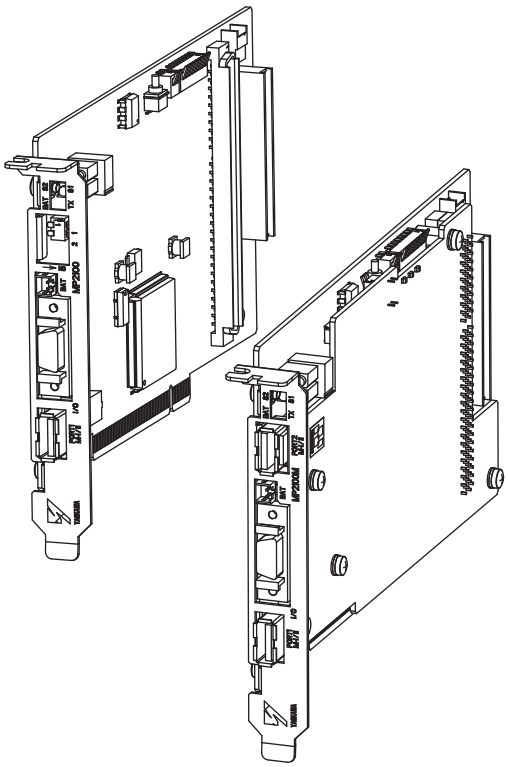
YASKAWA

Machine Controller

MP210□/MP210□M

USER'S MANUAL
Design and Maintenance

Model: JAPMC-MC2100, -MC2100-E, -MC2102-E
JAPMC-MC2140, -MC2140-E, -MC2142-E



MANUAL NO. SIEP C880700 01E

Using this Manual

The MP2100/MP2100M is a PCI-board-type Machine Controller that contains the CPU, I/O and MECHATROLINK-II communication functions in a single unit.

Read this manual carefully to ensure the proper use of the MP2100/MP2100M. Also, keep this manual in a safe place so that it can be referred to whenever necessary.

■ 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 Specifications and Functions	√	√	–	–	–	–
Chapter 3 Installation and Connection	–	√	√	√	–	–
Chapter 4 System Startup and Sample Programs	–	–	√	–	√	–
Chapter 5 Outline of Motion Control Systems	–	√	√	√	–	–
Chapter 6 Maintenance, Inspection, and Troubleshooting	–	–	–	–	√	√
Appendix A System Registers Lists	–	–	√	–	√	–
Appendix B Motion Parameter Lists	–	–	√	–	√	–
Appendix C Motion API	–	–	√	–	√	–

■ Basic Terms

Unless otherwise specified, the following definitions are used:

• MP2100	Machine Controller MP2100
• MP2100M	Machine Controller MP2100M
• PC	Programmable Logic Controller
• PP	Programming Panel
• MPE720	The Programming Device Software or a Programming Device (i.e., a personal computer) running the Programming Device Software
• MECHATROLINK-I, M-I	MECHATROLINK-I Communications
• MECHATROLINK-II, M-II	MECHATROLINK-II Communications

■ MPE720 Engineering Tool Version Number

In this manual, the operation of MPE720 is described using screen captures of MPE720 version 7.

For this reason, the screen captures and some descriptions may differ for MPE720 version 5 or version 6.

■ Visual Aids

The following aids are used to indicate certain types of information for easier reference.

IMPORTANT — Indicates important information that should be memorized or minor precautions, such as precautions that will result in alarms if not heeded.

Indicates supplemental information.

◀ EXAMPLE ▶ Indicates application examples.

Describes technical terms that are difficult to understand, or appear in the text without an explanation being given.

■ 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).

■ Indication of Reverse Signals

In this manual, the names of reverse signals (ones that are valid when low) are written with a forward slash (/) before the signal name, as shown in the following example:

Notation Examples

- $\overline{\text{S-ON}}$ = /S-ON
- $\overline{\text{P-CON}}$ = /P-CON

■ Copyrights

- EtherNet/IP and DeviceNet is a registered trademark of the ODVA (Open DeviceNet Vendor Association, Inc.).
- PROFIBUS is a trademark of the PROFIBUS User Organization.
- EtherCAT is a registered trademark of Beckhoff Automation GmbH, Germany.
- Ethernet is a registered trademark of the Xerox Corporation.
- PROFINET is a registered trademark of PROFIBUS & PROFINET International.
- CC-Link is a trademark of the Mitsubishi Electric Corporation.
- MPLINK is a registered trademark of the YASKAWA Electric Corporation.
- Microsoft, Windows, Windows NT, and Internet Explorer are registered trademarks of the Microsoft Corporation.
- Pentium is a registered trademark of the Intel Corporation.
- MECHATROLINK is a trademark of the MECHATROLINK Members Association.
- Other product names and company names are the trademarks or registered trademarks of the respective company. “TM” and the ® mark do not appear with product or company names in this manual.

■ Related Manuals

Refer to the following related manuals as required. Thoroughly check the specifications, restrictions, and other conditions of the product before attempting to use it.

Manual Name	Manual Number	Contents
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 optional SVB-01 module and the SVB module that is built into the MP2000 Series Machine Controllers.
Machine Controller MP2000 Series SVC-01 Motion Module USER'S MANUAL	SIEP C880700 41	Describes the functions, specifications, and application methods of the SVC-01, SVC Motion Module for the MP2000 Series Machine Controllers.
Machine Controller MP2000 Series SVA-01 Motion Module User’s Manual	SIEP C880700 32	Describes the functions, specifications, and application methods of the SVA-01, SVA Motion Module for the MP2000 Series.
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 PO-01, Pulse Output Motion Module for the MP2000 Series Machine Controllers.
Machine Controller MP2000 Series MPU-01 Multiple-CPU Module User’s Manual	SIEP C880781 05	Describes the functions, specifications, and application methods of the Multiple-CPU Module for the MP2000 Series.
Machine Controller MP2000 Series Communication Module User’s Manual	SIEP C880700 04	Describes the functions, specifications, and application methods of the MP2000 Series Communication Modules (217IF, 218IF, 260IF, 261IF).
Machine Controller MP2000 Series I/O Module User’s Manual	SIEP C880700 34	Describes the functions, specifications, and application methods of the I/O Module for the MP2000 Series Machine Controllers.
Machine Controller MP2000 Series Analog Input/Analog Output Module AI-01/AO-01 User’s Manual	SIEP C880700 26	Describes the functions, specifications, and application methods of the Analog Input Module AI-01 and Analog Output Module AO-01 for the MP2000 Series Machine Controllers.
Machine Controller MP2000 Series Counter Module CNTR-01 User’s Manual	SIEP C880700 27	Describes the functions, specifications, and application methods of the Counter Module CNTR-01 for the MP2000 Series Machine Controllers.
Machine Controller MP2000 Series 262IF-01 FL-net Communication Module User's Manual	SIEP C880700 36	Describes the 262IF-01 FL-net Communication Module that can be connected to MP2000 Series Machine Controllers.
Machine Controller MP2000 Series 263IF-01 EtherNet/IP Communication Module User's Manual	SIEP C880700 39	Describes the 263IF-01 EtherNet/IP Communication Module that can be connected to MP2000 Series Machine Controllers.

(cont'd)

Manual Name	Manual Number	Contents
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 User's Manual, Ladder Programming	SIEZ-C887-1.2	Describes the instructions used in MP2000 ladder programming.
Machine Controller MP2000 Series USER'S MANUAL for Motion Programming	SIEP C880700 38	Describes the motion program language used with the MP2000 Series Machine Controllers.
System Integrated Engineering Tool MPE720 Ver.7 USER'S MANUAL	SIEP C880761 03	Describes the operating procedures for MPE720 Ver.7, which is used as the engineering tool for controllers.
Engineering Tool for MP2000 Series Machine Controller MPE720 Version 6 User's Manual	SIEP C880700 30	Describes the installation and operation of the engineering tools for MP2000 Series Machine Controller MPE720 Version 6.
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 User's Manual MECHATROLINK System	SIEZ-C887-5.1	Describes MECHATROLINK distributed I/O for MP900/MP2000 Series Machine Controllers.

Safety Information

The following conventions are used to indicate precautions in this manual. These precautions are provided to ensure the safe operation of the MP2100/MP2100M 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 apply to inspection when products arrive, storage and transportation, mounting, wiring, operation and inspection, and disposal. These precautions are important and must be observed.

⚠ WARNING

- Before connecting the Machine Controller and starting operation, ensure that an emergency stop procedure has been provided and is working correctly.

  There is a risk of injury.

- Observe all the procedures and instructions given in this manual for trial operation.

  Operating mistakes while the servomotors and machine are connected may damage the machine, and in some cases result in accidents causing injury or death.

- Do not disconnect any cable when power is being supplied.

  There is a risk of machine failure or electrical shock.

- 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.

- 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 MP2100/MP2100M.

- Do not attempt to modify the MP2100/MP2100M in any way.

  There is a risk of injury or device damage.

- Do not approach the machine when there is a momentary power loss to the power supply. When the power is restored, operations may start suddenly. Take steps to ensure human safety in the event that operation restarts suddenly.

  There is a risk of injury.

■ Storage and Transportation

⚠ CAUTION

- Do not store or install the MP2100/MP2100M 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 MP2100/MP2100M during transportation.

  There is a risk of injury or an accident.
- 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.

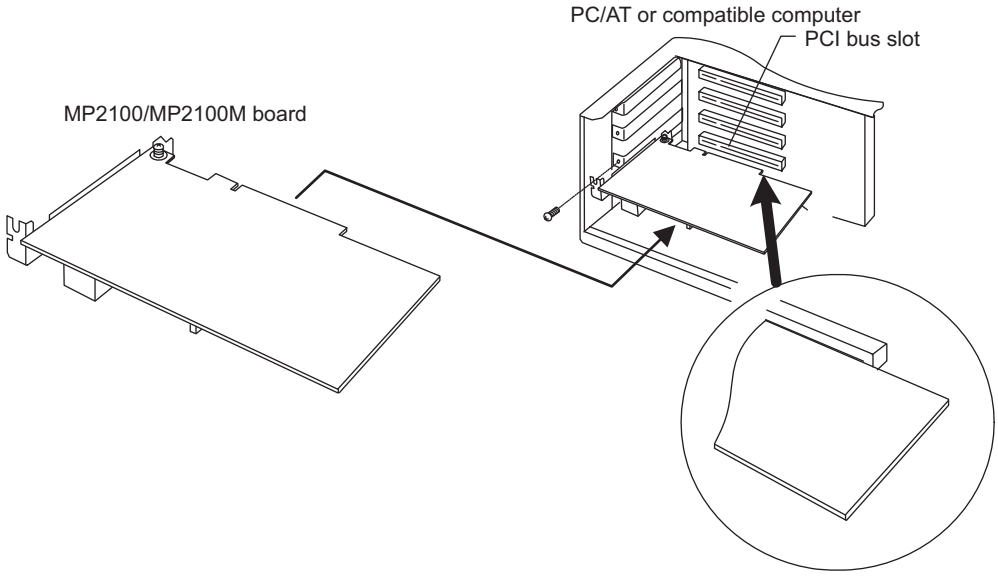
  Example: Heat treatment, where materials are kiln-dried to a core temperature of 56°C for 30 minutes or more.

  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

- The MP2100/MP2100M is mounted in the PCI slot of a standard personal computer (IBM PC/AT or compatible).



- To prevent the MP2100/MP2100M from being damaged by static electricity, discharge any static electricity by touching a grounded metal object.
- Before installing or removing the MP2100/MP2100M, always turn OFF the host computer's power supply and unplug the computer's power cord.
- When installing the MP2100/MP2100M, always press the Board firmly until it is fully seated in the PCI slot. If the Board is not fully inserted, the MP2100/MP2100M and/or host computer may be damaged or operate incorrectly. If the MP2100/MP2100M cannot be inserted into the PCI slot with firm pressure, do not try to force it into the slot. Remove the Board, align it properly, and try inserting it again.
- When handling the MP2100/MP2100M, hold the Board by its edges and never touch the components or soldered connections. Touching the components or leads can cause cuts or damage the MP2100/MP2100M or host computer.
- In some computers, it is necessary to secure the MP2100/MP2100M in the PCI slot with a screw or a clip after inserting the Board into the slot. Refer to the host computer's user manual for details on securing PCI Boards. If the MP2100/MP2100M is not secured, it may become loose and the MP2100/MP2100M and/or host computer may be damaged or operate incorrectly.
- Connect and secure the cables that connect the MP2100/MP2100M to the Servodrive and I/O Module. If the cable connectors are not fully inserted and secured, the MP2100/MP2100M, Servodrive, and/or I/O Module may operate incorrectly.
- Never use the MP2100/MP2100M 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 subject the MP2100/MP2100M 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 MP2100/MP2100M 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.

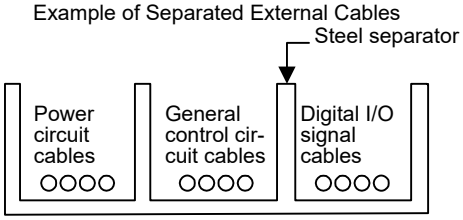
■ Selecting, Separating, and Laying External Cables

⚠ CAUTION

- Consider the following items when selecting the I/O signal lines (external cables) to connect the MP2100/MP2100M 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 MP2100/MP2100M.

  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 MP2100/MP2100M, transfer the programs and parameters from the old controller to the new controller before restarting operation.

  There is a risk of device damage.

■ Disposal Precautions

⚠ CAUTION

- Correctly discard the product and used batteries as stipulated by regional, local, and municipal laws and regulations. Be sure to include these contents in all labelling and warning notifications on the final product as necessary.

■ 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) 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 MP2100/MP2100M Machine Controller.

1.1 MP2100 Overview

1.1.1 MP2100 Features

The MP2100 is a Machine Controller with complete sequence control and motion control functionality integrated into a half-size PCI Board. Just install the MP2100 in a personal computer to provide high-speed communication with the servo drive, automatic setup, and full system support from design to maintenance.

(1) Reduced Size

The PCI bus and half-size configuration make the MP2100 compatible with almost any personal computer.

The size has been reduced to one fourth of that of the MP910.

(2) Powerful Computer Motion Functions

A large library with 50 motion API (software interface) is provided in programming languages such as C. This is a valuable resource when creating host computer applications.

(3) High Performance

- Control characteristics have been improved by increasing the CPU and Motion Network (MECHATROLINK-II) speed.
  MECHATROLINK-II baud rate: 2.5 times faster
- MECHATROLINK-II enables position control, speed control, and torque control and makes precise synchronous control possible. The control mode can also be changed while online, facilitating complicated machine operations.

(4) Easy to Use

- The time required to start the machine can be greatly reduced by using the self-configuration function that automatically detects devices connected to MECHATROLINK-II and automatically sets the required parameters.
- The Application Converter can use your previous software assets with their accumulated databanks of specific knowledge to improve the system further.

(5) Expandability

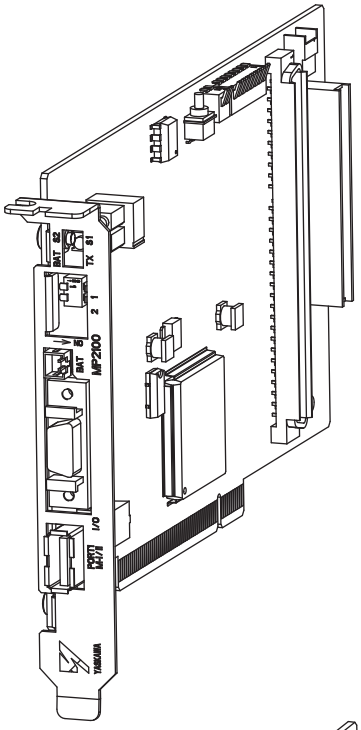
Up to four MP2100 Boards can be installed in one host computer. Each MP2100 can control up to 21 stations, including 16 servo axes and I/O devices. A single host computer with four MP2100 Boards can thus control up to 84 stations, including 64 servo axes and I/O devices.

(6) Addition of a High-speed Version

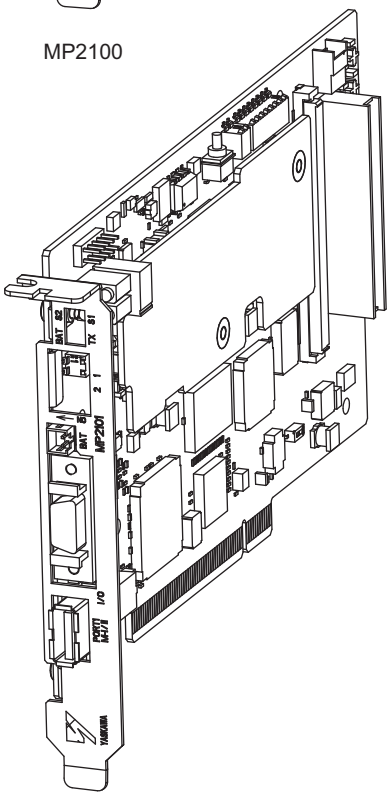
The MP2101 has been added as a high-speed version of the MP2100.

1.1.2 MP2100 Appearance

The following figures show the external appearance of the MP2100 and MP2101.



MP2100



MP2101

1.2 MP2100M Overview

1.2.1 MP2100M Features

The MP2100M is a Machine Controller that has a CPU Board equivalent to the MP2100 and an SVB Board mounted to the option connector.

The SVB Board provides one MECHATROLINK-II port, and can be used to control servo drives, inverters, and I/O devices.

The MP2100M has the following features in addition to those of the MP2100.

(1) Controls Up to 32 Servo Axes from a Single MP2100M

The two MECHATROLINK ports provided on the SVB built into the CPU Board and the SVB Motion Module can be used to control up to 32 Servo axes from a single MP2100M. Up to 42 stations can be controlled when including I/O devices. The CPU Module and SVB Motion Module can also be synchronized, enabling interpolation and synchronization applications between CPU Modules with Built-in SVB and SVB Modules.

(2) High-speed Motion Control of 0.5 ms

The SVB Motion Module's MECHATROLINK port can be used on its own to control up to four axes using a 0.5-ms motion control cycle.

Set whether to use the two MP2100M ports (SVB port built into the CPU Module and SVB Motion Module port) or high-speed processing (SVB Motion Module only) in the MPE720 Module Configuration Definition.

(3) Addition of a High-speed Version

The MP2101M has been added as a high-speed version of the MP2100M.

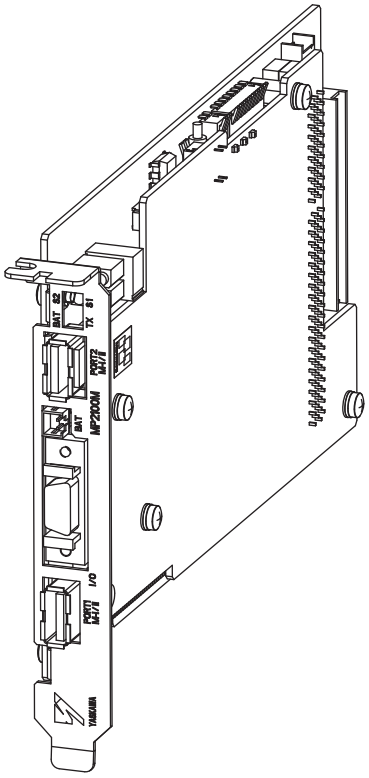
1.2.2 Comparing SVB Motion Module and SVB Built into the CPU Module

The following table shows the differences between the SVB built into the CPU Module and the separate SVB Motion Module.

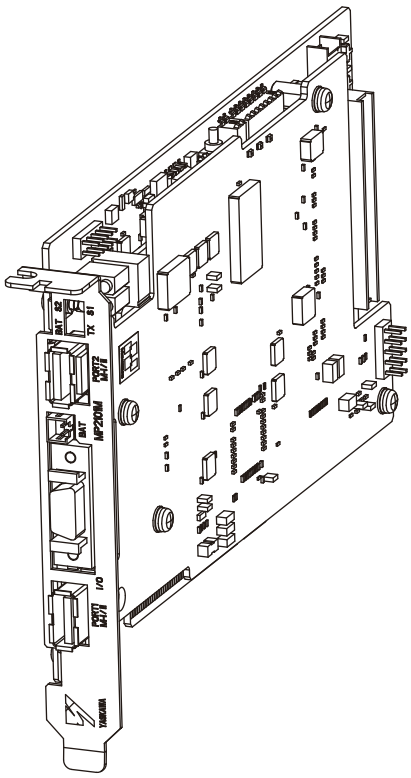
No.	Function/Characteristics	SVB Built-into CPU Module	SVB Motion Module
1	MECHATROLINK-II communication cycle	1 ms, 1.5 ms, or 2 ms	0.5 ms, 1 ms, 1.5 ms, or 2 ms
2	Minimum command delay time: Application to Servo (Scan delay + communication delay)	2 ms = 1 ms (high-speed scan) + 1 ms (communication)	1.5 ms = 1 ms (high-speed scan) + 0.5 ms (communication)
3	Minimum response delay time: Servo to Application (Scan delay + communication delay)	1 ms = 0 ms (high-speed scan) + 1 ms (communication)	1.5 ms = 1 ms (high-speed scan) + 0.5 ms (communication)
4	Slave functions	None	Supported (asynchronous only)

1.2.3 MP2100M Appearance

The following figures show the external appearance of the MP2100M and MP2101M.



MP2100M



MP2101M

1.3 MP2100/MP2100M Modules

The following table shows the names and descriptions of the MP2100/MP2100M Modules and Optional Modules.

Group		Name	Description	Model	Description
MP2100		Basic Module	MP2100	JAPMC-MC2100-E JAPMC-MC2100	MECHATROLINK-I/MECHATROLINK-II, 16 axes maximum 5 inputs, 4 outputs
MP2101		High-speed Version of the MP2100	MP2101	JAPMC-MC2102-E	
MP2100M		Basic Module	MP2100M	JAPMC-MC2140-E JAPMC-MC2140	MECHATROLINK-I/MECHATROLINK-II, 32 axes maximum 5 inputs, 4 outputs
MP2101M		High-speed Version of the MP2100M	MP2101M	JAPMC-MC2142-E	
Optional Modules*	Motion Modules	MECHATROLINK-II Motion-control Module	SVB-01	JAPMC-MC2310-E JAPMC-MC2310	MECHATROLINK-I, -II, 16 axes maximum
		MECHATROLINK-III Motion-control Module	SVC-01	JAPMC-MC2320-E	MECHATROLINK-III, 16 axes maximum (Anticipated)
		Analog Output Motion-control Module	SVA-01	JAPMC-MC2300-E JAPMC-MC2300	Analog output, 2 axes maximum
		Pulse Output Motion-control Module	PO-01	JAPMC-PL2310-E	Pulse output, 4 axes maximum
		Multiple-CPU Module	MPU-01	JAPMC-CP2700-E	Optional module (it has both CPU functions and the functions of a built-in SVC-01.)
	I/O Modules	I/O Module	LIO-01	JAPMC-IO2300-E JAPMC-IO2300	16 inputs, 16 outputs (sink mode output) 1 pulse input
			LIO-02	JAPMC-IO2301-E JAPMC-IO2301	16 inputs, 16 outputs (source mode output) 1 pulse input
			LIO-04	JAPMC-IO2303-E JAPMC-IO2303	32 inputs, 32 outputs (sink mode output)
			LIO-05	JAPMC-IO2304-E JAPMC-IO2304	32 inputs, 32 outputs (source mode output)
			LIO-06	JAPMC-IO2305-E	8 inputs, 8 outputs (sink mode output) Analog input, 1 channel Analog output, 1 channel 1 pulse input
		Analog Input Module	AI-01	JAPMC-AN2300-E JAPMC-AN2300	Analog input, 8 channels
		Analog Output Module	AO-01	JAPMC-AN2310-E	Analog output, 4 channels
		Input Module	DI-01	JAPMC-DI2300-E	64 inputs
		Output Module	DO-01	JAPMC-DO2300-E JAPMC-DO2300	64 outputs (sink mode output)
		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 10Base-T Ethernet communication
			218IF-02	JAPMC-CM2302-E	RS-232C and 100Base-TX/10Base-T Ethernet communication
		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)

(cont'd)

Group		Name	Description	Model	Description
Optional Modules*	Communication Modules	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 Slave Module	269IF-01	JAPMC-CM2308-E	CC-Link IE Field communication (CC-Link IE Field slave)
		MPLINK/CP-215 Communication Module	215AIF-01	JAPMC-CM2360-E JAPMC-CM2360	RS-232C, MPLINK, and CP-215 communication
	Communication Modules (From other manufacturers)	AnyWire-DB Master Module	AFMP-01	–	AnyWire-Master DB by Anywire Corporation
		CC-Link Interface Board	AFMP-02-C	–	CC-Link by Anywire Corporation
			AFMP-02-CA	–	CC-Link and AnyWire-Master DB by Anywire Corporation
		A-net/A-Link Mster Module	MPANL00-0	–	A-net/A-Link by ALGO System
		CUnet Master Unit Module	MPCU-NET-0	–	CUnet by ALGO System
		HLS Master Module	MPHLS-01	–	HLS by M-System Co.,Ltd.
	Expansion Interface Module	Connection Module between Racks	EXIOIF	JAPMC-EX2200-E JAPMC-EX2200	System expansion (Configuration of MP2100M + max. 3 racks)

* Optional Modules can only be used with the MP2100M/MP2101M.
To use Optional Modules with the MP2100M/MP2101M, the following devices are required.

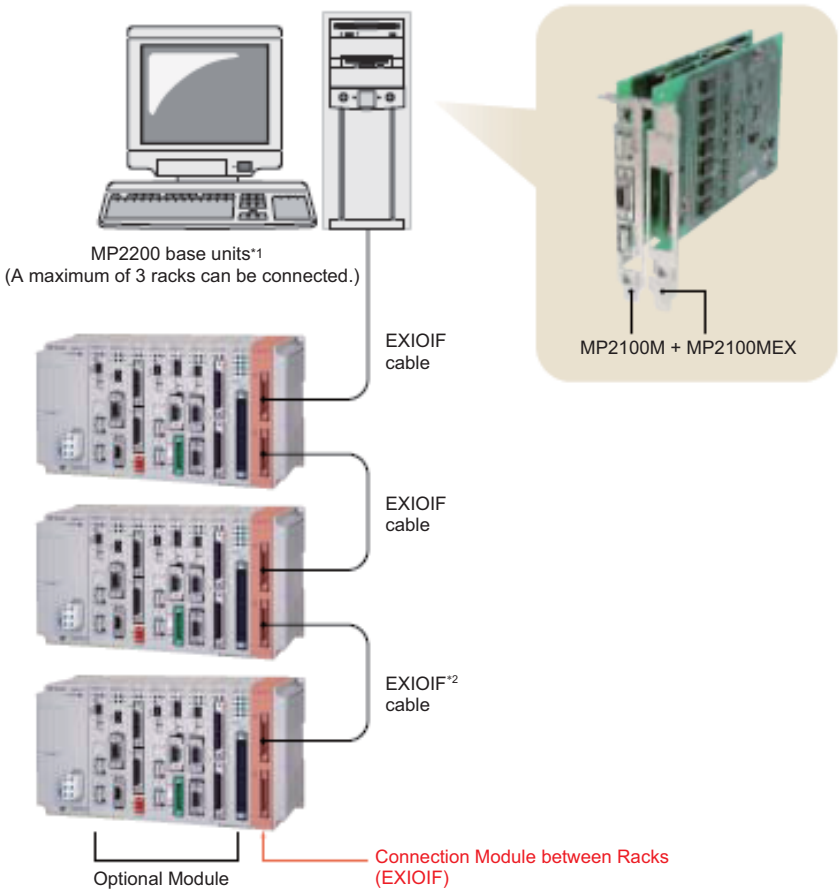
- MP2100MEX interface board (JAPMC-EX2100)
- MP2200 base unit (JEPMC-BU2200 (-E) 100V type, or JEPMC-BU2210 (-E) 24V type)
- EXIOIF connection module (JAPMC-EX2200)

For details on the MP2100 MEX, refer to the following section.

IMPORTANT

■ What is the MP2100MEX?

The MP2100MEX is an interface board for rack extension for use with the MP2100M/MP2101M. When the MP2100MEX is fitted to an MP2100M and they are installed in a computer, a maximum of three racks of MP2200 base units (a maximum of 24 Optional Modules) can be connected by using EXIOIF cables and EXIOIF Modules.



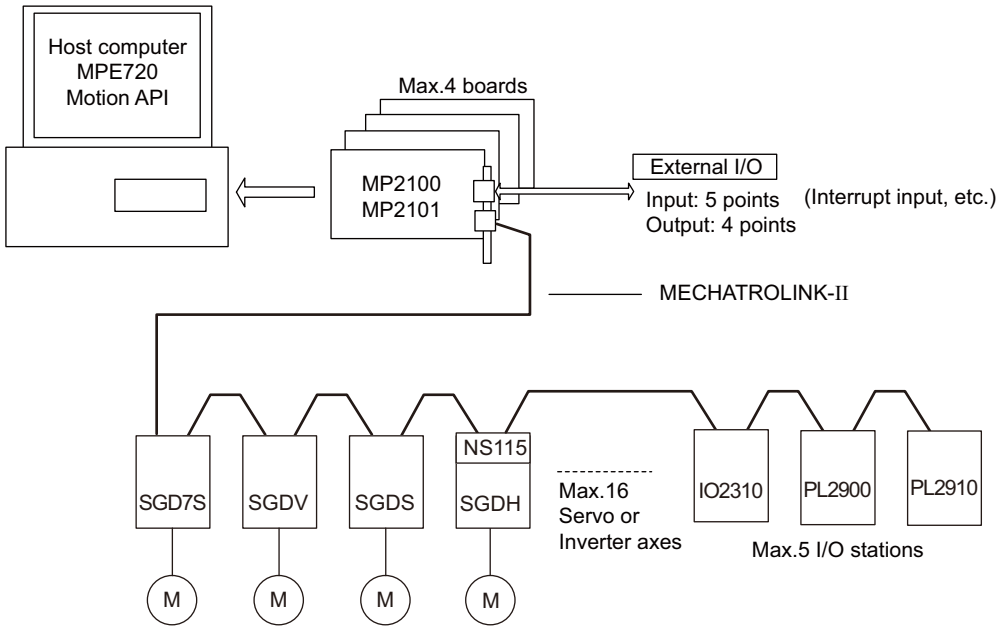
* 1. A maximum of 9 Optional Modules can be installed on each rack of the MP2200 base unit, making a maximum of 24 Optional Modules (excluding EXIOIF Modules) with 3 racks.

* 2. The total length of EXIOIF cable must not exceed 6.0 m.

1.4 Example of System Configuration

1.4.1 Example of MP2100 System Configuration

The following diagram shows an example of system configuration in which MP2100/MP2101 is used.

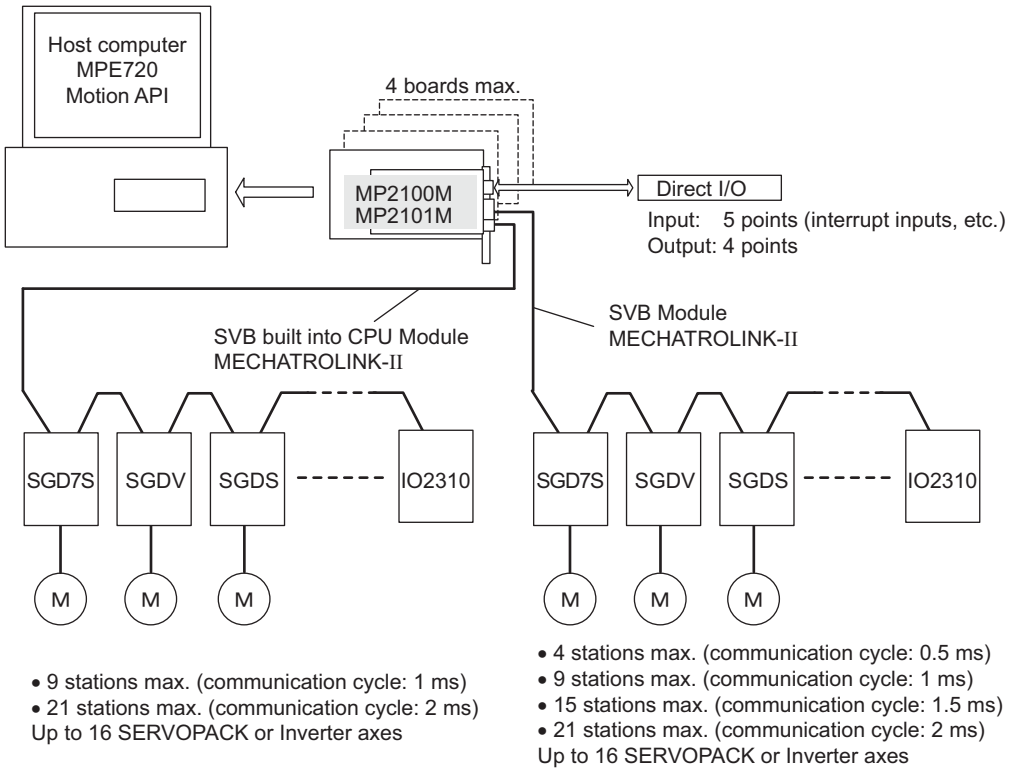


Note 1. Use MPE720 Ver.7, Ver.6 or Ver.5 (Version 5.10 or later), and Motion API Ver. 1.05 (driver: Ver. 1.1.3.0) or later.

2. Up to 21 devices can be connected to MECHATROLINK-II. (The SERVOPACKs and Inverters can be connected for to up to 16 axes.)
3. Up to 9 digital I/O points (5 inputs and 4 outputs) can be used.
4. 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.
5. Different SERVOPACKs are connected to MECHATROLINK-I and MECHATROLINK-II.
   Select the SERVOPACKs that can be connected by referring to *1.5 Devices Connectable to MECHATROLINK*.
6. If devices compatible with MECHATROLINK-I (4 Mbps) and with MECHATROLINK-II (10 Mbps) are used together, make the all settings for MECHATROLINK-I (4 Mbps).
7. When connecting SERVOPACKs via MECHATROLINK, connect the overtravel, zero point return deceleration limit switch and external latch signals to the SERVOPACKs.
   For the details on connection, refer to the SERVOPACK's manual.

1.4.2 Example of MP2100M System Configuration

The following diagram shows an example of system configuration in which MP2100M/MP2101M is used.



Note 1. Use MPE720 Ver.7, Ver.6 or Ver.5 (Version 5.10 or later), and Motion API Ver. 1.05 (driver: Ver. 1.1.3.0) or later.

2. Up to 42 devices can be connected to MECHATROLINK-II. (The SERVOPACKs and Inverters can be connected for to up to 32 axes.)
3. Up to 9 digital I/O points (5 inputs and 4 outputs) can be used.
4. 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.
5. Different SERVOPACKs are connected to MECHATROLINK-I and MECHATROLINK-II.
   Select the SERVOPACKs that can be connected by referring to *1.5 Devices Connectable to MECHATROLINK*.
6. If devices compatible with MECHATROLINK-I (4 Mbps) and with MECHATROLINK-II (10 Mbps) are used together, make the all settings for MECHATROLINK-I (4 Mbps).
7. When connecting SERVOPACKs via MECHATROLINK, connect the overtravel, zero point return deceleration limit switch and external latch signals to the SERVOPACKs.
   For the details on connection, refer to the SERVOPACK's manual.

1.5 Devices Connectable to MECHATROLINK

The devices that are compatible with MECHATROLINK and can be connected to the MP2100/MP2100M 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 Number	Details	MECHATROLINK-I	MECHATROLINK-II
JEPMC-IO350	64-point I/O Module 24 VDC, 64 inputs, 64 outputs (sink)	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

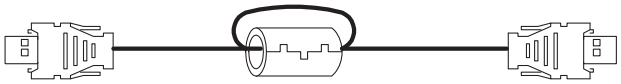
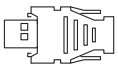
(cont'd)

Model Number	Details	MECHATROLINK-I	MECHATROLINK-II
JAMSC-IO2900-E	DC Input Module 24 VDC, 16 inputs	Yes	Yes
JAMSC-IO2910-E	DC Output Module 24 VDC, 16 outputs	Yes	Yes
JEPMC-IO2310 JEPMC-IO2310-E	64-point I/O Module 24 VDC, 64 inputs, 64 outputs (sink)	Yes	Yes
JEPMC-IO2330 JEPMC-IO2330-E	64-point I/O Module 24 VDC, 64 inputs, 64 outputs (source)	Yes	Yes
JEPMC-PL2900 JEPMC-PL2900-E	Counter Module Reversible counter, 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
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
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 YV250 Machine Vision System	Yes	Yes
JEVSA-YV260□□-E	MYVIS YV260 Machine Vision System	Yes	Yes
JEPMC-MC400	MP940 Machine Controller	Yes	No

1.6 Cables, Included Accessories, and Options

1.6.1 Cables

The following table shows the cables that can be connected to the MP2100/MP2100M.

Name and Specification/External View	Model Number	Length
MECHATROLINK Cable MECHATROLINK connector - MECHATROLINK connector 	JEPMC-W6002-A5-E	0.5m
	JEPMC-W6002-01-E	1 m
	JEPMC-W6002-03-E	3 m
	JEPMC-W6002-05-E	5 m
	JEPMC-W6002-10-E	10 m
	JEPMC-W6002-20-E	20 m
	JEPMC-W6002-30-E	30 m
	JEPMC-W6002-40-E	40 m
	JEPMC-W6002-50-E	50 m
MECHATROLINK Cable MECHATROLINK connector - MECHATROLINK connector, with ferrite core 	JEPMC-W6003-A5-E	0.5 m
	JEPMC-W6003-01-E	1 m
	JEPMC-W6003-03-E	3 m
	JEPMC-W6003-05-E	5 m
	JEPMC-W6003-10-E	10 m
	JEPMC-W6003-20-E	20 m
	JEPMC-W6003-30-E	30 m
	JEPMC-W6003-40-E	40 m
	JEPMC-W6003-50-E	50 m
Terminator (terminating resistor) 	JEPMC-W6022-E	-
I/O Cable 	JEPMC-W2062-□□-E	-

* Commercially-available USB cables cannot be used. Always use the above mentioned cable.

1.6.2 Accessories and Options

Name	Model	Remarks
Relay Cable for Battery	JEPMC-W2092-B8	This cable is included only with the MP2100M (model: JAPMC-MC2140). For details on installation, refer to *6.2 Replacing the Battery*.
Battery	JZSP-BA01	Option
Extended Cable for Battery	JEPMC-W2090-01-E	Option

1.7 Software

1.7.1 The Programming Device Software

Name	Model	Remarks
MPE720	CPMC-MPE720 (ver. 5.10 or later)	CD-ROM
MPE720 ver. 6	CPMC-MPE770D (all versions)	DVD-ROM
MPE720 ver. 7	CPMC-MPE780D (all versions)	DVD-ROM

1.7.2 Motion API

Name	Model	Remarks
Motion API	CPMC-MPA700	CD-ROM

2

Specifications and Functions

This chapter explains specifications in detail for the MP2100/MP2100M and gives an overview of its functions.

2.1 Specifications

2.1.1 Hardware Specifications

The following table shows the hardware specifications of the MP2100/MP2100M.

Item	Specifications			
Name	MP2100	MP2101	MP2100M	MP2101M
Model Number	JAPMC-MC2100-E JAPMC-MC2100	JAPMC-MC2102-E	JAPMC-MC2140-E JAPMC-MC2140	JAPMC-MC2142-E
Power Supply	+5 VDC: Supplied from PCI bus, approx. 0.5 A max.		+5 VDC: Supplied from PCI bus, approx. 1.0 A max.	
Flash Memory	MP2100/MP2100M 8 MBytes (User area 5.5 MBytes) MP2101/MP2101M 16 MBytes (User area 11.5 MBytes)			
SDRAM	16 MBytes			
SRAM	256 kBytes: M registers, S registers, trace memory, alarm history (battery backup)			
Motion Network MECHATROLINK	Motion network 1 channel SERVOPACK and I/O for up to 21 stations connectable (SERVOPACK for up to 16 axes) Transmission speed: 4 Mbps (MECHATROLINK-I) or 10 Mbps (MECHATROLINK-II)		Motion network 2 channels SERVOPACK and I/O for up to 42 stations connectable (SERVOPACK for up to 32 axes) Transmission speed: 4 Mbps (MECHATROLINK-I) or 10 Mbps (MECHATROLINK-II)	
I/O Signal	5 digital inputs (1 input also used as an interrupt input) 24 VDC, 4.1 mA (TYP) source mode/sink mode inputs 4 digital outputs 24 VDC, 100 mA (TYP) open-collector sink mode outputs			
Indicators	• Bi-level, Bi-color type LED S1: Green (RDY), Red (ERR) S2: Green (RUN), Red (ALM) • Bi-level type LED TX: green BAT: red		CPU Module • Bi-level, Bi-color type LED S1: Green (RDY), Red (ERR) S2: Green (RUN), Red (ALM) • Bi-level type LED TX: green BAT: red SVB Module 2-color 2-level indication LED D3: Green (RUN) Red (ERR) D4: Green (TX) Red (reserve)	
Switches	• Mode Switch 1 1: INT 2: STOP • Mode Switch 2 1: TEST 2: MON 3: CONFIG 4: SUP 5: SLOT0 6: SLOT1 • Reset Switch		CPU Module • Mode Switch 1 1: TEST 2: MON 3: CONFIG 4: INIT 5: SUP 6: STOP 7: SLOT0 8: SLOT1 • Reset Switch	
Dimensions (mm)	106.68 × 174.63 (W × D) (PCI half-size)			
Approx. Mass	140 g		210 g	

2.1.2 Specifications of Operating Environment

Item		Specifications
Ambient Temperature	Operating Temperature	0 to 55 °C
	Storage Temperature	–25 to 85 °C
Humidity		30% to 95% (with no condensation)
Power Supply Voltage		5 V ±5%
Vibration Resistance		Depends on the personal computer.
Shock Resistance		Depends on the personal computer.
Noise Resistance	Power supply noise (FT noise) Radiation noise (FT noise) Ground noise (impulse noise) Electrostatic noise (air discharge method)	Depends on the personal computer. 1000 V min., for one minute Depends on the personal computer. 4000 V min., 10 times
Ground		Conforming to the ground standard for the personal computer
Operating Altitude		2,000 m above sea level or lower
Conformity Standard		EN 55011, EN 61000-6-2, EN 61000-6-4 (Group1 ClassA)
Other		No corrosive gas
Reliability	Number of fits	500 fits
	Life	10 years max., at 40 °C

2.1.3 Function Lists

(1) PLC Function Specifications

The following table shows the PLC function specifications for MP2100/MP2100M.

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.
Scanning	Two scan levels: High-speed scan and low-speed scan • MP2100 High-speed scan time setting: 1 to 32 ms (Integral multiple of MECHATROLINK communication cycle) Low-speed scan time setting: 2 to 300 ms (Integral multiple of MECHATROLINK communication cycle) • MP2101/MP2100M/MP2101M High-speed scan time setting: 0.5 to 32 ms (0.5 ms units) Low-speed scan time setting: 2 to 300 ms (0.5 ms units)
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/sequence programs: Combined maximum of 256 (sequence programs to be compatible with Ver. 2.66 or later) 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: MP2100: 8 MBytes (User area: 5.5 MBytes) MP2101/MP2100M/MP2101M: 16 MBytes (User area: 11.5 MBytes) Files, ladder programs, motion programs, etc. Other than data backed up by battery Data memory: Battery backup: 512 kBytes, M registers, S registers, alarm history, trace data
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 number or symbol assignment

(cont'd)

Item	Specifications
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

(2) Motion Control Function Specifications

The following table shows the motion control function specifications for the MP2100/MP2100M.

Item			Specifications
Interface			MECHATROLINK-I, MECHATROLINK-II
Number of Controlled Axes/Module			MP2100: Up to 16 axes MP2100M: Up to 32 axes
Control Specifications	PTP Control		Linear, rotary, and infinite-length
	Interpolation		MP2100: Up to 16 linear axes, 2 circular axes, and 3 helical axes MP2100M: Up to 16 linear axes, 2 circular axes, and 3 helical axes
	Speed Reference Output		Yes (Only when using MECHATROLINK-II)
	Torque Reference Output		Yes (Only when using MECHATROLINK-II)
	Phase Control		Yes (Only when using MECHATROLINK-II)
	Position Control	Positioning	Yes
		External positioning	Yes
		Zero point return	Yes
		Interpolation	Yes
		Interpolation with position detection function	Yes
		JOG operation	Yes
		STEP operation	Yes
		Parameter changes during motion command execution	Yes (Only when using MECHATROLINK-II (32-byte mode))
Reference Unit			mm, inch, deg, pulse, or µm
Reference Unit Minimum Setting			1, 0.1, 0.01, 0.001, 0.0001, 0.00001
Maximum Programmable Value			–2147483648 to +2147483647 (signed 32-bit value)
Speed Reference Unit			Reference unit/s: mm/s, inch/s, deg/s, pulse/s, µm/s Reference unit/min.: mm/min, inch/ min, deg/min, pulse/min, µm/min Percentage: Percentage of rated speed
Acceleration/Deceleration Type			Linear, asymmetric, S-curve, exponent
Acceleration/Deceleration Reference Unit			Reference unit/s^2: mm/s^2, $inch/s^2$, deg/s^2, $pulse/s^2$, $\mu m/s^2$ Acceleration/deceleration time constant: Time from 0 to rated speed (ms)
Override Function			Positioning: 0.01% to 327.67% by axis
Coordinate System			Rectangular coordinates

(cont'd)

Item		Specifications
Zero Point Return	DEC1+ Phase-C pulse	Yes
	ZERO signal	Yes
	DEC1+ ZERO signal	Yes
	Phase-C pulse	Yes
	Only Phase-C pulse	Yes
	POT and Phase-C pulse	Yes
	POT	Yes
	Home limit switch and Phase-C pulse	Yes
	HOME	Yes
	NOT and Phase-C pulse	Yes
	NOT	Yes
	INPUT and Phase-C pulse	Yes
	INPUT	Yes

2.2 Functions of the MP2100/MP2100M Base Module

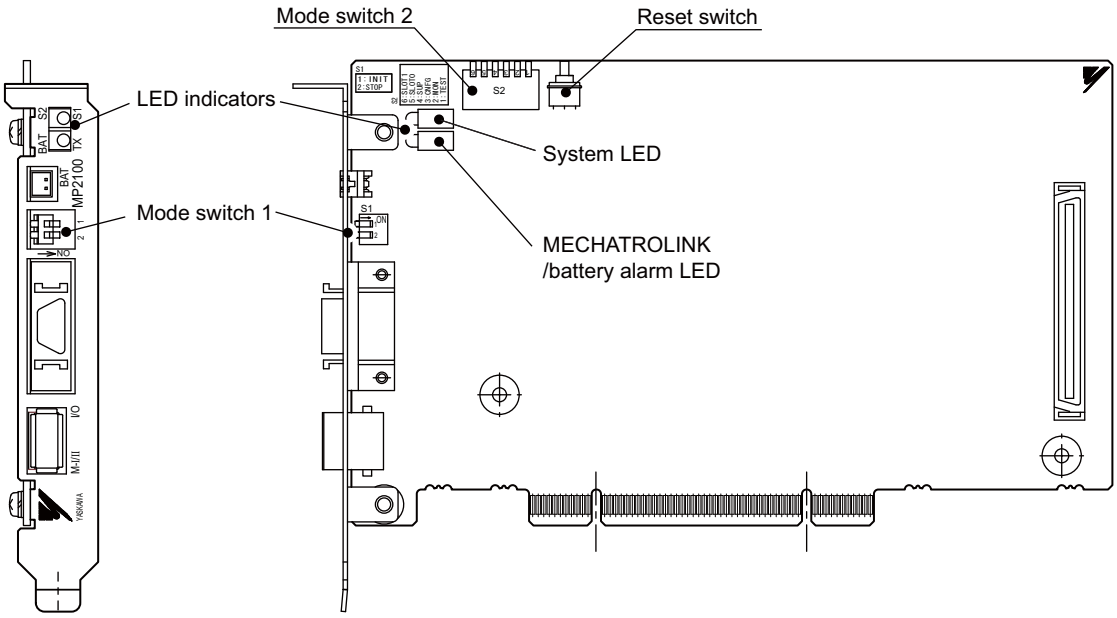
The MP2100/MP2100M Base Module is a PCI-board-type module that contains the CPU, Built-in SVB and I/O functions in a single unit, and it has both a motion control function and a sequence control function.

