 is offline. Therefore, confirm that the CPU Module that is being used supports the function to clear D registers at startup before using this function.

Appendices

A

E.2 Security

(1) Overview of Functions

Security functions set a password (security key) to prevent data saved in the Controller from being read by other parties.

(2) Supported Versions

The following versions support security functions.

- Controller: Version 2.32 or later
- MPE720: Version 5.21 or later
- MPE720 Version 6: All versions

(3) Description of Functions

[a] Security Settings

- Only one security condition can be set for each Controller.

  Note: 1. You cannot make separate security settings for each user that logs on.
  2. This security function has no relation to the identify function in terms of user names and passwords used to log on to the MPE720.
- Only users with writing privilege of level 7 or higher can make this security setting.

[b] Target data

- The following table shows the data that can be privacy protected.

Type	Description	Remarks
Ladder	Drawings (DWG) and user functions (FUNC)	All types of drawings, as well as parent, child, grandchild, and error drawings.
Motion	Motion program	
	Sequence program	Supported by the CPU-03 and CPU-04.
C Language	C language tasks and C language functions	Supported from version 2.31 of the embedded API.

- You can place restrictions (e.g., copy protection) only on reading from the Controller.

  Note: It is not possible to use the security functions to restrict writing to the Controller.
- You can place restrictions reading according to the reading privilege level of the application.

  Note: File rights: Reading is restricted when R (the reading privilege level) is greater than or equal to the restriction privilege level.

[c] Security Identification

- Security identification uses a security key of up to eight alphanumeric case-sensitive characters.

  Note: If you forget the key, the security setting cannot be released unless memory is cleared. Keep a record of the key in case you forget it.
- If the key that is entered is incorrect three consecutive times for the current security settings, it will not be possible to change any security settings without turning OFF and then ON again the power supply to the Controller.
- You can release the security without changing the security key setting.

  Note: You can enable or disable security while the security key is set by selecting whether to restrict reading of files. This is effective when you do not want other parties to make security settings.

[d] Precautions for Transfers

- The security settings are not transferred.
- The security settings are automatically canceled after batch transfers or batch loading from a CF card* are performed. After the batch job has been completed, be sure to return the security settings to their previous state.

  * A batch load from a CF card can be only executed when using a CPU-02 or a CPU-03 Module.

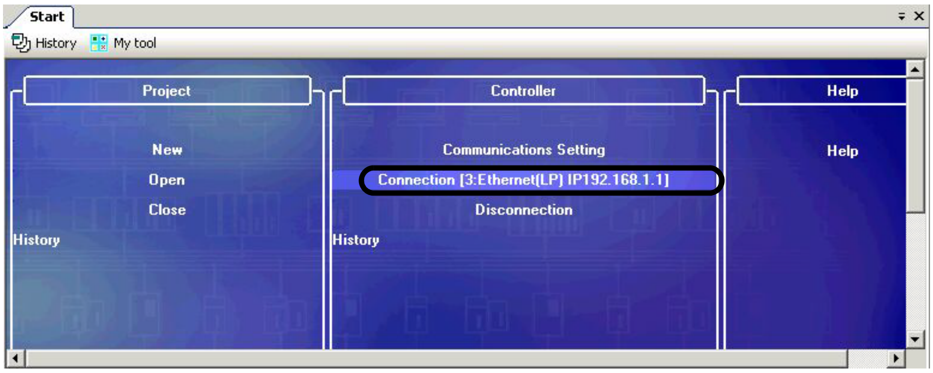
  Note: If using controller software version 2.76 or later, the security settings are not canceled after batch transfers or batch loading. If the INIT DIP switch for the controller is ON and then the power supply is turned ON, however, the security settings will also be released. For details, refer to *E.2 (6) Initialization*.
- With the CPU-02 or CPU-03, batch saving to the CF card is prohibited regardless of the restriction privilege level if reading is restricted.