2.2.1 MP2100 Appearance/LED Indicators and Switch Settings

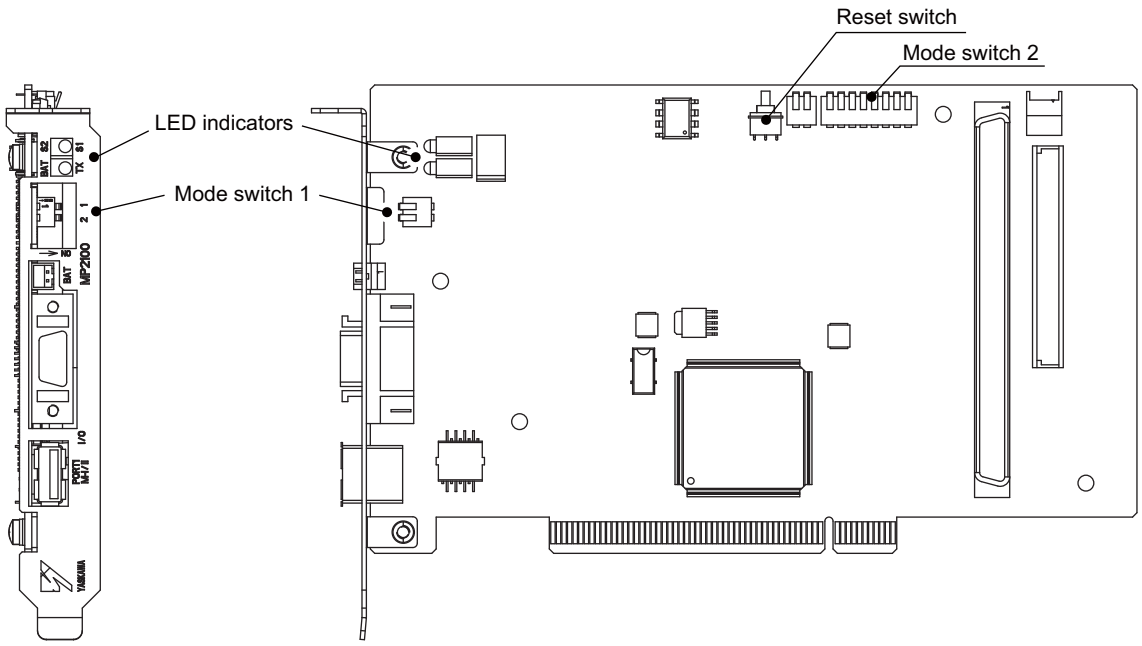
(1) Appearance

(a) JAPMC-MC2100 (non-RoHS-compliant)

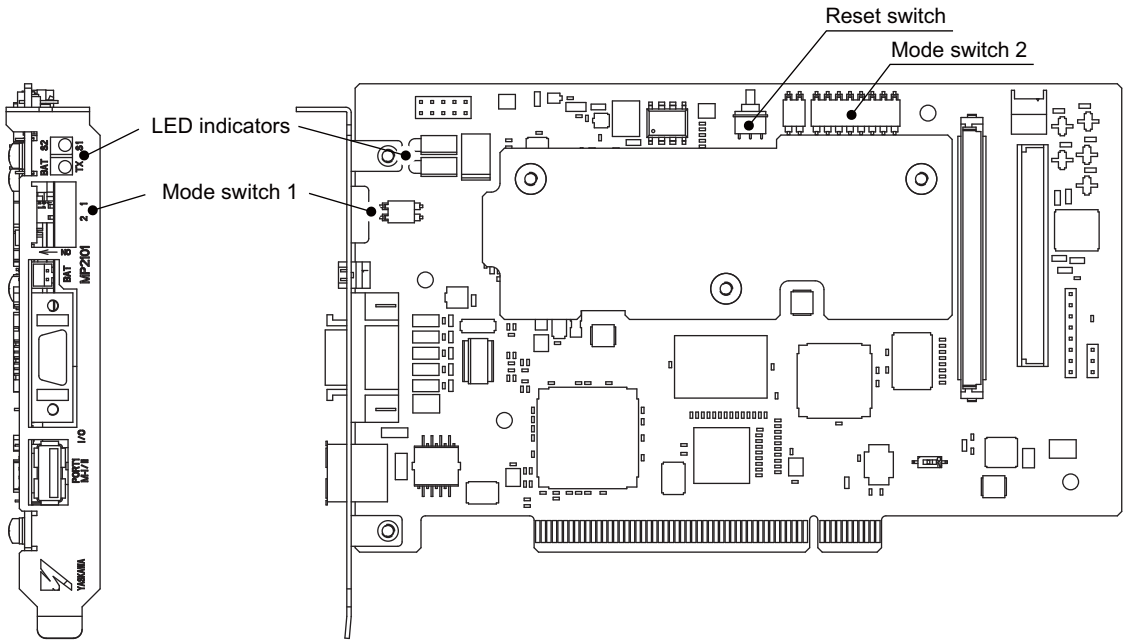
The following diagram shows the layout of the LED indicators and switches for the MP2100.



(b) JAPMC-MC2100-E



(c) JAPMC-MC2102-E



(2) LED Indicators

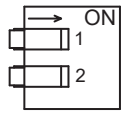
The following table shows the operating status and error status of the LED indicators of the MP2100.

Indicator	Display	Status
S1	Green	Normal operation (RDY)
	Red lit/flashing	Error (ERR)
S2	Not lit	User program stopped
	Green	Normal operation (RUN)
	Red lit/flashing	Alarm (ALM)
TX	Green	M-I/II data transmission in progress
BAT	Red	Battery alarm

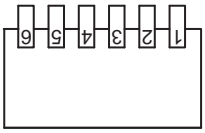
(3) Switch Settings

1. Use these switches to set the operating conditions for the JAPMC-MC2100.

Mode Switch 1 (S1)



Mode Switch 2 (S2)



Reset Switch

(a) Mode Switch 1 (S1)

No.	Name	Status*1	Function	Default Setting	Details
1	INIT	ON	Memory Clear	OFF	Set to ON to clear memory. The S registers and M registers are cleared to zero. If this switch is set to OFF, the program stored in flash memory will be executed.
		OFF	Normal operation		
2	STOP	ON	User program stopped		Set to ON to stop the user program before debugging the program.
		OFF	User program running		

(b) Mode Switch 2 (S2)

No.	Name	Status*1	Function	Default Setting	Details
1	TEST	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
2	MON	ON	System use		Always set to OFF.
		OFF	Normal operation		
3	CNFG	ON	Configuration mode		Set to ON to execute self-configuration for connected devices.
		OFF	Normal operation		
4	SUP	ON	System use		Always set to OFF.
		OFF	Normal operation		
5	SLOT0	SLOT1 - SLOT0 ON - ON ON - OFF	CP4 CP3		The CP number. These are used for MPE720 communications settings.*2
6	SLOT1	OFF - ON OFF - OFF	CP2 CP1		

* 1. ON indicates the switch has been pushed to the board side.

* 2. The CP number is used for **CPU No**. on the **Communications Setting** window when the MPE720 Ver.7 is connected. When using two or more MP2100s, set a different CP number for each MP2100.

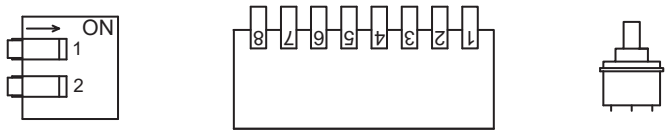
(c) Reset Switch

This switch resets the MP2100.

This switch can be used to turn ON and OFF the MP2100 power without turning the host computer ON and OFF.

2. Use these switches to set the operating conditions for the JAPMC-MC2100-E/JAPMC-MC2102-E.

Mode Switch 1 (S1) Mode Switch 2 (S2) Reset Switch



(a) Mode Switch 1 (S1)

No.	Name	Status*1	Function	Default Setting	Details
1	INIT	ON	Memory Clear	OFF	Set to ON to clear memory. The S registers and M registers are cleared to zero. If this switch is set to OFF, the program stored in flash memory will be executed.
		OFF	Normal operation		
2	STOP	ON	User program stopped		Set to ON to stop the user program before debugging the program.
		OFF	User program running		

(b) Mode Switch 2 (S2)

No.	Name	Status*1	Function	Default Setting	Details
1	TEST	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
2	MON	ON	System use		Always set to OFF.
		OFF	Normal operation		
3	CNFG	ON	Configuration mode		Set to ON to execute self-configuration for connected devices.
		OFF	Normal operation		
4	INIT	ON	Memory clear		Set to ON to clear memory. The S registers and M registers are cleared to zero. If this switch is set to OFF, the program stored in flash memory will be executed.
		OFF	Normal operation		
5	SUP	ON	System use		Always set to OFF.
		OFF	Normal operation		
6	STOP	ON	User program stopped		Set to ON to stop the user program before debugging the program.
		OFF	User program running		
7	SLOT0	SLOT1 - SLOT0 ON - ON ON - OFF OFF - ON OFF - OFF	 CP4 CP3 CP2 CP1		The CP number. These are used for MPE720 communications settings.*2
8	SLOT1				

* 1. ON indicates the switch has been pushed to the board side.

* 2. The CP number is used for **CPU No**. on the **Communications Setting** window when the MPE720 Ver.7 is connected. When using two or more MP2100s, set a different CP number for each MP2100.

(c) Reset Switch

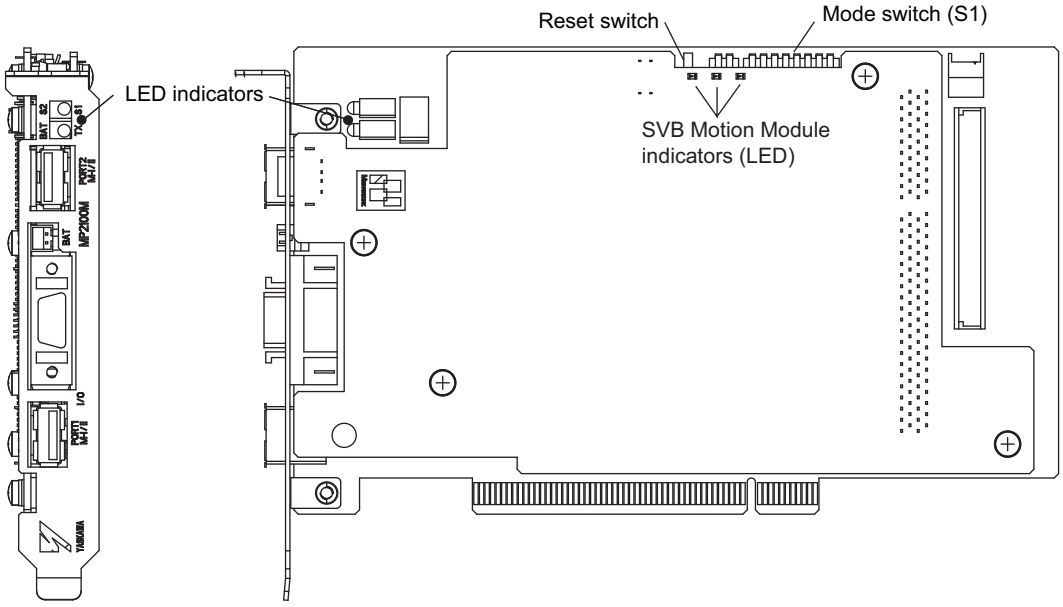
This switch resets the MP2100.

This switch can be used to turn ON and OFF the MP2100 power without turning the host computer ON and OFF.

2.2.2 MP2100M Appearance/LED Indicators and Switch Settings

(1) Appearance

The following diagram shows the layout of the LED indicators and switches for the MP2100M/MP2101M.



(2) LED Indicators

(a) CPU Module Status Indicators

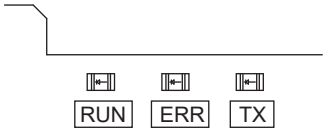
The following table shows the operating status and error status of the LED indicators of the MP2100M/ MP2101M.

.

Indicator	Display	Meaning When Lit
S1	Green	Normal operation (RDY)
	Red lit/flashing	Error (ERR)
S2	Not lit	User program stopped
	Green	Normal operation (RUN)
	Red lit/flashing	Alarm (ALM)
TX	Green	M-I/II data transmission in progress
BAT	Red	Battery alarm

Note: For details on indicator meanings, refer to *5.1.5 Indicator Patterns*.

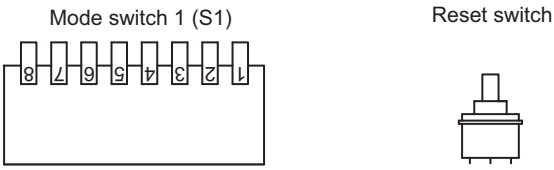
(b) SVB Module Status Indicators



Indicator	Display	Meaning When Lit
RUN	Green	Normal operation
	Not lit	Error
ERR	Red lit/flashing	Error
	Not lit	Normal operation
TX	Green	M-I/II data transmission in progress

(3) Switch Settings

Use these switches to set the operating conditions for the MP2100M/MP2101M when the power is turned ON.



(a) Mode Switch 1 (S1)

No.	Name	Status*1	Function	Default Setting	Details
1	TEST	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
2	MON	ON	System use		Always set to OFF.
		OFF	Normal operation		
3	CNFG	ON	Configuration mode		Set to ON to execute self-configuration for the connected devices.
		OFF	Normal operation		
4	INIT	ON	Memory clear		Set to ON to clear memory. The S registers and M registers are cleared to 0. If this switch is set to OFF, the program stored in flash memory will be executed.
		OFF	Normal operation		
5	SUP	ON	System use		Always set to OFF.
		OFF	Normal operation		
6	STOP	ON	User program stopped		Set to ON to Stop the user program before debugging the program.
		OFF	User program running		
7	SLOT0	SLOT1 - SLOT0			Use to set CP number.*2
8	SLOT1	ON - ON	CP4		
		ON - OFF	CP3		
		OFF - ON	CP2		
		OFF - OFF	CP1		

* 1. ON indicates the switch has been pushed to the board side.
* 2. The CP number is used for **CPU No**. on the **Communications Setting** window when the MPE720 Ver.7 is connected. When using two or more MP2100Ms, set a different CP number for each MP2100M.

(b) Reset Switch

This switch resets the MP2100M/MP2101M.

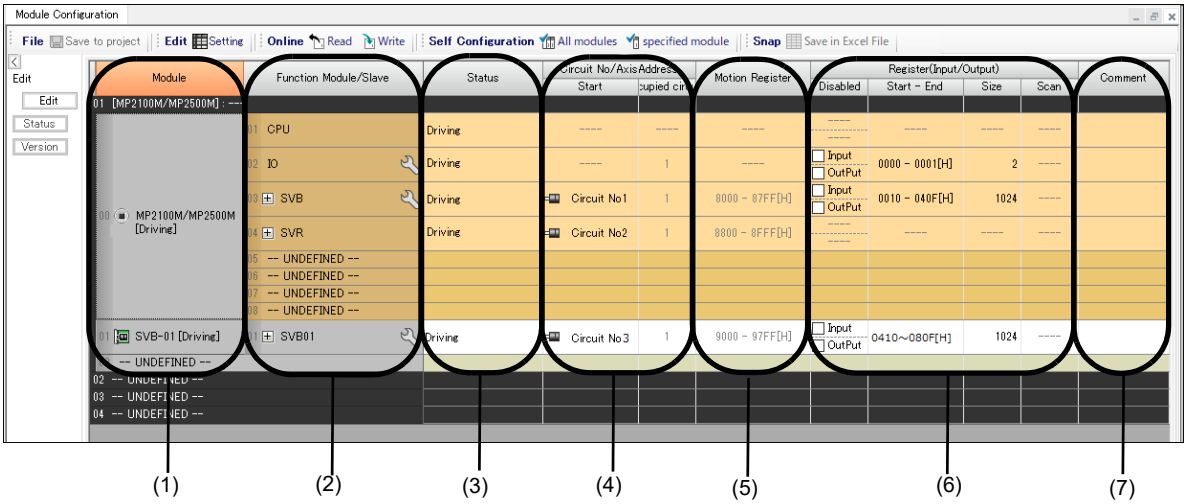
This switch can be used to turn ON and OFF the MP2100M power without turning the host computer ON and OFF.

2.2.3 Module Configuration Definitions

Configuration of the MP2100M including Optional Modules can be obtained and modified in the Module Configuration Definition Window.

(1) Module Configuration Definition Window Components

A typical MP2100M Module Configuration Definition Window is shown below.



Note: For information on how to open the Module Configuration Definition Window, refer to *4.2.2 (1) Setting and Saving Motion Fixed Parameters*.

The following table lists the items shown in the Module Configuration Definition Window.

No.	Item		Description
(1)	Module		Unit or module set for each slot is displayed.
(2)	Function Module/Slave		The functions that are built into the unit/module or slaves that are connected to the unit/module are displayed. Click the [ ] button to display the detail definitions window.
(3)	Status		The status of a function module and transmission status of a MECHATROLINK slave device are displayed.
(4)	Circuit No./Address	Start	The circuit number of a module is displayed.
		Occupied circuits	The number of circuits that can be used by the module is displayed.
(5)	Motion Register		The range of registers that can be used as motion parameters is displayed. You cannot change this value because it is automatically set from the circuit number.
(6)	Register (Input/Output)	Disabled	Select the relevant check box to disable inputs or outputs.
		Start - End	The range of registers used as the I/O area is displayed.
		Size	The size of the I/O area is displayed in words.
		Scan	When the I/O signal is processed is displayed.
(7)	Comment		You can set comments. <Setting Conditions> • Function module: 16 characters maximum. • MECHATROLINK slave: 32 characters maximum.

(2) Self-configuration

The self-configuration function automatically recognizes the Optional Modules mounted on the MP2100/MP2100M, creates definition files such as the module configuration definition file, MECHATROLINK detail definition file, and motion parameters. Also, module definition data can be automatically refreshed by executing the self-configuration function when starting the MP2100M or anytime thereafter.

Note: 1. For information on how to execute the self-configuration function, refer to *5.4.2 Procedure for Self-configuration Using Mode Switch* and *5.4.3 Procedure for Self-configuration Using MPE720*.
2. For details on definition information that is updated by the self-configuration function, refer to *5.4.4 Definition Data Updated by Self-configuration*.

2.2.4 CPU I/O (Built-in I/O) Module

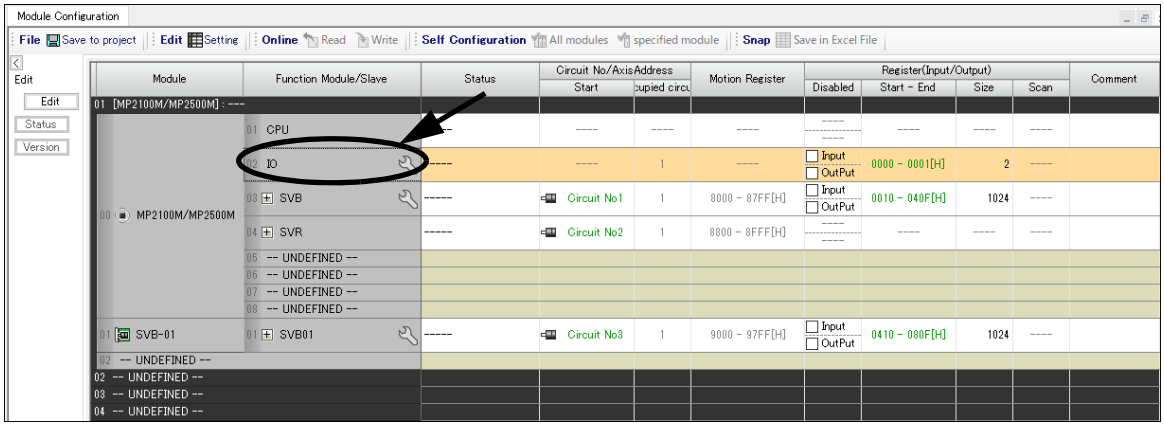
The CPU I/O Module is a digital I/O module built in the MP2100M, and provides five external input points and four external output points.

(1) CPU/IO Module Details

For CPU I/O Module details including specifications, refer to *3.2.3 I/O Connection*.

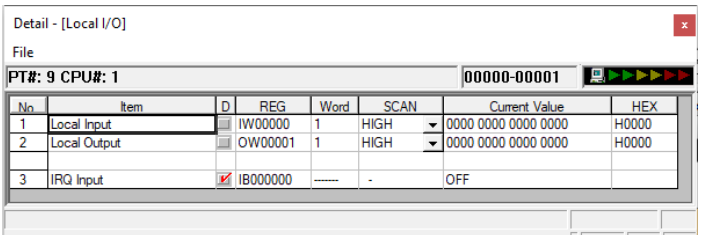
(2) Settings for CPU I/O Module (Local I/O Definition Window)

Double-click the **IO** cell on the Module Configuration Definition Window.



The message box appears and ask for confirmation to create a new file. Click the **OK** button to open the **Local I/O** window.

The following items are displayed in the **Local I/O** window. The local inputs (**Local Input**), local outputs (**Local Output**), and interrupt inputs (**IRQ Input**) can be set.



D : Enable or disable each item by clicking on the cell.

: Enabled, : Disabled

The register length is fixed at one word, i.e., 16 points are set for each input or output register.

REG : Displays the register number allocated to the inputs or outputs. It cannot be changed.

Word : Displays the word size of the register data. It cannot be changed.

SCAN : Select the speed from **HIGH**, **LOW**, or **NA** (none specified), for the scan that processes the inputs or outputs.

Current Value : The current value of the register will be displayed in binary when online. It will not be displayed when offline.

The outputs to external devices can be set by changing the current value of the local outputs. When the set value is confirmed, it is immediately saved in the register.

Other current values cannot be changed.

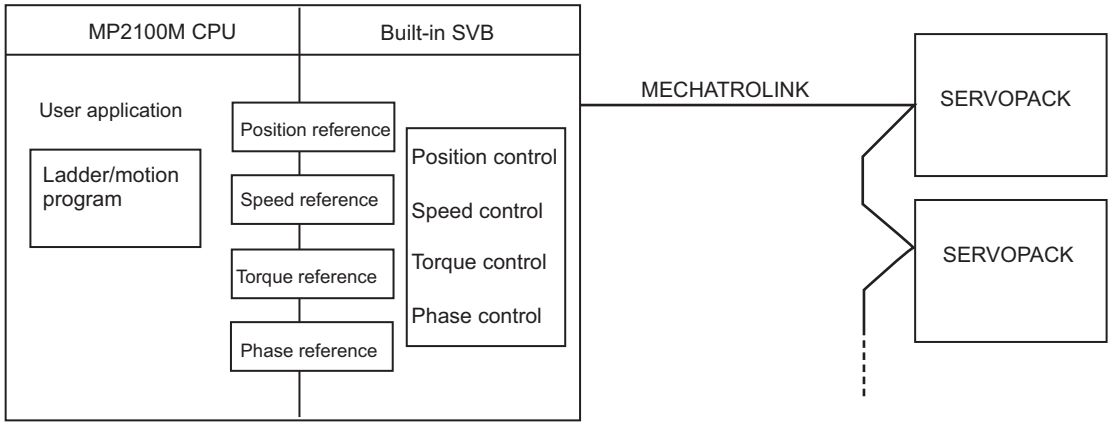
HEX : The current value of the register will be displayed in hexadecimal when online. It will not be displayed when offline.

After changing the local I/O configuration, save the definition data by selecting ***File - Save & Save into flash memory*** from the main menu.

2.2.5 Built-in SVB Modules

The SVB Module is a motion module used to control SERVOPACKs, stepping motor drivers, inverters, distributed I/O devices, etc. via MECHATROLINK interface MECHATROLINK-I/II.

The MECHATROLINK-II enables position, speed, torque, and phase control for highly accurate synchronized control. In addition, sophisticated machine operations can be performed by switching the control mode while the axis is moving.



The SVB Modules are of two types: The built-in SVB (hereinafter referred to as the Built-in SVB) and the slot-mounted optional SVB (hereinafter referred to as the Optional SVB).

The MP2100M includes a Built-in SVB Module as a standard feature.

Note: For details on the specifications of the Built-in SVB, refer to *2.2.5 (5) Specifications of Built-in SVB Modules*.

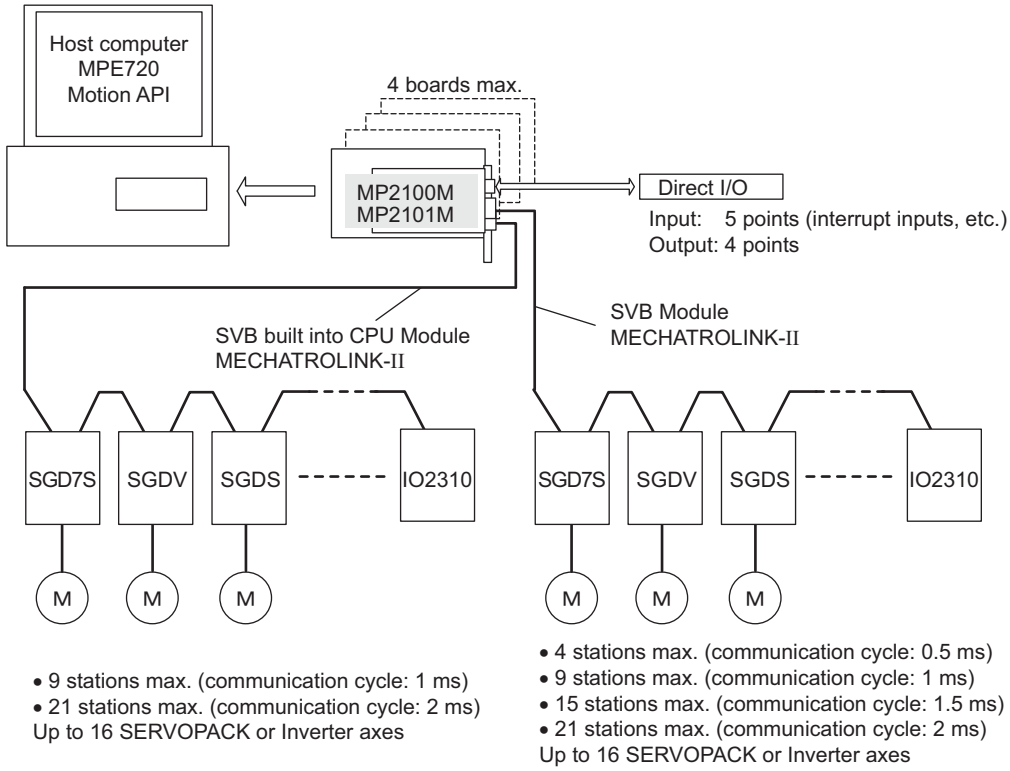
The Optional SVB is one of the Optional Modules for the MP2100/MP2100M. The SVB-01 Module is an Optional SVB that can be mounted on the MP2100.

(1) Features

- Up to 21 slave stations can be connected to a single SVB Module (Servo drives can be connected up to 16 axes).
  MP2100M: Up to two SVB-01 Modules can be mounted in optional slots.
  Including the Built-in SVB, a total of 48 axes can be controlled.
- Synchronization between Modules is also supported, making it suitable for both synchronous control and interpolation across Modules.
- An SVB-01 Module used as a slave can be connected to a host controller equipped with MECHATROLINK communication functions.
- Self-configuration enables automatic allocation of setting data for the slave device that is connected to MECHATROLINK.
- SERVOPACK parameters can be managed over networks.

(2) Example of System Configuration

The following diagram shows an example of system configuration.



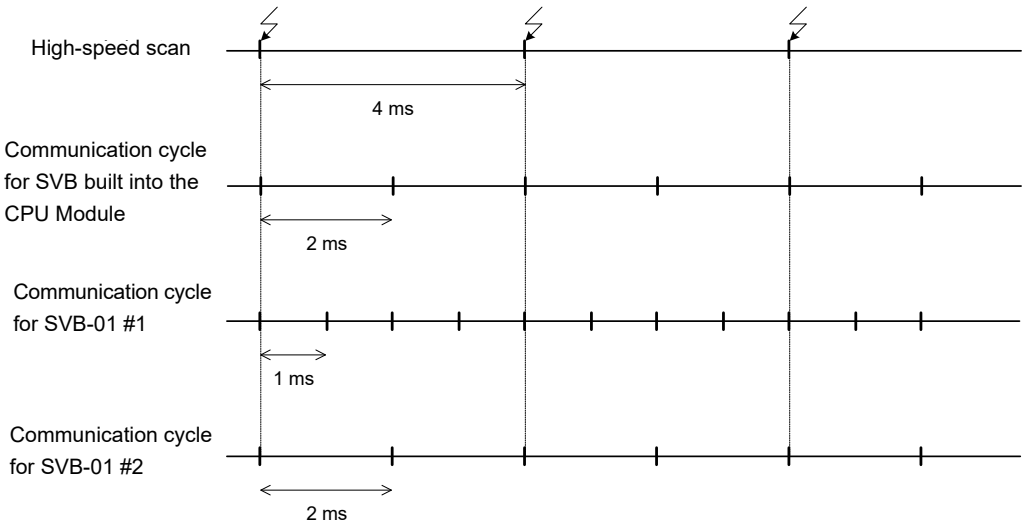
Note: 1. Use MPE720 Ver.7, Ver.6 or Ver.5 (Version 5.10 or later), and Motion API Ver. 1.05 (driver: Ver. 1.1.3.0) or later.
2. Use the connecting cables and connectors recommended by Yaskawa. Always check the device to be used and select the correct cable for the device by referring to *3.2.2 (3) Cables*.
3. Different SERVOPACKs are connected to MECHATROLINK-I and MECHATROLINK-II. Select the SERVOPACKs that can be connected by referring to *1.5 Devices Connectable to MECHATROLINK*.
4. If devices compatible with MECHATROLINK-I (4 Mbps) and with MECHATROLINK-II (10 Mbps) are used together, make the all settings for MECHATROLINK-I (4 Mbps).
5. When connecting SERVOPACKs via MECHATROLINK, connect the overtravel, zero point return deceleration limit switch, and external latch signals to the SERVOPACKs. For the details on connection, refer to the SERVOPACK's manual.

(3) Synchronization between Modules

(a) Overview

The MP2100M has a hardware-based function for synchronizing the CPU and Optional Modules.

This function makes it possible to synchronize the high-speed scan and MECHATROLINK communication, which means that synchronization between a Built-in SVB Module and an SVB-01 Module, and synchronization among multiple SVB-01 Modules, are possible.



In the synchronous mode, the start timings for the high-speed scan and each communication cycle are synchronized as shown in the figure above, so no discrepancy arises. This means that the timing in which commands in high-speed scanning are transferred to communication cycle processing is constant, and the distribution processing for interpolation commands is simplified.

(b) Conditions Under which Synchronization Is Possible

When the combination of the high-speed scan time and the MECHATROLINK communication cycle is one of those indicated by "✓" in the table below, synchronization is established among the modules.

High-speed Scan (RTC: 0.5 ms)	MECHATROLINK Communication Cycle			
	0.5 ms	1 ms	1.5 ms	2 ms
1.0 ms	✓	✓	–	✓
1.5 ms	✓	–	✓	–
2.0 ms	✓	✓	–	✓
2.5 ms	✓	–	–	–
3.0 ms	✓	✓	✓	–
3.5 ms	✓	–	–	–
4.0 ms	✓	✓	–	✓
4.5 ms	✓	–	✓	–
5.0 ms	✓	✓	–	–
5.5 ms	✓	–	–	–
6.0 ms	✓	✓	✓	✓
:				

(c) Timing at which Modules are Synchronized

Modules are automatically synchronized when the power supply is turned OFF and ON again.

(d) Operation when High-speed Scan Cycle is Changed

If the high-speed scan is changed, SVB Module MECHATROLINK communication is continued, but the speed waveform is disrupted when interpolation commands are issued. Change the high-speed scan cycle while the CPU is in the STOP status or while no motion commands are being executed.

After changing the high-speed scan settings and switching operation from synchronous to asynchronous or from asynchronous to synchronous, save the settings in the flash memory and then turn the power OFF and back ON.

(e) Operation when the MECHATROLINK Communication Cycle is Changed

- When the MECHATROLINK communication cycle of a Built-in SVB Module is changed

Even if the combination of high-speed scan setting and communication cycle setting is one that permits synchronization, there is a risk that asynchrony will result from the change operation. After making a change, save the settings in the flash memory and then turn the power OFF and back ON.

- When the MECHATROLINK communication cycle of an SVB-01 Module is changed

If the combination of high-speed scan setting and communication cycle setting is one that permits synchronization, synchronization is performed automatically. In this case, the power does not have to be turned OFF and then back ON.

(f) Conditions when the Power Supply must be Turned OFF and ON

When any of the following operations is performed, save the settings to flash memory and then turn the power supply OFF and ON.

- When executing a self-configuration command from the MPE720 after turning ON the power supply
- When loading a Module definition after turning ON the power supply
- When the communication cycle of the Built-in SVB Module is changed after turning ON the power supply
- When operation changes from synchronous to asynchronous or from asynchronous to synchronous as a result of changing the high-speed scan setting

(4) Setting SVB Module

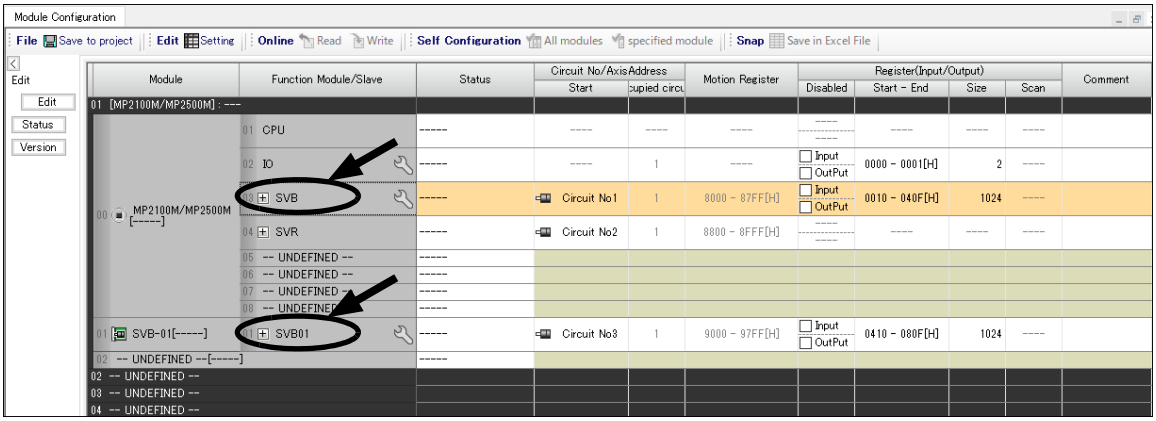
A SERVOPACK connected to MECHATROLINK can be controlled by setting MECHATROLINK detail definitions with the MPE720 Module Configuration Definition Window.

(a) MECHATROLINK Detail Definition

- How to Open the MECHATROLINK Detail Definition Window

To make or confirm settings for the built-in SVB Module, double-click the **SVB** cell on the Module Configuration Definition Window to display MECHATROLINK detail definition window.

To make or confirm settings for the SVB-01 Module, double-click the **SVB01** cell on the Module Configuration Definition Window to display MECHATROLINK detail definition window.



- The MECHATROLINK Detail Definition Window

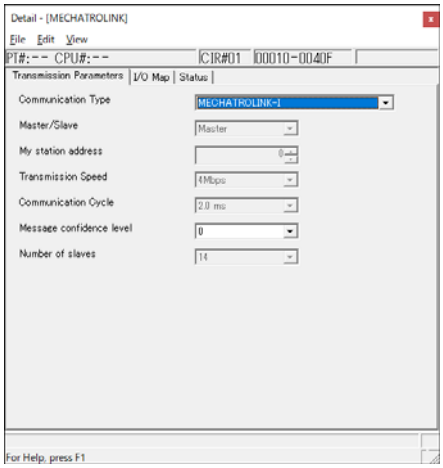
The **MECHATROLINK Detail Definition** window has three tabs: **Transmission Parameters**, **I/O Map**, and **Status**. Click the tab to view the contents.

1. Transmission Parameters Tab

   The parameters required to use the MECHATROLINK transmission system are displayed.

<Communication Type: MECHATROLINK-II>

<Communication Type: MECHATROLINK-I>



The items shown on the **Transmission Parameters** tab are as follows. For editable items, the settings can be changed. Always save the settings to the flash memory after changing them.

Item	Display during Self-configuration	Options and Precautions on Settings
Communication Type	Displays the detected communication type.	Select from among “MECHATROLINK-II (32 Byte Mode),” “MECHATROLINK-II (17 Byte Mode)” and “MECHATROLINK-I.”
Master/Slave	Displays whether the selected SVB Module is used as a master station or slave station.	Select either “Master” or “Slave.” For Built-in SVB Modules (slot “00”), this is fixed as “Master.”
My station address (Local station address)	Displays the local station address set by using the rotary switches.	For master station, fixed to 0. For a slave station you make any setting in the range from 1 to the number of slave stations.
Transmission Speed	Displays the transmission speed for communication. The indication will be as follows depending on the communication type. MECHATROLINK-II (32 Byte Mode): 10 Mbps MECHATROLINK-II (17 Byte Mode): 10 Mbps MECHATROLINK-I: 4 Mbps	Cannot be set.
Transmission Byte (Not displayed for MECHATROLINK-I)	Displays the number of transmission bytes. The indication differs depending on the communication type and the master/slave distinction. For details, refer to *The number of transmission bytes/communication cycle/ number of retry stations/number of slave stations* on the next page.	Cannot be set.

(cont'd)

Item	Display during Self-configuration	Options and Precautions on Settings
Communication Cycle	Displays the communication cycle. The indication differs depending on the communication type and the master/slave distinction, and so on. For details, refer to *The number of transmission bytes/communication cycle/number of retry stations/number of slave stations* below.	This can only be set when the communication type is MECHATROLINK-II and the Master/Slave selection is "Master." The values that can be set differ for Built-in SVB Modules and Optional SVB Modules.
Message confidence level (Not displayed for MECHATROLINK-II)	This is a setting that is not used with MECHATROLINK transmission.	Use the default value (0).
SigmaWin (Not displayed for MECHATROLINK-I)	Displays whether or not SigmaWin+ can be used via a MECHATROLINK-II adapter such as JUSP-NP115 in MECHATROLINK communications.	Select either "Used" or "Not Used."
Number of retry to slaves (Not displayed for MECHATROLINK-I)	Displays the maximum number of stations to which transmissions (retries) are sent in one transmission cycle when a normal response has not been received from a slave station.	A value in the range 0 to 7 can be set, for "Master" stations only. No setting can be made for "Slave" stations.
Number of slaves	Displays the number of slave stations that can be connected. The indication differs depending on the communication type, communication cycle, whether or not SigmaWin+ is used, and the number of retry stations.	Cannot be set.
Slave synchronous function	Enable or disable the slave synchronous function.	Cannot be set.
Wait for monitoring information to be refreshed (Not displayed with Built-in SVB Modules)	Displays whether CPU processing waits for Optional SVB Module monitor parameter scan delays (Enable) or not (Disable).	Select either "Enable" or "Disable." For details of the processing, refer to *Waiting for monitor information to be refreshed* on the next page.

- The number of transmission bytes/communication cycle/number of retry stations/number of slave stations

The number of transmitted bytes, communication cycle, number of retry stations and number of slave stations when self-configuration is executed are automatically set based on conditions including the communication type, Master/Slave status, and highest slave station number (the highest among the detected slave station numbers).

<With "Master" Setting>

Item	MECHATROLINK-II (32-Byte Mode)				MECHATROLINK-II (17-Byte Mode)		MECHATROLINK-I
Maximum slave station number	1 to 8	9	10 to 16	17 to 21	1 to 14	15	
Transmission bytes	31 Bytes				16 Bytes		–
Communication cycle	1 ms	1 ms	2 ms	2 ms	1 ms	1 ms	2 ms
Number of retry stations	1	0	5	21 - highest station number	1	0	14
Number of slaves	8	9	16	Highest station number	14	15	14

<With "Slave" Setting>

Item	MECHATROLINK-II (32-Byte Mode)	MECHATROLINK-II (17 Byte Mode)	MECHATROLINK-I
Transmission bytes	–	–	–
Communication cycle	1 ms	1 ms	2 ms
Number of retry stations	30	30	15
Number of slaves	30	30	15

- Settable communication cycles

The values that can be set for the communication cycle are as follows depending on whether dealing with a Built-in SVB Module or an Optional SVB Module, and the communication type.

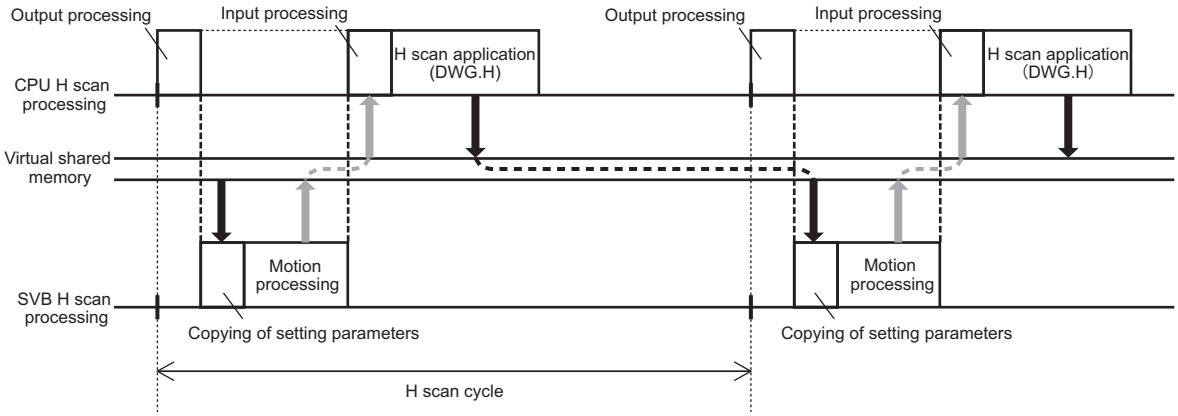
Built-in/Optional	Built-in SVB Module		Optional SVB Module	
MECHATROLINK-II communication type	32-byte mode	17-byte mode	32-byte mode	17-byte mode
Settable communication cycles	1 ms/1.5 ms/2 ms	Fixed at 1 ms	0.5 ms/1 ms/1.5 ms/2 ms	0.5 ms/1 ms

Note: 1. Setting is only possible when "Master" is selected.
2. With MECHATROLINK-I, the value is fixed as "2 ms".

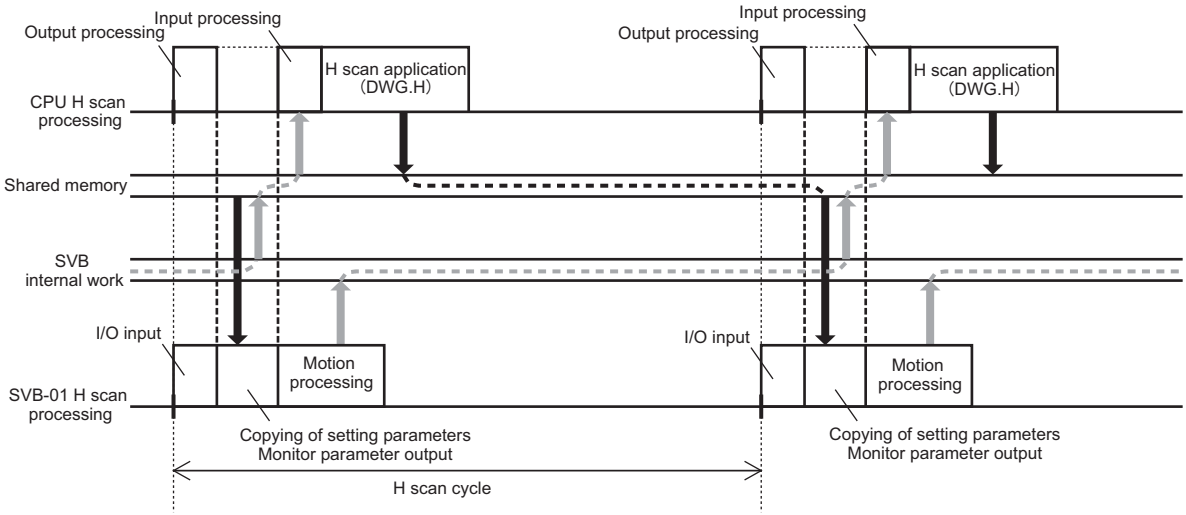
- Waiting for monitor information to be refreshed

An Optional SVB Module (SVB-01) communicates data with the CPU of the Machine Controller using actual shared memory, but the time lapse until it becomes possible to monitor the motion parameters created at the SVB-01 Module with the CPU's application is one scan longer than if the same processing were done with a Built-in SVB Module (see the figures below).

<Progress of Processing at a Built-in SVB Module>



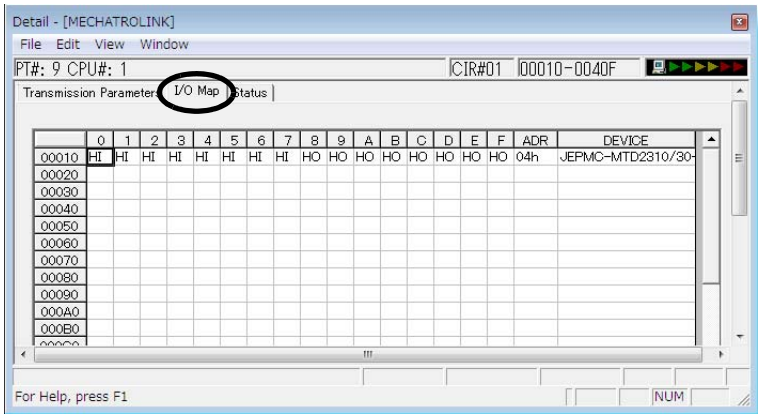
<Progress of Processing at an Optional SVB Module>



When the "wait for monitoring information to be refreshed" mode is enabled, this delay is eliminated and monitoring of motion parameters with the same timing as a Built-in SVB Module is enabled. However, since the start of execution of the CPU's application is delayed until completion of the SVB-01 Module's motion processing, the CPU's high-speed scan processing time is lengthened.

2. I/O Map Tab

The status allocated to I/O registers is displayed.



<Displayed Meaning>

HI: High-speed scan input

HO: High-speed scan output

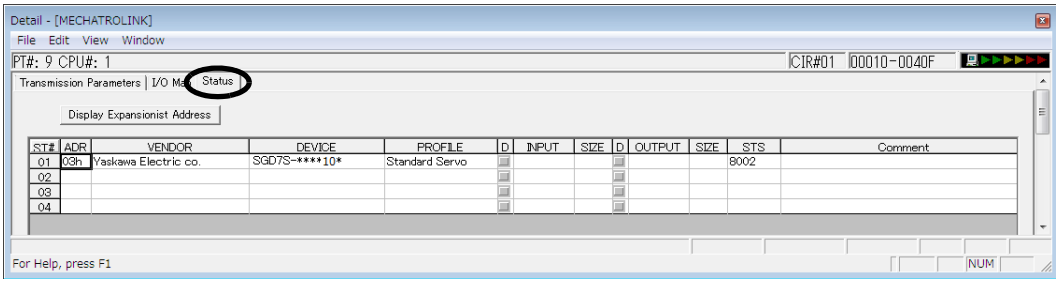
LI: Low-speed scan input

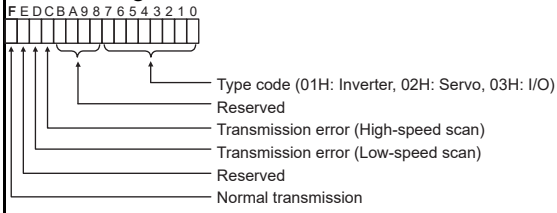
LO: Low-speed scan output

Note: The I/O Map tab is used for monitoring (read-only). Do not change the displayed settings.

3. Status Tab

The MECHATROLINK transmission status is displayed. The displayed settings cannot be changed.



Item	Description
ST #	The station number. A number of lines corresponding to the number of slave stations set on the parameter setting window is displayed.
AD	Displays the station address of a slave station. When the local station is set as a slave station, the address specified on the parameter setting window is displayed.
ExADR	This is displayed on clicking the **Display Expansionist Address** button, and hidden on clicking the **Hide Extended Address** button. When multi-station modules where a number of stations are grouped together as a single node are used, the individual extended addresses are set in this field.
VENDOR	Displays the vendor name of the device.
DEVICE	Displays the slave model.
PROFILE	Display the profile to be used.
INPUT	Displays the leading register number of the input area. When **Standard Servo** is set for **PROFILE**, this is invalid.
OUTPUT	Displays the leading register number of the output area.
D	I/O register's enable/disable status  : Enabled  : Disabled
SIZE	Displays the input/output size in word units. When **Standard Servo** is set for **PROFILE**, this is invalid.
STS	In online mode, MECHATROLINK transmission status information is displayed in hexadecimal. The meaning of each bit is shown below. 
Comment	Displays a comment of your choice.

(b) Motion Parameters

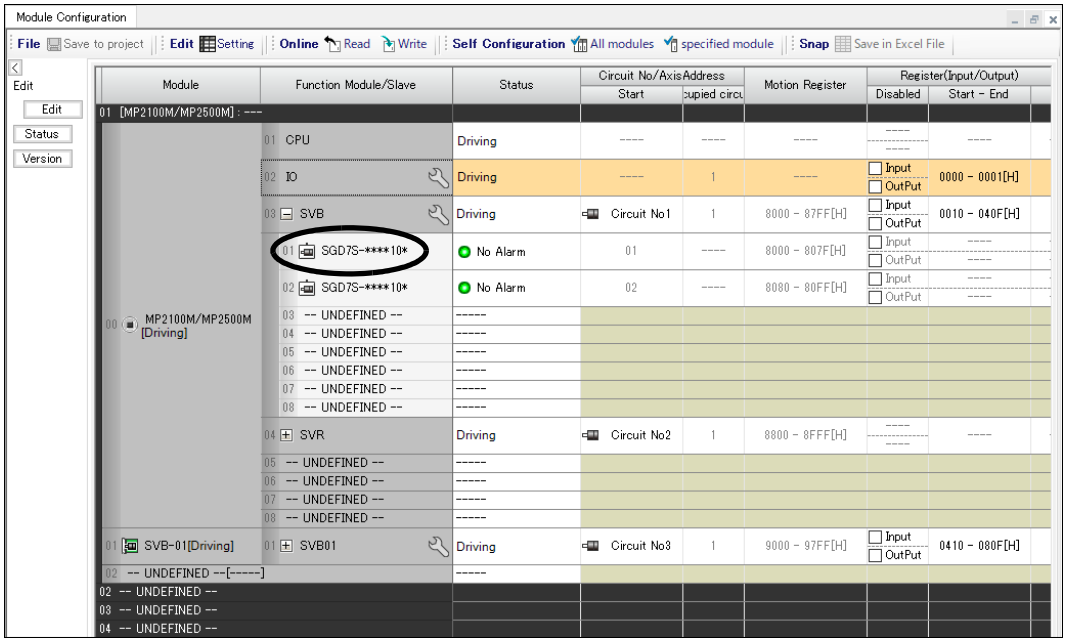
The motion parameters are parameters that control motion axes (e.g., SERVOPACKs, inverters, and steppers), and "motion parameter" is used as the general term for fixed parameters, setting parameters, and monitor parameters

Note: Refer to *Appendix B.1 Motion Parameter Lists* for details on motion parameters.

- Display the Motion Parameter Window

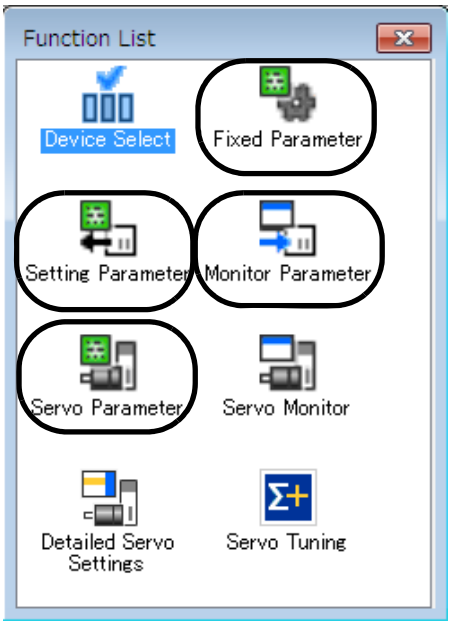
Open the motion parameters window by the following procedure.

1. On the Module Configuration Definition Window, double-click the cell of the SERVOPACK for which you want to display the motion parameters window. (refer to *2.2.3 (1) Module Configuration Definition Window Components* on page 2-13).



The **Function List** window opens.

2. Click the one of the motion parameters icons ("Fixed Parameter", "Setting Parameter", "Servo Parameter", or "Monitor Parameter") to display the relevant settings tab page.



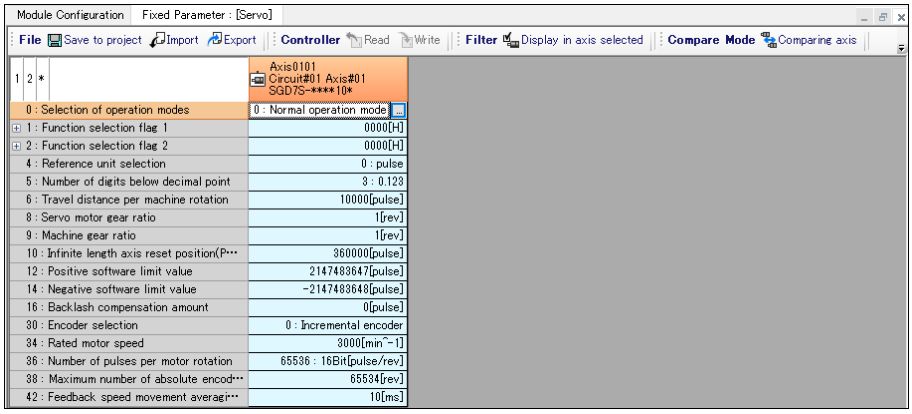


Fig. 2.1 Fixed Parameter Tab Page

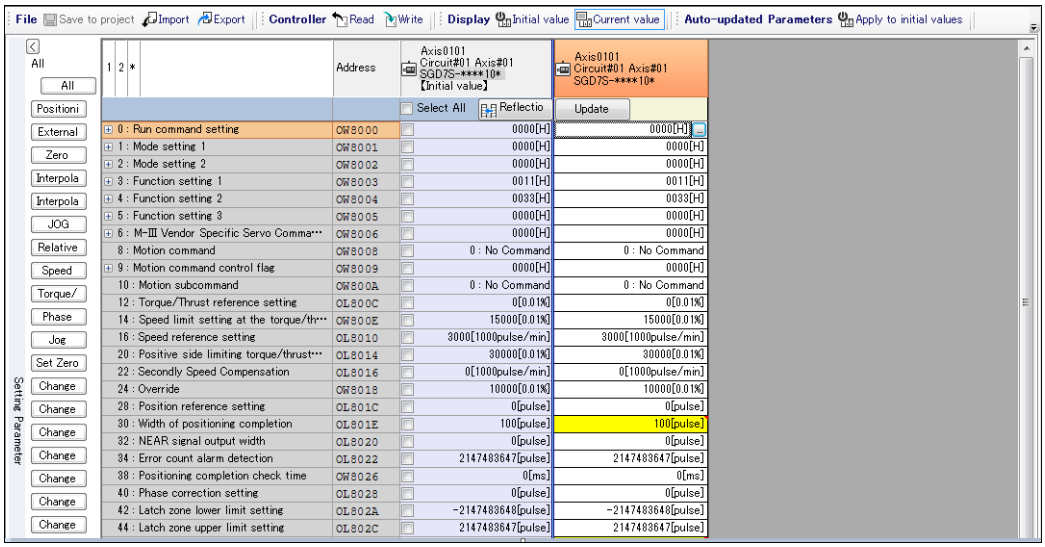


Fig. 2.2 Setting Parameter Tab Page

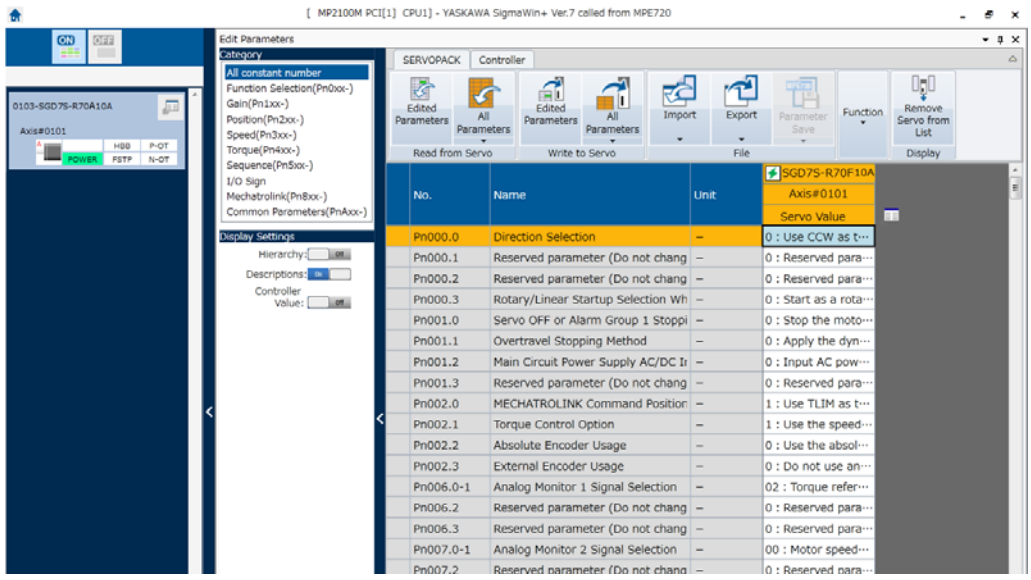


Fig. 2.3 Servo Parameter Window

Note: Refer to the relevant SERVOPACK user's manual for information on SERVOPACK parameters.

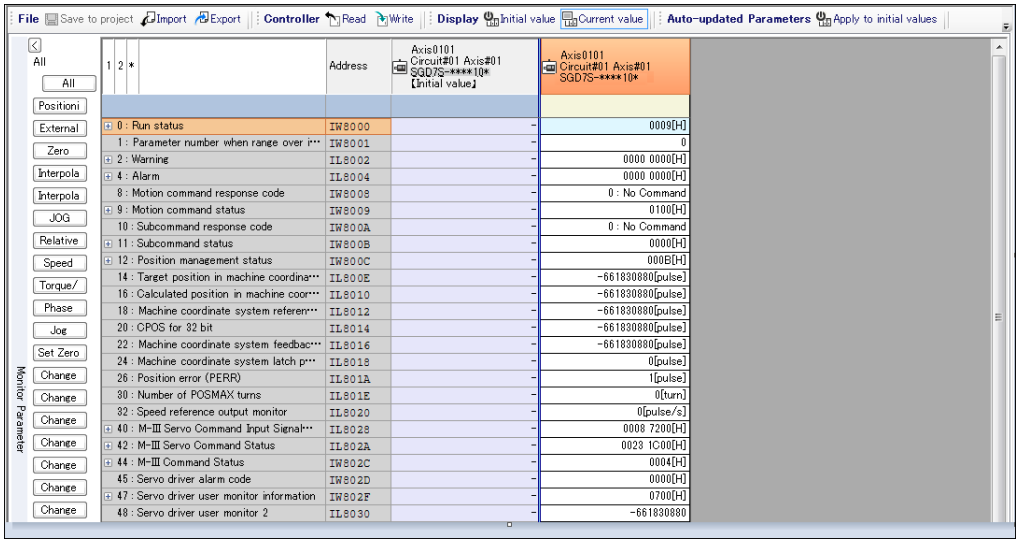


Fig. 2.4 Monitor Parameter Tab Page (read-only)

(5) Specifications of Built-in SVB Modules

(a) MECHATROLINK Communication Specifications

Item	MECHATROLINK-I Specifications	MECHATROLINK-II Specifications
Transmission path type	Bus type	Bus type
Transmission path	Electric bus	Electric bus
Transmission distance	50 m	50 m (When using repeaters*: Maximum extension of 100 m possible)
Transmission speed	4 Mbps	10 Mbps
Communication cycle	2 ms	1 ms, 2 ms
Maximum number of connected stations	14	21
Transmission control method	Cyclic method	Cyclic method
Access control method	1: N	1: N (When SigmaWin+ is used, 2: N)
Transmission mode	Control communication	Control communication
Error control	CRC check	CRC check

* Compatible repeater: JEPMC-REP2000

(b) MECHATROLINK Transmission Settings and Maximum Number of Slave Stations

MECHATROLINK Transmission Settings			Maximum Number of Slave Stations
Communication Type	Transmission Speed	Communication Cycle	
MECHATROLINK-I	4 Mbps	2 ms	14
MECHATROLINK-II (17-byte mode)	10 Mbps	1 ms	15
MECHATROLINK-II (32-byte mode)	10 Mbps	1 ms	9
		1.5 ms	15
		2 ms	21 (maximum of 16 axes for SERVOPACKs)

Note: For details on how to set the MECHATROLINK transmission settings, refer to *8.8.6 MECHATROLINK Definitions* of the *Machine Controller MP900/MP2000 Series MPE720 Software for Programming Device User's Manual* (Manual No.: SIEP C880700 05).

(c) Transmission Distance and Maximum Number of Slave Stations

Communication Type	Transmission Distance (Total Network Length)	Maximum Number of Slave Stations
MECHATROLINK-I	50 m	14
MECHATROLINK-II	30 m (When using repeaters: Maximum extension of 100 m possible)	16 (21)*
	50 m (When using repeaters: Maximum extension of 100 m possible)	15 (21)*

* The figures in parentheses apply when repeaters (JEPMC-REP2000) are used. In MECHATROLINK-II communications, repeaters (JEPMC-REP2000) are required if the number of slave stations is 17 or greater.

2.2.6 Built-in SVR Virtual Motion Module

(1) Overview

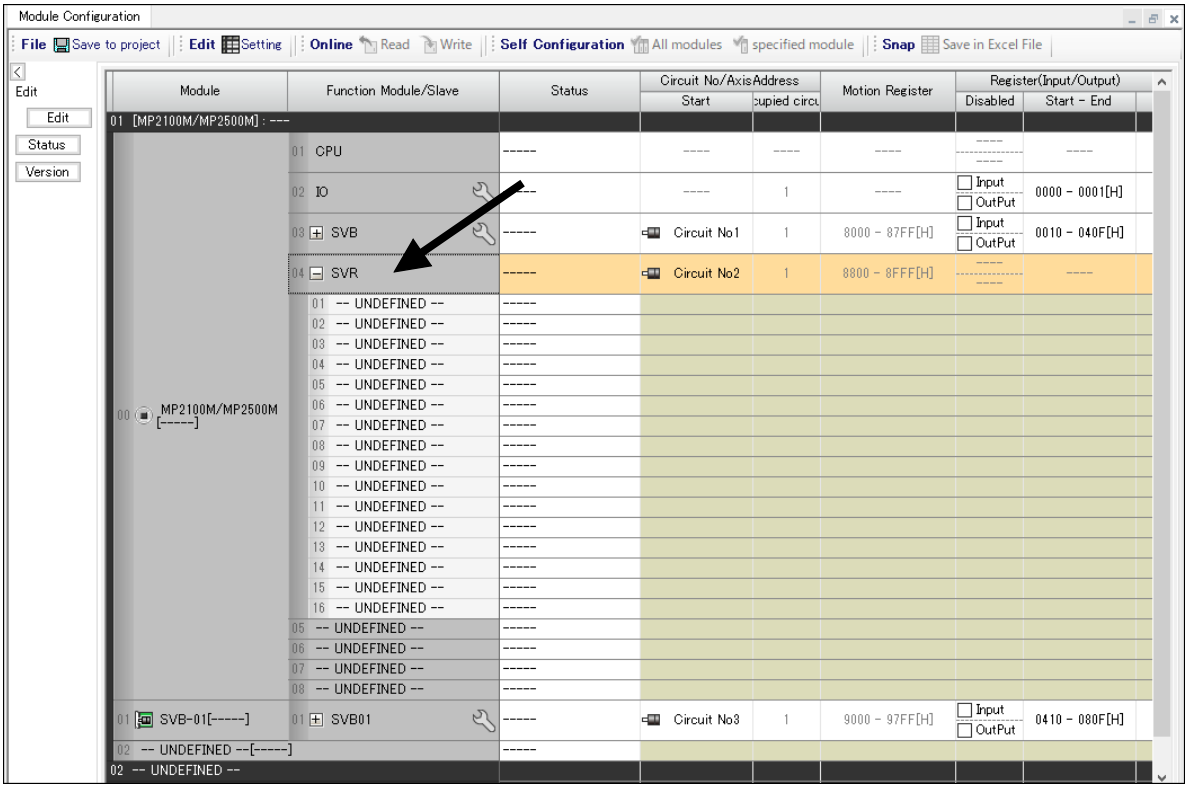
The Virtual Motion Module (SVR) is a Software Module that provides an interface for virtual axes that are not actually connected to Servomotors. The SVR is standard feature with the MP2100M.

The SVR is configured in the same way as the Built-in SVB Module with fixed parameters, setting parameters, and monitoring parameters, and can be accessed from application programs using I/O registers.

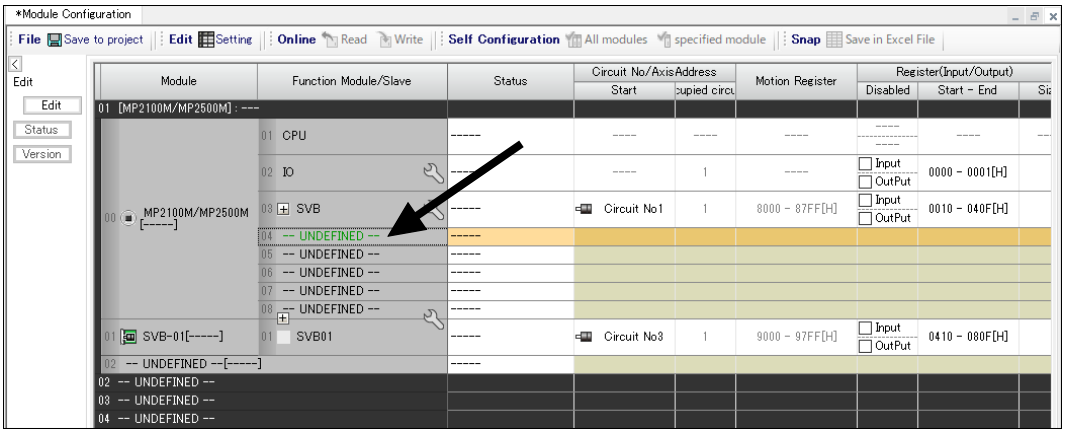
Note: For details on the motion parameters and motion commands of the SVR, refer to *Machine Controller MP2000 Series Built-in SVB/SVB-01 Motion Module User's Manual* (Manual No.: SIEP C880700 33).

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

With the MP2100M, in the Module Configuration Definition Window, the fourth slot is the SVR.



Note: If the SVR is not used, the MP2100 processing time can be reduced by deleting **SVR** in the Module Configuration Definition Window.



(2) Example of Use of the SVR

The SVR is usually used for the following two applications.

- Program testing: Results can be obtained simply, without having to install motors.
- Generating commands: When the SVR is used in situations where a Motion Module just for generating commands is required, such as where the master axis is in phase control or synchronized control of multiple axes, Motion Modules associated with actual axes are not needed.

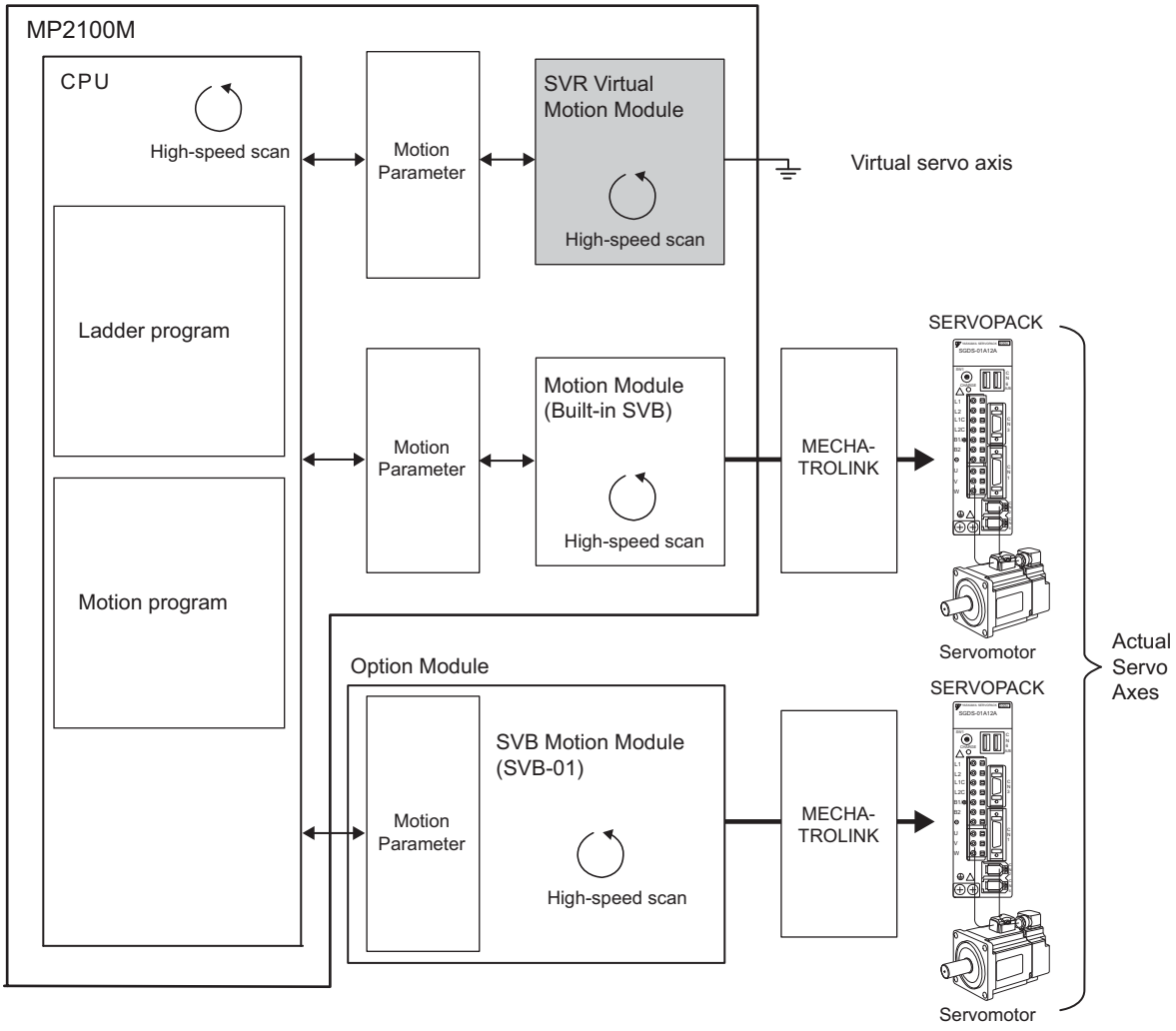
The following table lists example applications of the SVR.

No.	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 operates with a sine curve command.

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

(3) Example of System Configuration

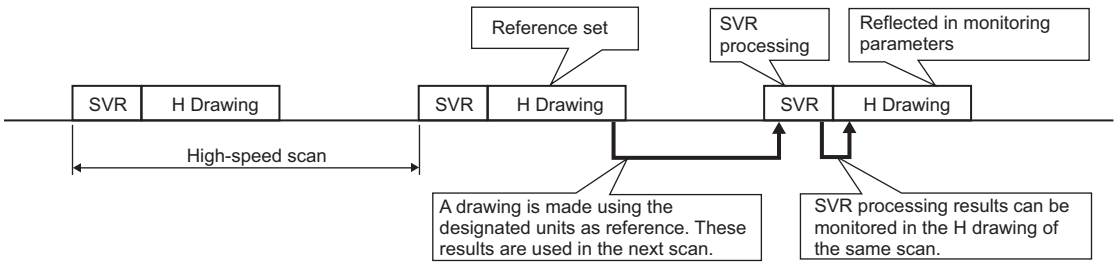
The following diagram shows an example of system configuration in which SVR is used.



(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 the processing for the drawing has been completed, and the processing results are reflected in the monitoring parameters.



(b) Processing Time

When fixed parameter 0 (Run Mode) is set to 0 (Normal Running), service on each of the SVR Module's virtual axes (total of 16 axes) is started.

Note: The default setting for the Run Mode is "1" (axes not used).

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

Command	MP2100
NOP	35 + 14 × number of axes (μs)
POSING	35 + 36 × number of axes (μs)

Note: Number of axes:
This is the number of axes (1 to 16) when Normal Running (0) is set for Set Run Mode (fixed parameter 0).
The formulae above do not apply when the number of axes is 0.

INFO

■ Comparison with SVB Simulation Mode

In the simulation mode, no positioning function is available, so the position data is updated to the final target position for one scan. In contrast, the SVR uses its positioning function to perform distribution and therefore, in the same way as an actual module, it refreshes the position data for every scan up to the final target position.

2.2.7 BUS Interface Module (BUSIF)

(1) Overview

The BUS Interface Module (BUSIF) is a virtual module that provides data exchange between the user application program at the host computer and the application program in the MP2100/MP2100M through a BUS interface. The BUSIF module is a standard feature with the MP2100/MP2100M. The BUSIF Module can be used with the following software versions.

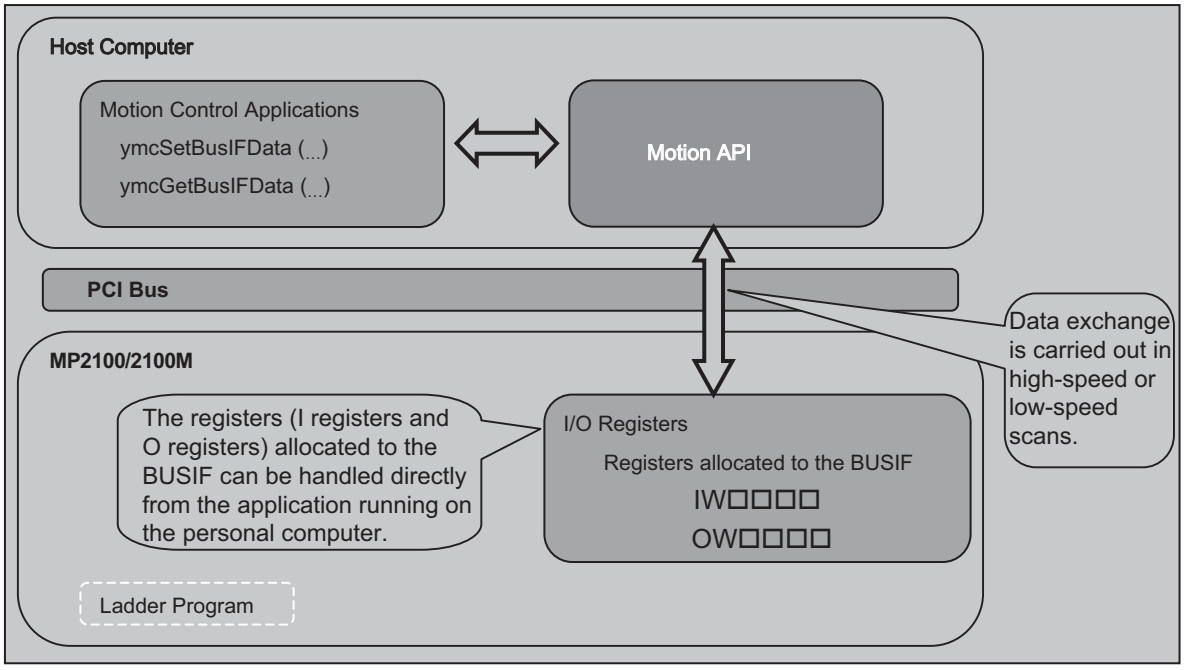
Software	Version
MP2100/MP2100M	Ver. 2.46 or later
MPE720 Ver.5	Ver. 5.34 or later
MPE720 Ver.6	All versions
MPE720 Ver.7	All versions
Motion API (PCAPI)	Ver. 1.20 or later

Data between the user application program at the host computer and the application program in the MP2100/MP2100M can be exchanged without using the BUSIF, but the BUSIF enables high-speed access to registers from the host computer.

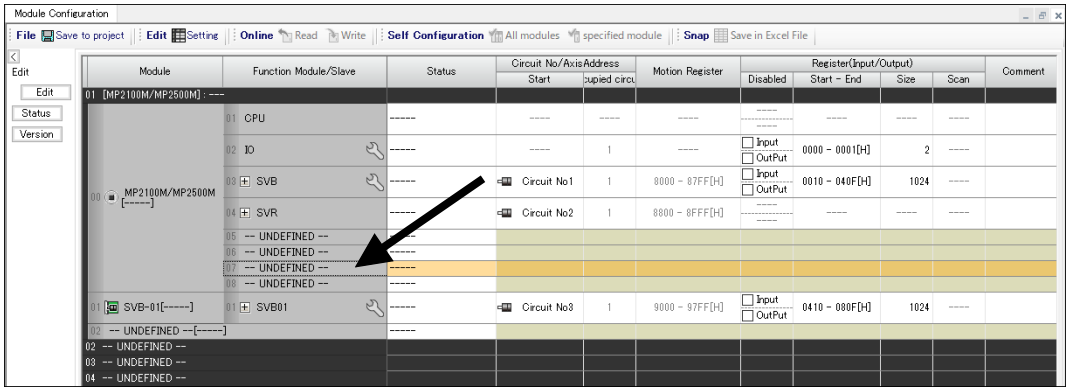
Data exchange by means of the BUS interface uses the I registers and O registers allocated to the BUSIF Module. The I registers and O registers allocated to the BUSIF Module can be accessed from the application program in the MP2100/MP2100M. These I registers and O registers can be accessed from the application program at the host computer by using API functions* for the BUS interface.

* Two API functions are available for the BUS interface: ymcSetBusIFData() and ymcGetBusIFData(). If these API functions are not used for the BUS interface, the registers can be accessed by using ymcSetRegister() and ymcGetRegister().
Refer to the help file that accompanies the Motion API Package "MotionAPI(PCAPI)_J.chm/MotionAPI(PCAPI)_E.chm" as the reference for the API.

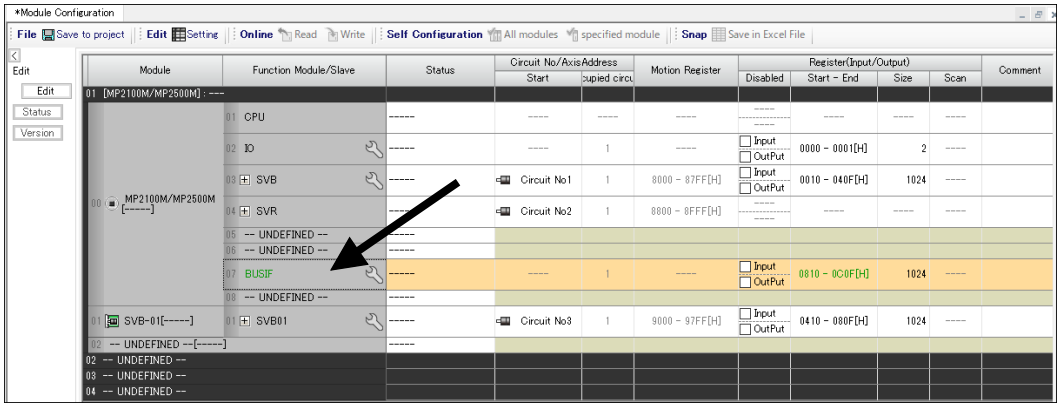
Two timings are available for exchanging data: high-speed scanning synchronization and low-speed scanning synchronization. Either of these can be selected. The following diagram shows the BUSIF in schematic form.



With the MP2100/MP2100M, in the Module Configuration Definition Window, the seventh slot is the BUSIF. The default setting for the BUSIF Module is **UNDEFINED**.

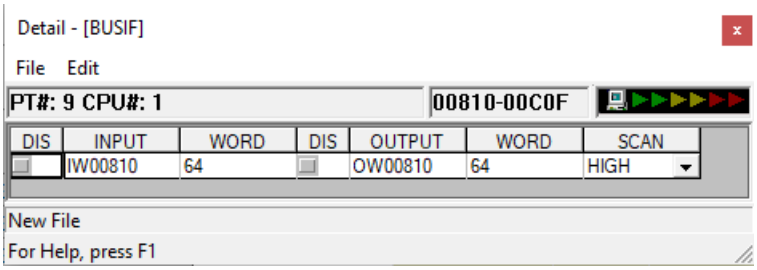


Assign **BUSIF** only when you are going to use the BUSIF. Using the BUSIF Module increases the processing time of the MP2100/MP2100M.



(2) Details of BUSIF Definition

The following items are displayed in the **BUSIF** window, and the I/O registers, I/O size, and the I/O refresh timing can be set.



Item	Contents
DIS	Enable or disable each item by clicking on the cell. : Enabled, : Disabled
INPUT	Displays the register number allocated to the input registers. The register numbers can be changed.
OUTPUT	Displays the register number allocated to the output registers. The register numbers can be changed.
WORD	Displays the word size of the register data. It can be set in the range from 1 to 1024 words. The default size is 64 words.
SCAN	Set the HIGH or LOW. The default is HIGH. When HIGH is set, data is updated in synchronization with the high-speed scan. When LOW is set, data is updated in synchronization with the low-speed scan.

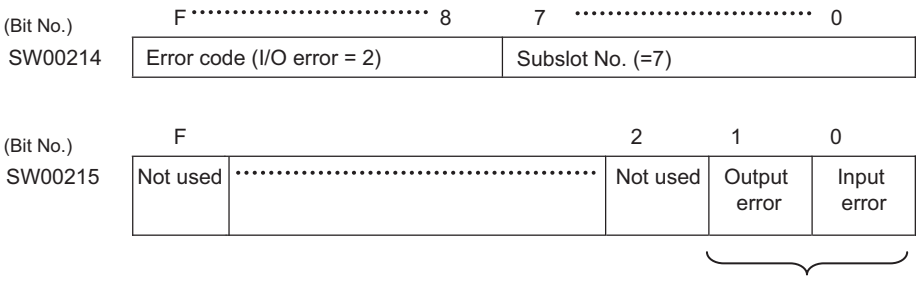
(3) Errors of the BUSIF Module

The BUSIF Module carries out exclusive access control with the API for the BUS interface at the host computer.

While the API for the BUS interface is being processed in the application program at the host computer, the I/O service of the MP2100/2100M's BUSIF Module is put on hold.

If I/O service cannot be accomplished within the stipulated time, the BUSIF Module detects a timeout error and outputs the error status to the system registers as shown in the table below.

BUSIF Module Error Status



Bit	Meaning
Bit 0	0 : Normal, 1 : Input timeout error
Bit 1	0 : Normal, 1 : Output timeout error

IMPORTANT

When an I/O error occurs at the BUSIF Module, make one of the following corrections.

- Reduce the BUSIF Module's number of I/O words. 64 words or fewer is recommended.
- Set a scan time setting that is substantially bigger than the maximum value for the scan time.
- Reduce the I/O size that is read or written at one time with the API for the BUS interface.
- If the API for the BUS interface is called repeatedly, reduce the frequency of API calls.

2.2.8 M-EXECUTOR Module (Motion Program Executor)

This section gives an overview of the functions of the M-EXECUTOR Module (Motion Program Executor) and explains the detail window.

The M-EXECUTOR Module can be used with the following software versions.

Software	Version
MP2100/MP2100M	Ver. 2.66
MPE720 Ver.5	Ver. 5.44
MPE720 Ver.6	Ver. 6.10
MPE720 Ver.7	All versions

(1) Function Overview of the M-EXECUTOR Module

- Executing a motion program without using a ladder program

To execute a motion program, an MSEE command is usually needed in a ladder program. The M-EXECUTOR Module allows the motion program to be executed without using an MSEE command into the ladder program.

Note: An MSEE command can be used in the ladder program as usual.

- Controlling a motion program without using a ladder program

Any register can be mapped to the control signal of the motion program registered in the M-EXECUTOR Module. So, without a ladder program, this allows direct control of a motion program from a host PLC or other device.

- Describing sequence control in motion language

Sequence programming has been added as a new programming method.

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.

The sequence program can be used as an alternative to the ladder program.

For information about commands available in the sequence program, refer to *Machine Controller MP2000 Series Users Manual Motion Programming* (Manual No.: SIEP C880700 38).

(2) Specifications of the M-EXECUTOR Module

(a) Programs Registerable in M-EXECUTOR

The following table shows the programs that can be registered 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	Up to 16 programs in total
	L Scan	16	Up to 16 programs in total

(b) Program Control Method

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

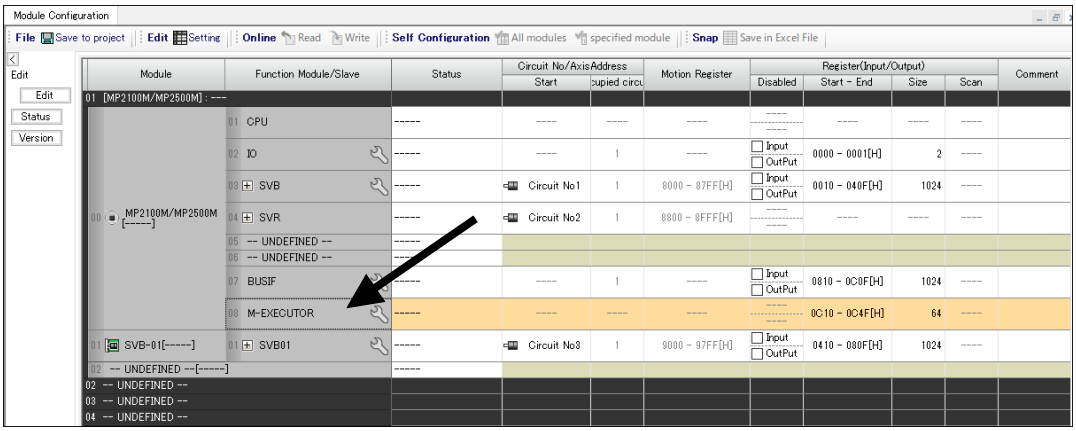
Item	Motion Program	Sequence Program
Execution Method	Sequential Execution	Startup: Event driven H Scan: Scan execution L Scan: Scan execution
System Work	1:1 correspondence between the definition number and system work (The number of program definitions is set in the **MPE720** window.) Definition No.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

(3) Module Configuration Definition

(a) Details of Module Configuration Definition Window

The M-EXECUTOR module is the No. 8 cell in the **Function Module/Slave** area on the Module Configuration Definition Window.

To check or set the M-EXECUTOR module, double-click the **M-EXECUTOR** cell on the Module Configuration Definition Window to display the M-EXECUTOR detail definition window.



For details on the module configuration definitions, refer to *2.2.3 Module Configuration Definitions*.

- 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

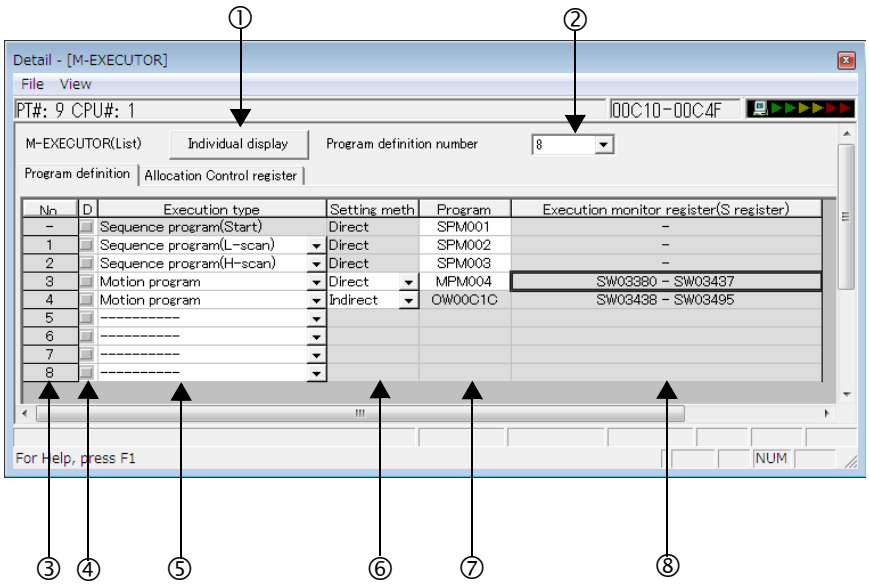
M-EXECUTOR Output Register	Item	
Ow□□□□ + 0	Definition No.1	Program number
Ow□□□□ + 1		Control signal
Ow□□□□ + 2		Override
Ow□□□□ + 3		Spare
Ow□□□□ + 4	Definition No.2	Program number
Ow□□□□ + 5		Control signal
Ow□□□□ + 6		Override
Ow□□□□ + 7		Spare
	⋮	
Ow□□□□ + 3C	Definition No.16	Program number
Ow□□□□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

(4) Detailed Window

This section describes the M-EXECUTOR detail window.

- Program definition tab (M-EXECUTOR (list display) window)

The **Program definition** tab page registers a motion or sequence program to be executed.
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

Shows M-EXECUTOR (individual display) window.

② Program definition number

Sets the number of program definitions registered in the M-EXECUTOR Module.
The valid range is 0-16 (8 by default).

③ No.

Shows the program execution order. The program will be 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 Condition
----------	None	None (select this to delete the definition)
Sequence Program (startup)	Sequence program	Power-up (during power-up, run only once)
Sequence Program (L scan)		Periodical startup (Executes each time a low-sped scan is performed)
Sequence Program (H scan)		Periodical startup (Executes each time a high-speed scan is performed)
Motion Program	Motion program	Turns ON the program operation start request of the control signal (Executes when the program operation start request is ON).

⑥ Setting

Sets the method by which a program can be selected.
The way to select a program may differ according to the program.

Designation Method	Motion Program	Sequence Program	Remarks
Direct Designation	Enabled	Enabled	Select the program by entering the program number. Example: MPM001, SPM002.
Indirect Designation	Enabled	Disabled	Select the program by entering the number of the register where the program is stored. Example: OW0C0C ("1" in register OW0C0C refers to program MPM001.)

⑦ Program

Sets a program number.

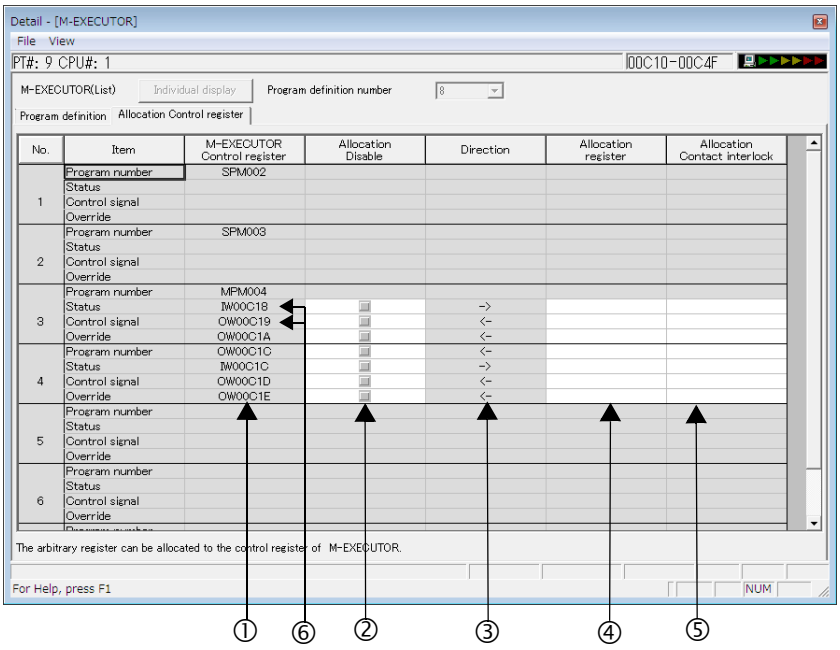
Execution Type	Remarks
Sequence Program (startup, L scan, H scan)	Enter "1" and press ENT to automatically input "SPM001." An unregistered program can be saved or this window closed without making any settings (blank). If so, the program will not be executed.
Motion Program	Direct designation: Enter "1" and press ENT to automatically input "MPM001." An unregistered program can be saved or this window closed without making any settings (blank). If so, 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 *5.2.4 (5) Monitoring Motion Program Execution Information Using S Registers*.