(4) Operation Procedure

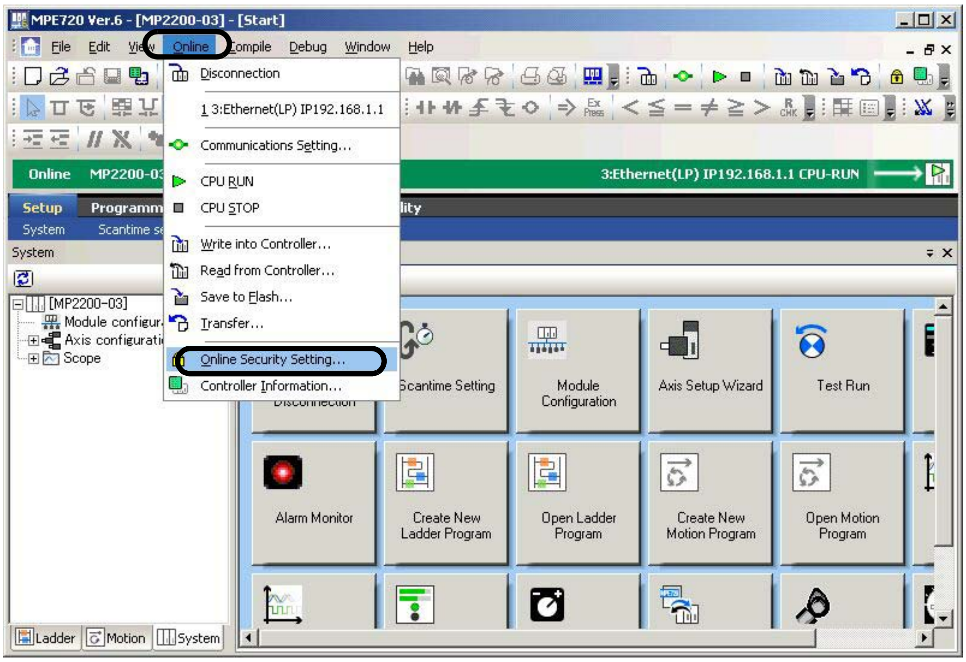
[a] Opening the Security Setting Dialog Box

Make the security settings using the MPE720. The **Security Setting** Dialog Box will start when the Engineering Tool goes online with the Controller.

1. Connect to the Controller online.



2. Select ***Online - Online Security Setting***.



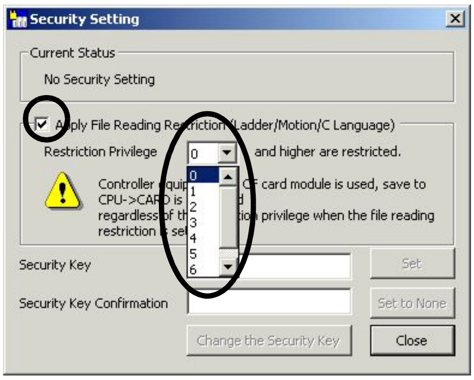
The following dialog box will be displayed in environments that support security settings.

The figure above shows the default status.

[b] New Security Settings

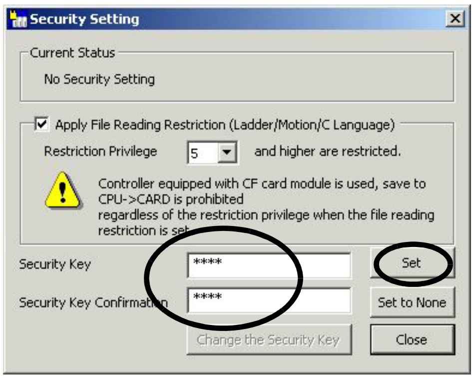
Make a new security setting in the **Security Setting** Dialog Box.

1. Select the **Apply File Reading Restriction (Ladder/Motion/C Language)** Check Box, and then select a restriction privilege level from 0 to 7.

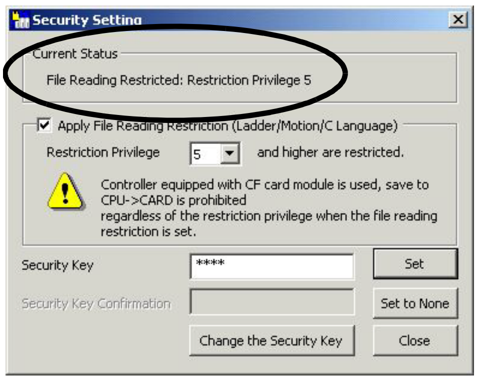


Note: The display depends on the conditions that are set.

2. Enter a password of eight or less alphanumeric characters in the **Security Key** Field. For confirmation, enter the same password in the **Security Key Confirmation** Field. To apply the security settings to the Controller, click the **Set** Button.



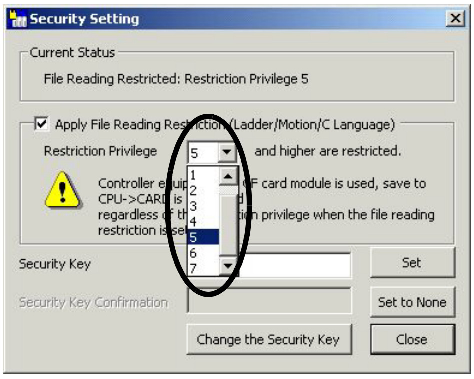
The following figure will be displayed if the security settings have been normally applied to the Controller.



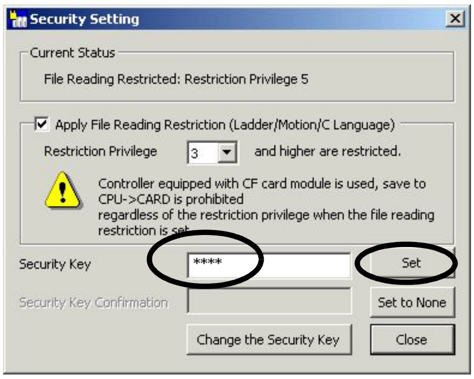
[c] Changing the Restriction Privilege

Change the restriction privilege with the security settings made.

1. Open the **Security Setting** Dialog Box and change the restriction privilege level to a number from 0 to 7.



2. Enter the password in the **Security Key** Field. To apply the security settings to the Controller, click the **Set** Button.

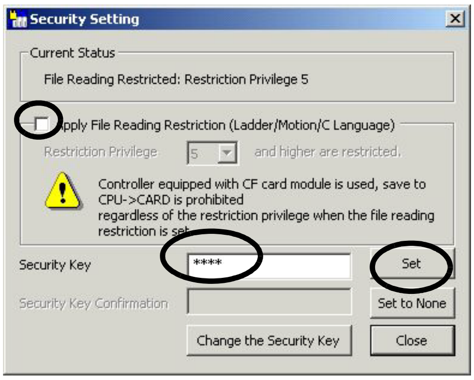


If a change in restriction privileges has been normally applied to the Controller, the new restriction privilege level will be displayed in the **Current Status** Area of the **Security Setting** Dialog Box.

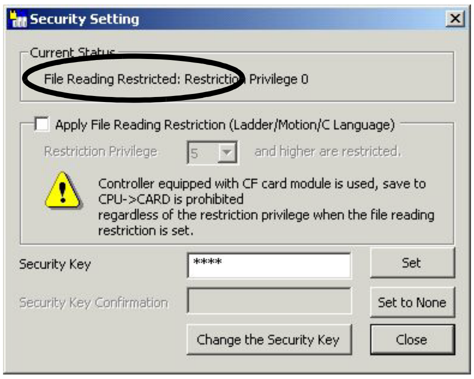
[d] Changing the File Reading Restriction

Change the file reading restriction with the security settings made. In this example, we will release the restriction. To set the restriction, use the same procedure. Having no file reading restriction set is the same as security being released.

1. Select or clear the **Apply File Reading Restriction (Ladder/Motion/C Language)** Check Box. Enter the password in the **Security Key** Field, and then click the **Set** Button.



2. If the change in file reading restriction has been normally applied to the Controller, the status will change from **File Reading Restricted** to **No Security Setting** in the **Current Status** Area.



[e] Changing the Security Key

The security key can be changed with the security settings made. This procedure will change only the security settings.

1. Open the **Security Setting** Dialog Box, and then click the **Change the Security Key** Button.

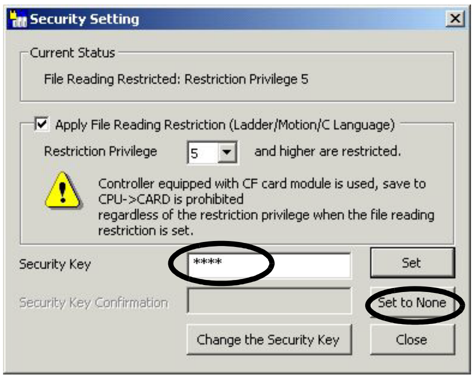
A dialog box for changing the security key will be displayed as shown in the following figure.