- Allocation Control register tab

The **Allocation Control register** tab page sets an allocation register.
A white cell can be set by the user, and a gray 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 an indirect designation has been set.
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 more information on the M-EXECUTOR control register, refer to *2.2.8 M-EXECUTOR Module (Motion Program Executor)*.

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 allocation and M-EXECUTOR control registers in real-time.
Any register can be mapped to the allocation register.

Registers that can be set as an 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)

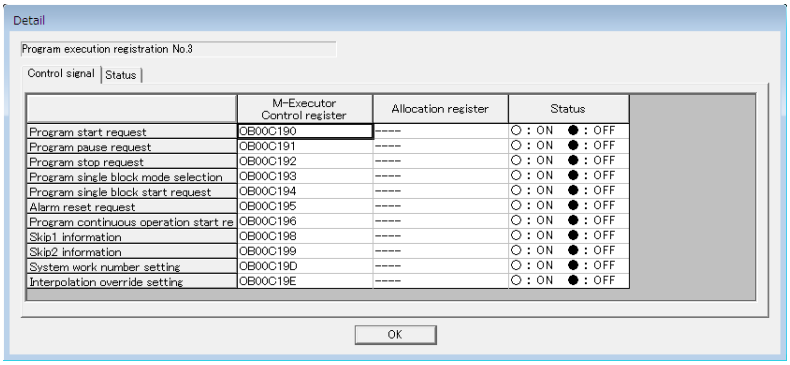
IMPORTANT

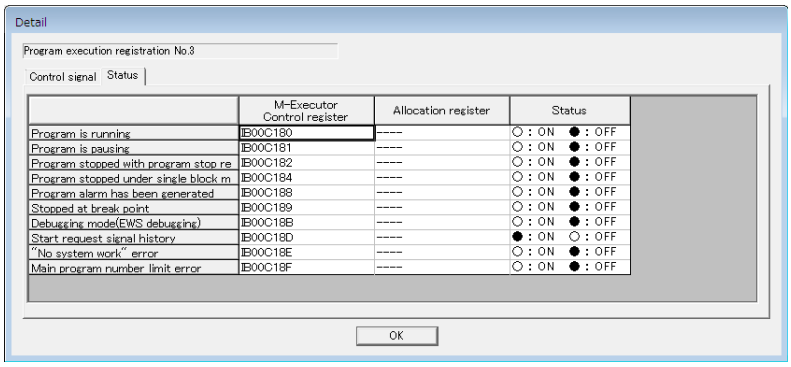
An allocation contact interlock is used to interlock the operation of a motion program.

When setting an allocation register, be sure to set the allocation contact interlock.

⑥ Status and Control Register

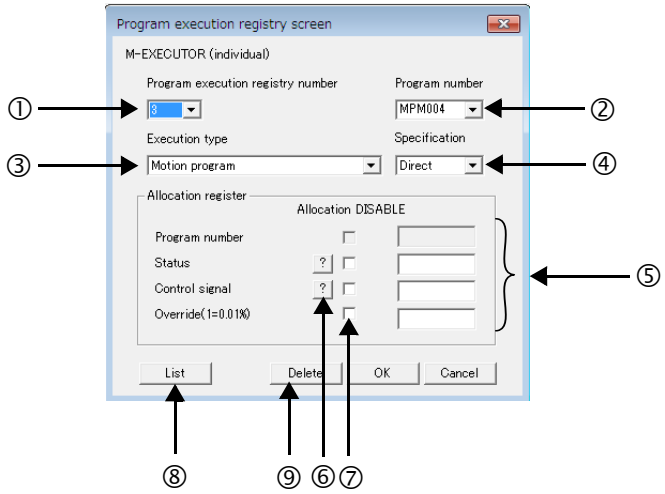
Double-click **Status** or **Control register** to display the bit detail window.
The signal sequence and status can be checked here. You can switch between the **Control signal** tab and the **Status** tab.





- Program execution registry screen (M-EXECUTOR (Individual display) window)

This window is displayed by clicking the **Individual display** button in the M-EXECUTOR (list display) window.



① Program execution registry number

Select the program execution registration number.
Programs will be executed in order, from the smallest program execution registration number.

② Program number

Set the program number.

③ Execution type

Select a program execution type from the drop-down menu:

Execution Type	Program to be Executed	Execution Conditions
Sequence program (Start)	Sequence programs	At power ON (Executed once at power ON.)
Sequence program (L scan)		Cyclic start (Executed in low-speed scan cycle.)
Sequence program (H scan)		Cyclic start (Executed in high-speed scan cycle.)
Motion program	Motion programs	When the Program operation start request bit of control signal turns ON. (Executed when the Program operation start request bit turns ON.)

④ Specification

Select the program designation method: Direct or Indirect

The applicable designation methods differ depending on whether the program is a motion program or a sequence program.

Designation Method	Motion Programs	Sequence Programs	Description
Direct	Applicable	Applicable	Designate the program number. Example: MPM001, SPM002
Indirect	Applicable	N/A	Designate the register number that stores the program number. Example: OW0C0C (When 1 is stored in OW0C0C, MPM001 is executed.)

⑤ Register allocation

Allocate the registers. The allocated registers exchange data in real time with the M-EXECUTOR control registers. I, O, and M registers can be allocated.

⑥ Status, Control signal (icons)

Click the corresponding icon to display the bit assignment of the status flag or control signal.

⑦ Allocation DISABLE

Click to select or clear the corresponding check box to enable or disable the allocated register. When the check box is cleared, the allocated register is enabled.

⑧ List button

Click the List button to display the M-EXECUTOR (list) window.

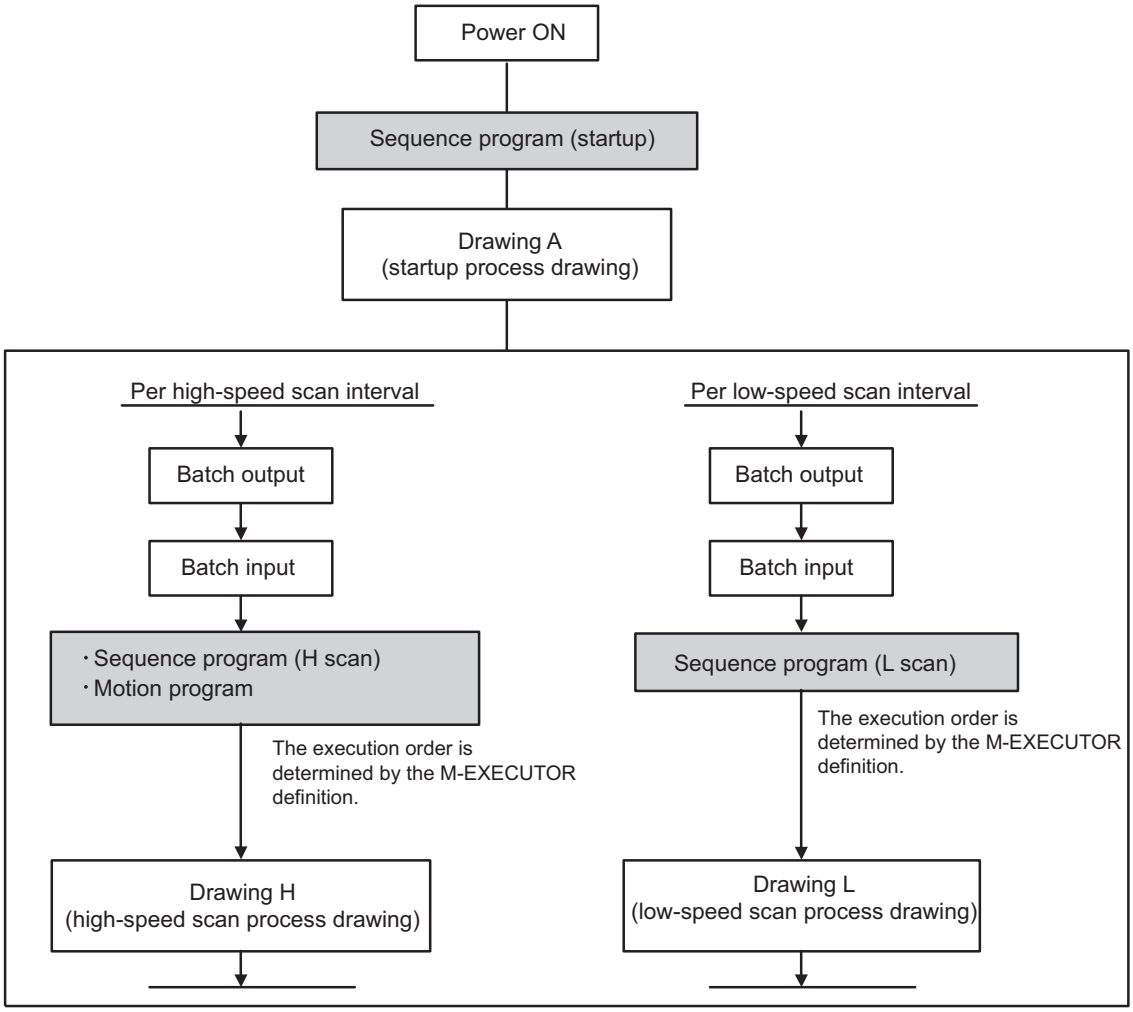
⑨ Delete button

Delete the definition.

(5) Execution Scheduling

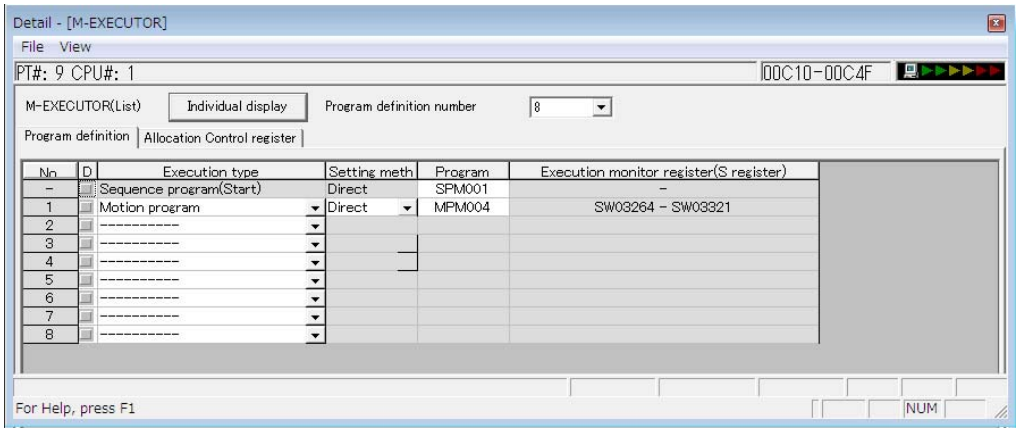
Programs registered in M-EXECUTOR are executed based on their priorities (execution type).

Programs registered in M-EXECUTOR are executed immediately before ladder processing.



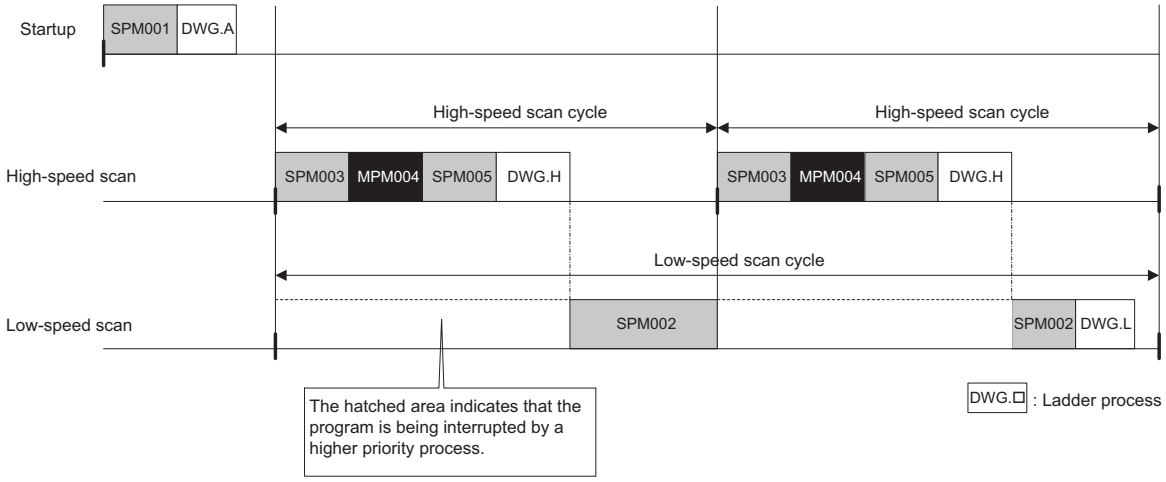
An execution example is as follows:

- M-EXECUTOR detail definitions



- Execution scheduling

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



3

Installation and Connection

This chapter explains how to install the MP2100/MP2100M in the host computer and how to connect each module.

3.1 Installing the MP2100/MP2100M

3.1.1 Computer Specifications

The following tables show the specifications for the host computer in which the MP2100/MP2100M is installed.

(1) Hardware Specifications

The MP2100/MP2100M occupies one of the host computer's PCI slots. Up to four MP2100/MP2100M Boards can be installed in one personal computer.

Item	Specification	Notes
Model	IBM PC/AT or compatible	NEC 9800 series computers are not compatible.
CPU	Pentium 200 MHz min.	Recommended: Pentium 400 MHz or faster
Memory	64 MBytes min.	Recommended: 128 MBytes or more
Available Hard Disk Space	500 MBytes min.	–
Display Resolution	800 × 600 min.	Recommended: 1024 × 768 or more
Expansion Slot*	One half-size PCI slot	–
Interrupt*	Uses 1 level	IRQ sharing is possible.
I/O Memory*	Uses 32 kBytes of shared memory.	–

* These are the specifications when one MP2100/MP2100M Board is being used.

When two or more MP2100/MP2100M Boards are installed in a computer, multiply the resources listed above by the number of boards.

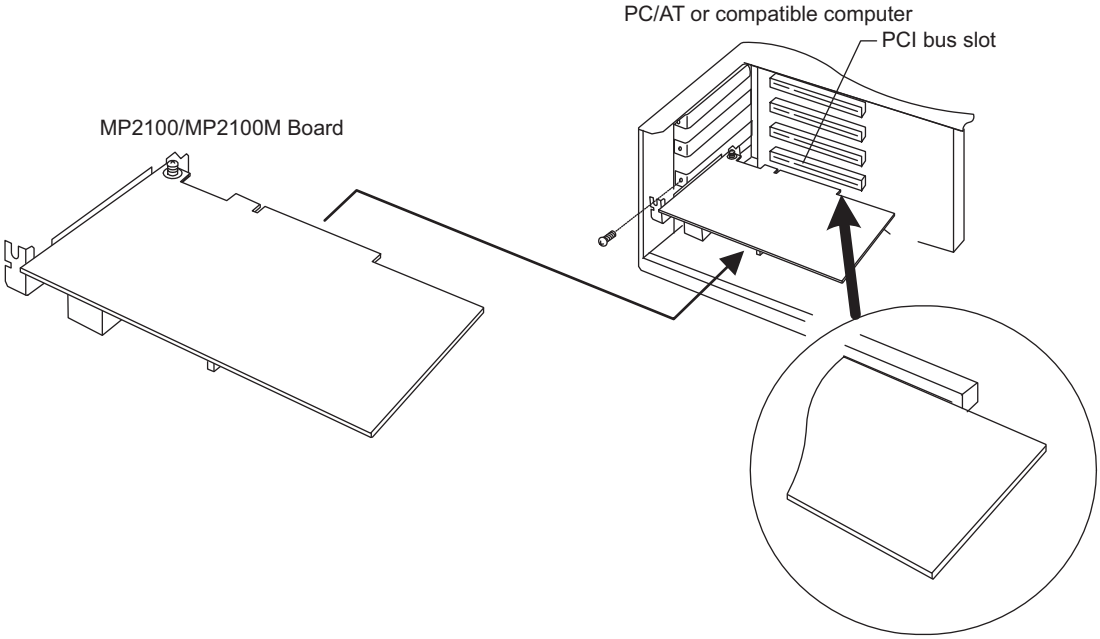
(2) Software Specifications

The following table shows the host computer software specifications.

Item	Specification	Notes
OS	Windows 7 (32-bit or 64-bit edition)	–
	Windows 8 (32-bit or 64-bit edition)	–
	Windows 8.1 (32-bit or 64-bit edition)	–
	Windows 10 (32-bit or 64-bit edition)	–
Browser	Microsoft IE 5.5	SP2 or later
Development Language	Microsoft Visual C/C++/Basic 6.0	SP5 or later
	Microsoft Visual C++/Basic/C#2003	–
	Microsoft Visual C++/Basic/C#2005	–
	Microsoft Visual C++/Basic/C#2008	–
	Microsoft Visual C++/Basic/C#2010	–
	Microsoft Visual C++/Basic/C#2012	–
	Microsoft Visual C++/Basic/C#2013	–
	Microsoft Visual C++/Basic/C#2015	–

3.1.2 Installing the MP2100/MP2100M

Install the MP2100/MP2100M to a PCI slot of the host computer. The MP2100/MP2100M occupies one half-size PCI slot.



⚠ CAUTION

- To prevent the MP2100/MP2100M from being damaged by static electricity, discharge any static electricity by touching a grounded metal object.
- Before installing or removing the MP2100/MP2100M, always turn OFF the host computer's power supply and unplug the computer's power cord.
  There is a risk of electrical shock or fire.
- When installing the MP2100/MP2100M, always press the Board firmly until it is fully seated in the PCI slot. If the Board cannot be inserted into the PCI slot with firm pressure, do not try to force it into the slot. Remove the Board, align it properly, and try inserting it again.
  If the Board is not fully inserted, the MP2100/MP2100M and host computer may be damaged or operate incorrectly.
- When handling the MP2100/MP2100M, hold the Board by its edges and never touch the components or soldered connections.
  Touching the components or leads can cause cuts or damage the MP2100/MP2100M and host computer.
- In some computers, secure the MP2100/MP2100M in the PCI slot with a screw or a clip after inserting the Board into the slot. Refer to the host computer's user's manual for details on securing PCI Boards.
  If the MP2100/MP2100M is not secured, it may become loose and the MP2100/MP2100M and host computer may be damaged or operate incorrectly.
- Connect and secure the cables that connect the MP2100/MP2100M to the servo drive and I/O Module.
  If the cable connectors are not fully inserted and secured, the MP2100/MP2100M, servo drive, and I/O Module may operate incorrectly.
- Never use the MP2100/MP2100M 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 subject the MP2100/MP2100M to strong shock.
  There is a risk of an accident.

3.1.3 Installing the Drivers

Use the following procedure to install the Windows drivers for the MP2100/MP2100M.

The driver installation procedure varies with different operating systems (OS), so verify which OS is being used in the host computer. This example explains how to install the drivers for Windows 10.

In this case, the DVD-ROM drive is drive “D.” If necessary, replace the “D:” drive letter with the actual DVD-ROM drive letter in your host computer.

The MP2100/MP2100M drivers are included in the MPE720 Ver.7 DVD-ROM.

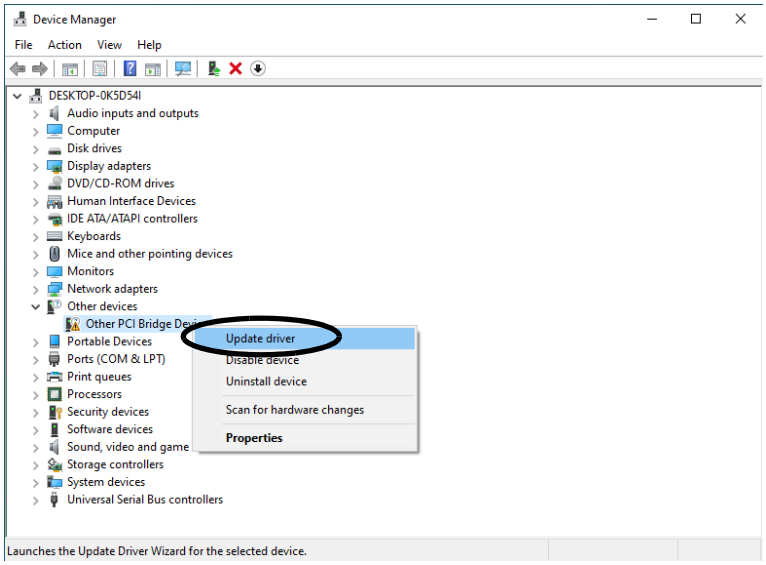
1. Turn OFF the power supply to the host computer and install the MP2100/MP2100M in a PCI slot. Secure the bracket of the MP2100/MP2100M with a screw.

Note: Use the driver with Ver.1.1.3.0 or later for the MP2100M.

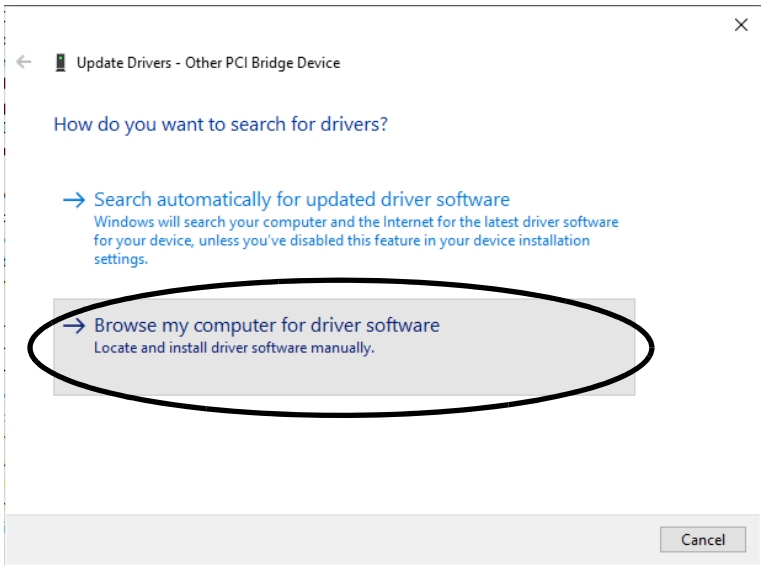
2. Turn ON the host computer and start Windows 10.

3. Right-click the Windows Start button and select **Device Manager**.

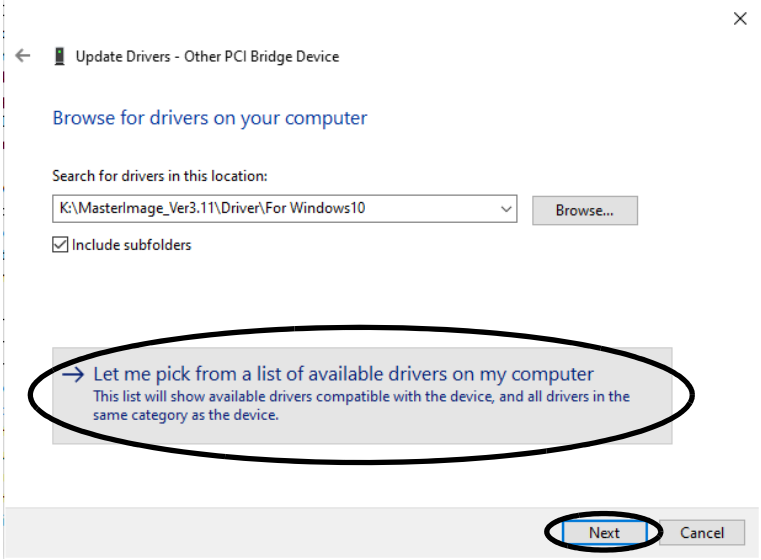
4. Right-click **Other PCI Bridge Device** and select **Update driver**.



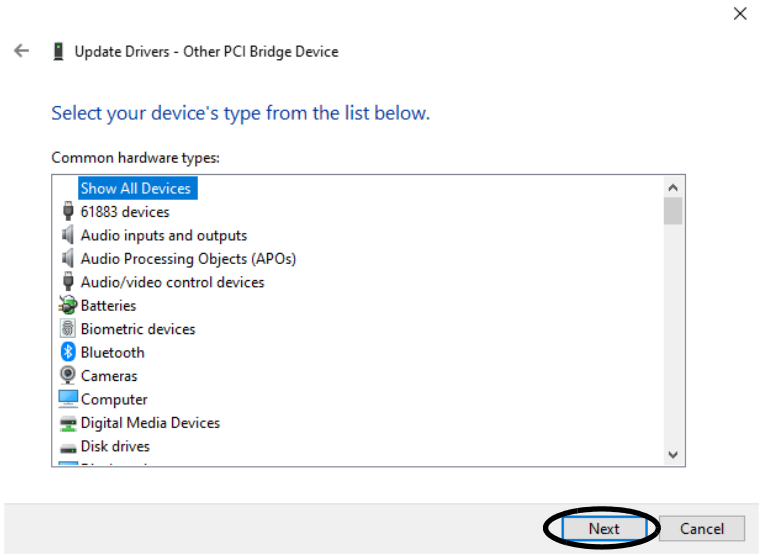
5. Click **Browse my computer for driver software**.



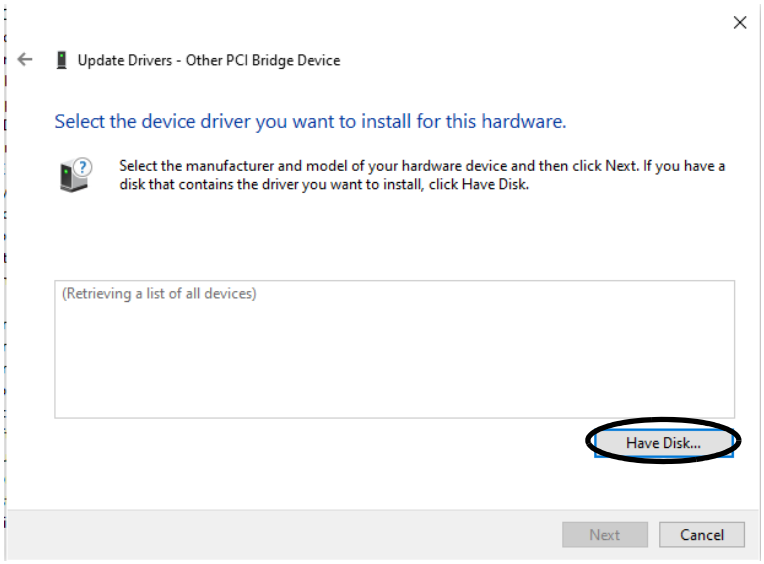
6. Select **Let me pick from a list of available drivers on my computer** and click the **Next** button.



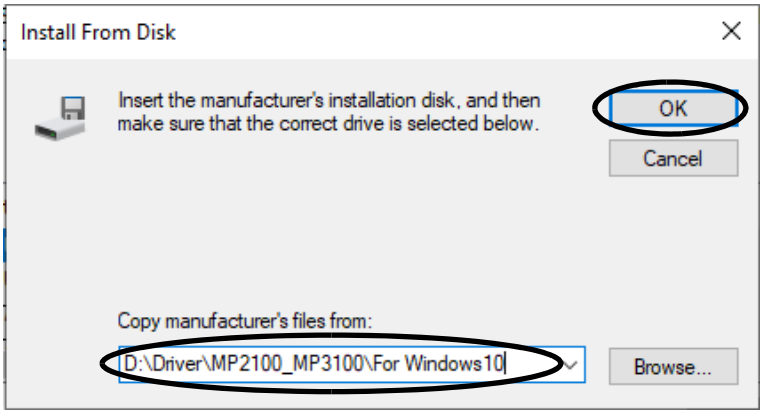
7. Click the **Next** button.



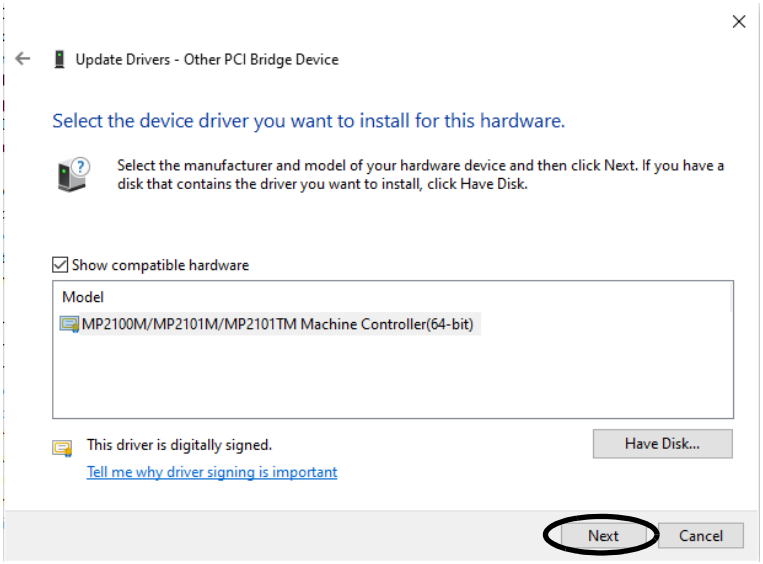
8. Click the **Have Disk** button.



9. Specify the path to the install CD. Specify "Driver\MP2100_MP3100\For Windows 10" on the installation CD and click the OK button.



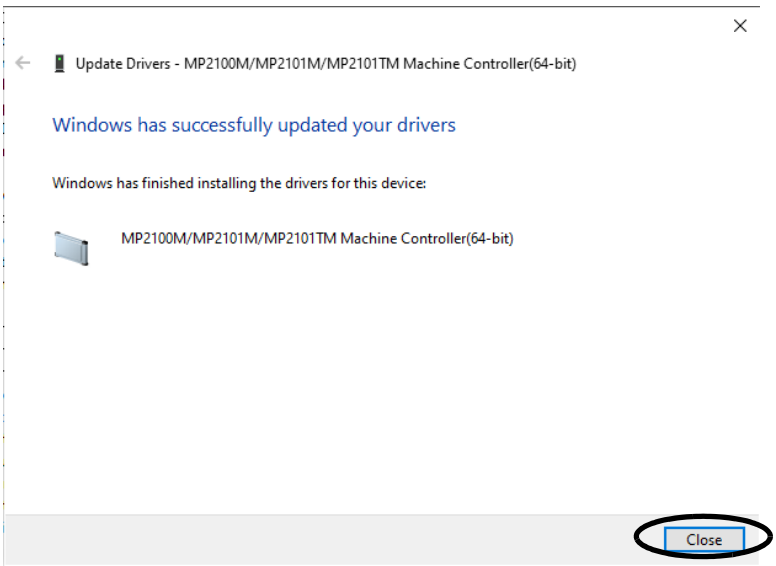
10. Click the **Next** button.



11. Click the **Install** button.

    The required files start to be copied. Wait until the files are finished being copied.

12. When the files have finished being copied, an installation complete message is displayed. Click the **Close** button.



This completes the driver installation.

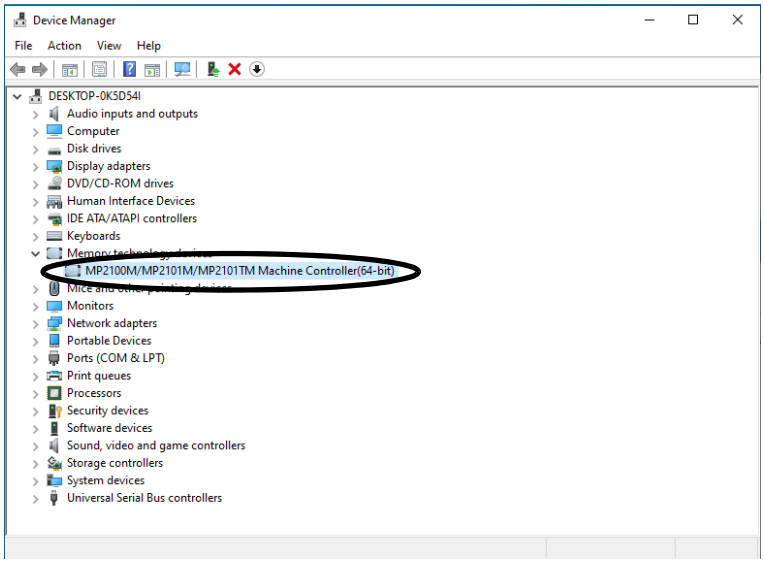
3.1.4 Verifying Driver Installation

Use the following procedure to verify that the MP2100/MP2100M is recognized properly by the system and the drivers are installed properly.

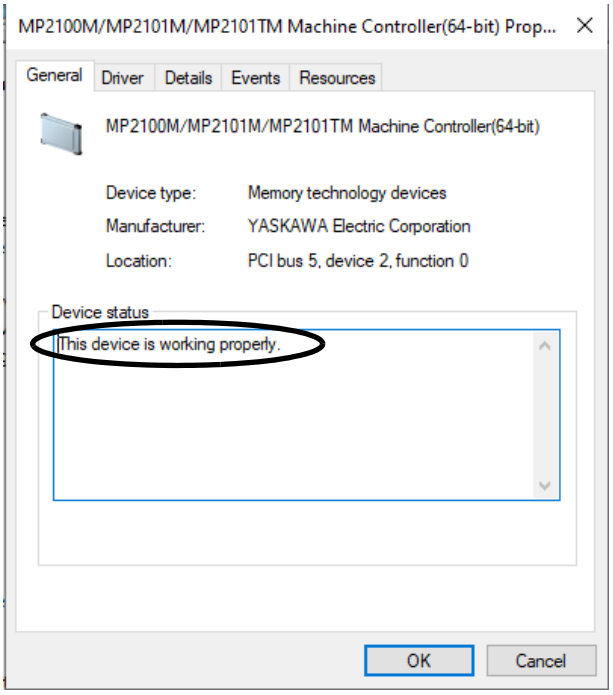
1. Click the Windows Start button and select *Settings* - *Devices* - *Connected devices*.

2. Select Device Manager under Related settings.

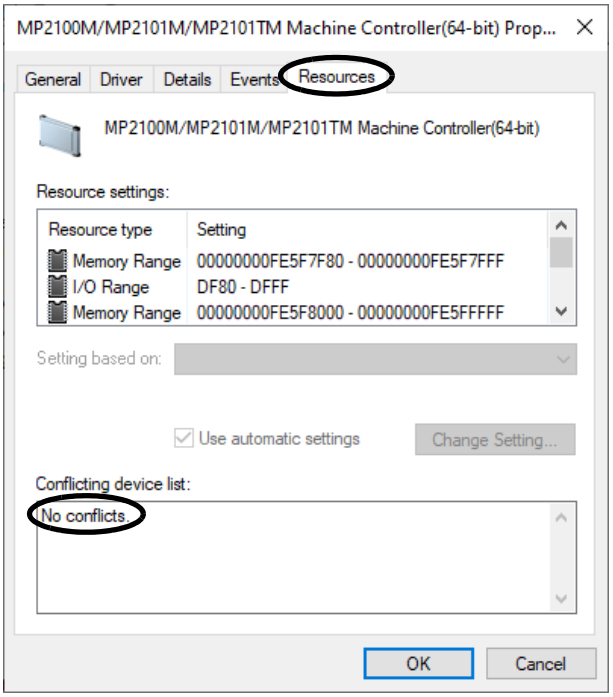
3. Open the **Memory technology drivers** folder and double-click the **MP2100M/MP2101M/MP2101TM Machine Controller(32-bit)** icon.
   (In the 64-bit version of Windows, "(64-bit)" is added to the end of the driver name.)



4. Check the **Device status** information and verify that it says **This device is working properly**.



5. Click the **Resources** tab and confirm that "No conflicts" is displayed in the **Conflicting device list**.



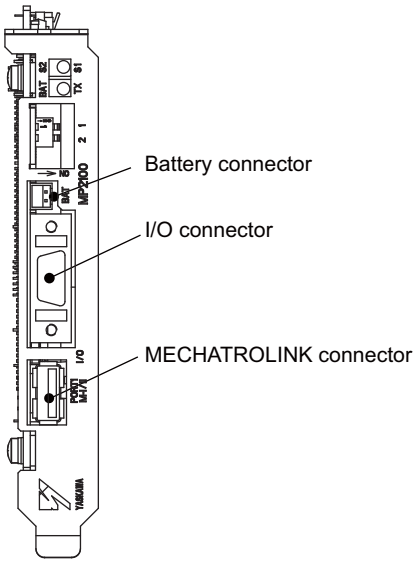
This completes the verification of the driver installation. The MP2100/MP2100M can be used if no problem has occurred up to this point.

If a problem has been identified, perform the installation again.

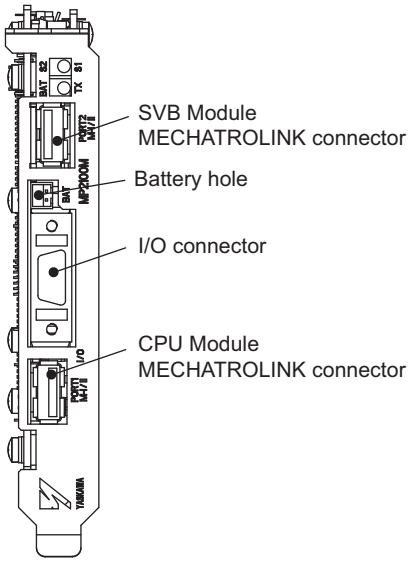
3.2 MP2100/MP2100M Connections

3.2.1 Connectors

The following diagram shows the connectors for the MP2100/MP2100M.



MP2100

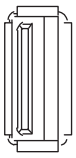


MP2100M

3.2.2 MECHATROLINK-I/II Connection

(1) MECHATROLINK-I/II Connector (M-I/II)

MECHATROLINK-I/II connector is used to connect the MP2100/MP2100M and the SERVOPACKs and distributed I/O Module.

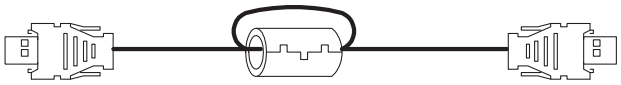
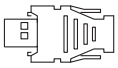


Pin Number	Signal Name	Description
1	(NC)	Not used.
2	/DATA	Signal – side
3	DATA	Signal + side
4	SH	Not used.
Shell	Shield	Connects the shield wire.

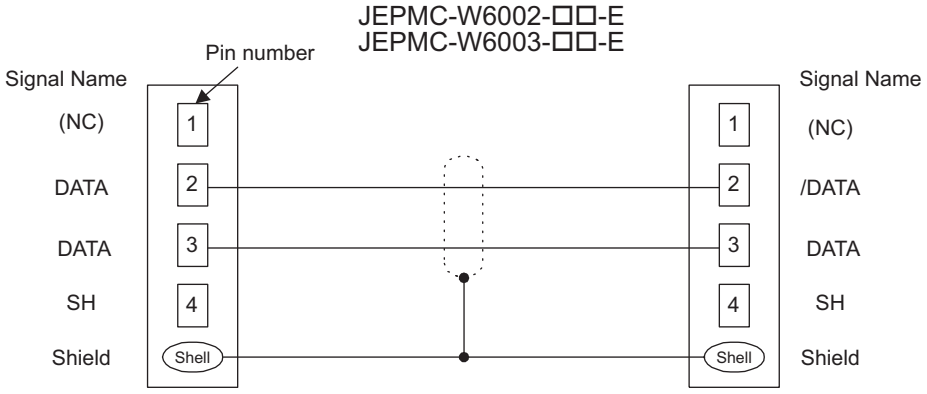
(2) Connector Specifications

Name	Connector Name	No. of Pins	Connector Model		
			Module	Cable	Manufacturer
MECHATROLINK Connector	M-I/II	4	USB-AR41-T11	DUSB-APA41B1-C50	Daiichi Denshi Kogyo K.K.

(3) Cables

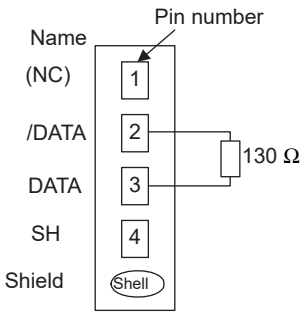
Name and Specification/External View	Model Number	Length
MECHATROLINK Cable MECHATROLINK connector - MECHATROLINK connector 	JEPMC-W6002-A5-E	0.5m
	JEPMC-W6002-01-E	1 m
	JEPMC-W6002-03-E	3 m
	JEPMC-W6002-05-E	5 m
	JEPMC-W6002-10-E	10 m
	JEPMC-W6002-20-E	20 m
	JEPMC-W6002-30-E	30 m
	JEPMC-W6002-40-E	40 m
	JEPMC-W6002-50-E	50 m
MECHATROLINK Cable MECHATROLINK connector - MECHATROLINK connector, with ferrite core 	JEPMC-W6003-A5-E	0.5 m
	JEPMC-W6003-01-E	1 m
	JEPMC-W6003-03-E	3 m
	JEPMC-W6003-05-E	5 m
	JEPMC-W6003-10-E	10 m
	JEPMC-W6003-20-E	20 m
	JEPMC-W6003-30-E	30 m
	JEPMC-W6003-40-E	40 m
	JEPMC-W6003-50-E	50 m
Terminator (terminating resistor) 	JEPMC-W6022-E	-

(4) Cable Connections between the MP2100/MP2100M and I/O Units or SERVOPACKs

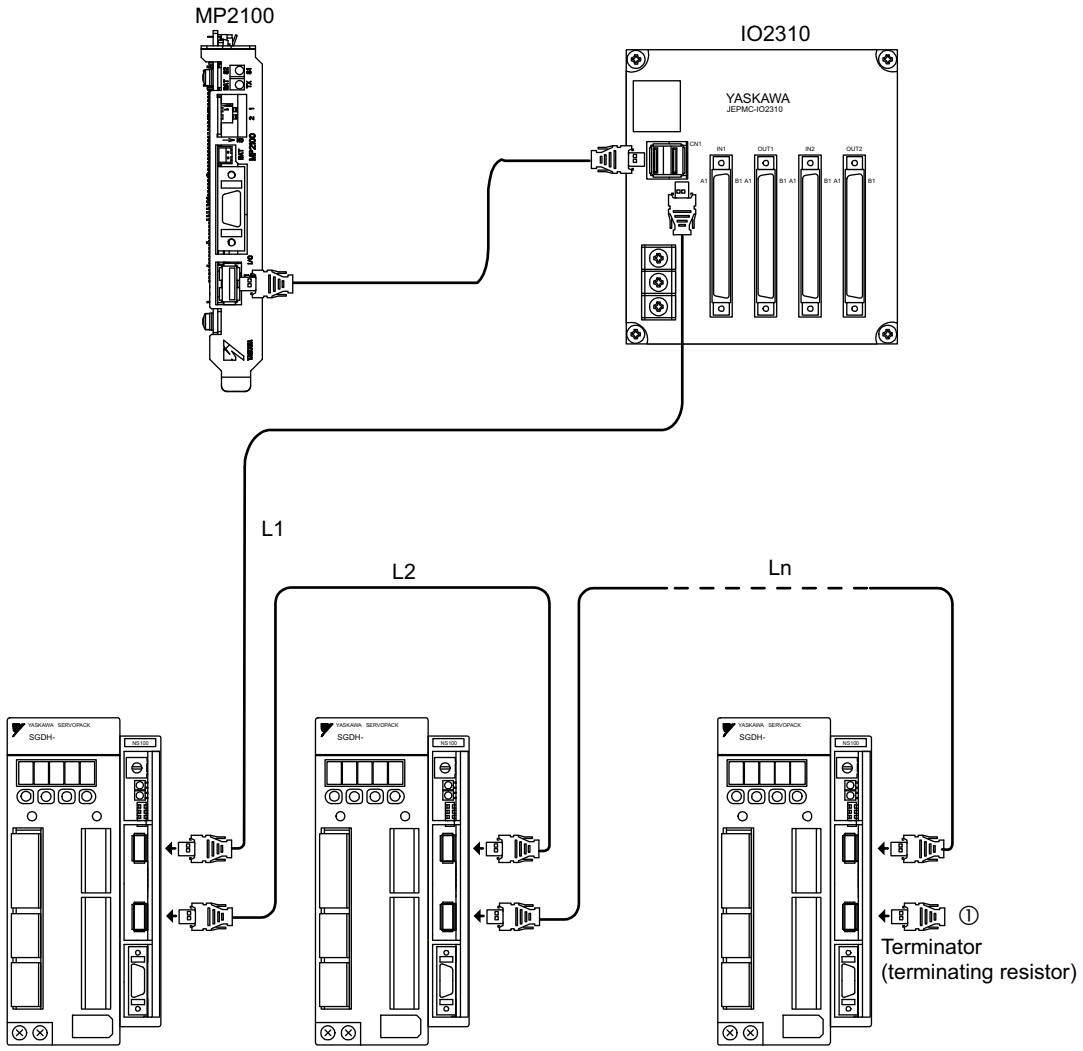


Note: The JEPMC-W6003-□□-E cable has a ferrite core.

(5) Terminator Connections



(6) Connection Example between MP2100/MP2100M, SERVOPACK, and IO2310



Note: 1. Use standard cables between units.
2. Use under the condition that L1 + L2 + L3 + . . . + Ln ≤ 50 m (100 m if repeaters are used).

IMPORTANT

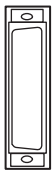
The MP2100/MP2100M has a built-in terminator.
Insert a JEPMC-W6022-E terminator into ① in the above diagram.

3.2.3 I/O Connection

(1) I/O Connector

I/O connector is used to connect the MP2100/MP2100M and external I/O signals.

External input: 5 points; External output: 4 points



(2) Connector Specifications

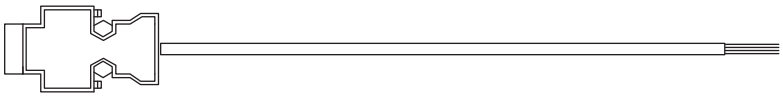
Name	Connector Name	No. of Pins	Connector Model		
			Module	Cable	Manufacturer
I/O Connector	I/O	14	10214-52A2JL	• 10114-3000PE Connector • 10314-52A0-008 or 10314-52F0-008 shell	Sumitomo 3M Limited.

(3) Cables

Name	Model Number	Length
I/O Cable (loose wires)	JEPMC-W2062-A5-E	0.5 m
	JEPMC-W2062-01-E	1 m
	JEPMC-W2062-03-E	3 m

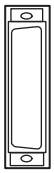
(4) External Appearance of I/O Cable

JEPMC-W2062-□□-E



(5) Connector Pin Arrangement

The following table shows the connector pin arrangement.



Pin Number	Signal Name	I/O	Remarks	Pin Number	Signal Name	I/O	Remarks
1	DI_24 V	P	Input common	8	DI_24V	P	Input common
2	DI_00	I	Input 00	9	DI_02	I	Input 02
3	DI_01	I	Input 01	10	DI_03	I	Input 03
4	DI_04	I	Input 04	11	DO_24V	P	+24V input
5	DO_COM	P	Output common	12	DO_COM	P	Output common
6	DO_00	O	Output 00	13	DO_02	O	Output 02
7	DO_01	O	Output 01	14	DO_03	O	Output 03

Note: P: Power input; I: Input signal; O: Open-collector output

(6) Input Circuits

The following table shows the I/O Connector input circuit specifications.

Item	Specifications		
Inputs	5 points	DI-00	General-purpose input (shared with interrupts)
		DI-01 to DI-04	General-purpose input
Input Format	Sink mode/source mode input		
Isolation Method	Photocoupler		
Input Voltage	±24 VDC, ±20%		
Input Current	4.1 mA (typ.)		
ON Voltage/Current	15 VDC min./2.0 mA min.		
OFF Voltage/Current	5 VDC max./1.0 mA max.		
ON Time/OFF Time	ON: 1 ms max. OFF: 1 ms max.		
Number of Commons	5 points		

Note: DI-00 is shared with an interrupt input. If DI-00 is turned ON while interrupts are enabled, the interrupt processing drawing is executed.

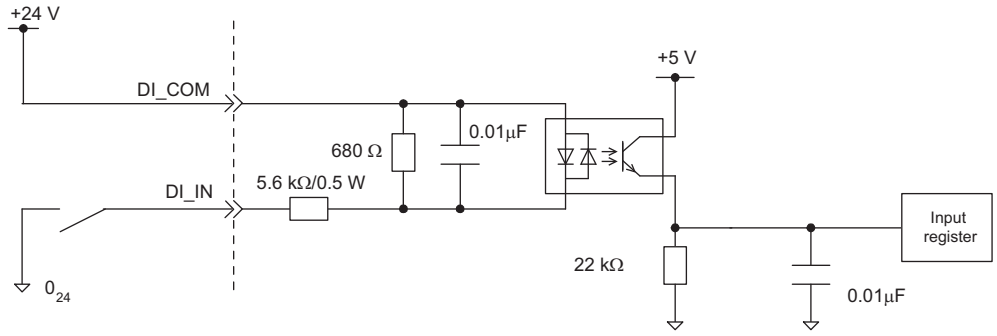


Figure 3.1 Digital Input Circuit (Source mode Input)

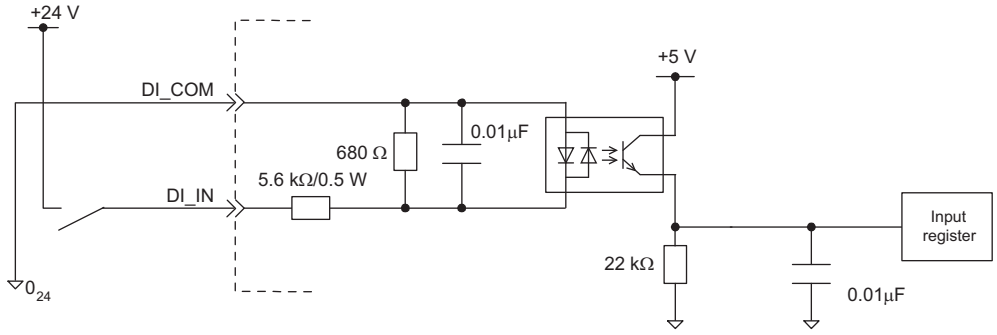


Figure 3.2 Digital Input Circuit (Sink mode Input)

(7) Output Circuit

The following table shows the I/O Connector output circuit specifications.

Item	Specifications
Outputs	4 points
Output Format	Transistor, open-collector, sink mode output
Isolation Method	Photocoupler
Output Voltage	+24 VDC, ±20%
Output Current	100 mA max.
Leakage Current When OFF	0.1 mA max.
ON Time/OFF Time	ON: 1 ms max. OFF: 1 ms max.
Number of Commons	4 points
Protection Circuit	Fuse The fuse is not, however, for circuit protection. It is for protecting against fire during output shorts. Attach a fuse externally to each output if circuit protection is required.

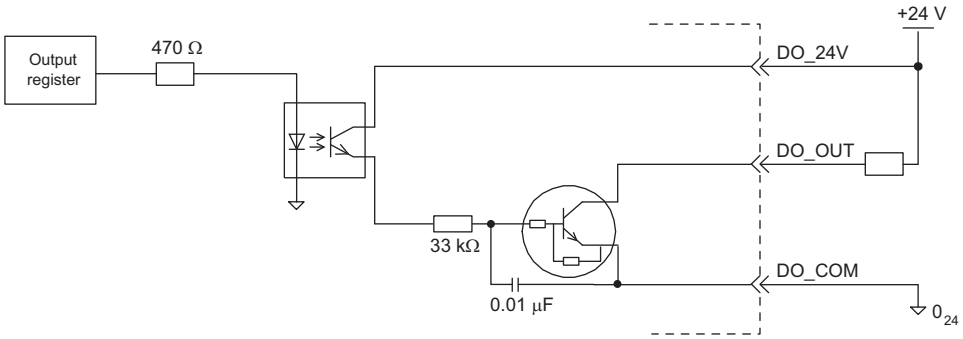
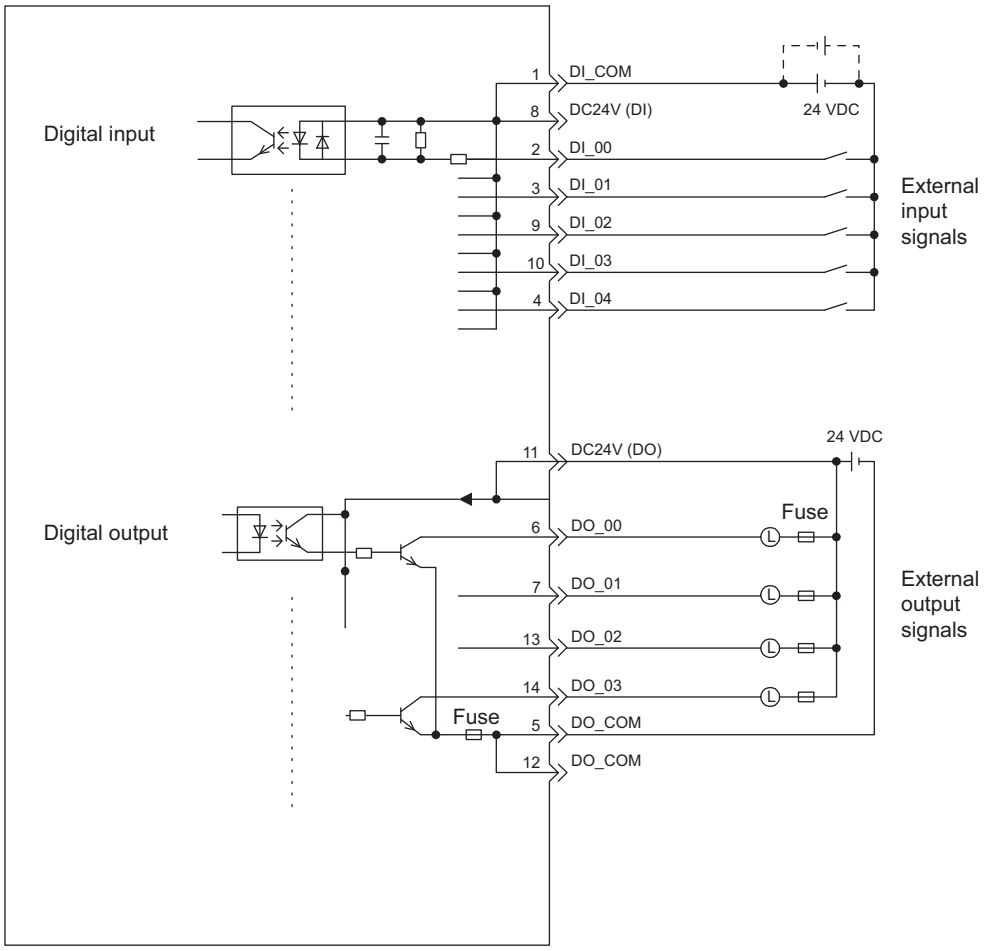


Figure 3.3 Digital Output Circuit (Sink Mode Output)

(8) I/O Connector Connections

The following diagram shows the connections for the I/O connector.



Note: Connect a fuse suitable for the load specifications in the output signal circuit in series with the load. If an external fuse is not connected, load shorts or overloads may result in fire, destruction of the load device, or damage to the output element.

4

System Startup and Sample Programs

This chapter explains the procedure for starting the MP2100/MP2100M system and sample programs for typical operation and control.
For details on MP2100M, refer to *4.5 MP2100M Startup*.

4.1 Startup Procedure for a Model System

This section explains the system startup procedure using the MP2100 when the sample program on the MPE720 installation disk is used. Details on the machine system design have been omitted here.

For differences in the procedure for MP2100M, refer to *4.5 MP2100M Startup*.

4.1.1 Flowchart for Model System Startup

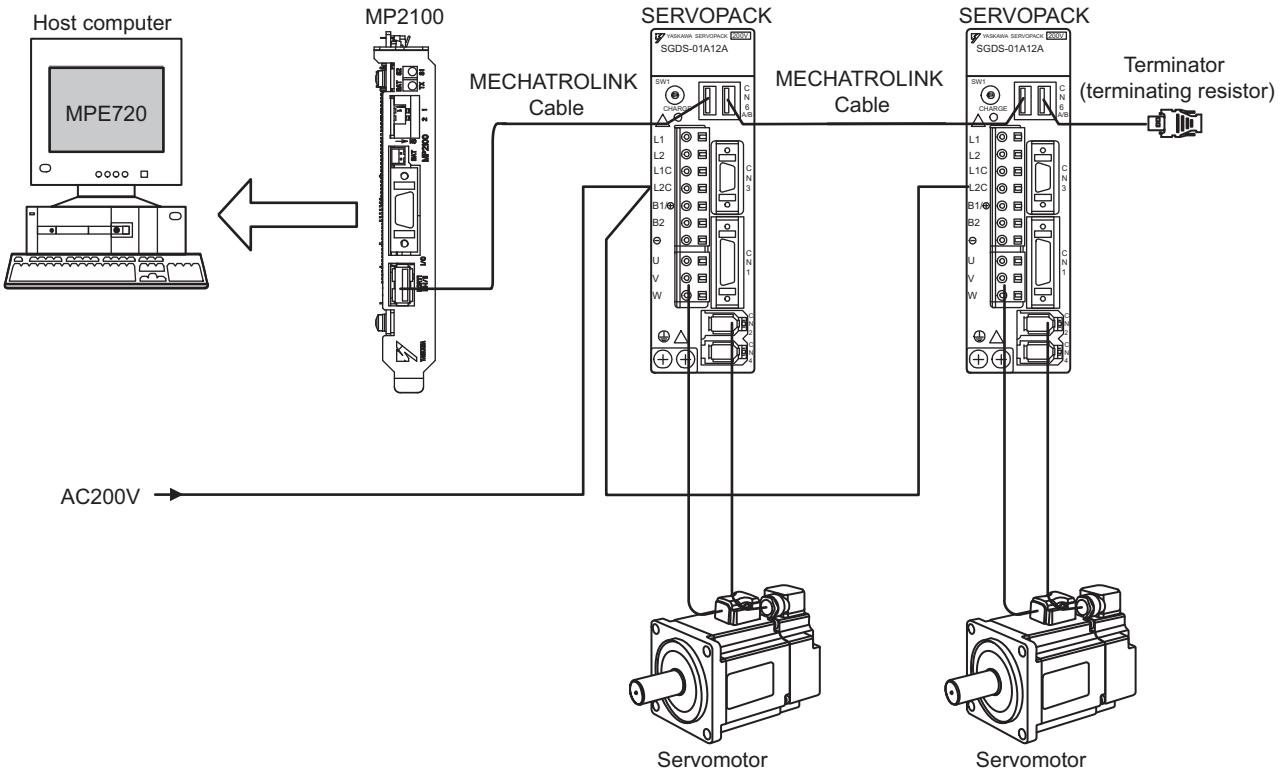
The system startup procedure is outlined below. Refer to the references given in italics for more information.

1	Equipment Preparation Prepare the equipment required for testing. → *4.1.2 System Configuration Model and Necessary Devices*
⇩	
2	Mounting the MP2100 Mount the MP2100 to the host computer. → *3.1 Installing the MP2100/MP2100M*
⇩	
3	Connecting and Wiring the System Wire the Servomotors and SERVOPACKs. → *4.1.2 System Configuration Model and Necessary Devices*
⇩	
4	Initializing the SERVOPACKs Initialize the SERVOPACKs. → *4.1.4 Initializing SERVOPACK Parameter Set Values*
⇩	
5	Setting and Saving the Communication Platform Use the Communication Platform to start communications and use the contents to make settings or save the information of each port. → *4.1.5 Setting and Saving Communication Platform*
⇩	
6	Self-configuration of the MP2100 The connected devices are automatically confirmed. → *4.1.6 Executing the MP2100 Self-configuration*
⇩	
7	Reading the Sample Programs and Starting MPE720 Ver.7 Read the sample programs from the installation disk and start MPE720 Ver.7. → *4.2.1 Copying and Transferring Sample Program Files*
⇩	
8	Setting and Saving the Sample Programs After transferring the sample programs to the MP2100, set the fixed parameters and other settings, then save them in flash memory together with the configuration definition. → *4.2.2 Setting Motion Fixed Parameters and Adjusting the Settings for Servo Control*, *4.2.3 Saving and Transferring Data*
⇩	
9	Checking Operation Execute the program and check the test operation.→ *4.3 Checking Sample Program Operation*

4.1.2 System Configuration Model and Necessary Devices

The system startup in this section is explained by reference to the following configuration model.

Prepare the devices indicated in (1) to (3) below and connect them in the manner shown in the following diagram.

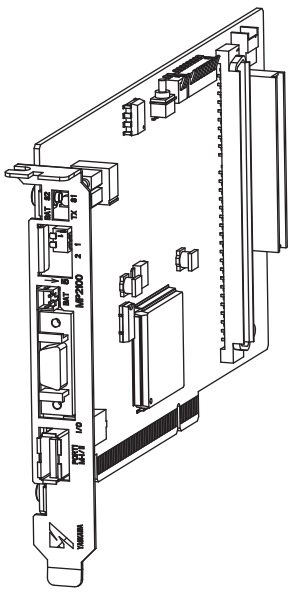


Set the MECHATROLINK station numbers of SERVOPACK to 1 and 2.

The sample program is designed to operate with station numbers 1 and 2.

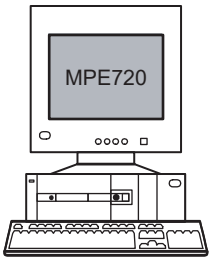
(1) Controller-related Equipment

Name	Model	Quantity
MP2100	JEPMC-MP2100	1
MECHATROLINK Cables (1 m)	JEPMC-W6002-01-E	2
Terminator (terminating resistor)	JEPMC-W6022-E	1



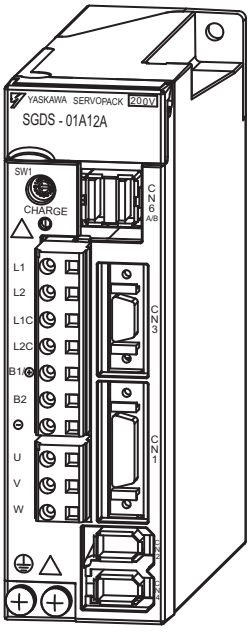
(2) Host Computer

Name	Model	Quantity
Computer	Commercially-available product, with PCI half-size slot	1
MPE720	CPMC-MPE780D (All versions)	1
Motion API	CPMC-MPA700	1

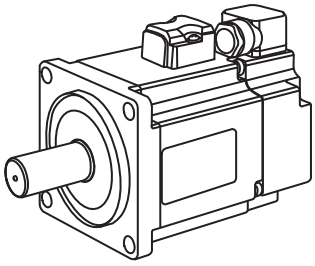


(3) Servodrive-related Equipment

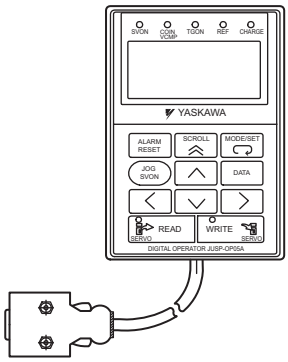
Name	Model	Quantity
Σ-III SERVOPACKs	SGDS-01A12A	2
Σ-III Servomotors	SGMAS-01ACA21	2
Motor Cables (3 m)	JZSP-CSM01-03	2
Encoder Cables (3 m)	JZSP-CSP01-03	2
Digital Operator	JUSP-OP05A-1-E	1



SERVOPACK



Servomotor



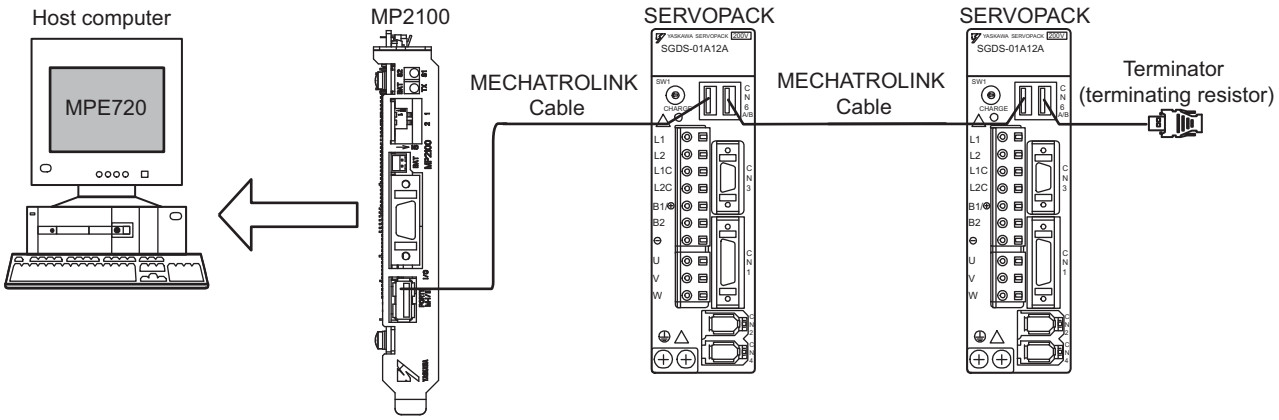
Digital Operator

4.1.3 Connecting and Wiring the System

(1) Connecting the MP2100 and SERVOPACKs

Use a MECHATROLINK Cable to connect the MP2100 and SERVOPACKs.

For the MECHATROLINK cables to use, refer to *3.2.2 MECHATROLINK-I/II Connection*.

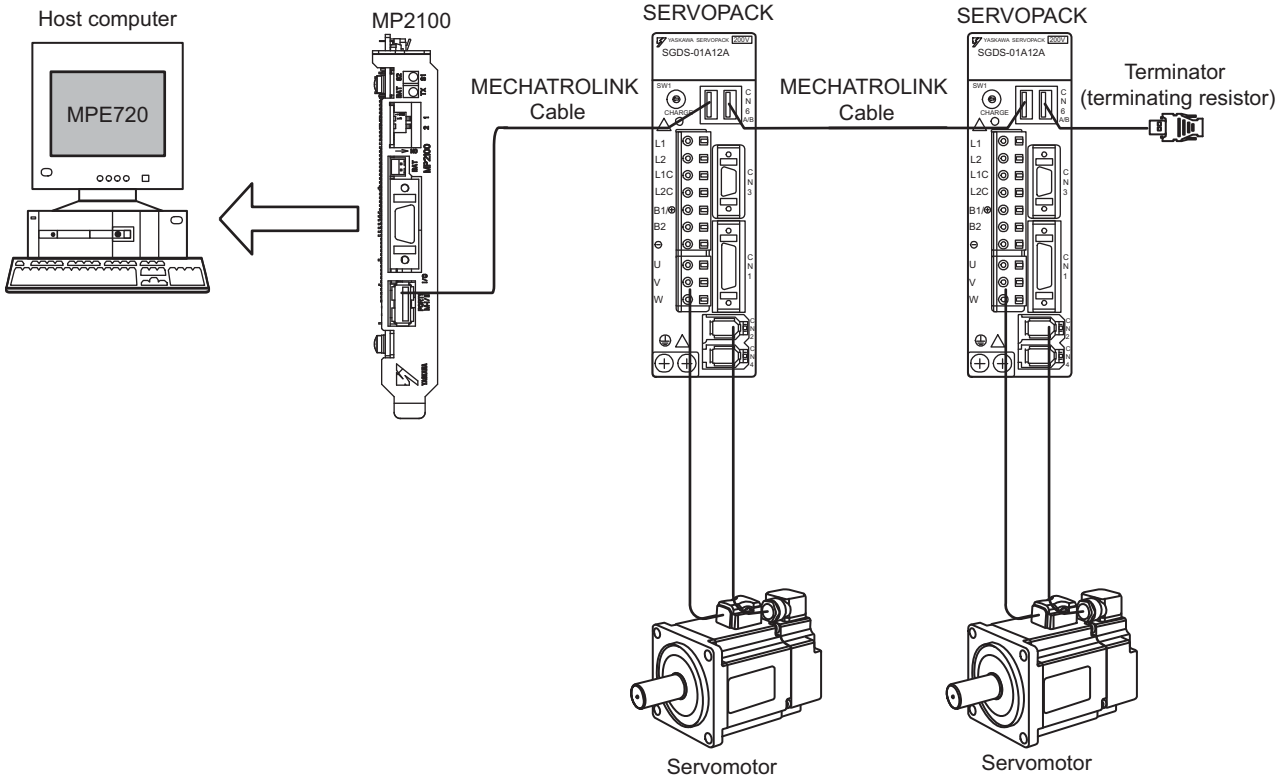


Set the SERVOPACK MECHATROLINK station numbers to 1 and 2.

The sample program is designed to operate with station numbers 1 and 2.

(2) Connecting the SERVOPACKs and Servomotors

Use the motor cable and encoder cable to connect the SERVOPACKs and Servomotors.



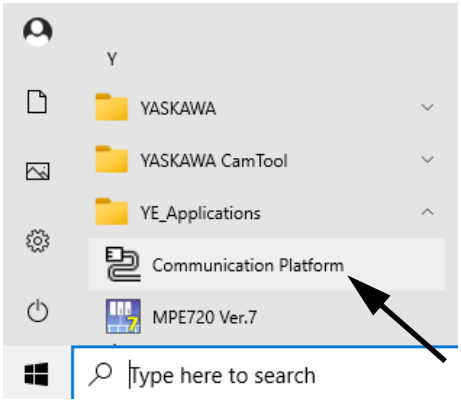
4.1.4 Initializing SERVOPACK Parameter Set Values

Refer to the manual for your SERVOPACK and initialize the parameter set values.

4.1.5 Setting and Saving Communication Platform

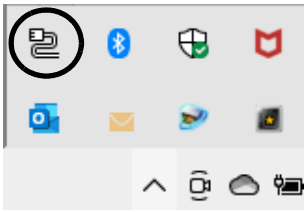
Use the following procedure to set the communication process between the personal computer (MPE720) and the MP2100 using the Communication Platform. These settings are not required if the communication settings have already been made.

1. Select ***Start - All Programs - YE_Applications - Communication Platform*** from the Windows Start Menu to start the Communication Platform.



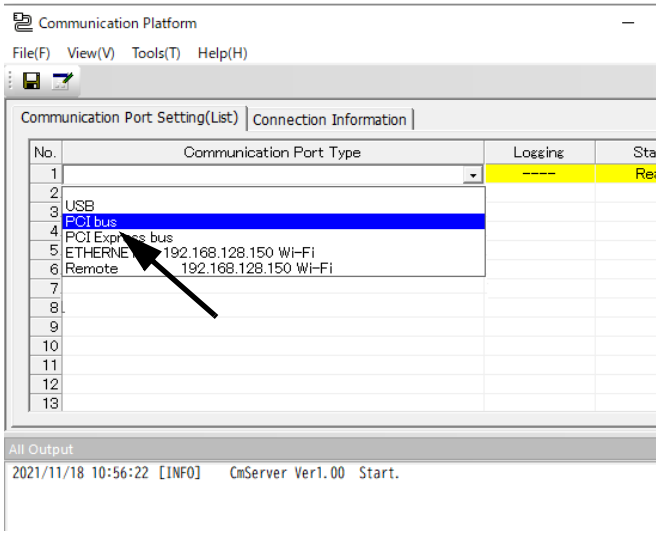
   The **Communication Platform** icon appears on the task tray at the bottom right of the window.

2. Double-click the **Communication Platform** icon on the task tray to open the **Communication Platform** window.



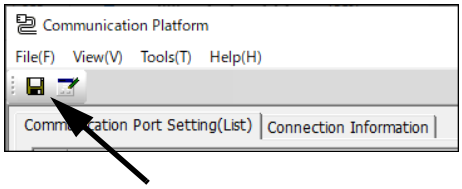
Communication Platform Icon

3. Set communications port **No.1** to **PCI bus**.

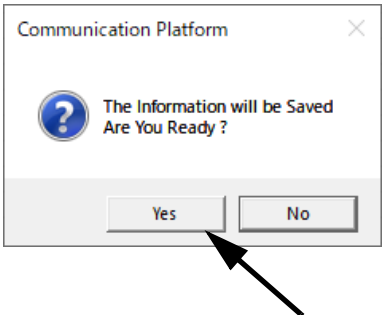


4. Save the Communication Port Settings
These settings will be used as the communication port information whenever the Communication Platform is started. Use the following procedure to save the communication port settings.

a) Click the **Save** Button.

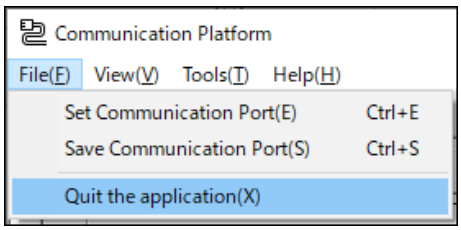


b) A save confirmation window is displayed. Click the **Yes** button.

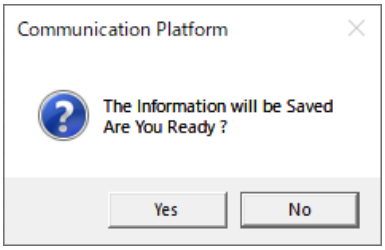


5. Restart Communication Platform
After the Communication Platform settings are configured or changed, the Communication Platform must be restarted.

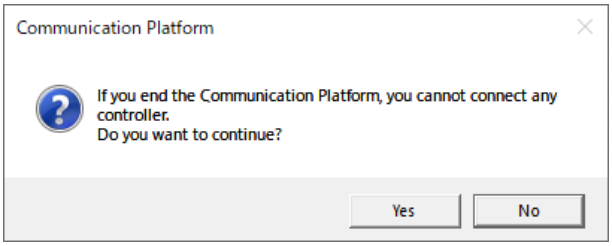
a) Select *File* - *Quit the application* to close the Communication Platform window.



b) A save confirmation window is displayed. Click the **Yes** button.



c) An exit confirmation window is displayed. Click the **Yes** button.



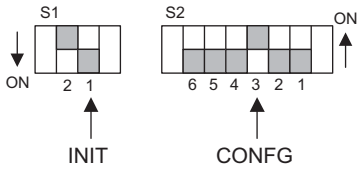
d) Restart the **Communication Platform** window with step 1 of this procedure.

4.1.6 Executing the MP2100 Self-configuration

Execute self-configuration to automatically recognize the devices connected to the MECHATROLINK connector.

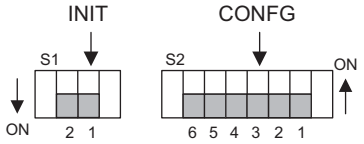
This section explains the method for self-configuration of the JAMC-MC2100. For other Modules, refer to *2.2 Functions of the MP2100/MP2100M Base Module*.

1. Check that the power supply to the Σ-III SERVOPACK is turned ON.
2. Turn ON the 1 (INIT) of the mode switch 1 (S1) and 3 (CNFG) of the mode switch 2 (S2).



3. Turn ON the reset switch and check that the LED indications of the MP2100 change as shown below.

4. Turn OFF the 1 (INIT) of the mode switch 1 (S1) and 3 (CNFG) of the mode switch 2 (S2).



For mode switches, refer to *2.2.1 (3) Switch Settings* .

IMPORTANT

- Mode switch INIT

  RAM data will be cleared if 1 (INIT) of the mode switch 1 on the MP2100 is turned ON and the reset switch is turned ON.

  Flash memory data is read when pin 1 (INIT) of the mode switch 1 is turned OFF and the reset switch is turned ON. Therefore, always save data to the MP2100 flash memory before turning OFF the power when writing or editing programs.

  Refer to *4.2.3 (1) Saving to Flash Memory* for information on how to save data to flash memory.

- Turning OFF Power after Executing Self-configuration

  Do not turn OFF the power supply of the host computer after executing self-configuration until the definitions data has been saved to flash memory in the MP2100. If the power is turned OFF somehow before the data is saved to flash memory, execute self-configuration again.

4.2 Starting Sample Program - MPE720 Ver.7

This section explains how to copy the sample program file from MPE720 Ver.7 installation disk, how to start MPE720 Ver.7 to transfer the sample program to the MP2100, and how to set and save parameters, according to the following flowchart.

The descriptions assume that the MPE720 is already installed on the computer.

1	Copy the sample program file onto the hard disk of the computer from the installation disk contained in the CD-ROM.	→ *4.2.1 (1) Copying the Sample Program File*
⇩		
2	Start MPE720 Ver.7 and transfer the sample program file to the MP2100.	→ *4.2.1 (2) Starting MPE720 Ver.7 to Transfer the Sample Program to the MP2100*
⇩		
3	Open the Module Configuration Definition Window to set and save the motion fixed parameters.	→ *4.2.2 (1) Setting and Saving Motion Fixed Parameters*
⇩		
4	Adjust and save the SERVOPACK parameters.	→ *4.2.2 (2) Making Servo Adjustments and Saving SERVOPACK Parameters*
⇩		
5	Save the parameter settings stored in the MP2100 RAM to the flash memory.	→ *4.2.3 (1) Saving to Flash Memory*
⇩		
6	Transfer the MP2100 data to the computer.	→ *4.2.3 (2) Transferring Data from the MP2100 to the Computer*

4.2.1 Copying and Transferring Sample Program Files

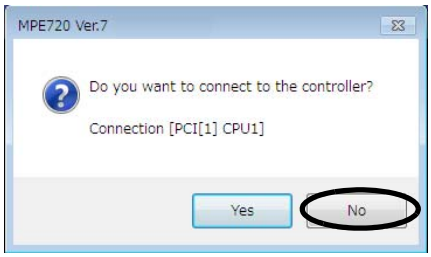
(1) Copying the Sample Program File

The sample program is stored in the following folder on the MPE720 Ver.7 installation disk. Copy the sample program to any folder on the computer.
\MPE720 Ver.7\Installer\AppendFiles\Sample\ENG\2100SMPL_E.YMW

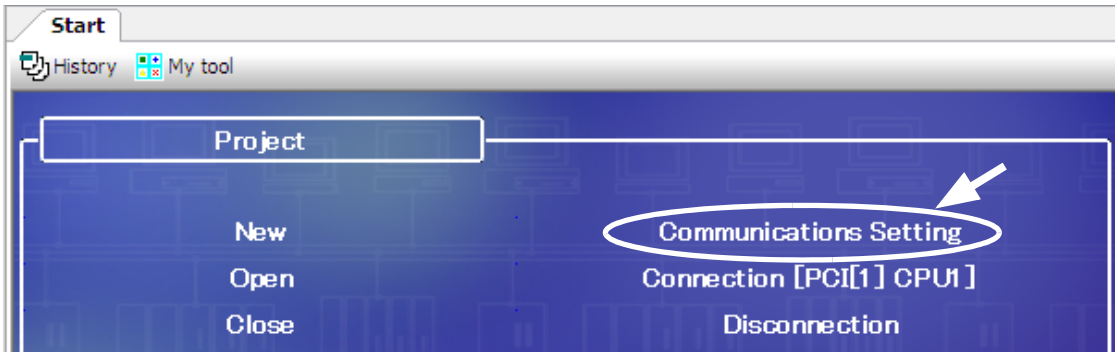
(2) Starting MPE720 Ver.7 to Transfer the Sample Program to the MP2100

1. Double-click the **2100SMPL_E.YMW** file copied to the hard disk.

   MPE720 Ver.7 starts up and the following message asks for confirmation to connect to the controller. Click the **No** button.



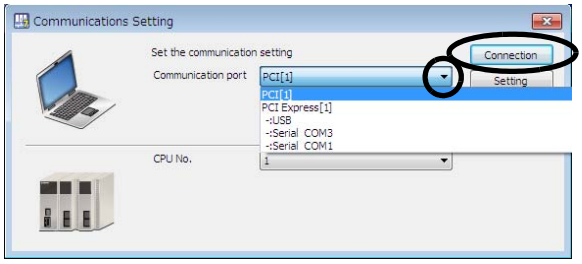
2. Click **Communications Setting** in the MPE720 Ver.7 main window.



   The **Communications Setting** dialog box appears.

3. Click the ▼ button, and select the slot where the MP2100 has been connected from the dropdown list box. Then, click the **Connection** Button.

   Note: In the drop-down list box, the details of each slot set in the procedure described in *4.1.5 Setting and Saving Communication Platform* are displayed.

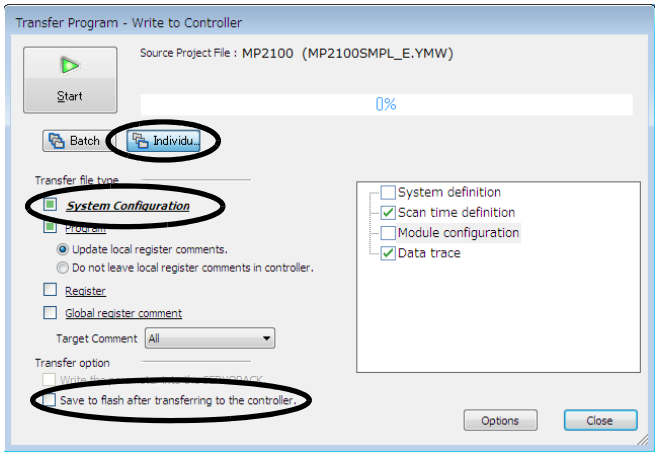


   The MP2100 and your computer is connected online.

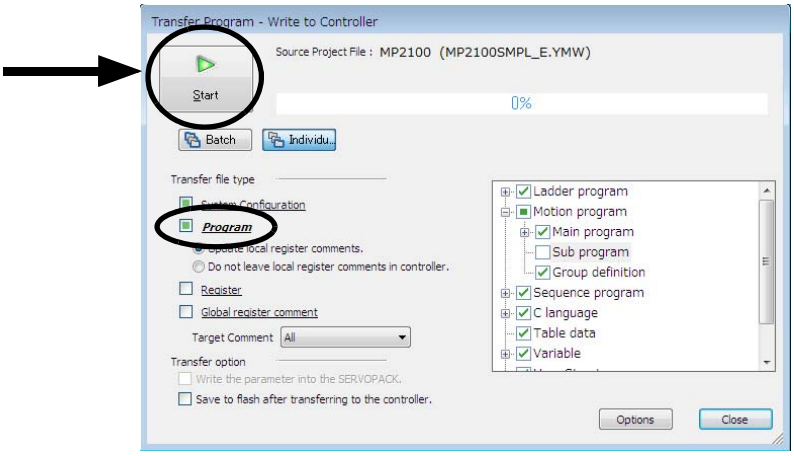
4. Select ***Online*** - ***Write into Controller*** in the menu bar.

   The **Transfer Program - Write into Controller** dialog box appears.

5. Click the **Individual** button to select an individual file transfer, and click the **Save to flash after transferring to the controller** check box to clear it. Then, select the **System Configuration** check box.
   Select the **Scan time definition** and **Data trace** check boxes from the list of files that is displayed on the right.

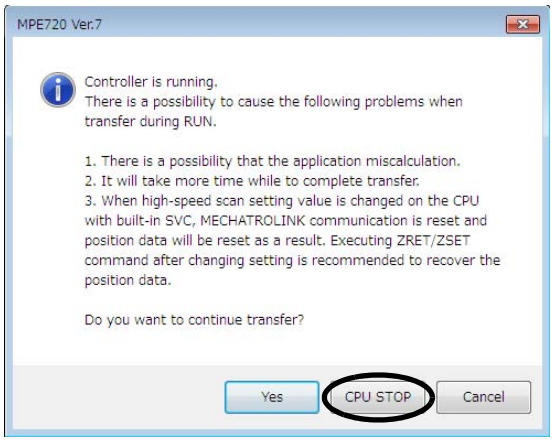


6. Select the **Program** check box. Then, select the **Main program** and **Group definition** check boxes under **Ladder Program** and **Motion Program**. Click the **Start** button.



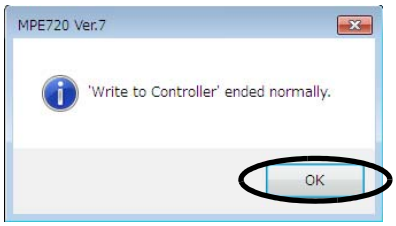
   A message box appears and ask whether or not to stop the MP2100 to allow transfer of the files.

7. Click the **CPU STOP** button.



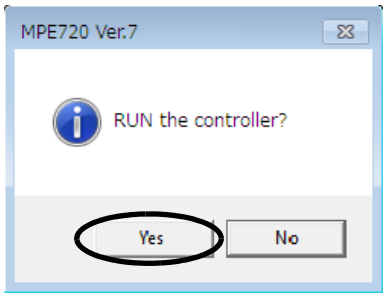
   The MP2100 stops running and MPE720 starts transferring the selected sample program files. When the transfer is complete, a message box appears to tell you that the sample program files have been successfully written to the MP2100.

8. Click the **OK** button in the message box.



Another message box appears asking whether or not to run the MP2100.

9. Click the **Yes** button to run the MP2100.



The MP2100 starts running.

This completes the transfer of the sample program files to the MP2100.

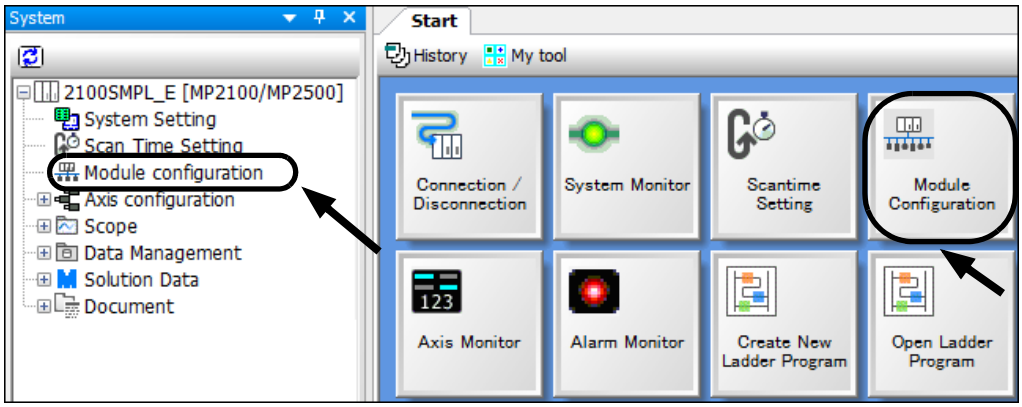
4.2.2 Setting Motion Fixed Parameters and Adjusting the Settings for Servo Control

(1) Setting and Saving Motion Fixed Parameters

This section explains how to set and save MP2100 motion fixed parameters for Axes 1 and 2 according to the sample program.

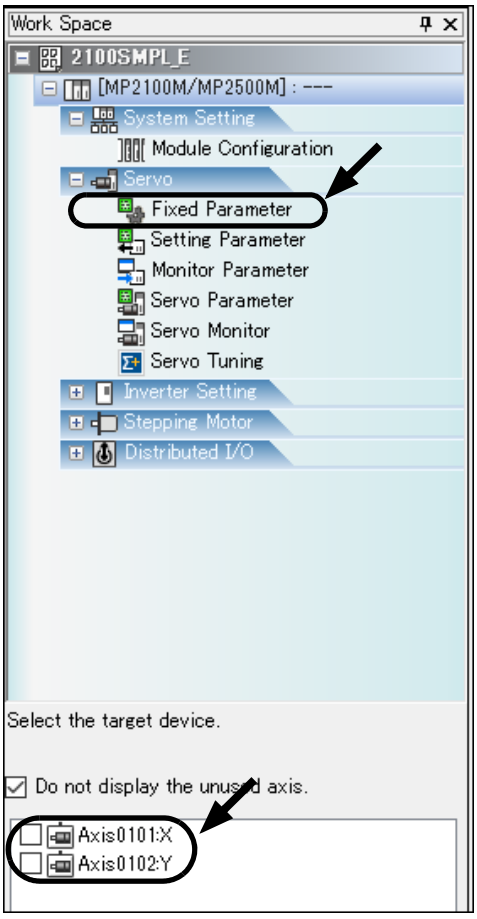
Note: To use the program, set the motion fixed parameters according to the connected system. For details on the motion fixed parameters, refer to *Machine Controller MP2000 Series Built-in SVB/SVB-01 Motion Module User's Manual* (Manual No.: SIEP C880700 33).

1. Click **Module configuration** in the main window of MPE720 Ver.7.



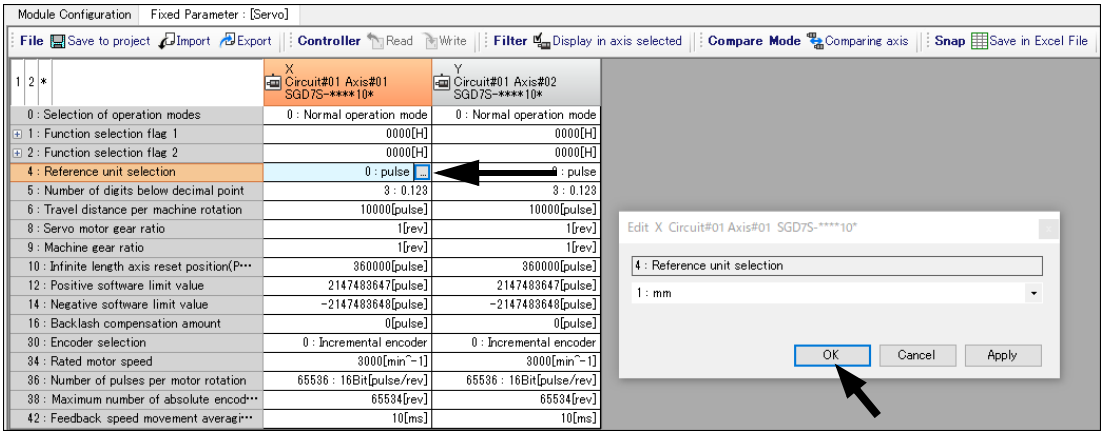
The Module Configuration Definition Window will be displayed.

2. Select the check boxes for the SERVOPACK axes for which you want to display the parameters and double-click **Fixed Parameter**.

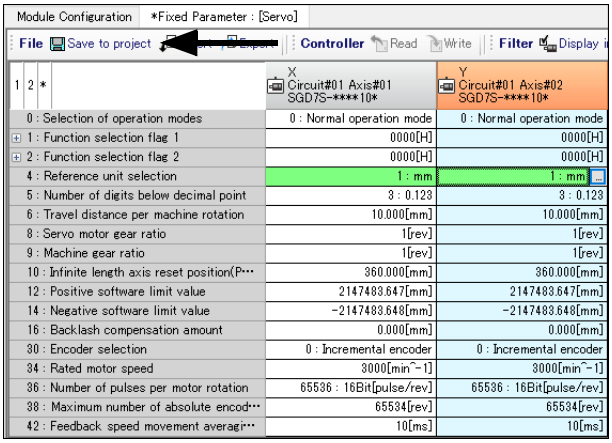


The **Fixed Parameter** tab page is displayed.

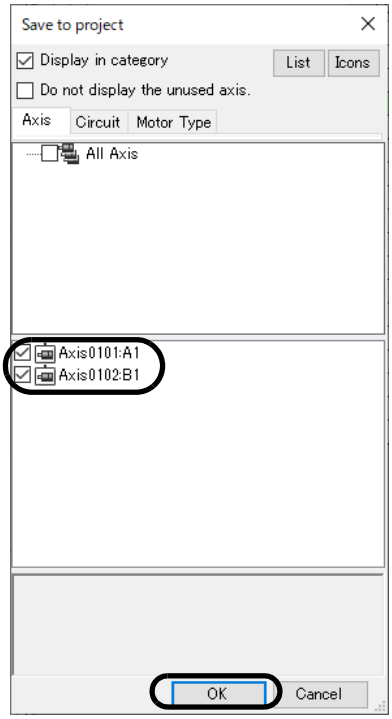
3. Set the fixed parameters.
   Click **...** for **Reference Unit Selection** of the first axis, select **mm**, and click the **OK** button. Set the second axis the same way.



4. Click **Save to project** to save the set values.



5. Select the check boxes for the SERVOPACK to save and click the **OK** button.



This completes the process for saving fixed parameter settings. Next, save the SERVOPACK parameters.

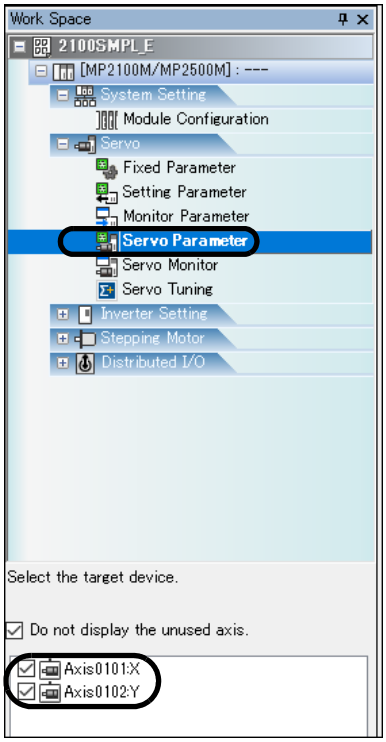
(2) Making Servo Adjustments and Saving SERVOPACK Parameters

This section explains how to make servo adjustments and save the SERVOPACK parameters for each axis to the MP2100.