2. Enter the set password in the **Current Security Key** Field. Enter the new password in the **New Security Key** Field and **New Security Key Confirmation** Field. Click the **OK** Button to apply the change to the Controller.

[f] Releasing Security

The security can be released with the security settings made. This procedure will initialize the security settings.

1. Enter the set password in the **Security Key** Field, and then click the **Set to None** Button.

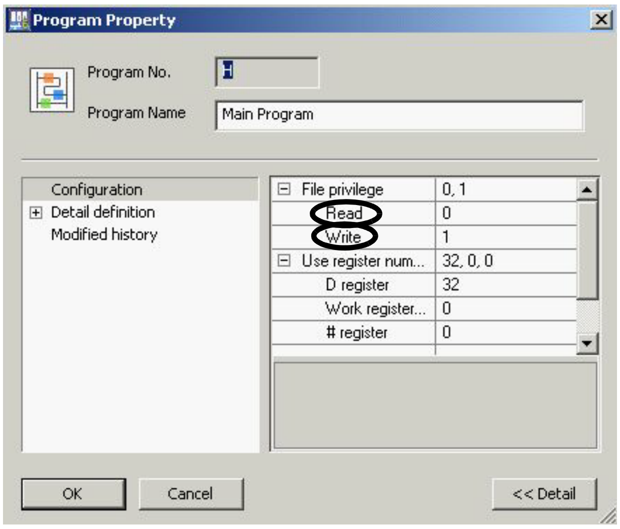


The text **No Security Setting** will be displayed in the **Current Status** Area if the security release has been normally applied to the Controller.

(5) Security Operation

[a] Privilege Settings in User Applications

Privilege settings enable individually set reading privilege levels and writing privilege levels in the properties for each application (i.e., ladder, motion, or C language).



The figure above shows the dialog box for ladder (DWG) properties.

[b] File Reading with Security Settings

The system will activate a security check if an attempt is made to read a user application with the MPE720 after security settings have been made. If the reading privilege level of the application is equal to or higher than the security restriction privilege level, an error will occur, and you will not be able to read the application. The same will occur when a comparison is performed between the CPU Unit and MPE720.

Fig. E.1 This Error Message is Displayed If an Attempt is Made to Open a User Application (i.e., ladder, motion, or C language).

(6) Initialization

Set ON the INIT of DIP switch for the Controller and then turn ON the power supply. Memory will be cleared and operation will start. The security settings will also be initialized.

(7) Security-related System Registers

The security setting status can be checked by using the following system registers.

<table>
<tr><th>Name</th><th>Register Number</th><th colspan="2">Description</th></tr>
<tr><td>Security Status</td><td>SW00506</td><td colspan="2">0: No security, 1: Security set</td></tr>
<tr><td rowspan="4">Security Reading Restriction Information</td><td rowspan="4">SW00507</td><td>SB005070 to SB005073</td><td>File Reading Restriction Privilege
SW00507=***xH
x: Restriction privilege level (0 to 7)</td></tr>
<tr><td>SB005074 to SB005076</td><td>Reserved by system.</td></tr>
<tr><td>SB005077</td><td>0: File reading not restricted
1: File reading restricted</td></tr>
<tr><td>SB005078 to SB00507F</td><td>Reserved by system.</td></tr>
</table>

(8) Application Precautions

- The security settings data is saved in flash memory.
  When starting from flash memory, an operation will be performed according to the status when data was saved to flash memory. Therefore, be sure to save the data to flash memory after making the security settings if the Controller will be started from flash memory.
- Security function does not apply when offline (i.e., when the MPE720 is used alone).
  You cannot make security settings even if you log on offline. Manage offline data so that applications are protected from being disclosed.
- Always enter at least one alphanumeric character for the security key.
  To ensure security, at least one alphanumeric character must be entered in the security key. Although no key can guarantee foolproof security, you can increase the security of your key by entering a mixture of uppercase, lowercase, and numeric characters (e.g., A to Z, a to z, and 0 to 9).

E.3 Battery Backup for Table Data

(1) Overview of Functions

For the MP2000 Series, one table consists of the following three types of data.