1. Execute servo gain and other adjustments for each servo.

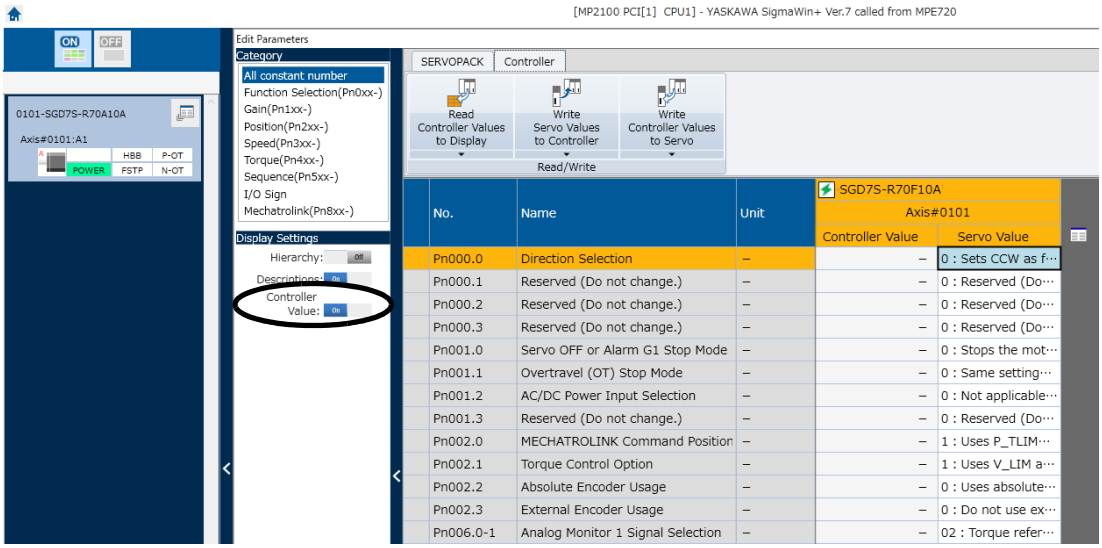
   Note: Refer to each SERVOPACK manual for information on the servo adjustments.

2. Select the check boxes for the SERVOPACK axes for which you want to display the parameters and double-click **Servo Parameter.**

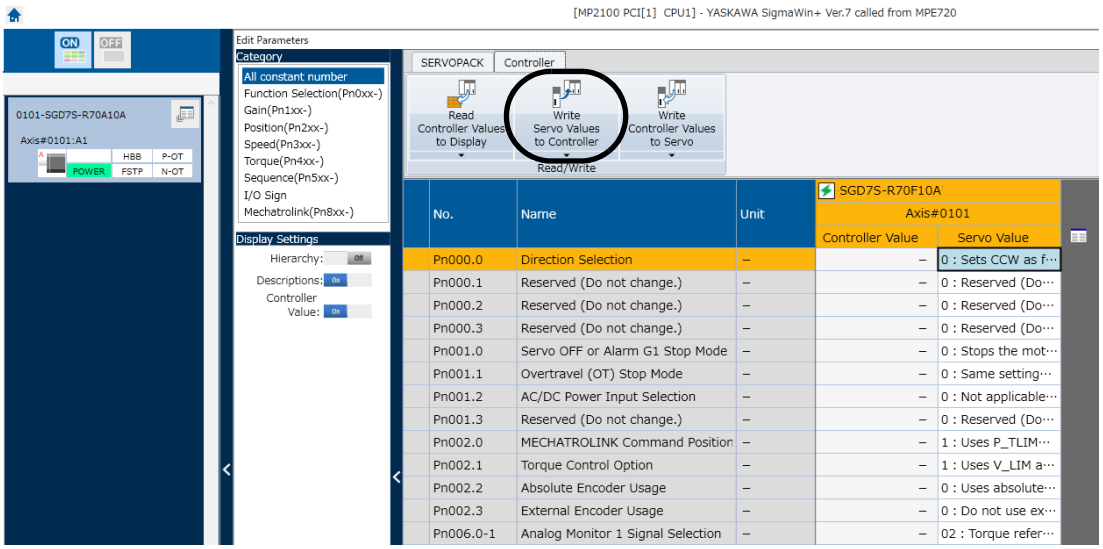


3. The SigmaWin+ is started by the SigmaWin+ linked function.

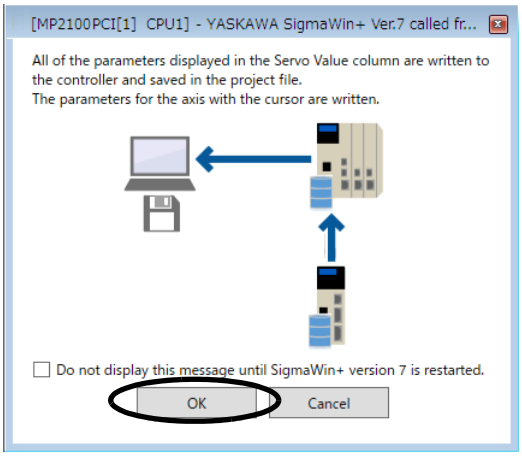
4. In the **Display Settings** area, set **Controller Value** to **ON**.



5. Select the **Controller** tab, and click **Write Servo Values to Controller**.

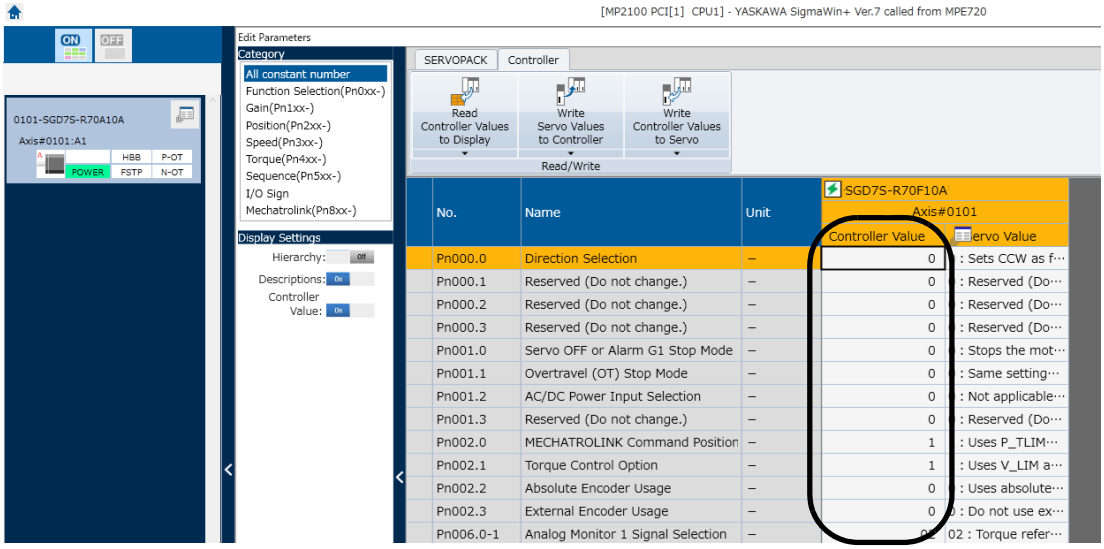


6. The following window is displayed. Click the **OK** button.



The servo values will be written to the Controller.

7. Confirm that the same values as **Servo Value** are displayed in **Controller Value**.



8. Use the same procedure as SERVOPACK axis 1 to write the values of axis 2 to the controller.
9. Switch to the Module Configuration Definition Window, and click the **Save to project** button.

This completes the adjustment of the SERVOPACK and the saving of SERVOPACK parameters.

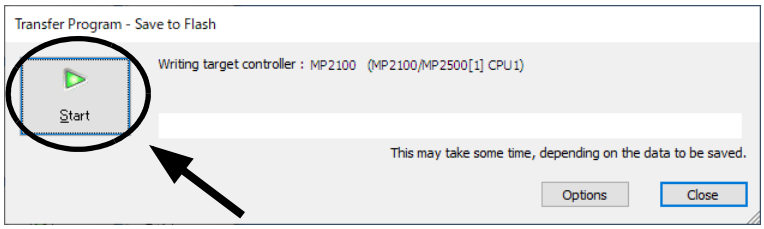
4.2.3 Saving and Transferring Data

The module configuration files that were automatically created by the MP2100 during self-configuration and edited program files can be saved in flash memory. This data can be transferred to your computer for synchronization of the data in the MP2100 with the data in the computer.

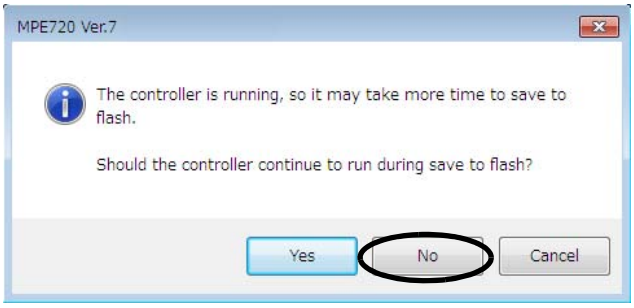
(1) Saving to Flash Memory

Use the following procedure to save the data from the MP2100 RAM to flash memory.

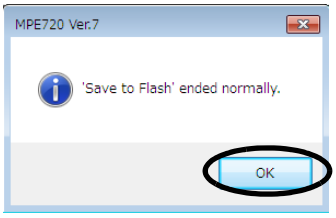
1. Select ***Online*** - ***Save to Flash*** in the menu bar of the MPE720 Ver.7 main window.
   The **Transfer Program - Save to Flash** dialog box appears.
2. Check to see that the target controller displayed in the dialog box is correct, and then click the **Start** button.



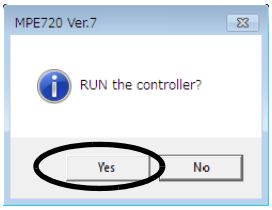
   A message box appears and ask whether or not to stop the MP2100 while saving data to the flash memory.
3. Click the **No** button to stop the MP2100 and start saving the data to the flash memory.



   When the data has been successfully saved in the flash memory, a message box appears to inform you that saving process is complete.
4. Click the **OK** button in the message box.



   Another message box appears asking whether or not to run the MP2100.
5. Click the **Yes** Button to run the MP2100.

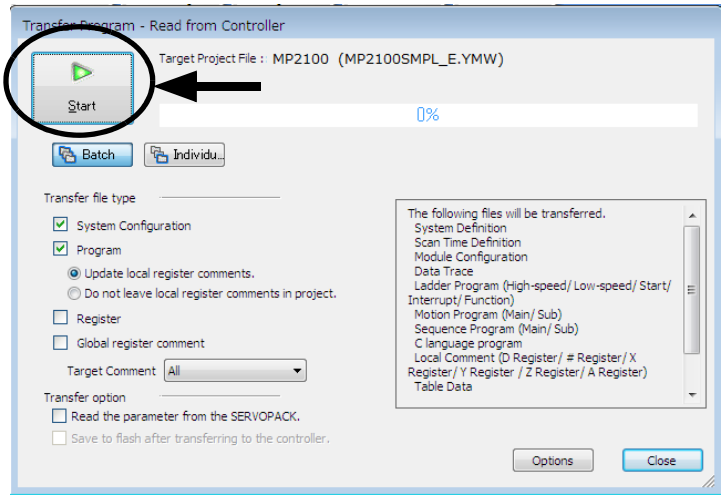


The data has been saved in the flash memory.

(2) Transferring Data from the MP2100 to the Computer

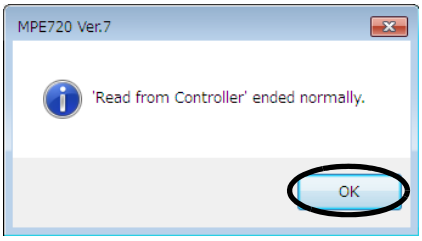
Use the following procedure to transfer data from the MP2100 RAM to the computer.

1. Select ***Online*** - ***Read from Controller*** from the menu bar of the MPE720 Ver.7 main window.
   The **Transfer Program - Read from Controller** dialog box appears.
2. Check the selected items and displayed contents to see if they are correct, and then click the **Start** button.



The project file (sample file) saved in the computer will be overwritten with the data transferred from the MP2100. When the overwriting is complete, a message box appears.

3. Click the **OK** button.



This completes the transfer of data from the MP2100 to the computer.

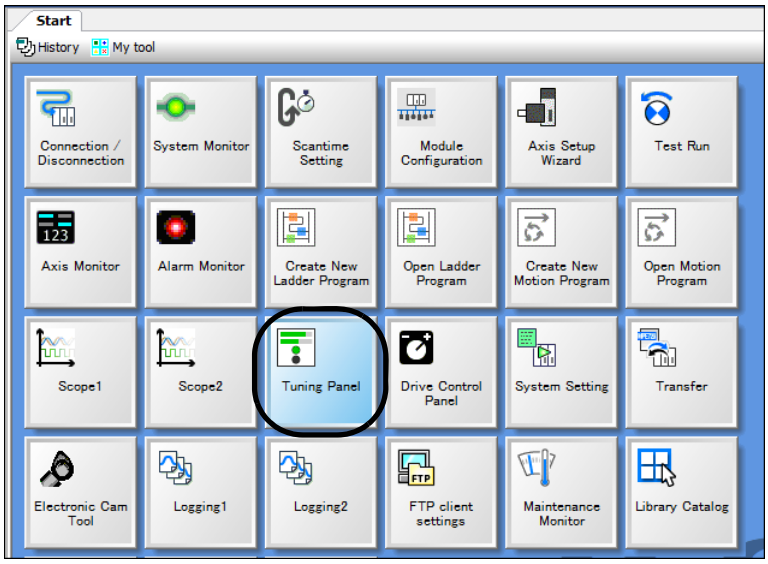
4.3 Checking Sample Program Operation

This section explains how to check four operations in the model system applying sample programs started in 4.2 by using the **Tuning Panel** window.

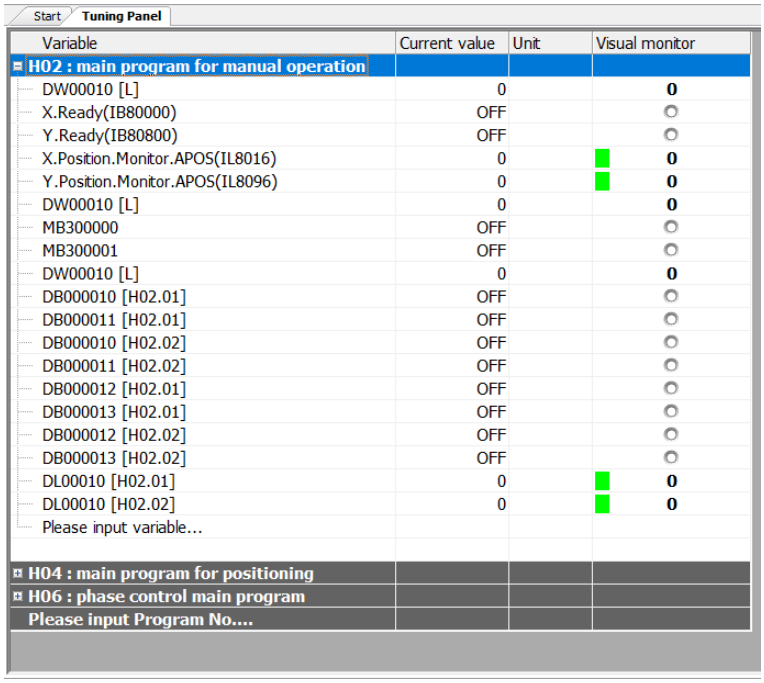
4.3.1 Opening the Tuning Panel Window

Use the following procedure to open the **Tuning Panel** window.

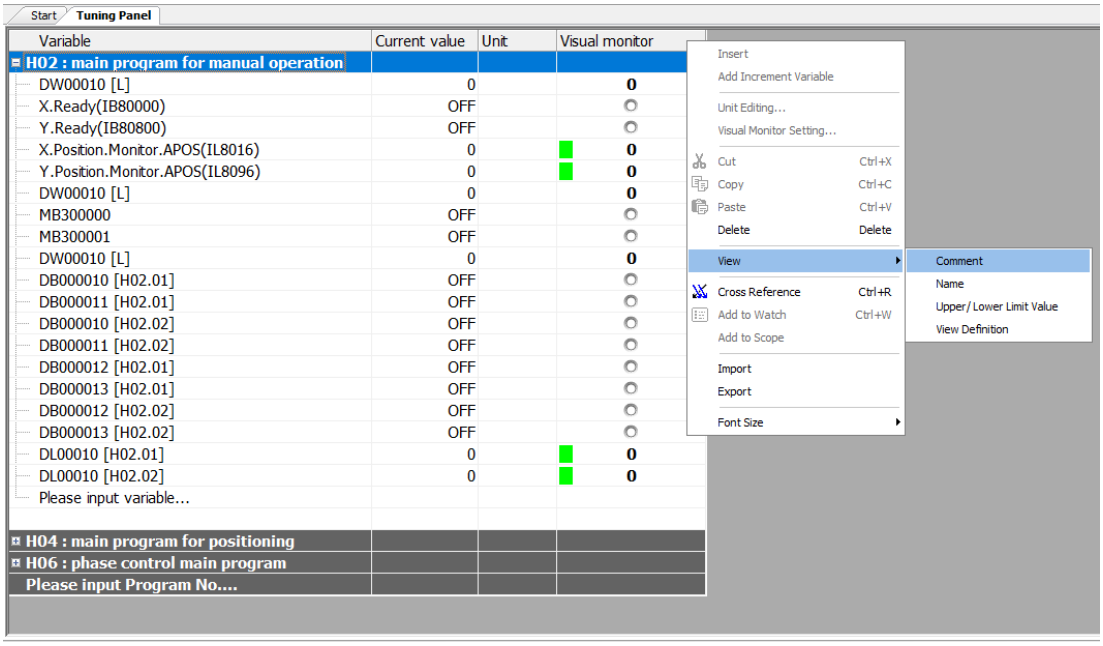
1. Start up MPE720 Ver.7, and click **Tuning Panel** in the **My tool** window.

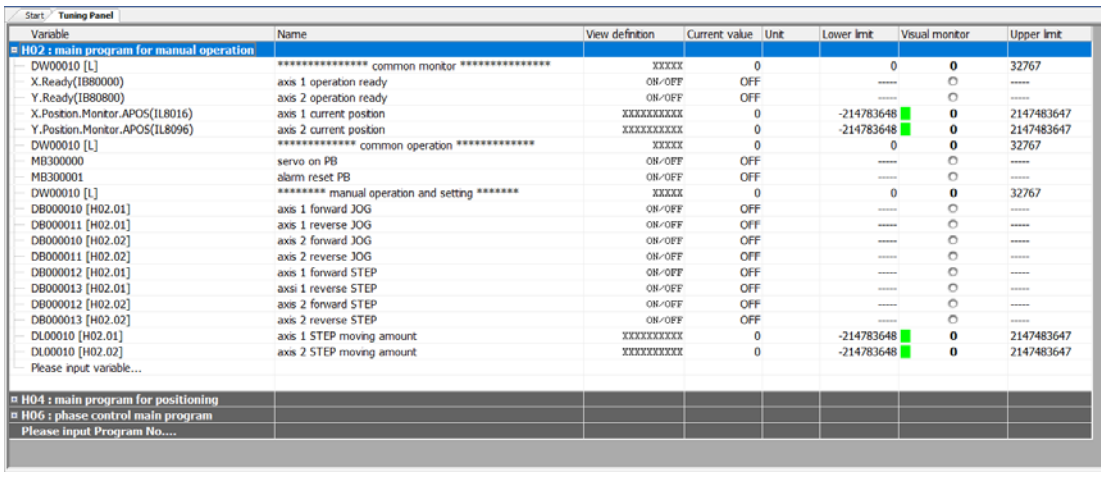


The **Tuning Panel** window appears.



2. Right-click on the Tuning Panel and select **View**.
   Select **Comment**, **Name**, **Upper/Lower Limit Value**, and **View Definition** as required.

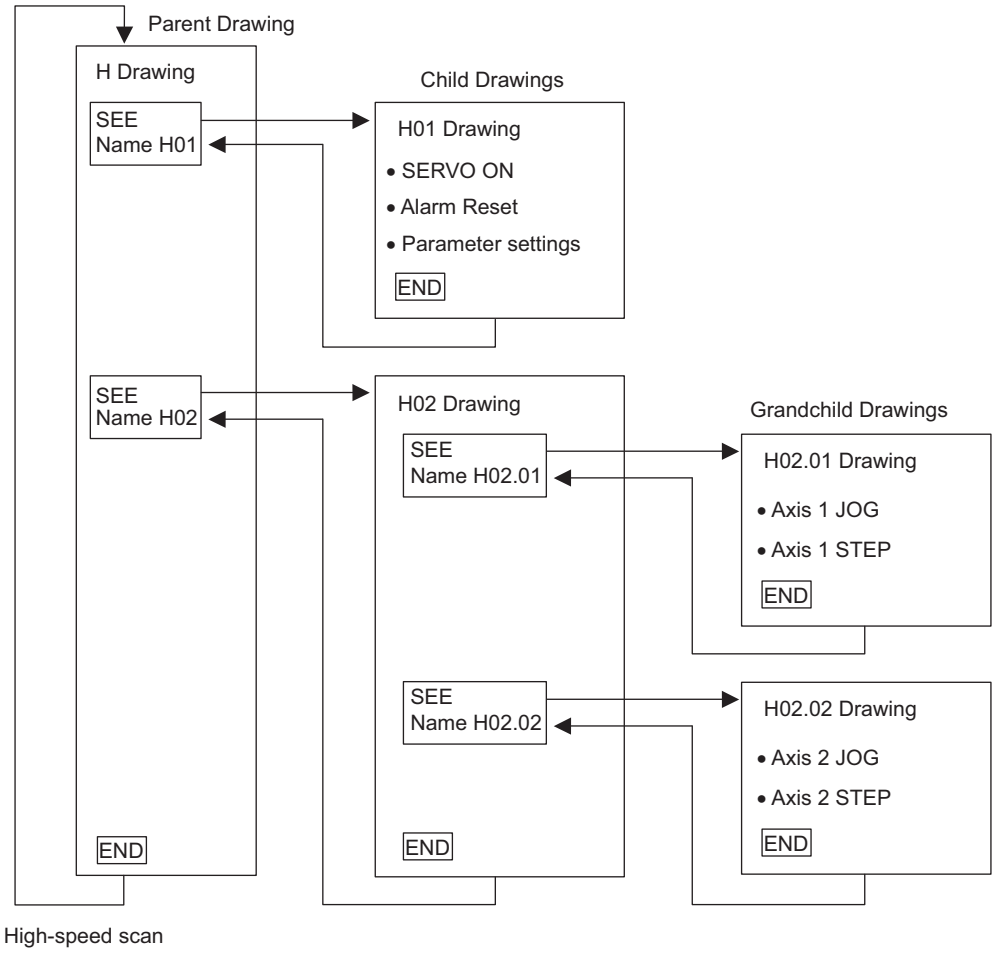




4.3.2 Operation Check 1: Manual Operation

(1) Program Outline

This section explains how to execute JOG and STEP operations for Servomotor 1 or 2 (axis 1 or 2) using a ladder program such as the one shown below.

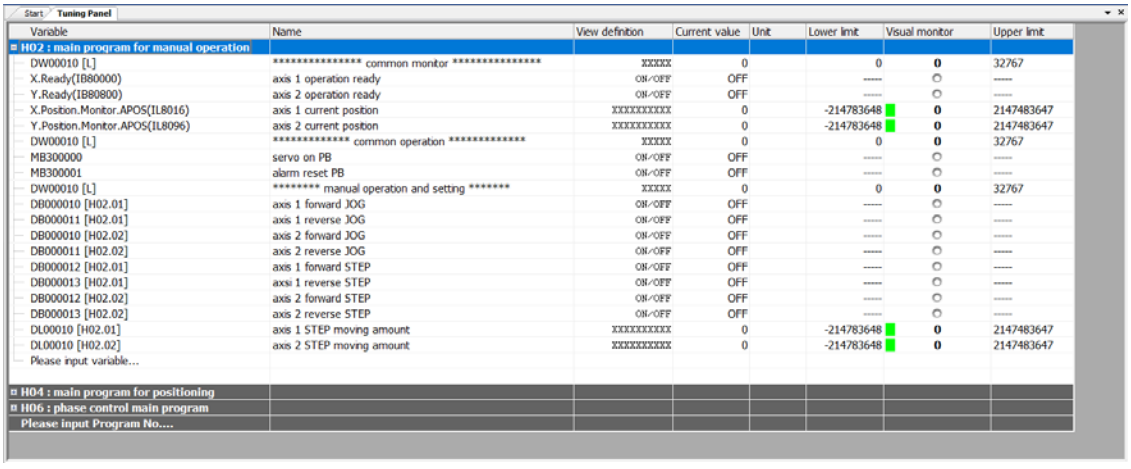


Note: 1. Refer to *4.3.2 (4) Sample Program Details* for details of each program (drawing).
2. A simple device is used in this example to describe the MP2100 system startup.
3. This device has no power OFF circuit for the SERVOPACK in the event of emergency stops or overtravel. Include a proper emergency stop circuit in actual devices.

(2) H02 Drawing Tuning Panel

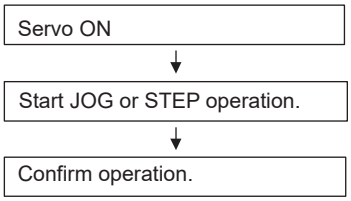
Display the H02 Drawing Tuning Panel as shown in *4.3.1 Opening the Tuning Panel Window*.

Model system operation can be controlled by writing the current values for **Common operation** and **Manual Operation and Setting** from the Tuning Panel.



(3) Procedure

Use the following procedure to confirm operation.



The following table gives an outline of the operation when the **Tuning Panel** window is used.

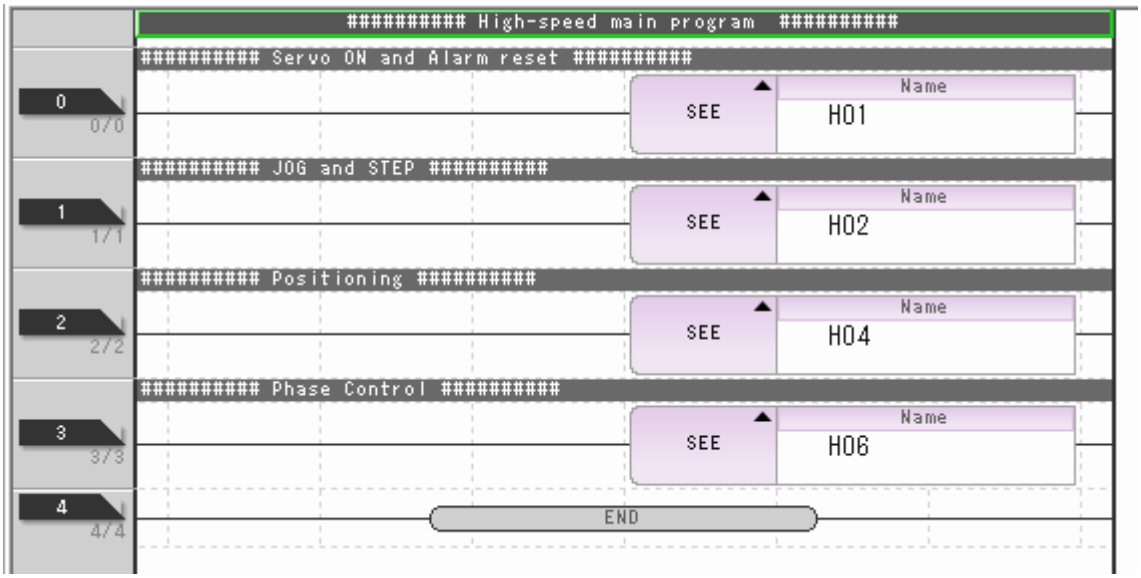
Data Name	Current Value Operation	Operation Outline
Servo ON PB	OFF → ON	The servomotor turns ON and the Servo will be clamped.
	ON → OFF	Servo turned OFF.
Axis 1 Forward JOG	OFF → ON	Axis 1 rotates forward.
	ON → OFF	Axis 1 stops.
Axis 1 Reverse JOG	OFF → ON	Axis 1 rotates in reverse direction.
	ON → OFF	Axis 1 stops.
Axis 2 Forward JOG	OFF → ON	Axis 2 rotates forward.
	ON → OFF	Axis 2 stops.
Axis 2 Reverse JOG	OFF → ON	Axis 2 rotates in reverse direction.
	ON → OFF	Axis 2 stops.
Axis 1 Forward STEP	OFF → ON	Axis 1 starts rotating forward for the moving amount set under Axis 1 STEP moving amount.
	ON → OFF	STEP operation stops. Always stop after executing stepping.
Axis 1 Reverse STEP	OFF → ON	Axis 1 starts rotating in reverse for the moving amount set under Axis 1 STEP moving amount.
	ON → OFF	STEP operation stops. Always stop after executing stepping.
Axis 2 Forward STEP	OFF → ON	Axis 2 starts rotating forward for the moving amount set under Axis 2 STEP moving amount.
	ON → OFF	STEP operation stops. Always stop after executing stepping.
Axis 2 Reverse STEP	OFF → ON	Axis 2 starts rotating in reverse for the moving amount set under Axis 2 STEP moving amount.
	ON → OFF	STEP operation stops. Always stop after executing stepping.
Axis 1 STEP Moving Amount	Enter any value.	Sets the STEP moving amount for axis 1.
Axis 2 STEP Moving Amount	Enter any value.	Sets the STEP moving amount for axis 2.

Note: Create routines within the actual application program in order to monitor and control the registers corresponding to the signals and data listed in the table above.

(4) Sample Program Details

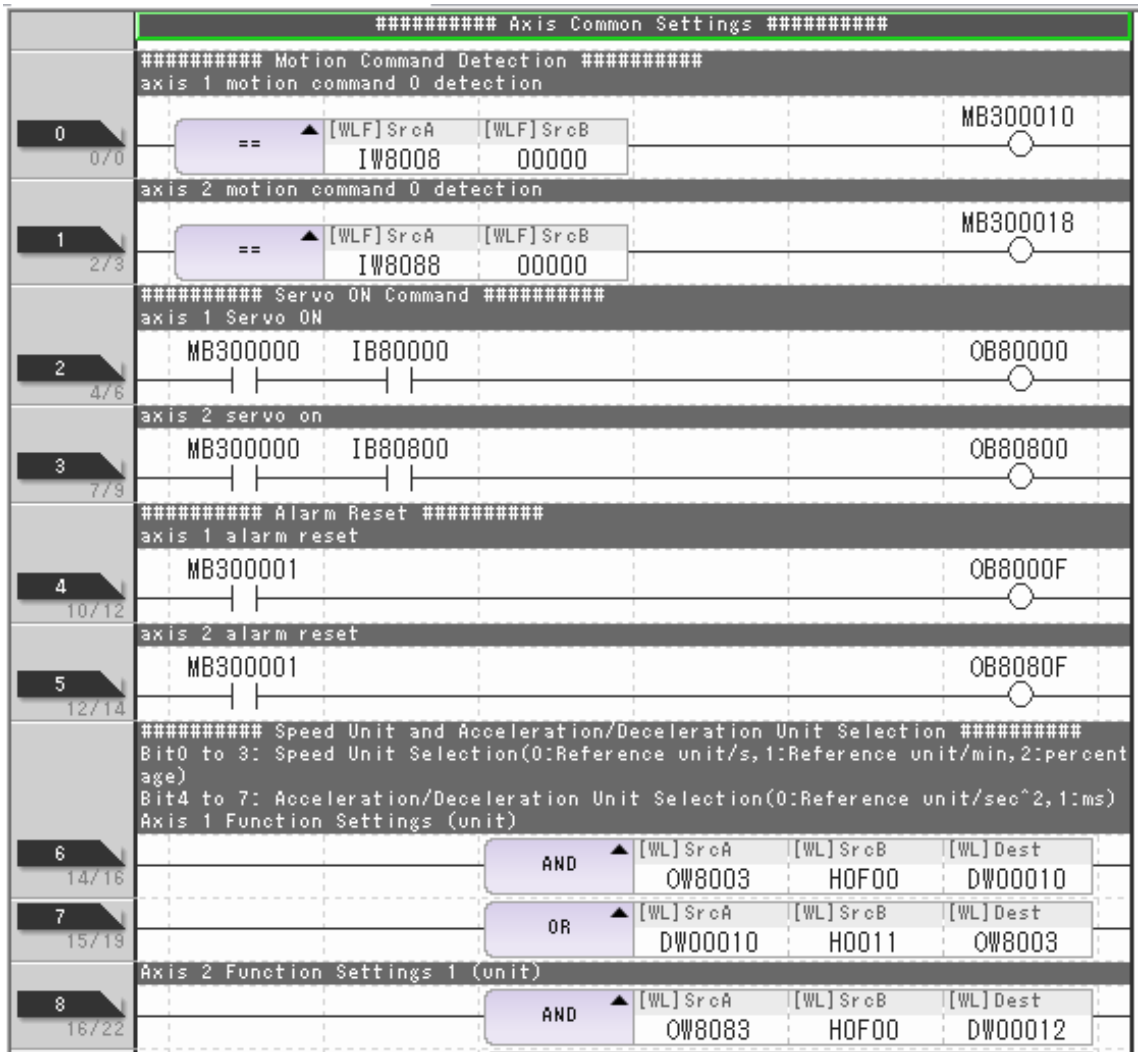
(a) H Drawing

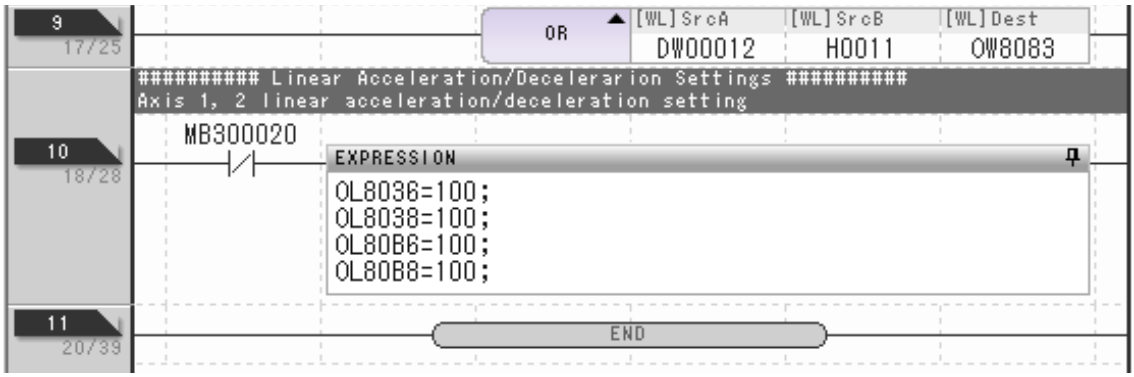
The H parent drawing controls the overall sample program.



(b) H01 Drawing

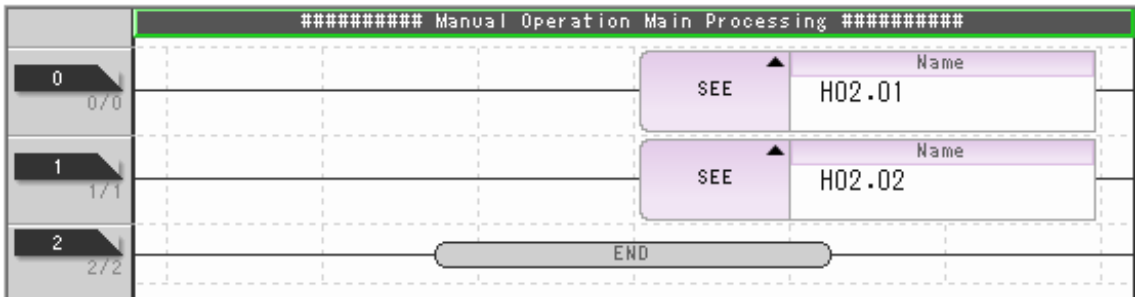
The H01 child drawing turns ON the Servo, resets alarms, and sets common parameters.





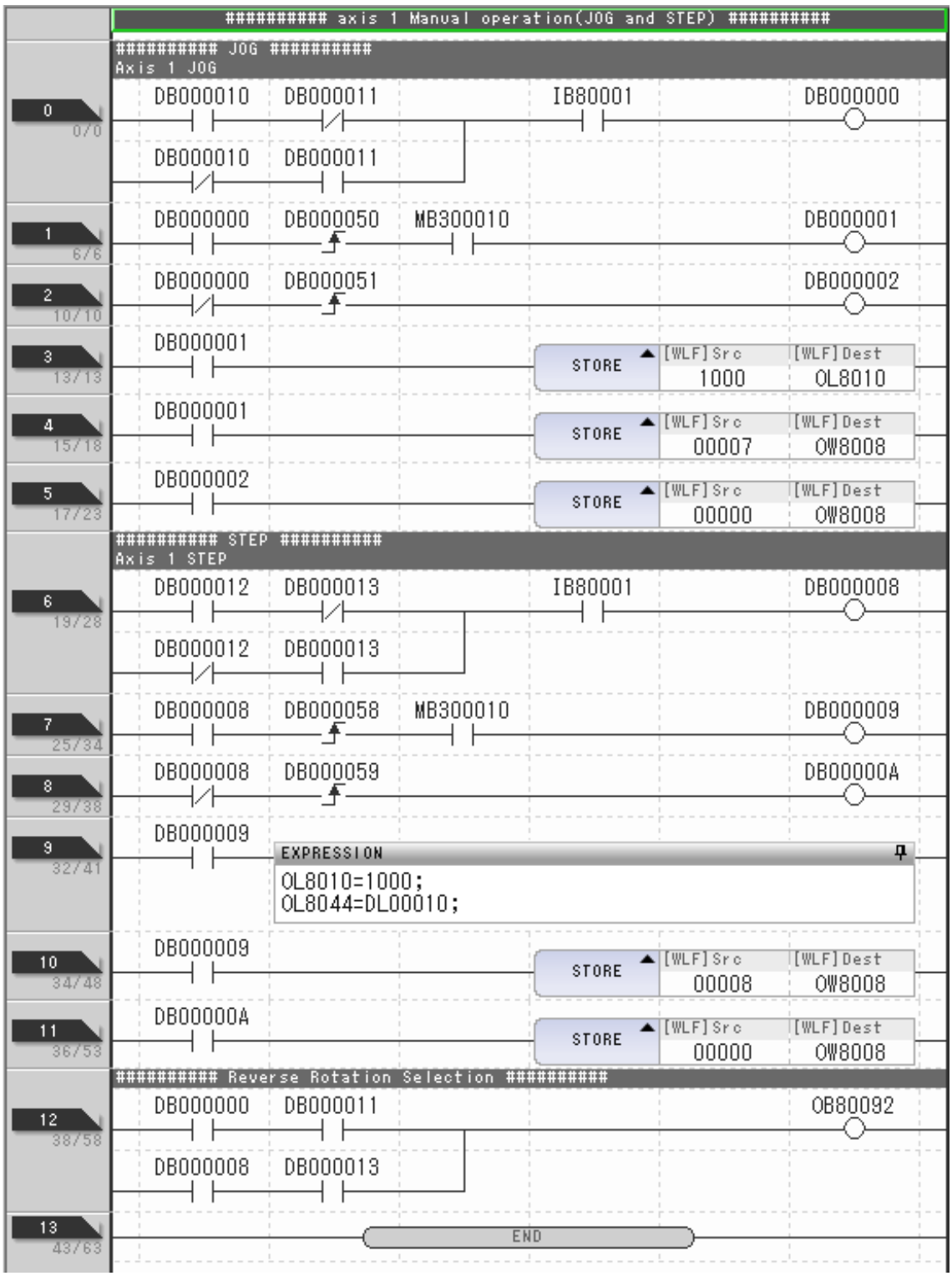
(c) H02 Drawing

The H02 child drawing controls JOG and STEP operation.



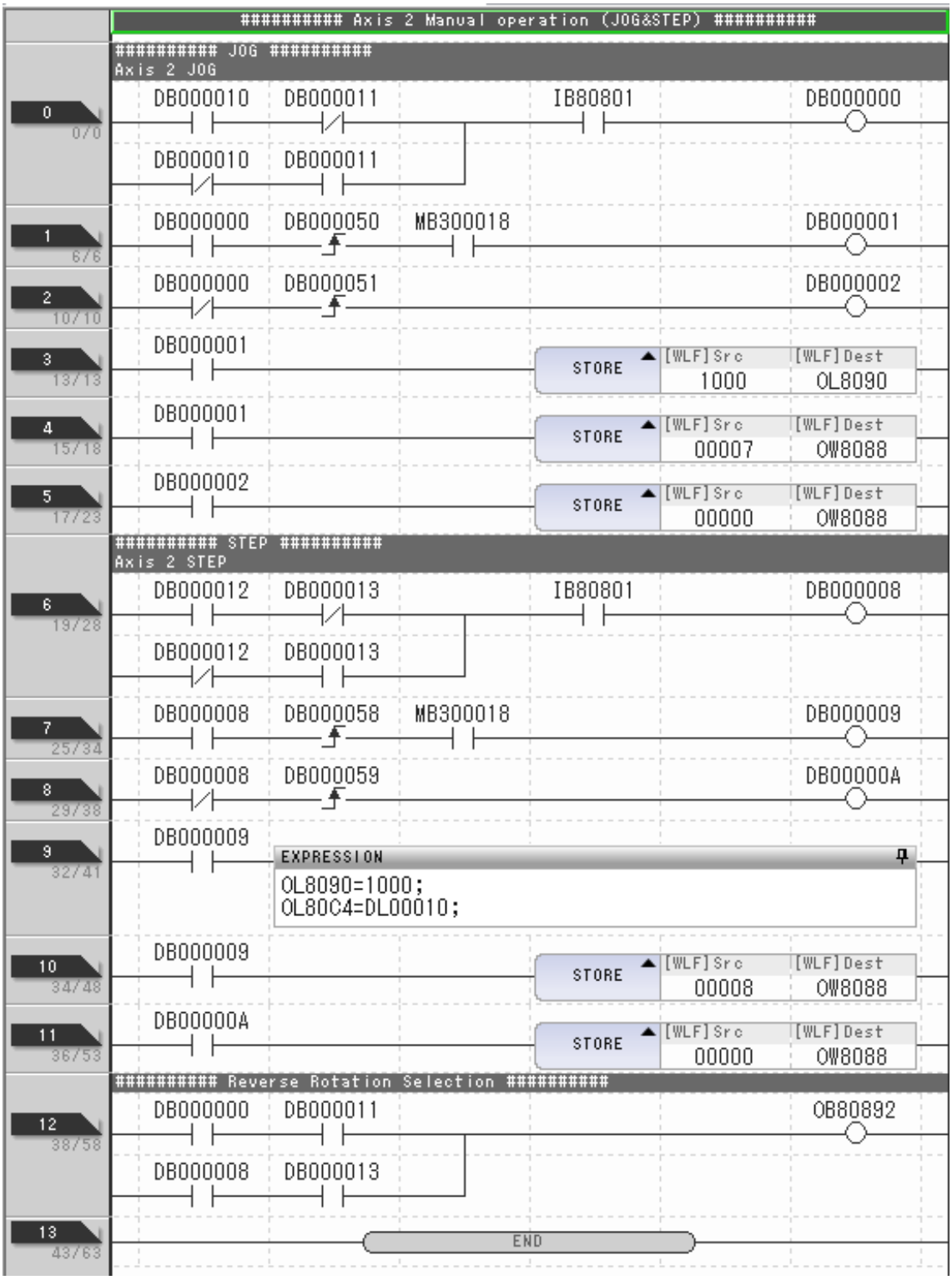
(d) H02.01 Drawing

The H02.01 grandchild drawing controls JOG and STEP operation for axis 1.



(e) H02.02 Drawing

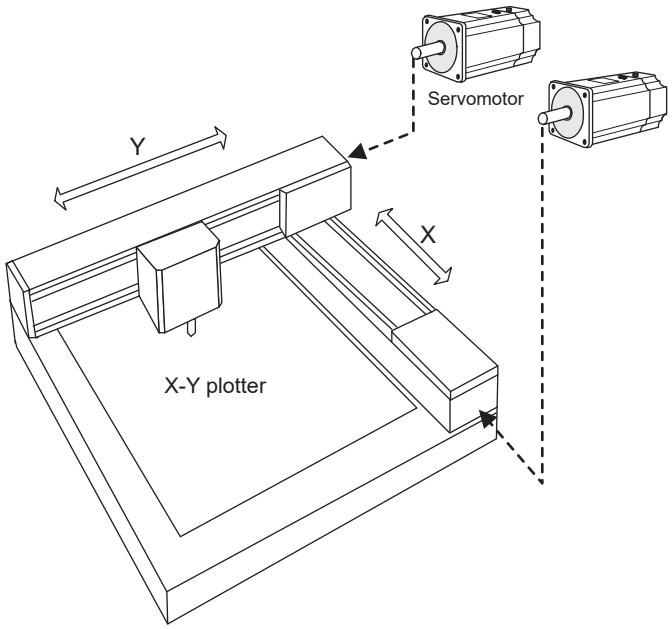
The H02.02 grandchild drawing controls JOG and STEP operation for axis 2.



4.3.3 Operation Check 2: Position Control

(1) Operation Outline

In this example, an X-Y plotter like the one shown in the figure is operated by ladder and motion programs.



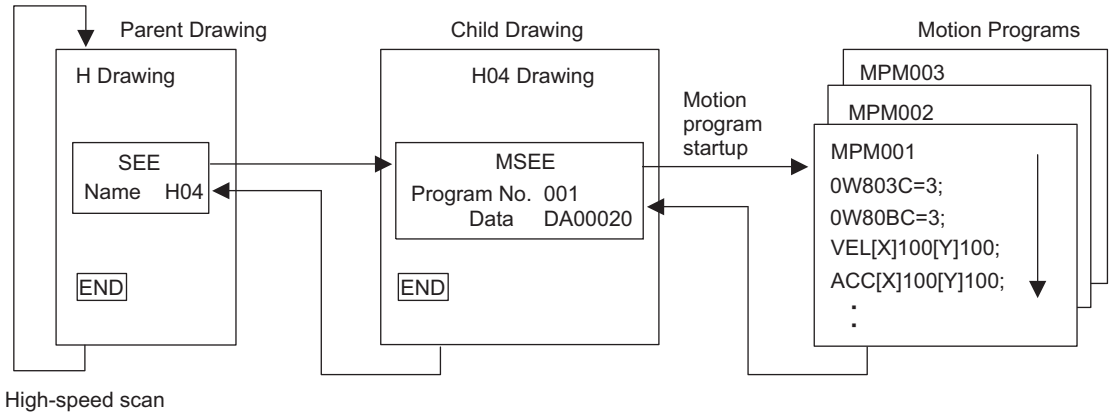
(2) Program Outline

Use the ladder program (H04 Drawing) and three prepared sample programs (MPM001, MPM002, and MPM003) to check the operation, as shown in the figure.

Programs MPM001 to MPM003 perform the following operations.

- MPM001: Zero point return using Servomotor phase-C
- MPM002: Axis 2 positioning and interpolation (with interval timer)
- MPM003: Axis 2 positioning and interpolation (without interval timer)

Motion programs are written in text format, and the listed commands and operations are executed in listed order.