1. Table definitions (e.g., definitions of table name, table type, number of columns, and number of rows)
2. Attribute definitions (e.g., definitions for column names, data type (size), and table type)
3. Table data

Normally, the data listed above is placed in volatile memory in the same way as drawings. Therefore, data will not be held the next time operation is started unless the data has been saved to flash memory. To back up the data when the power supply is turned OFF, you can place the table data (item number three above) into memory that is backed up with a battery. This is called “battery backup memory.” You can access table data that has been backed up using this function by following normal table data operation instructions.

(2) Applicable Models

- CPU-02/CPU-03/CPU-04

(3) Supported Versions

- Controller: Version 2.32 or later
- MPE720: Version 5.21 or later
- MPE720 Version 6: All versions

(4) Battery Backup Memory Capacity

- CPU-02: 1 MB (including data used by the system)
- CPU-03/CPU-04: 3 MB (including data used by the system)

(5) System Registers for Battery Backup Memory Capacity

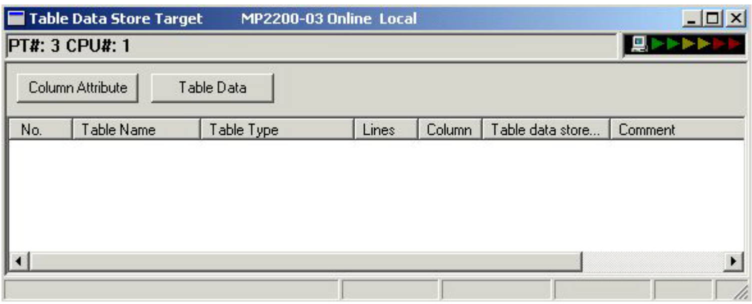
Data related to battery backup memory is stored in the system registers given in the following table.

Name	Register Number	Remarks
Battery Backup Memory Remaining Capacity	SL00640	Bytes
Battery Backup Memory Total Capacity	SL00642	Bytes

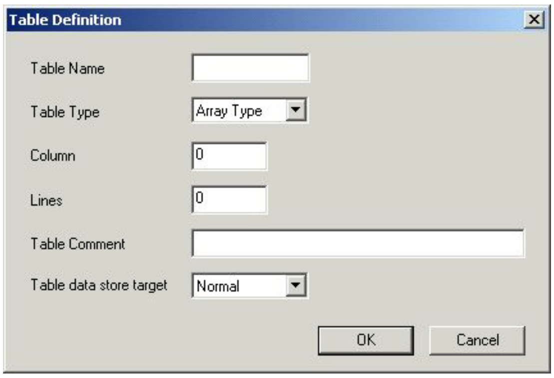
(6) Setting Procedure for Using Battery Backup Memory

Use the following procedure.

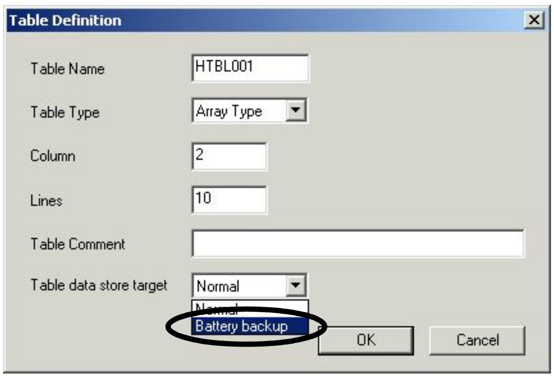
1. Open the **Table Data Store Target** Window from the MPE720.



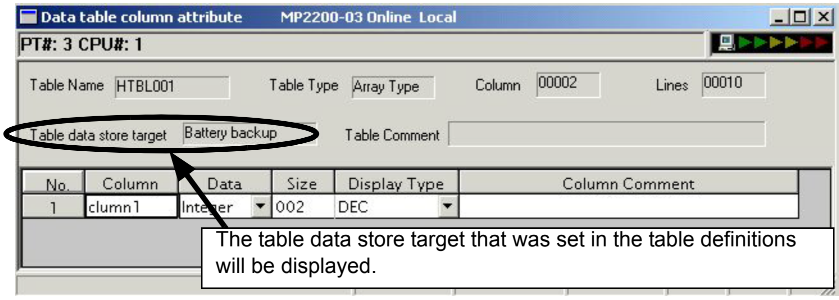
2. Select ***File - Create New*** to open the **Table Definition** Dialog Box.



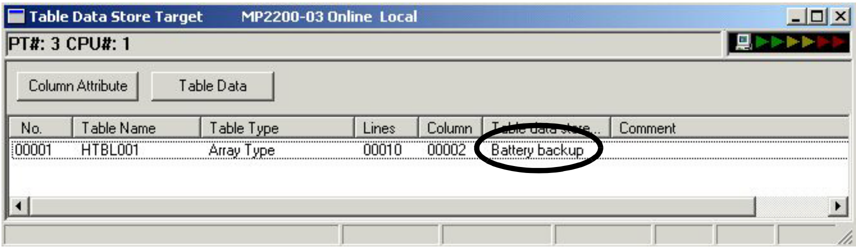
3. Select **Battery Backup** in the **Table data store target** List.



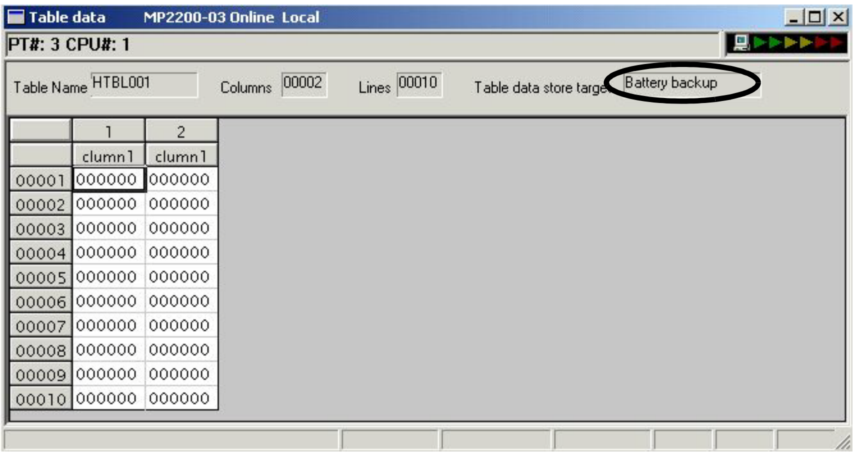
4. When you finish setting the table definitions, the window for setting the column attribute definitions will be displayed to allow you to enter the required data.



5. After saving the attribute definitions and returning to the **Table Data Store Target** Window, the following figure will be displayed.



6. Table data will be displayed as in the following figure.



(7) Application Precautions

a) Always save data to the flash memory before turning OFF the power supply if new table data is created or definitions are changed by using this function. Table definitions and column attribute definitions will be saved to volatile memory even if battery backup memory is specified for the table data store target. Therefore, the table itself will be deleted the next time the system is started if data is not saved to the flash memory after table definitions have been created. Also, access will be normally performed if the definitions have been changed and the table names before and after the change are the same. Actually, however, you will be accessing the table data store target with the status from before the change, and the entire battery backup memory data will be incorrect.

b) Tables with their table data store target set to battery backup memory will start with table data in the status that existed before the power supply was turned OFF (when starting from flash memory). The table data, however, will be unstable (same as for M registers) if the data has not been held because the battery is not mounted or the battery voltage is low. In this case, you can use the system normally by initializing the table data.

c) If you turn ON the INIT of DIP switch pin and then turn ON the power supply, the battery backup memory data will be cleared to all zeroes. To recover the data, dump it in advance, and then reload it.

d) If the table data store target is set to battery backup memory, areas allocated to table data will be allocated in block units. Empty blocks may occur if you repeatedly delete and save definitions. To delete these empty blocks, dump the data, initialize the battery backup memory, and then reload the data.

e) If the table data store target is changed, the table data will be cleared to all zeros even if there is no change in the size.

Appendix F Installing MPE720 Version 6

MPE720 Version 6 (CPMC-MPE770) is the Engineering Tool for the MP2000-series Machine Controllers.
With this Tool installed on a computer and with a connection to a Machine Controller through a Communication Module, you can program, control, and monitor the Controller from computer windows.
When you purchase MPE720 version 6, you will receive an installation CD-ROM.
Use this installation CD-ROM to install MPE720 version 6.
For the installation procedure, refer to the *Engineering Tool for MP2000 Series Machine Controller MPE720 Version 6 User's Manual* (manual no.: SIEP C880700 30).
If you register online as a user of MPE720 version 6, you can download upgraded versions of the product free of charge for two years from the time of purchase from the Yaskawa e-Mecha website (http://www.e-mechatronics.com).
We will also send you e-mail notification of version upgrades to the product as they are released. You must register as a member of the e-Mecha website before registering as an online user. If you are unable to use the online download service, please use the MPE720 Version 6 CD Dispatch Application that is included with the installation CD-ROM.