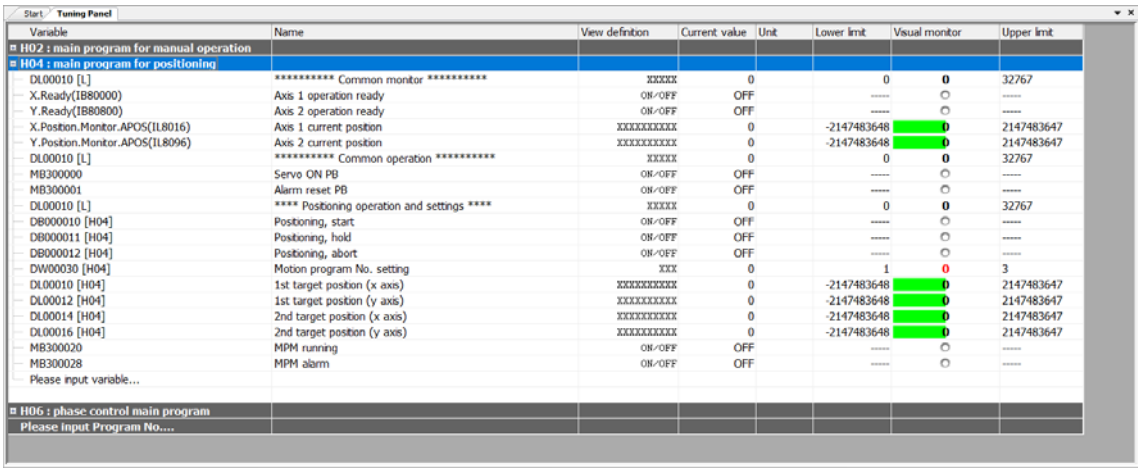


Note: 1. Refer to *4.3.3 (5) Sample Program Details* for details of each program.
2. A simple device is used in this example to describe the MP2100 system startup.
3. This device has no power OFF circuit for the SERVOPACK in the event of emergency stops or overtravel. Include a proper emergency stop circuit in actual devices.

(3) H04 Drawing Tuning Panel

Display the H04 Drawing Tuning Panel as shown in *4.3.1 Opening the Tuning Panel Window*.

Model system operation can be controlled by writing the current values for **Common operation** and **Positioning operation and settings** from the Tuning Panel.



Variable	Name	View definition	Current value	Unit	Lower limit	Visual monitor	Upper limit
H02 : main program for manual operation							
H04 : main program for positioning							
DL00010 [L]	********** Common monitor **********	XXXXX	0		0	0	32767
X.Ready(IB80000)	Axis 1 operation ready	ON/OFF	OFF		-----		-----
Y.Ready(IB80800)	Axis 2 operation ready	ON/OFF	OFF		-----		-----
X.Position.Monitor.APOS(IL8016)	Axis 1 current position	XXXXXXXXXX	0		-2147483648	0	2147483647
Y.Position.Monitor.APOS(IL8096)	Axis 2 current position	XXXXXXXXXX	0		-2147483648	0	2147483647
DL00010 [L]	********** Common operation **********	XXXXX	0		0	0	32767
MB300000	Servo ON PB	ON/OFF	OFF		-----		-----
MB300001	Alarm reset PB	ON/OFF	OFF		-----		-----
DL00010 [L]	**** Positioning operation and settings ****	XXXXX	0		0	0	32767
DB000010 [H04]	Positioning, start	ON/OFF	OFF		-----		-----
DB000011 [H04]	Positioning, hold	ON/OFF	OFF		-----		-----
DB000012 [H04]	Positioning, abort	ON/OFF	OFF		-----		-----
DW00030 [H04]	Motion program No. setting	XXX	0		1	0	3
DL00010 [H04]	1st target position (x axis)	XXXXXXXXXX	0		-2147483648	0	2147483647
DL00012 [H04]	1st target position (y axis)	XXXXXXXXXX	0		-2147483648	0	2147483647
DL00014 [H04]	2nd target position (x axis)	XXXXXXXXXX	0		-2147483648	0	2147483647
DL00016 [H04]	2nd target position (y axis)	XXXXXXXXXX	0		-2147483648	0	2147483647
MB300020	MPM running	ON/OFF	OFF		-----		-----
MB300028	MPM alarm	ON/OFF	OFF		-----		-----
Please input variable...							
H06 : phase control main program							
Please input Program No....							

(4) Procedure

Use the following procedure to operate the Tuning Panel and check operation.

1. Servo ON

   Change the **Current Value** for **Servo ON PB** from **OFF** to **ON**.

   The servomotor turns ON and the Servo will be clamped.

2. Motion program No. setting

   Enter a value from **1** to **3** as the **Current Value** for the **Motion Program No. setting** to specify the motion program to be executed.

   001 (=MPM001): Program for executing zero point return using servomotor phase-C. When this program is executed, X axis (axis 1) and Y axis (axis 2) rotate a set distance once a phase-C pulse has been input, and then return home.

   002 (= MPM002): Repeats the following two operations with an interval time.

   1.) Moves to target position 1 in incremental mode, performs linear interpolation to target position 2, and then repeats this operation 5 times.

   2.) In absolute mode, performs counterclockwise circular interpolation from current position to home (0,0) (once).

   003 (= MPM003): Repeats the same operation as MPM002, but without an interval time.

   Note: No programs have been created for numbers **004** and higher. An MPM alarm will occur if **004** or higher numbers are entered and operation is started.

3. Set Target Position 1 and Target Position 2

   Enter any value for the following settings. These settings determine the target position for positioning when Motion Program No. 2 and No. 3 are executed.

   1st target position (X axis)

   1st target position (Y axis)

   2nd target position (X axis)

   2nd target position (Y axis)

4. Positioning start

   Change the **Current Value** for **Positioning start** from **OFF** to **ON**.

   The program is executed and the model system operates according to the motion program set in step 2. Once the operation has been checked, input **OFF** to stop the system.

   When a motion program starts, the **Current Value** for **MPM running** changes to **ON**. Also, when the Servo axis rotates, all **Current Values** changes.

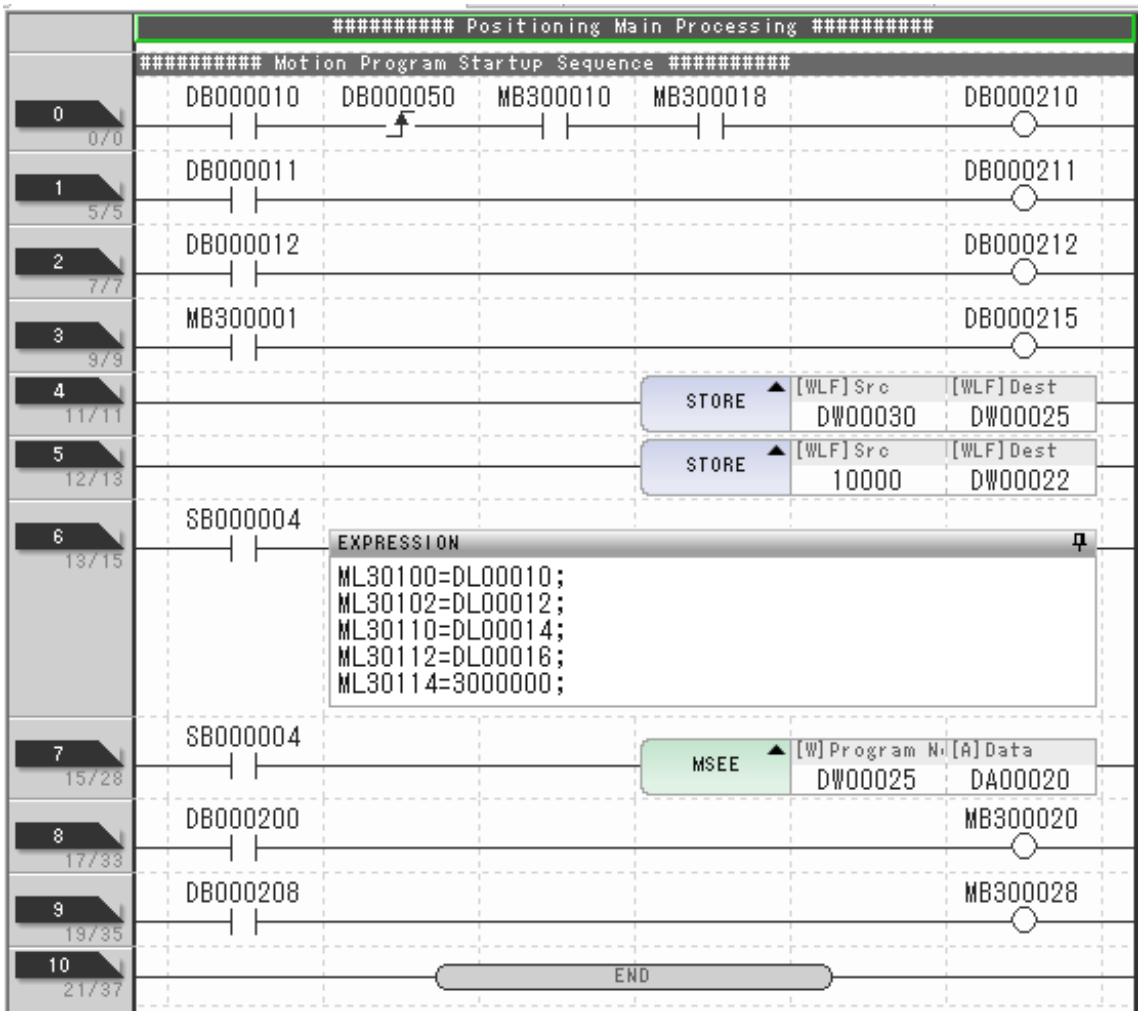
Note: 1. If an error occurs during motion program execution, the current value for **MPM alarm** in the Tuning Panel changes to **ON**. Use the following procedure to clear alarms.
   1. Set the **Current Value** for **Positioning abort** to **ON** then **OFF**.
   2. Set the **Current Value** for **Alarm reset PB** to **ON** then **OFF**.

2. Create routines within the actual application program in order to monitor and control the registers corresponding to the signals and data listed in the table above.
3. The register numbers that correspond to the signals used in this sample program are the register numbers displayed under **REG-No**. next to **DWG** at the right of the **Tuning Panel** window.

(5) Sample Program Details

(a) H04 Drawing

The H04 child drawing shows the ladder program for managing and controlling MPM motion programs.



(b) Motion Program MPM001

The MPM001 motion program uses the Servomotor phase-C pulse to perform home return.

```
00001  "MPM001";
00002  OW803C=3;                "X axis home return method selection (3: Phase C)"
00003  OW80BC=3;                "Y axis home return type selection (3: Phase C)"
00004  VEL [X]1000 [Y]1000;     "Travel speed setting for positioning command"
00005  ACC[X]100[Y]100;         "Acceleration time setting"
00006  DCC[X]100[Y]100;         "Deceleration time setting"
00007  OW803E=100;              "X axis approach speed (mm/min)"
00008  OW8040=50;               "X axis creep speed (mm/min)"
00009  OL8042=10000;            "X axis final travel distance (0.001 mm)"
00010  OW80BE=100;              "Y axis approach speed (mm/min)"
00011  OW80C0=50;               "Y axis creep speed (mm/min)"
00012  OL80C2=10000;            "Y axis final travel speed (0.001 mm)"
00013  ZRN[X]00[Y]00;           "Home return command"
00014  END;
```

(6) Motion Programs MPM002 and MPM003

Motion programs MPM002 and MPM003 perform positioning, linear interpolation, and circular interpolation for two axes.

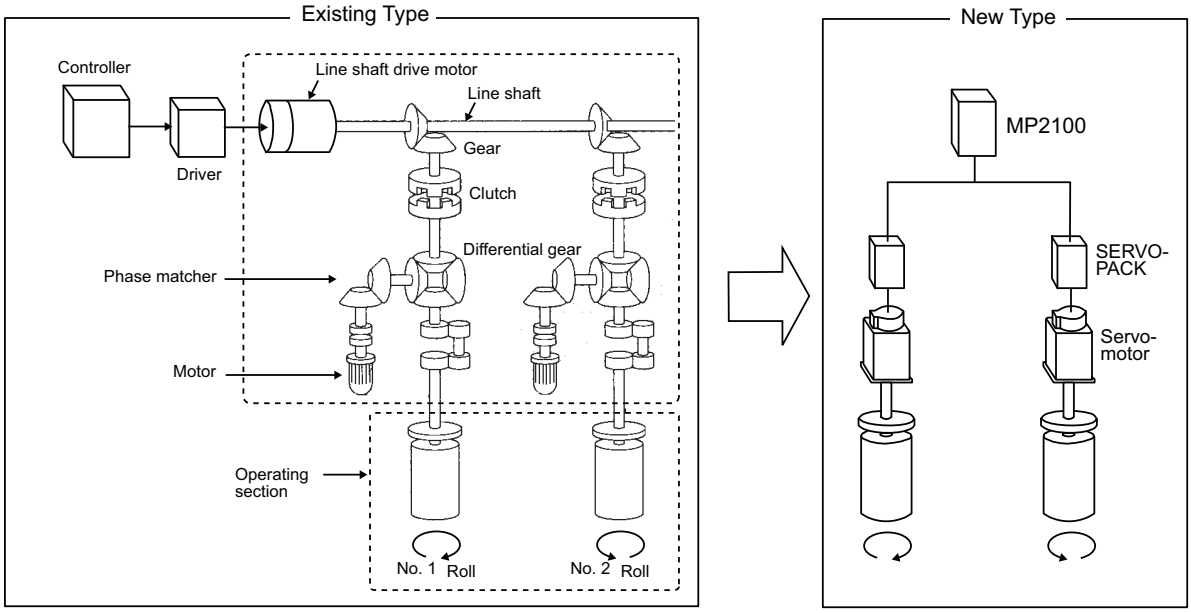
MPM002 inserts a timer command between each travel command to indicate operation divisions. MPM003 continuously executes travel commands, without the timer commands of MPM002, as shown in the following figure.

```
00001  "MPM002";
00002  "Data Setting";
00003  VEL [X]1000 [Y]1000;          "Travel speed setting for positioning command"
00004  FMX T50000000;                "Composite speed upper limit setting for interpolation command"
00005  IAC T500;                     "Acceleration time setting for interpolation command"
00006  IDC T500;                     "Deceleration time setting for interpolation command"
00007  PLN [X][Y];                   "Plane specification for circular interpolation command"
00008  INC;                          "Increment position command setting"
00009  TIM T100;
00010  "Repeat Operation"
00011  DW10 =0;
00012  WHILE DW10 <5;                "No. repeats = 5"
00013  MOV [X]ML30100 [Y]ML30102 ; "Positioning command"
00014  TIM T100;
00015  MVS [X]ML30110 [Y]ML30112 FML30114; "Linear interpolation command"
00016  TIM T100;
00017  ABS;                          "Absolute positioning command setting"
00018  MCC [X]0 [Y]0 R1000.0 FML30114; "Circular interpolation command"
00019  TIM T100;
00020  DW10 =DW10 +1;
00021  WEND;
00022  "End Repeat Operation";
00023  END;
```

4.3.4 Operation Check 3: Phase Control - Electronic Shaft

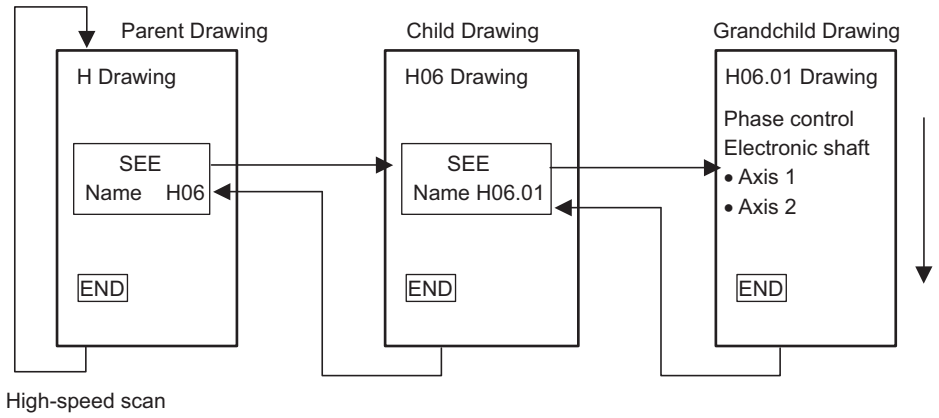
(1) Machine Outline

As shown in the following figure, the servomotor performs the same operation as rolls No. 1 and No. 2 connected to the line shaft. No phase matching, however, is used.



(2) Program Outline

Use the ladder program (H06.01 Drawing) to check the above operation. The two axes synchronize to a virtual master axis according to the entered speed settings, and axis 1 and axis 2 rotate in exactly the same way.

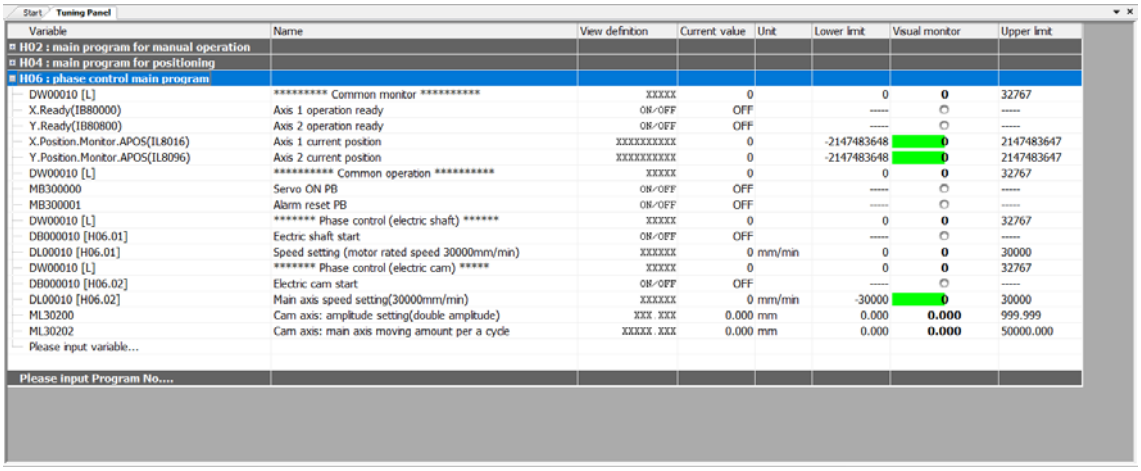


Note: 1. Refer to *4.3.4 (5) Sample Program Details* for details of H06.01 Drawing.
2. A simple device is used in this example to describe the MP2100 system startup.
3. This device has no power OFF circuit for the SERVOPACK in the event of emergency stops or overtravel. Include a proper emergency stop circuit in actual devices.

(3) H06 Drawing Tuning Panel

Display the H06 Drawing Tuning Panel as shown in *4.3.1 Opening the Tuning Panel Window*.

Model system operation can be controlled by writing the current values for **Common operation** and **Phase control (electric shaft)** from the Tuning Panel.



Variable	Name	View definition	Current value	Unit	Lower limit	Visual monitor	Upper limit
H02 : main program for manual operation							
H04 : main program for positioning							
H06 : phase control main program							
DW00010 [L]	********* Common monitor **********	XXXXX	0		0	0	32767
X.Ready(IB80000)	Axis 1 operation ready	ON/OFF	OFF		-----		-----
Y.Ready(IB80800)	Axis 2 operation ready	ON/OFF	OFF		-----		-----
X.Position.Monitor.APOS(IL8016)	Axis 1 current position	XXXXXXXXX	0		-2147483648	0	2147483647
Y.Position.Monitor.APOS(IL8096)	Axis 2 current position	XXXXXXXXX	0		-2147483648	0	2147483647
DW00010 [L]	********** Common operation **********	XXXXX	0		0	0	32767
MB300000	Servo ON PB	ON/OFF	OFF		-----		-----
MB300001	Alarm reset PB	ON/OFF	OFF		-----		-----
DW00010 [L]	******* Phase control (electric shaft) ******	XXXXX	0		0	0	32767
DB000010 [H06.01]	Electric shaft start	ON/OFF	OFF		-----		-----
DL00010 [H06.01]	Speed setting (motor rated speed 30000mm/min)	XXXXXX	0	mm/min	0	0	30000
DW00010 [L]	******* Phase control (electric cam) *****	XXXXX	0		0	0	32767
DB000010 [H06.02]	Electric cam start	ON/OFF	OFF		-----		-----
DL00010 [H06.02]	Main axis speed setting(30000mm/min)	XXXXXX	0	mm/min	-30000	0	30000
ML30200	Cam axis: amplitude setting(double amplitude)	XXX.XXX	0.000	mm	0.000	0.000	999.999
ML30202	Cam axis: main axis moving amount per a cycle	XXXXX.XXX	0.000	mm	0.000	0.000	50000.000
Please input variable...							
Please input Program No....							

(4) Procedure

Use the following procedure to operate the Tuning Panel and check operation.

1. Servo ON

   Change the **Current Value** for **Servo ON PB** from **OFF** to **ON**.

   The servomotor turns ON and the Servo is clamped.

2. Electronic Shaft Start

   Change the **Current Value** for **Electric shaft start** from **OFF** to **ON**.

   The mode changes to the phase control (electronic shaft) mode. Enter **OFF** to exit phase control (electric shaft) mode.

3. Enter Speed Settings

   Enter any value within the setting range (0 to 30000) as the **Current Value** of **Speed setting (motor rated speed 30,000 mm/min)**.

   This operation synchronizes the speed for both axes to the speed of the virtual master axis and operation starts.

   Change the **Current Value** for **Electric shaft start** from **ON** to **OFF** when the check operation has been completed.

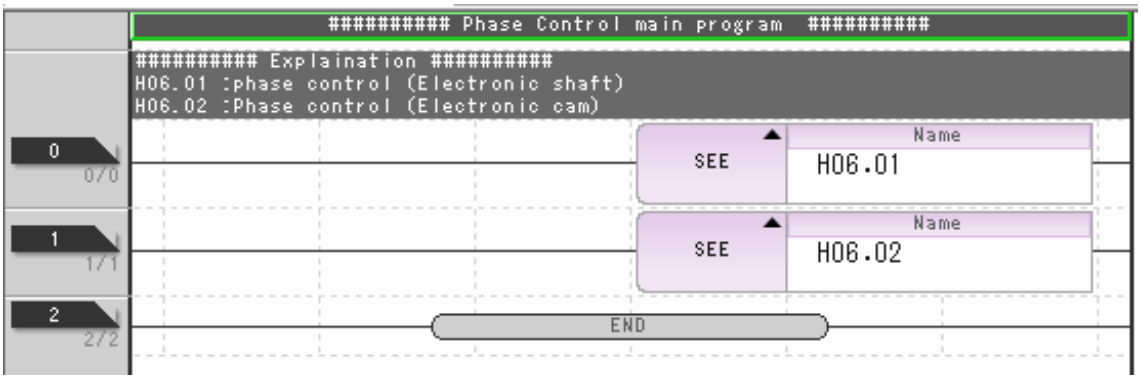
   Note: 1. Create routines within the actual application program in order to monitor and control the registers corresponding to the signals and data listed in the table above.

   2. The register numbers that correspond to the signals used in this sample program are the register numbers displayed under **REG-No**. next to **DWG** at the right of the **Tuning Panel** window.

(5) Sample Program Details

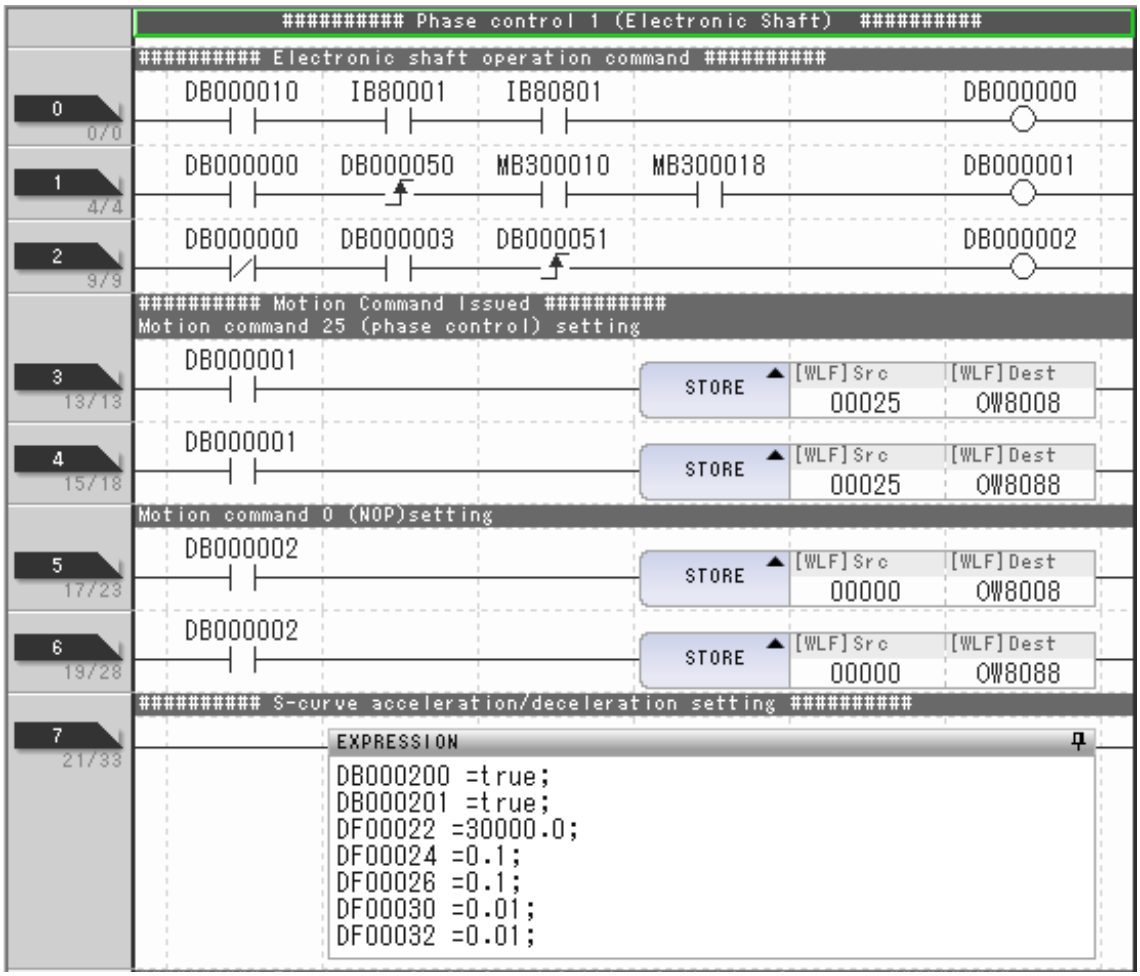
(a) H06 Drawing

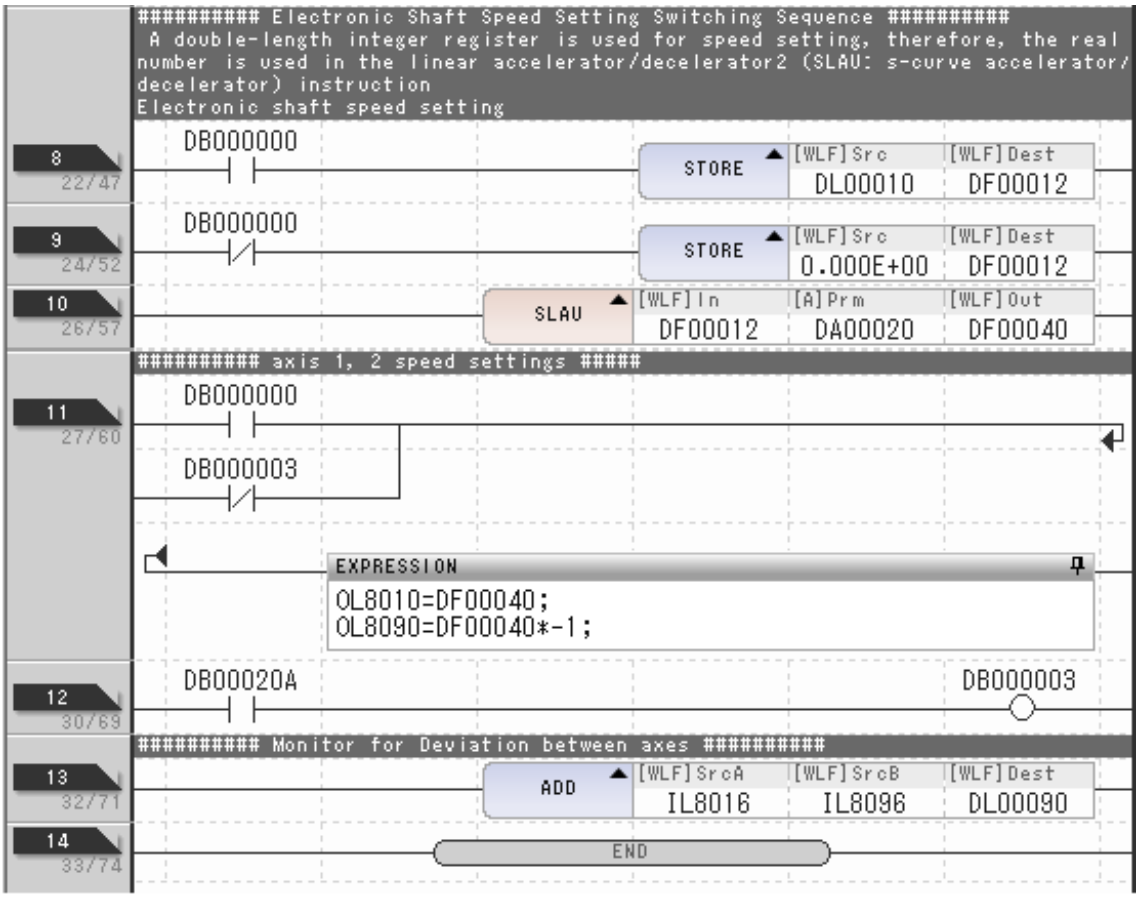
The H06 child drawing controls phase control (electronic shaft).



(b) H06.01 Drawing

The H06.01 grandchild drawing shows the ladder program for controlling phase control (electronic shaft) operation.

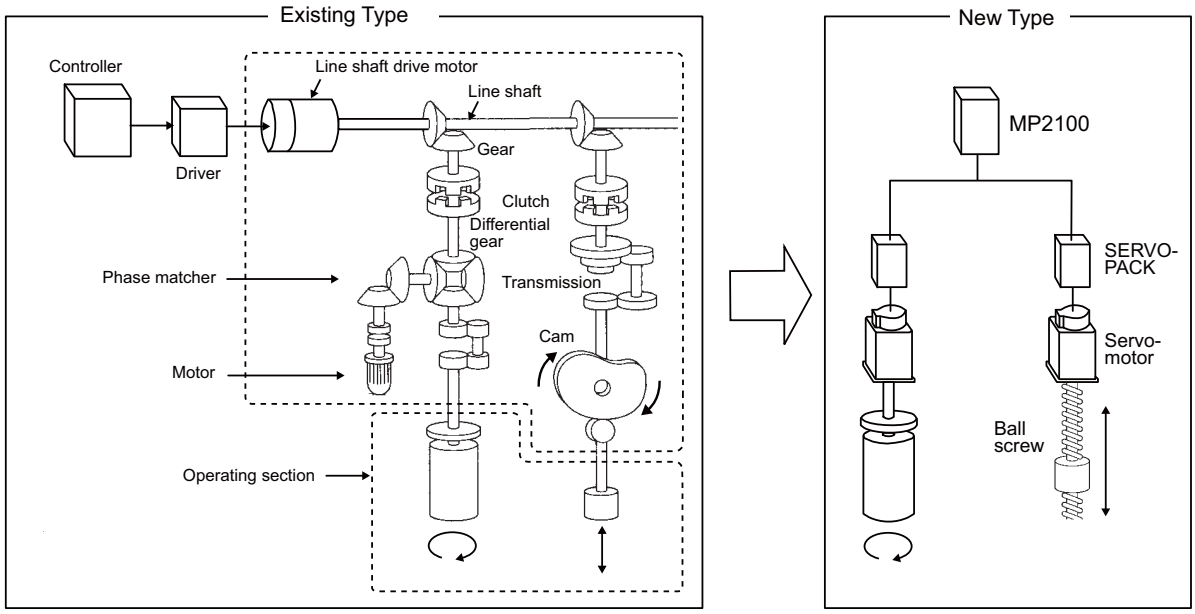



########## Electronic Shaft Speed Setting Switching Sequence ##########
A double-length integer register is used for speed setting, therefore, the real number is used in the linear accelerator/decelerator2 (SLAU: s-curve accelerator/decelerator) instruction
Electronic shaft speed setting
8
22/47
DB000000
STORE
[WLF]Src
DL00010
[WLF]Dest
DF00012
9
24/52
DB000000
STORE
[WLF]Src
0.000E+00
[WLF]Dest
DF00012
10
26/57
SLAU
[WLF]In
DF00012
[A]Prm
DA00020
[WLF]Out
DF00040
########## axis 1, 2 speed settings #####
11
27/60
DB000000
DB000003
EXPRESSION
OL8010=DF00040;
OL8090=DF00040*-1;
12
30/69
DB00020A
DB000003
########## Monitor for Deviation between axes ##########
13
32/71
ADD
[WLF]SrcA
IL8016
[WLF]SrcB
IL8096
[WLF]Dest
DL00090
14
33/74
END

4.3.5 Operation Check 4: Phase Control - Electronic Cam

(1) Machine Outline

As shown in the following figure, the servomotor performs the same operation as the mechanical cam synchronized to a roller connected to the line shaft. No phase matching is used.

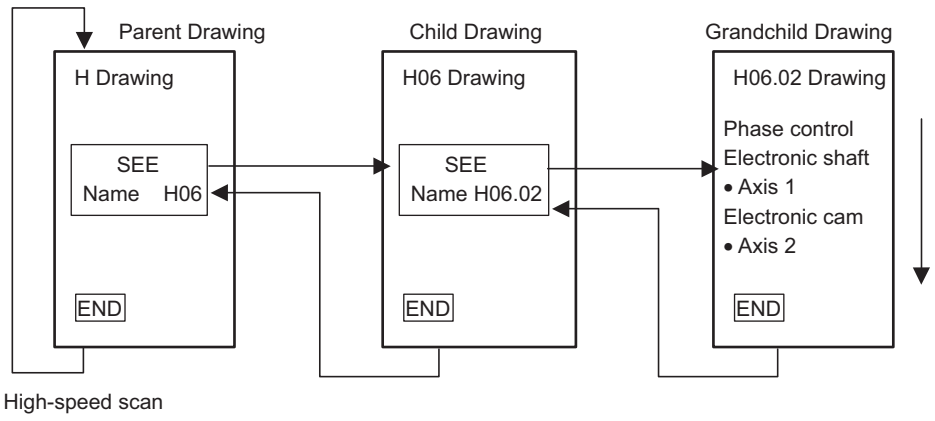


(2) Program Outline

Use the ladder program (H06.02 Drawing) to check the above operation.

The two axes rotate synchronized with the input speed setting. Axis 1 is the roll axis (Master axis) and axis 2 is the cam axis (Slave axis, which moves in COS cam pattern against Master axis).

Cam pattern data is created using a ladder program (L06 Drawing).



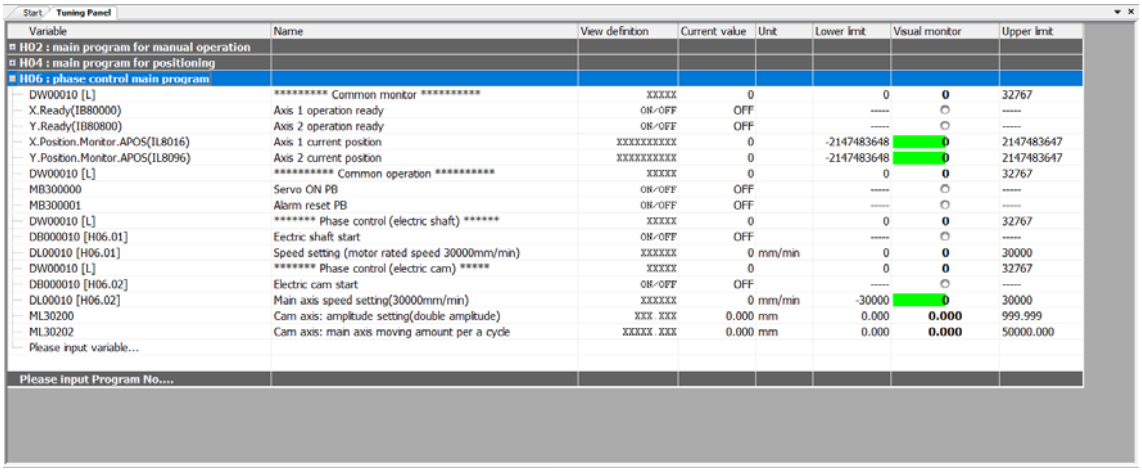
Note: 1. Refer to *4.3.5 (5) Sample Program Details* for details of H06.02 Drawing.

2. A simple device is used in this example to describe the MP2100 system startup. Caution is required because actual applications will be different.
3. This device has no power OFF circuit for the SERVOPACK in the event of emergency stops or overtravel. Include a proper emergency stop circuit in actual applications.

(3) H06 Drawing Tuning Panel

Display the H06 Drawing Tuning Panel as shown in *4.3.1 Opening the Tuning Panel Window*.

Model system operation can be controlled by writing the current values for **Common operation** and **Phase control (electric shaft)** from the Tuning Panel.



Variable	Name	View definition	Current value	Unit	Lower limit	Visual monitor	Upper limit
H02 : main program for manual operation							
H04 : main program for positioning							
H06 : phase control main program							
DW00010 [L]	********** Common monitor **********	XXXXX	0		0	0	32767
X.Ready(IB80000)	Axis 1 operation ready	ON/OFF	OFF		-----		-----
Y.Ready(IB80800)	Axis 2 operation ready	ON/OFF	OFF		-----		-----
X.Position.Monitor.APOS(IL8016)	Axis 1 current position	XXXXXXXXXX	0		-2147483648	0	2147483647
Y.Position.Monitor.APOS(IL8096)	Axis 2 current position	XXXXXXXXXX	0		-2147483648	0	2147483647
DW00010 [L]	********** Common operation **********	XXXXX	0		0	0	32767
MB300000	Servo ON PB	ON/OFF	OFF		-----		-----
MB300001	Alarm reset PB	ON/OFF	OFF		-----		-----
DW00010 [L]	******* Phase control (electric shaft) ******	XXXXX	0		0	0	32767
DB000010 [H06.01]	Eectric shaft start	ON/OFF	OFF		-----		-----
DL00010 [H06.01]	Speed setting (motor rated speed 30000mm/min)	XXXXXX	0	mm/min	0	0	30000
DW00010 [L]	******* Phase control (electric cam) ******	XXXXX	0		0	0	32767
DB000010 [H06.02]	Electric cam start	ON/OFF	OFF		-----		-----
DL00010 [H06.02]	Main axis speed setting(30000mm/min)	XXXXXX	0	mm/min	-30000	0	30000
ML30200	Cam axis: amplitude setting(double amplitude)	XXX XXX	0.000	mm	0.000	0.000	999.999
ML30202	Cam axis: main axis moving amount per a cycle	XXXXX XXX	0.000	mm	0.000	0.000	50000.000
Please input variable...							
Please input Program No....							

(4) Procedure

Use the following procedure to operate the Tuning Panel and check operation.

1. Servo ON

   Change the **Current Value** for **Servo ON PB** from **OFF** to **ON**.

   The servomotor turns ON and the Servo will be clamped.

2. Enter Cam Data

   Enter any value within the setting range to **Cam axis: amplitude setting (double amplitude)** and **Cam axis: main axis moving amount per cycle**. These settings create the cam pattern.

   - **Cam axis: amplitude setting (double amplitude)**, Setting range: 0 to 999.999
   - **Cam axis: main axis moving per cycle**, Setting range: 0 to 50000.000

   Cam pattern data is not changed when the **Current Value** for **Electric shaft start** is set to **ON**.

3. Start Electronic Cam Operation

   Change the **Current Value** for **Electric cam start** from **OFF** to **ON**.

   Axis 2 changes to **phase control (electric cam)** mode. Enter **OFF** to exit phase control (electric cam) mode.

4. Enter Main Axis Speed Settings

   Enter any value within the setting range (–30000 to 30000) as the **Current Value** for **Main axis speed setting**.

   This operation sets the master axis speed and starts operation.
   Change the **Current Value** for **Electric cam start** from **ON** to **OFF** when the check operation has been completed.

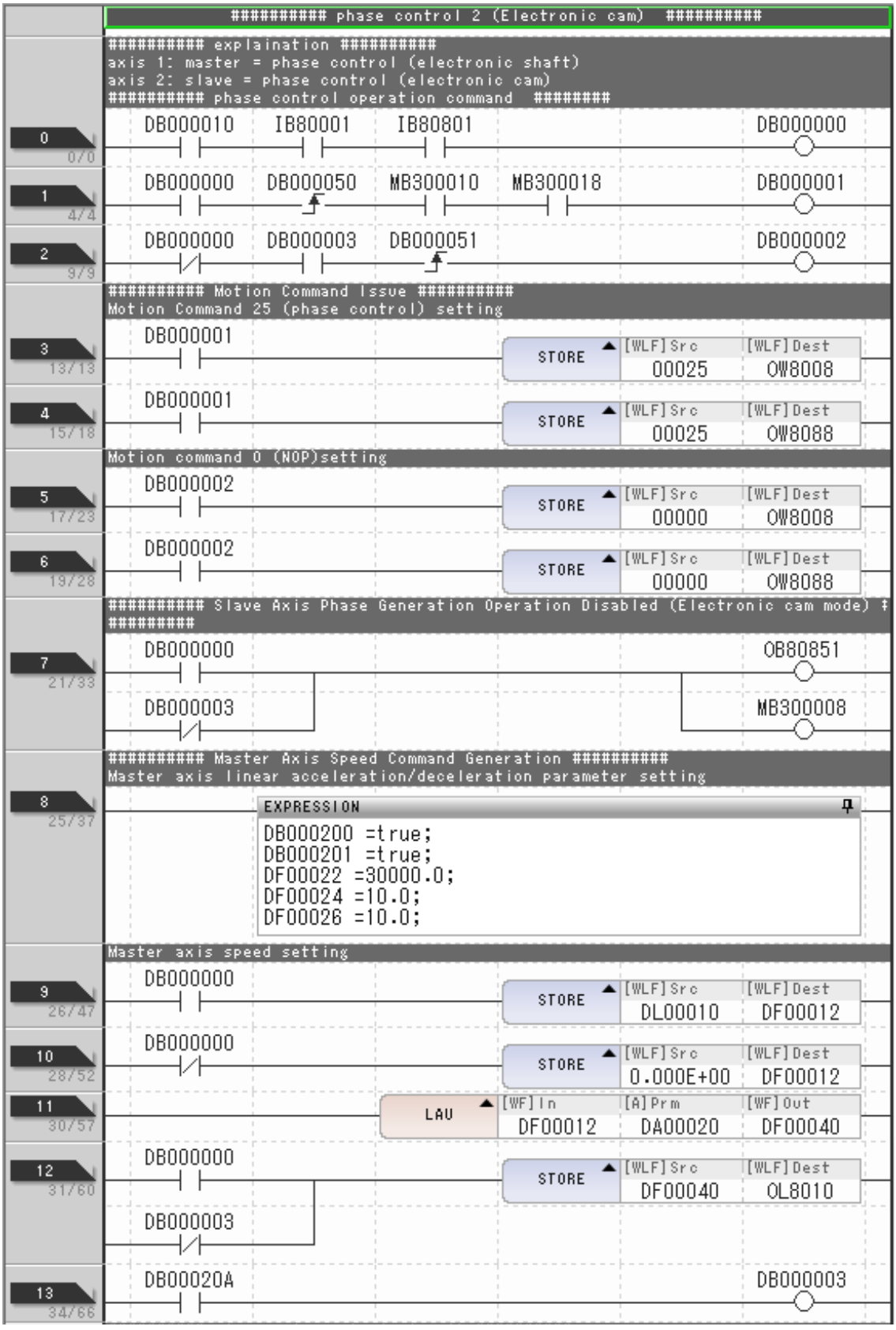
Note: 1. Create routines within the actual application program in order to monitor and control the registers corresponding to the signals and data listed in the table above.

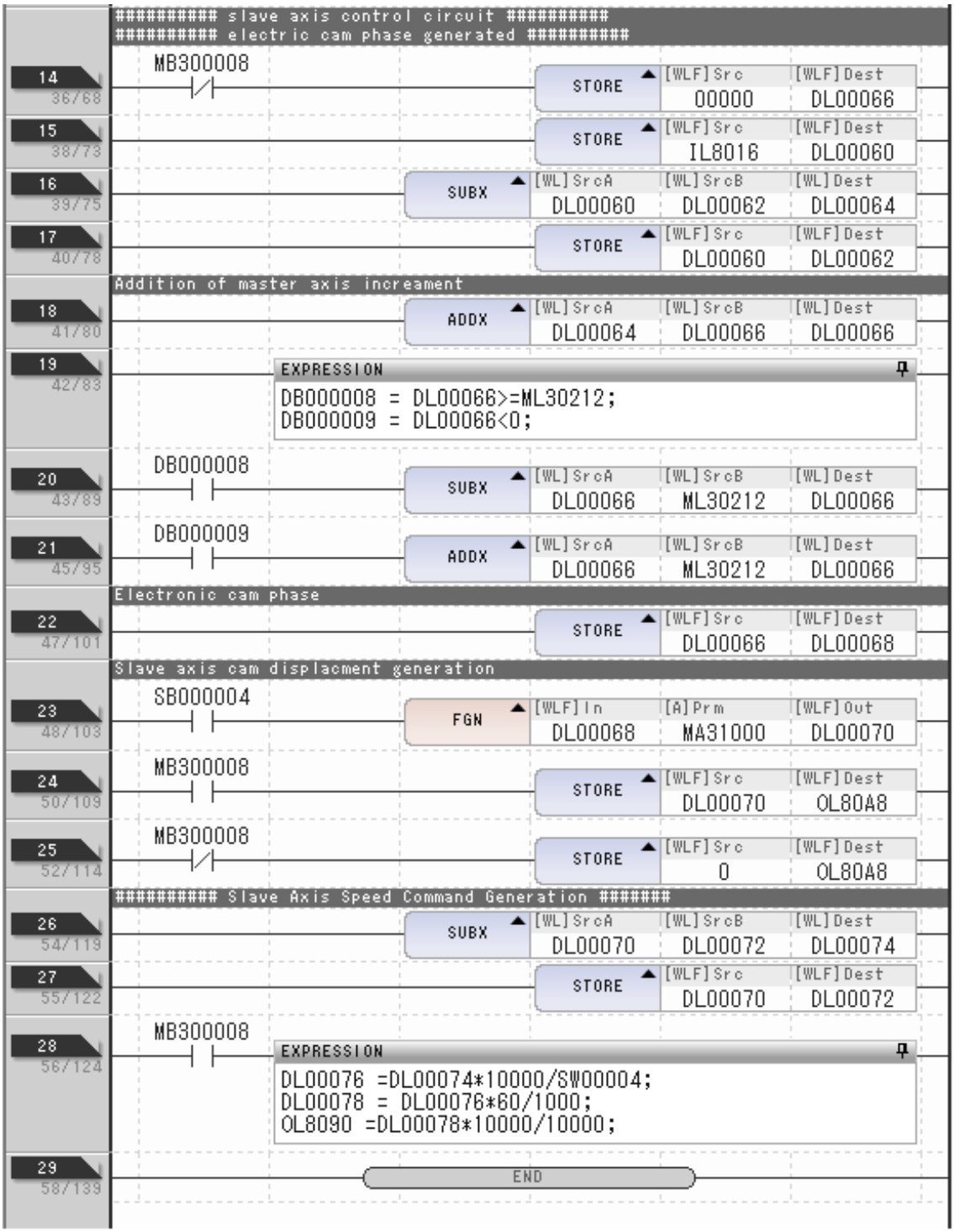
2. The register numbers that correspond to the signals used in this sample program are the register numbers displayed under **REG-No**. next to **DWG** at the right of the **Tuning Panel** window.