Index

Symbols

Numerics

A

B

C

N

O

P

R

S

Revision History

The revision dates and numbers of the revised manuals are given on the bottom of the back cover.

MANUAL NO. SIEP C880700 14C <4>-1
WEB revision number
Revision number

Published in Japan April 2010
Date of publication

Date of Publication	Rev. No.	WEB Rev. No.	Section	Revised Contents
November 2018	<20>	0	Back cover	Revision: Address
February 2018	<19>	0	2.1.1	Revision: Noise resistance
December 2017	<18>	0	1.2.2, 1.5.1	Addition: Description of CC-Link IE Field Communication Module (269IF-01)
			1.5.1	Addition: Description of PO-01Cable for PO-01Module (JEPMC-W6060-□□)
June 2017	<17>	0	Back cover	Revision: Address
March 2017	<16>	0	–	Printed version of the manual that is available on the web (web version: SIEP C880700 14H<15>-1)
January 2017	<15>	1	5.3.1 (1)	Revision: Notes for DWG Registers
			Appendix E.1 (1)	Revision: Description of Clearing D Registers at Startup
			Back cover	Revision: Address
December 2016		0	–	Printed version of the manual that is available on the web (web version: SIEP C880700 14H<14>-1)
July 2016	<14>	1	1.2.2	Revision: MP2200 Modules
			1.3	Revision: Optional modules
			1.4	Revision: Devices Connectable to MECHATROLINK-I/II/III
			1.5	Revision: Cables, Accessories and Options, and Software
November 2015		0	–	Printed version of the manual that is available on the web (web version: SIEP C880700 14G<13>-1)
			Back cover	Revision: Address
August 2015	<13>	1	1.5.1	Revision: Sumitomo 3M → 3M Japan Limited
			2.2.3	Addition: Description of the allowable frequency range
April 2015		0	Front cover, spine	Revision: Format
			Back cover	Revision: Address and format
January 2014	<12>	0	–	SIEP C880700 14F<11>-1, available on the Web.
November 2013	<11>	1	1.5.2	Revision: Power connector model
			2.9.1 (1)	Addition: Explanation of illustration (cable-end connector), notes for cable-end connector and handle for power connector
			2.9.1 (2)	Revision: Notes for cable-end connector
September 2013		0	Back cover	Revision: Address
November 2012	<10>	0	Back cover	Revision: Address
March 2012	<9>	0	Back cover	Revision: Address
November 2011	<8>	0	–	SIEP C880700 14E<7>-3, available on the Web.
			1.2.2, 1.3.1 (2)	Addition: Notes for EXIOIF Cable
September 2011	<7>	3	D.1.1, D.2.1	Revision: Communication device type Ethernet (218IFA/218IFC) = 16 → Ethernet (218IFA/218IFB/218IFC) = 16
July 2011		2	1.5.3	Revision: Supported version of MPE720 for CPU-02 Version 5.30 or higher → Version 5.31B or later
		1	2.2.3, 8.1.2	Revision: Input voltage 85 to 276 VAC → 85 to 132 VAC / 198 to 276 VAC
May 2011		0	–	SIEP C880700 14D<6>-1, available on the Web.
April 2011	<6>	1	8.2.4 (5) [b]	Revision: Description of *1 to *3
February 2011		0	–	SIEP C880700 14C<5>-3, available on the Web.

Date of Publication	Rev. No.	WEB Rev. No.	Section	Revised Contents
December 2010	<5>	3	Preface, 1.2.2	Addition: Description of MPU-01 Module (JAPMC-CP2700-E)
			1.5.1	Revision: Cable-end connector model 10114-3000VE → 10114-3000PE
			8.2.4 (5) [b]	Addition: 0051H (Module synchronous error)
October 2010		2	Front cover, back cover	Revision: Format
			E.2 [d]	Addition: Description of a controller software version
August 2010		1	3.2.1 (2) [b]	Revision: Size of twisted-pair cable
			A.1 (1)	Addition: Description of * of SB000005 and SB000006
			Back cover	Revision: Address
May 2010		0	–	SIEP C880700 14C<4>-1, available on the Web.
April 2010	<4>	1	Preface	Addition: ■ Terms Used to Describe “Torque”
			2.2.3	Addition: Allowable Power Loss Time
			6.3.1 (1)	Revision: The other device to be connected in step 4 ⑥ and ⑦ MP2300 → touch panel
November 2009		0	All chapters	Completely revised
			Back cover	Revision: Address
October 2006	<3>	–	–	Based on Japanese user’s manual, SIJP C880700 14C<4>-0, printed in October 2006.
			4.4	New section added: CPU-02 Module
			2.4.2, 7.2	Revision: Accessory battery model ZZK000064 → JZSP-BA01
April 2006	<2>	–	–	Based on Japanese user’s manual, SIJP C880700 14B<2>-1, available on the Web.
August 2005	<1>	–	Back cover	Revision: Address
September 2004	–	–	–	First edition

Machine Controller MP2200

USER'S MANUAL

IRUMA BUSINESS CENTER (SOLUTION CENTER)
480, Kamifujisawa, Iruma, Saitama, 358-8555, Japan
Phone: +81-4-2962-5151 Fax: +81-4-2962-6138
http://www.yaskawa.co.jp

YASKAWA AMERICA, INC.
2121, Norman Drive South, Waukegan, IL 60085, U.S.A.
Phone: +1-800-YASKAWA (927-5292) or +1-847-887-7000 Fax: +1-847-887-7310
http://www.yaskawa.com

YASKAWA ELÉTRICO DO BRASIL LTDA.
777, Avenida Piraporinha, Diadema, São Paulo, 09950-000, Brasil
Phone: +55-11-3585-1100 Fax: +55-11-3585-1187
http://www.yaskawa.com.br

YASKAWA EUROPE GmbH
Hauptstraβe 185, 65760 Eschborn, Germany
Phone: +49-6196-569-300 Fax: +49-6196-569-398
http://www.yaskawa.eu.com E-mail: info@yaskawa.eu.com

YASKAWA ELECTRIC KOREA CORPORATION
35F, Three IFC, 10 Gukjegeumyung-ro, Yeongdeungpo-gu, Seoul, 07326, Korea
Phone: +82-2-784-7844 Fax: +82-2-784-8495
http://www.yaskawa.co.kr

YASKAWA ASIA PACIFIC PTE. LTD.
30A, Kallang Place, #06-01, 339213, Singapore
Phone: +65-6282-3003 Fax: +65-6289-3003
http://www.yaskawa.com.sg

YASKAWA ELECTRIC (THAILAND) CO., LTD.
59, 1st-5th Floor, Flourish Building, Soi Ratchadapisek 18, Ratchadapisek Road, Huaykwang, Bangkok, 10310, Thailand
Phone: +66-2-017-0099 Fax: +66-2-017-0799
http://www.yaskawa.co.th

YASKAWA ELECTRIC (CHINA) CO., LTD.
22F, Link Square 1, No.222, Hubin Road, Shanghai, 200021, China
Phone: +86-21-5385-2200 Fax: +86-21-5385-3299
http://www.yaskawa.com.cn

YASKAWA ELECTRIC (CHINA) CO., LTD. BEIJING OFFICE
Room 1011, Tower W3 Oriental Plaza, No.1, East Chang An Ave.,
Dong Cheng District, Beijing, 100738, China
Phone: +86-10-8518-4086 Fax: +86-10-8518-4082

YASKAWA ELECTRIC TAIWAN CORPORATION
12F, No. 207, Sec. 3, Beishin Rd., Shindian Dist., New Taipei City 23143, Taiwan
Phone: +886-2-8913-1333 Fax: +886-2-8913-1513 or +886-2-8913-1519
http://www.yaskawa.com.tw

YASKAWA

YASKAWA ELECTRIC CORPORATION

In the event that the end user of this product is to be the military and said product is to be employed in any weapons systems or the manufacture thereof, the export will fall under the relevant regulations as stipulated in the Foreign Exchange and Foreign Trade Regulations. Therefore, be sure to follow all procedures and submit all relevant documentation according to any and all rules, regulations and laws that may apply.

Specifications are subject to change without notice for ongoing product modifications and improvements.

MANUAL NO. SIEP C880700 14H <20>-0

Published in Japan November 2018
18-10-15