(5) Sample Program Details

(a) H06.02 Drawing

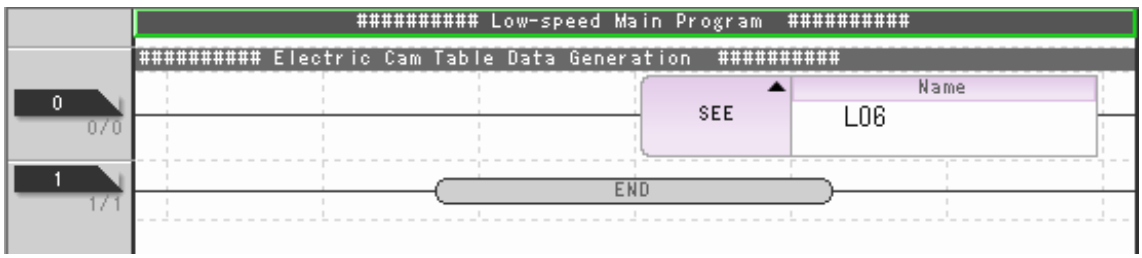
The H06.02 grandchild drawing controls phase control (electronic cam) operation.





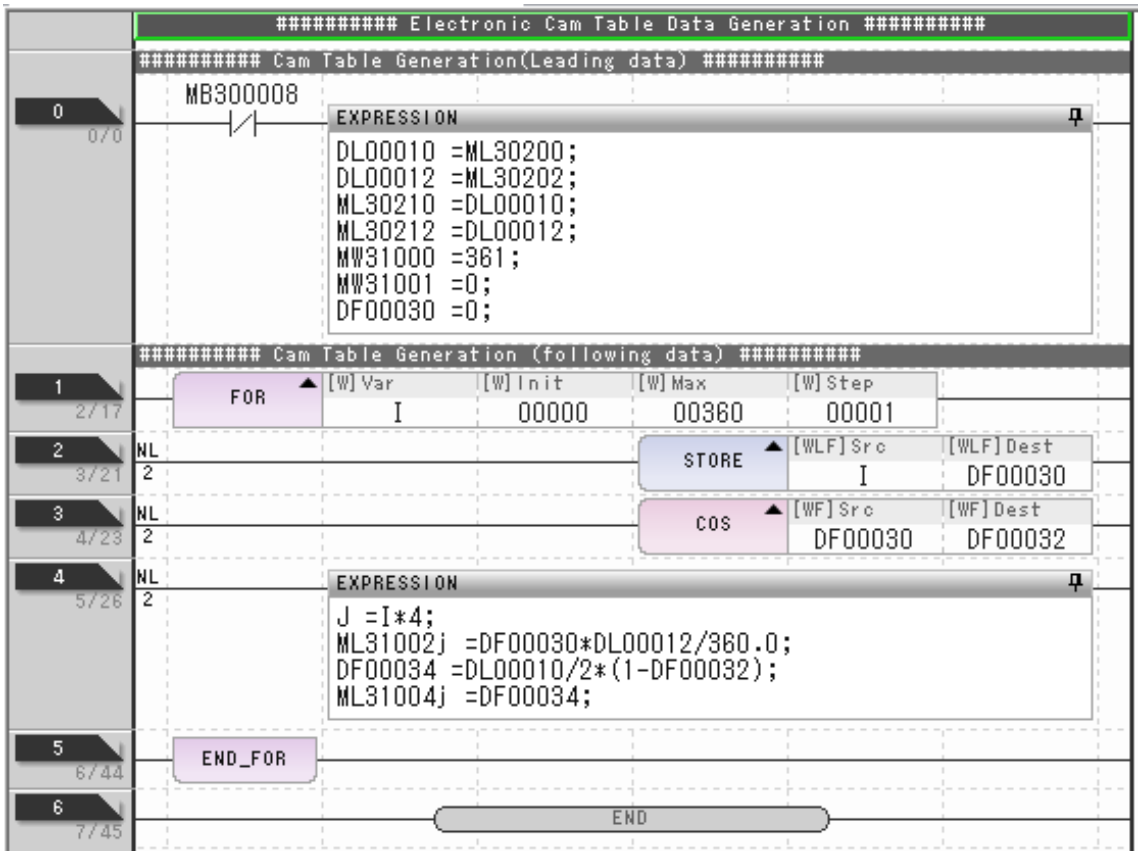
(b) L Drawing

The L parent drawing manages the low-speed scan that controls the overall sample program.



(c) L06 Drawing

The L06 child drawing creates cam pattern data for phase control (electronic cam).



4.4 System Startup Using Self-configuration

System startup time can be reduced by using self-configuration.

This section explains system startup using self-configuration, in the following three circumstances.

- Starting the system for the first time
- Adding an electronic device (e.g., SERVOPACK or Optional Module)
- Replacing electronic devices

4.4.1 Starting the System for the First Time

Use the following procedure to start up a new system.

1. Wire and Connect Electronic Devices

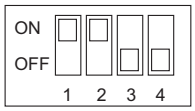
   Correctly wire and connect all electronic devices to be used.

2. Make Switch Settings for MECHATROLINK Slaves

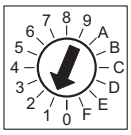
   Set the MECHATROLINK communication specifications using the DIP switch and the station address on the rotary switch on each MECHATROLINK slaves.

   Example of SERVOPACK Settings (SGDS-□□□1□□/SGDV-□□□□1□□□□□)

SW2	Name	Setting	Details	Setting on Shipment
1	Communication speed setting	OFF	4Mbps (MECHATROLINK-I)	ON
		ON	10Mbps (MECHATROLINK-II)	
2	Number of transmission bytes setting	OFF	17 Bytes	ON
		ON	32 Bytes	
3	Station address setting	OFF	Station address = 40h + SW1	OFF
		ON	Station address = 50h + SW1	
4	Reserved for system use	OFF	–	OFF



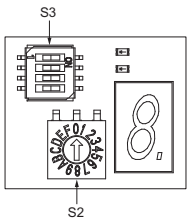
SW2 (Setting on shipment)



SW1 (Setting on shipment)

   Example of SERVOPACK Settings (SGDS-□□□□10A□□□)

S3	Name	Setting	Details	Setting on Shipment
1	Communication speed setting	OFF	4Mbps (MECHATROLINK-I)	ON
		ON	10Mbps (MECHATROLINK-II)	
2	Number of transmission bytes setting	OFF	17 Bytes	ON
		ON	32 Bytes	
3	Station address setting	OFF	Station address = 40h + SW1	OFF
		ON	Station address = 50h + SW1	
4	Reserved for system use	OFF	–	OFF



   Note: Refer to each slave's manual for details on the setting.

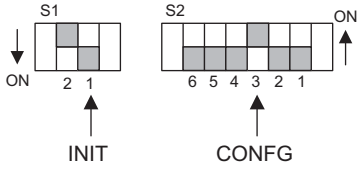
3. Start Up MECHATROLINK Slaves

   Turn ON the power to the MECHATROLINK slaves and check that the electronic devices start up normally.

   Note: 1. If using a new absolute encoder, the absolute encoder needs to be initialized.
   2. Adjustments such as servo gain adjustment can be made at this stage, but they can also be made after executing self-configuration.

4. Switch Settings of the MP2100

Set the mode switches of the MP2100 as follows.



Make the required settings, such as communication settings and station address, with the switches of each Optional Module mounted on the MP2100 too.

5. Execute Self-configuration

Make sure that all the MECHATROLINK slave devices have started, and then execute self-configuration.

Check that all of the MECHATROLINK slave devices have started up normally, turn the power to the MP2100 ON and execute self-configuration.

The LED indications on the MP2100 will change as follows.

Through self-configuration, the MP2100 recognizes the connected MECHATROLINK slave devices and Optional Modules, and assigns I/O registers. The motion parameters are automatically set to enable the minimum standard operations.

INFO

- The SERVOPACK's overtravel function will automatically be disabled by executing self-configuration, because the self-configuration is intended to enable immediate operation of slave devices including servo drives.
  For details on the overtravel function, refer to *11.2 Overtravel Function* in *Machine Controller MP2000 Series Built-in SVB/SVB-01 Motion Module User's Manual* (Manual No.: SIEP C880700 33).
  Before operating the machine after execution of self-configuration, enable each SERVOPACK's overtravel function by setting the parameters.

6. Make Parameter Settings to Match Machinery

Start MPE720, set the fixed parameters that relate to reference units (fixed parameters No. 4, No. 5, No. 6, No. 8, No. 9) and save them.

If the servo gain has not been adjusted in step 3, adjust the servo gain and make any other required adjustments.

Note: 1. For the procedure from the startup of MPE720 to setting and saving of the fixed parameters, refer to the following sections.
- *4.2.1 (2) Starting MPE720 Ver.7 to Transfer the Sample Program to the MP2100*
- *4.2.2 Setting Motion Fixed Parameters and Adjusting the Settings for Servo Control*

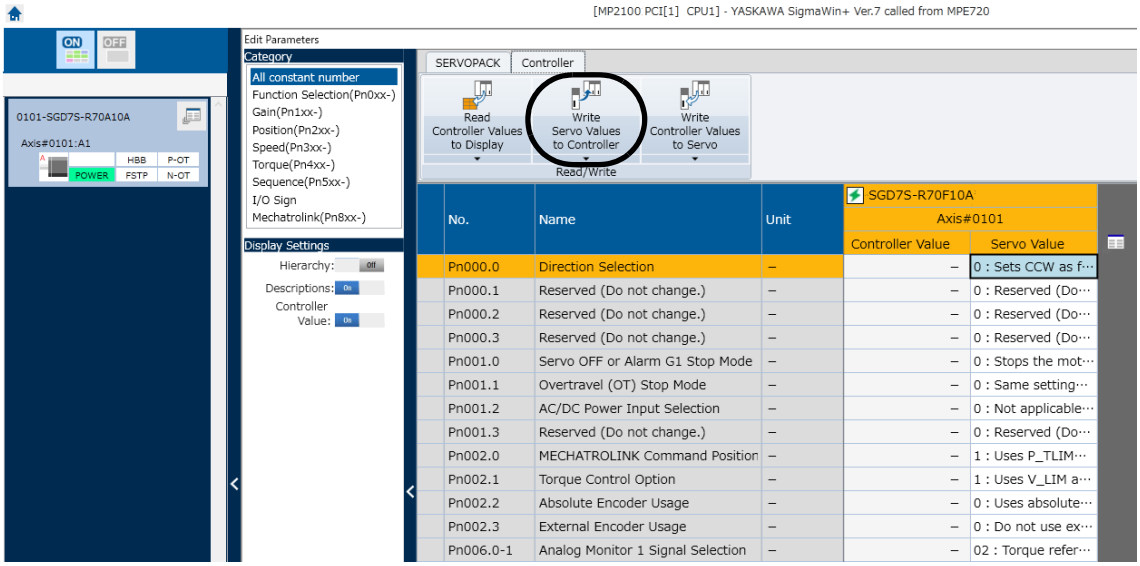
2. For the list of fixed parameters, refer to *Appendix B Motion Parameter Lists*.
   For details on fixed parameters and setting parameters matched to machines, refer to *Machine Controller MP2000 Series Built-in SVB/SVB-01 Motion Module User's Manual* (Manual No.: SIEP C880700 33).
3. For details on servo adjustment, refer to the relevant SERVOPACK manual.

7. Save SERVOPACK Parameters

   After servo tuning has completed, save the SERVOPACK parameters for each axis to the MP2100.

   a) After you select the axes on the Module Configuration Definition Window (refer to *4.2.2 Setting Motion Fixed Parameters and Adjusting the Settings for Servo Control*), the SigmaWin+ is started by the SigmaWin+ linked function.

   b) Select the **Controller** tab, and click Write **Servo Values to Controller**.



   Note: The data in the **Controller Value** column shows the SERVOPACK data saved to the MP2100. The data in the **SERVOPACK Value** column shows the data currently set in the SERVOPACK.

   c) Switch to the Module Configuration Definition Window, and click the **Save to project** button to save the SERVOPACK values to the MP2100.

8. Save MP2100 Data to Flash Memory

   Return to the main window of the MPE720 and save the data to flash memory.

   Note: For details on how to save the data to flash memory, refer to *4.2.3 (1) Saving to Flash Memory*.

9. Save Ladder Programs and Restart MP2100

   Transfer the ladder program to the MP2100 and save to flash memory, and then turn the power from OFF to ON to restart the MP2100.

   Note: For details on transferring ladder programs, refer to *4.2.1 (2) Starting MPE720 Ver.7 to Transfer the Sample Program to the MP2100*.

This completes the system startup procedure.

IMPORTANT

- Always save to flash memory when applications have been changed, e.g., the ladder program changed or parameters set. The added information will be lost if not saved to flash memory and the power is turned OFF. If the information is lost, load the application remaining on the hard disk of the personal computer to the MP2100 and save to flash memory.
- Yaskawa recommends that you back up the applications at appropriate times. Applications can be backed up as follows:
  MPE720 Ver.7: Select ***Online - Read from Controller*** in the main window.

4.4.2 System Startup when Adding Electronic Devices

Use the following procedure to start the system when adding SERVOPACKs, Optional Modules, and other electronic devices.

1. Back Up Applications

   Before adding the electronic devices, backup the application.

   Note: For details on how to create a backup, refer to *4.2.3 (2) Transferring Data from the MP2100 to the Computer*.

2. Turning off power to the host computer

   Once the application has been backed up, log off from MP2100 or disconnect the communication, and turn OFF the power to the host computer.

3. Start the Electronic Device to Be Added

   Make the DIP and rotary switch settings for the device to be added.

   When using MECHATROLINK slaves, turn ON the power to that device only and check that it starts up normally. Once normal startup has been confirmed, turn OFF the power supply.

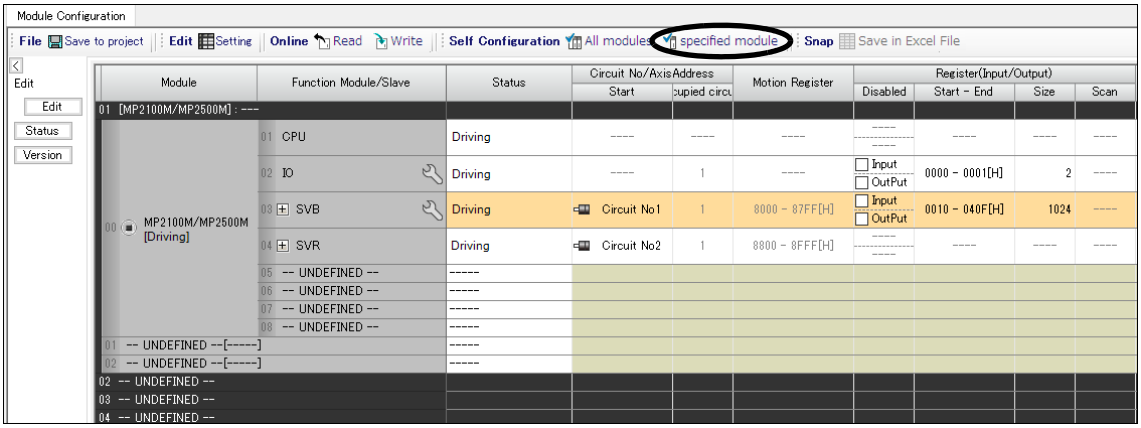
4. Connect the Electronic Device

   Connect the electronic device to the MP2100 and turn ON the power to all the MECHATROLINK slaves.

5. Execute Self-configuration.

   Executing self-configuration for the specified module

   Turn ON the power supply to the host computer, make an online connection to the MP2100, and log in. Select the optional module that was added or the SVB module that added SERVOPACKs. Click the **specified module** button for self-configuration to execute self-configuration for the specified module.



Note: 1. For details on how to execute self-configuration of MP2100 Modules, refer to *5.4.3 (2) Self-configuration of Each Module*.

2. If **All modules** is executed when MP2100 module is selected, all the modules will be self-configured.

3. With the specified module self-configuration function, existing definitions for SERVOPACKs are not refreshed and existing parameters are saved. However, SERVOPACKs need to be started up normally.

IMPORTANT

- If I/O addresses are changed for an existing application using MPE720 after the initial self-configuration has been executed, the I/O addresses are updated when self-configuration is subsequently executed. If SVR is set to disabled, the setting will return to enabled. Yaskawa recommends that you check the setting again including settings for existing electronic devices, after self-configuration has been executed.

Refer to steps 6 to 9 under *4.4.1 Starting the System for the First Time* for details of the rest of this procedure (steps 6 to 9).

6. Make Parameter Settings to Match Machinery.
7. Save SERVOPACK Parameters.
8. Save MP2100 Data to Flash Memory.
9. Save Ladder Programs and Restart the MP2100

This completes the system startup procedure when electronic devices have been added.

4.4.3 System Startup when Replacing Electronic Devices

Use the following procedure to start the system when replacing SERVOPACKs, Optional Modules, and other electronic devices because of malfunctions and other problems.

1. Back Up Applications

   Before replacing the electronic devices, backup the application using MPE720.

   Note: For details on how to create a backup, refer to *4.2.3 (2) Transferring Data from the MP2100 to the Computer*.

2. Turning OFF Power to the Host Computer

   Once the application has been backed up, log off from MP2100 or disconnect the communication, and turn OFF the power to the host computer.

3. Start the Electronic Device to Be Added

   Make the DIP and rotary switch and other settings for the new electronic device.
   When using MECHATROLINK slaves, turn ON the power to that device only and check that it starts up normally. Once normal startup has been confirmed, turn OFF the power supply.

4. Replace the Electronic Device

   Remove the electronic device to be replaced, connect the new device to the MP2100, and turn ON the power to all MECHATROLINK slaves.

5. Turning Host Computer ON

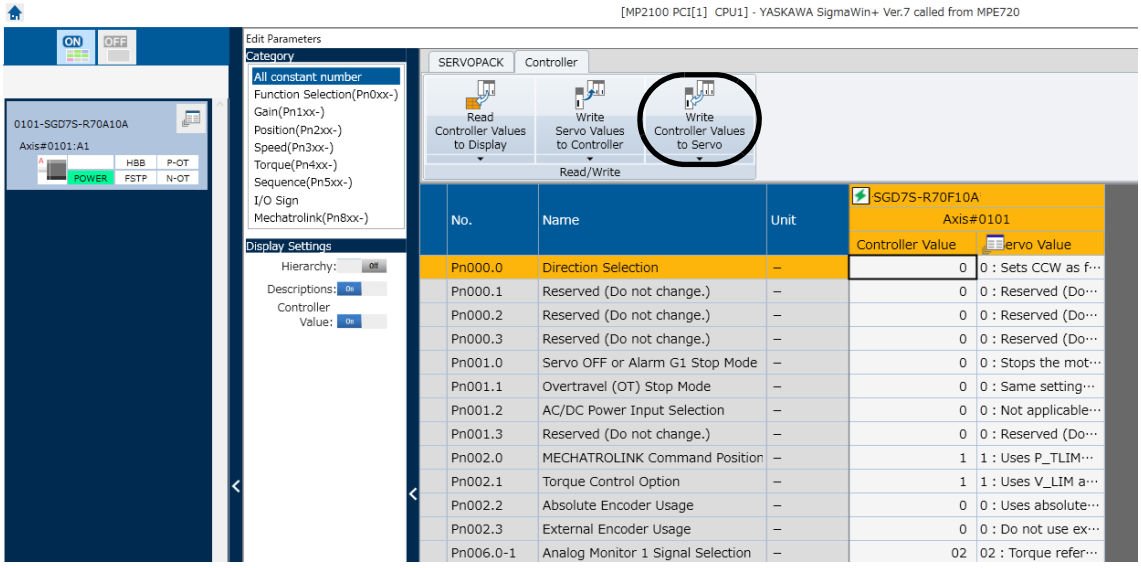
   Turn ON the power to the host computer.

6. Save SERVOPACK Parameters

   If a SERVOPACK has been replaced, use the following procedures to write the SERVOPACK parameters saved to the MP2100 to the new SERVOPACK.

   a) After you select the axes on the Module Configuration Definition Window (refer to *4.2.2 Setting Motion Fixed Parameters and Adjusting the Settings for Servo Control*), the SigmaWin+ is started by the SigmaWin+ linked function.

   b) Click **Write Controller Values to Servo** to write the controller values to the SERVOPACK.



   Note: When **Write Controller Values to Servo** is executed, ALL the SERVOPACK settings data in the MP2100 is written to the SERVOPACK.

7. Turn ON the Host Computer and SERVOPACKs

   Turn ON (OFF to ON) the power to the host computer and SERVOPACKs and then enable the parameters written to the SERVOPACKs.

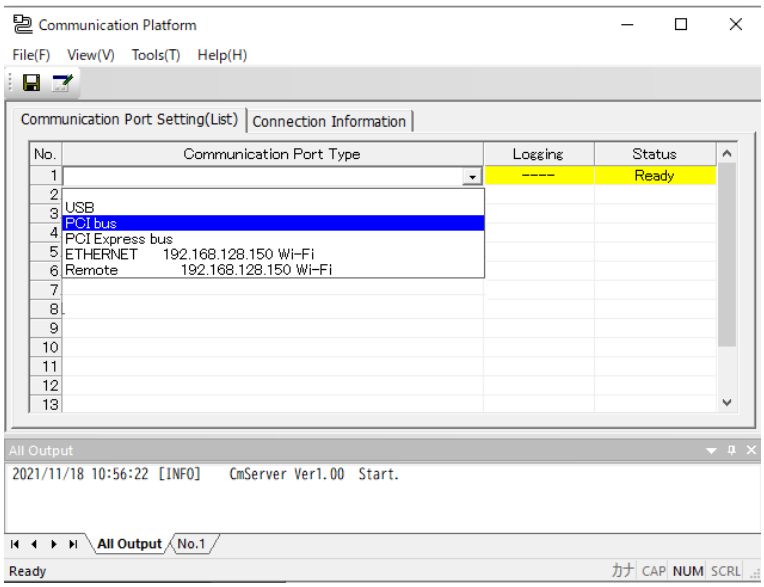
This completes the restarting of the system when devices have been replaced.

4.5 MP2100M Startup

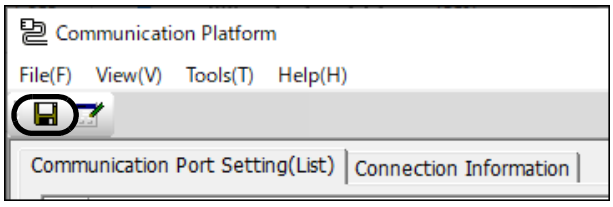
This section describes the MPE720 setting method when using the MP2100M.

4.5.1 Communication Platform Settings

1. Select **PCI bus** under **Communication Port Type** in the **Communication Platform** window.



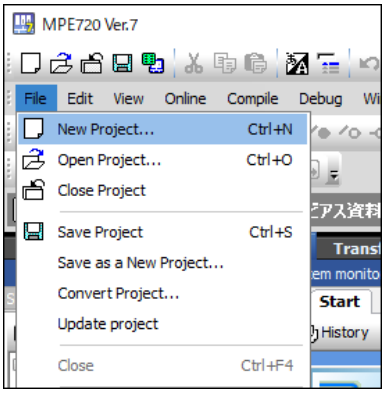
2. Click the **Save** button.



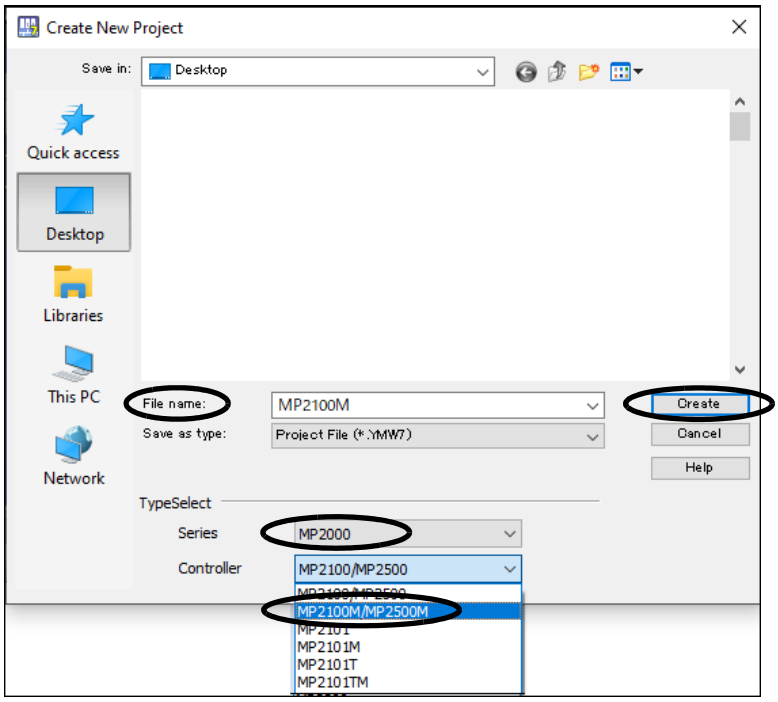
4.5.2 Create Project

Create an MP2100M project in the MPE720 Ver.7.

1. Select **New Project** from the **File** menu.

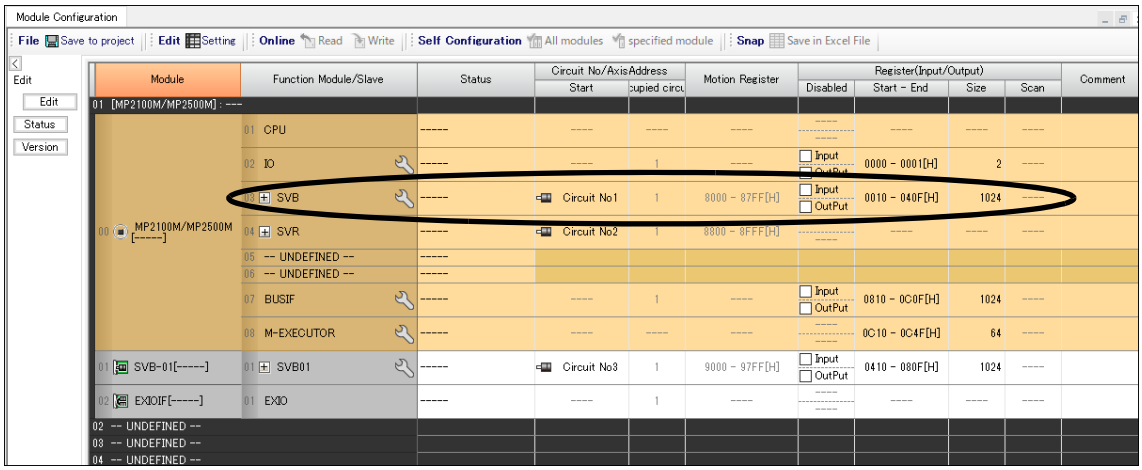


2. Specify the folder in which the project is to be saved, input the File name, select **MP2000** for **Series** and **MP2100M/MP2500M** for **Controller**, then click the **Create** button.

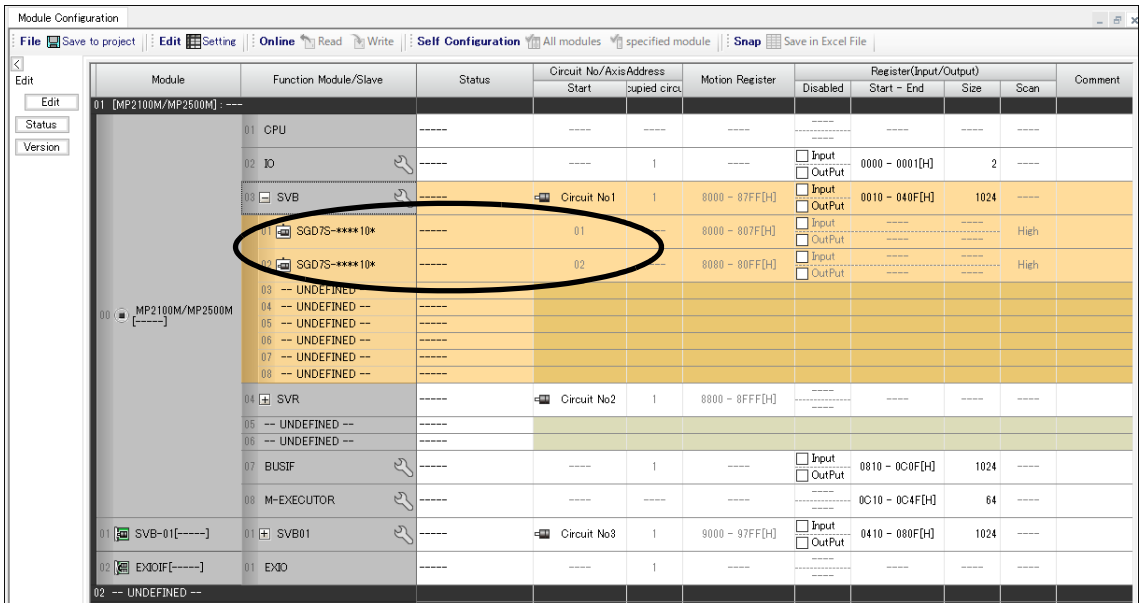


4.5.3 Module Configuration Definition Settings

1. Set the device that was connected to the MECHATROLINK connector on the CPU board to No. 3 **SVB** in the **Function Module/Slave** area on the Module Configuration Definition Window.
   Double-click the **SVB** cell to display the MECHATROLINK detail definition window.



2. Set the device that was connected to the MECHATROLINK connector on the SVB board under **SVB**.



5

Outline of Motion Control Systems

This chapter explains the basic operation of MP2100/MP2100M Motion Control Systems and provides an outline of user programs and registers.

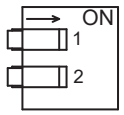
5.1 Startup Sequence and Basic Operation

This section explains the startup sequence and basic operation of the MP2100/MP2100M. The methods for setting the mode switch, the types of self-diagnosis, and the indicator patterns are also explained.

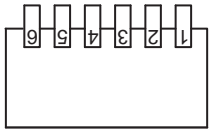
5.1.1 MP2100 Mode Switch Settings

The mode switch is used to control the startup sequence for the MP2100/MP2101.

Mode Switch 1 (S1)

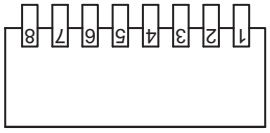


Mode Switch 2 (S2)



JAPMC-MC2100

Mode Switch 1 (S1)



JAPMC-MC2100-E/JAPMC-MC2102-E

The following tables show the function of each switch.

(1) Mode Switch 1 (S1)

No.	Name	Status	Function	Default Setting	Details
1	INIT	ON	Memory clear	OFF	Set to ON to clear memory. The S registers and M registers are cleared to zero. If this switch is set to OFF, the program stored in flash memory will be executed.
		OFF	Normal operation		
2	STOP	ON	User program stopped		Set to ON to stop the user program before debugging the program.
		OFF	User program running		

(2) Mode Switch 2 (S2)

JAPMC-MC2100

No.	Name	Status	Function	Default Setting	Details
1	TEST	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
2	MON	ON	System use		Always set to OFF.
		OFF	Normal operation		
3	CNFG	ON	Configuration mode		Set to ON to execute self-configuration for connect devices.
		OFF	Normal operation		
4	SUP	ON	System use		Always set to OFF.
		OFF	Normal operation		
5	SLOT1	SLOT1 - SLOT0			The CP number. These are used for MPE720 communications settings.*
		ON - ON	CP4		
6	SLOT0	ON - OFF	CP3		
		OFF - ON	CP2		
		OFF - OFF	CP1		

* The CP number is used for **CPU No.** on the **Communications Setting** window when the MPE720 Ver.7 is connected. When using two or more MP2100s, set different CP number.

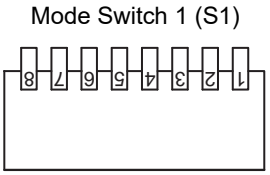
JAPMC-MC2100-E/JAPMC-MC2102-E

No.	Name	Status	Function	Default Setting	Details
1	TEST	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
2	MON	ON	System use		Always set to OFF.
		OFF	Normal operation		
3	CNFG	ON	Configuration mode		Set to ON to execute self-configuration for connect devices.
		OFF	Normal operation		
4	INIT	ON	Memory clear		Set to ON to clear memory. The S registers and M registers are cleared to zero. If this switch is set to OFF, the program stored in flash memory will be executed.
		OFF	Normal operation		
5	SUP	ON	System use		Always set to OFF.
		OFF	Normal operation		
6	STOP	ON	User program stopped		Set to ON to stop the user program before debugging the program.
		OFF	User program running		
7	SLOT0	SLOT1 - SLOT0 ON - ON ON - OFF OFF - ON OFF - OFF	 CP4 CP3 CP2 CP1		The CP number. These are used for MPE720 communications settings.*
8	SLOT1				

* The CP number is used for **CPU No.** on the **Communications Setting** window when the MPE720 Ver.7 is connected. When using two or more MP2100Ms, set different CP number.

5.1.2 MP2100M Mode Switch Settings

The mode switches are used to control the startup sequence for the MP2100M/MP2101M. The MP2100M/MP2101M has the switches indicated in the figure below. The function of each switch is shown in the table below.



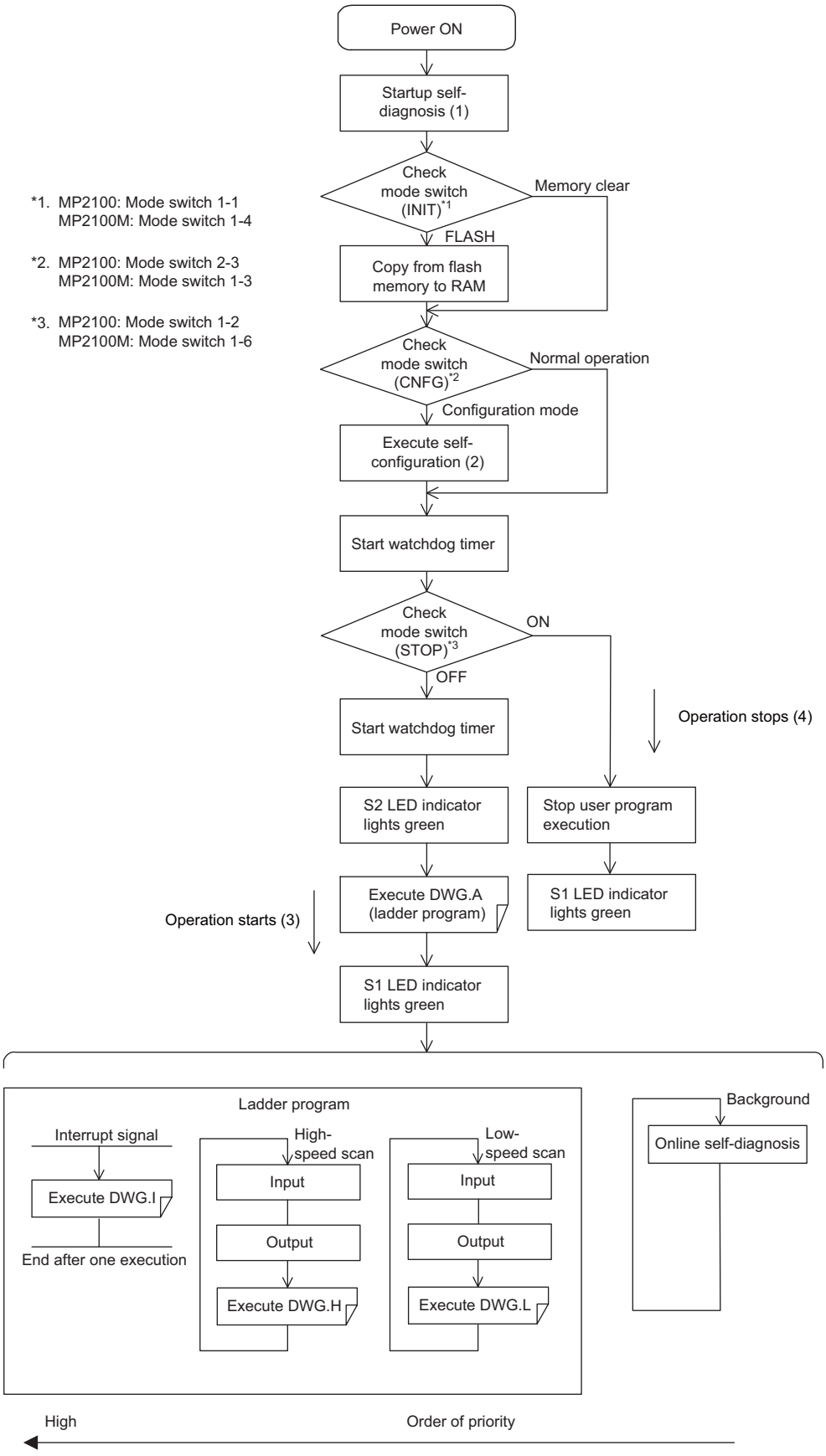
The following table shows the function of each switch.

No.	Name	Status	Function	Default Setting	Details
1	TEST	ON	System use	OFF	Always set to OFF.
		OFF	Normal operation		
2	MON	ON	System use		Always set to OFF.
		OFF	Normal operation		
3	CNFG	ON	Configuration mode		Set to ON to execute self-configuration for connect devices.
		OFF	Normal operation		
4	INIT	ON	Memory clear		Set to ON to clear memory. The S registers and M registers are cleared to zero. If this switch is set to OFF, the program stored in flash memory will be executed.
		OFF	Normal operation		
5	SUP	ON	System use		Always set to OFF.
		OFF	Normal operation		
6	STOP	ON	User program stopped		Set to ON to stop the user program before debugging the program.
		OFF	User program running		
7	SLOT0	SLOT - SLOT0			The CP number. These are used for MPE720 communications settings.*
		ON - ON	CP4		
		ON - OFF	CP3		
8	SLOT1	OFF - ON	CP2		
		OFF - OFF	CP1		

* The CP number is used for **CPU No.** on the **Communications Setting** window when the MPE720 Ver.7 is connected. When using two or more MP2100Ms, set a different CP number.

5.1.3 Startup Sequence

A basic outline of the startup sequence and basic operation of the MP2100/MP2100M is shown below.



5.1.4 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 S1 and S2 LED indicators will flash red for the specified number of times. For details, refer to *5.1.5 Indicator Patterns*.

(2) Self-configuration

Optional Modules are recognized and definition files are generated automatically.
For details, refer to *5.4 Self-configuration*.

The S2 LED indicator will flash green during execution of self-configuration.

(3) Starting an Operation

When the STOP switch is set to OFF (RUN) or changed from ON (STOP) to OFF (RUN), the CPU starts the watchdog timer and then executes DWG.A (startup processing drawing) in the ladder program.

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) Stopping an Operation

MP2100/MP2100M stops motion control operation when mode switch STOP is ON (STOP) and in the following circumstances.

Cause	Countermeasure
Power supply turned OFF	Turn ON the power again.
Power interruption	
Fatal error	Determine the error by the indicator status and turn the power OFF and ON.
STOP executed from MPE720	Execute RUN from MPE720.

5.1.5 Indicator Patterns

The MP2100/MP2100M performs a variety of diagnostics at startup. If an error is found, the S1 LED indicator flashes red. The number of times the indicators flash differs depending on the error details, so error details can be determined from counting the number of flashes. The following table shows details of MP2100/MP2100M LED indicator.

Note: 1. MPE720 cannot be operated when the indicators are flashing.
2. For information on errors and countermeasures, refer to *6.3.3 LED Indicators*.

Type	Indicator					Indicator Details	Remarks
	S1 (Green)	S2 (Green)	S1 (Red)	S2 (Red)	BAT		
Normal	Not lit	Not lit	Lit	Lit	Not lit	Hardware reset status	If this status continues for 10 seconds or longer after turning ON the power, the hardware is faulty.
	Not lit	Not lit	Not lit	Not lit	Not lit	Initial execution	
	Not lit	Lit	Not lit	Not lit	Not lit	Execute of drawing A	
	Lit	Not lit	Not lit	Not lit	Not lit	The user program is stopped (offline stop mode).	This status is established when a STOP operation is performed using a switch or the MPE720.
	Lit	Lit	Not lit	Not lit	Not lit	The user program is executing normally.	–
Error	Not lit	Not lit	Lit	Not lit	Not lit	A serious error has occurred.	The S1 LED indicator lights (Red) when a failure occurs in the CPU.
	Not lit	Not lit	Flashing	Not lit	Not lit	Number of flashes when software error occurs: 3: Address read error 4: Address write error 5: FPU error 6: General illegal command error 7: Slot illegal command error 8: General FPU suppression error 9: Slot FPU suppression error 10: TLB multi-bit error 11: LTB read error 12: LTB write error 13: LTB protection violation (read) 14: LTB protection violation (write) 15: First page write error	The S1 LED indicator flashes (Red) when an error occurs.
	Not lit	Not lit	Flashing	Flashing	Not lit	Number of flashes when hardware error occurs: 2: RAM diagnosis error 3: ROM diagnosis error 4: CPU function diagnosis error 5: FPU function diagnosis error	The S1 and S2 LED indicators flash (Red) when a self-diagnosis failure occurs.
Alarm	–*	–*	–*	–*	Lit	Battery alarm	The BAT LED indicator lights when the battery voltage drops.
	Lit	Not lit	Not lit	Lit	Not lit	Calculation error I/O error	The S2 LED indicator lights (Red) when a calculation or I/O error is detected.

* A dash (–) means not specified.

5.2 User Programs

This section explains the basic operation of the user program. The MP2100/MP2100M's user programs include ladder program and motion program. For details, refer to the following manuals.

- *Machine Controller MP2000 Series User's Manual, Ladder Programming* (Manual No.: SIEZ-C887-1.2)
- *Machine Controller MP2000 Series User's Manual for Motion Programming* (Manual No.: SIEP C880700 38)

5.2.1 Drawings (DWGs)

User programs are managed in units of ladder drawings, which are identified by drawing numbers. These drawings form the basis of user programs.

(1) Types of Drawings

Ladder drawings include parent drawings, child drawings, grandchild drawings, and operation error processing drawings. In addition to drawings, some functions can also 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 executed by using a SEE command from a parent drawing.

- Grandchild Drawings

  Grandchild drawings are executed by 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 executed from parent, child, and grandchild drawings using the FSTART command.

(2) Drawing Types and Order 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	Number of Drawings
DWG.A (drawing A)	Startup processing	1	Power ON (Executed once only, when the power is turned ON)	64
DWG.I (drawing I)	Interrupt processing	2	External interrupts, such as Optional Module DI interrupts or counter interrupts.	64
DWG.H (drawing H)	High-speed scan processing	3	Scheduled cycle startup (Executed during each high-speed scan of a scheduled cycle startup)	200
DWG.L (drawing L)	Low-speed scan processing	4	Scheduled cycle startup (Executed during each low-speed scan of a scheduled cycle startup)	500

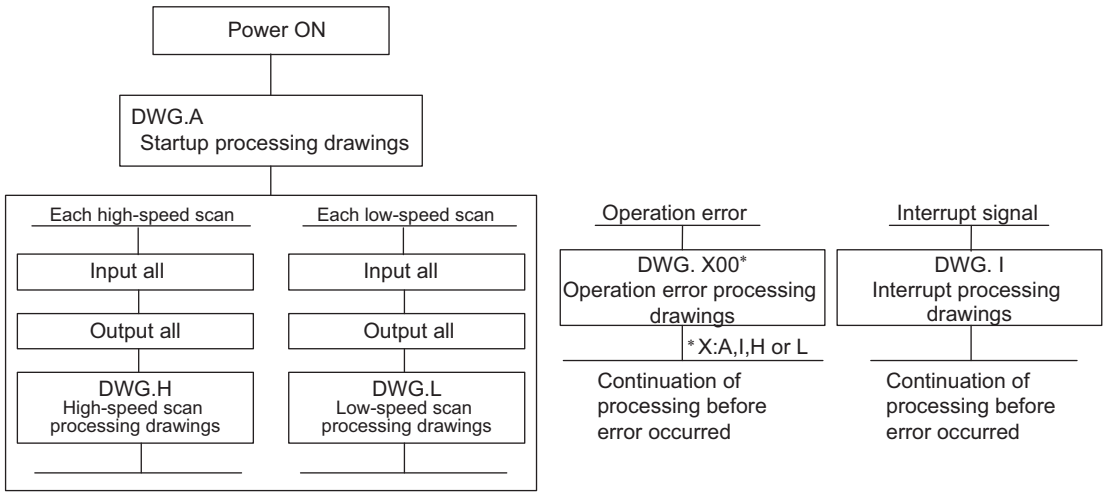
The following table provides details of the number of drawings for each drawing.

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

5.2.2 Execution Control of Drawings

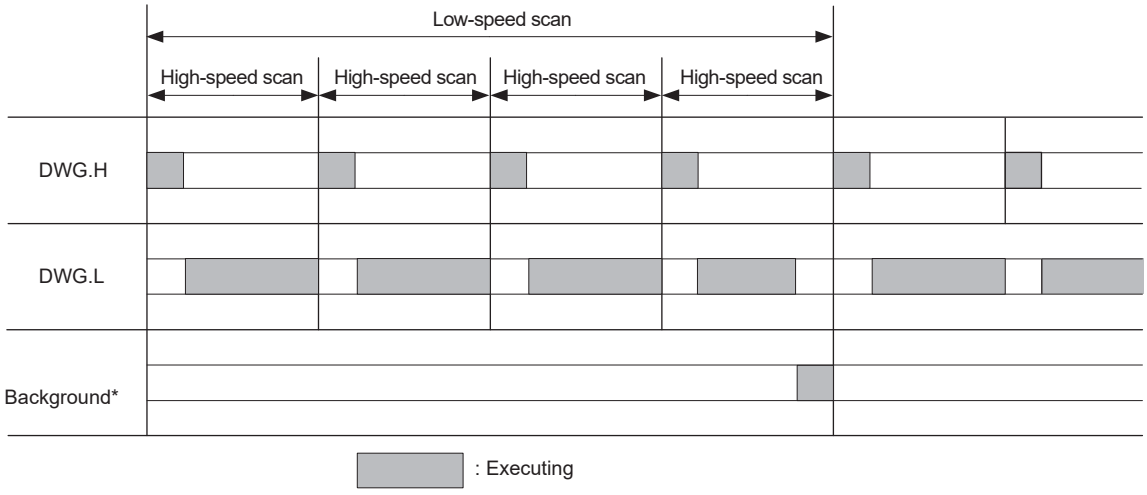
(1) Execution Control

The following table shows when each drawing is executed based on the order of priority.



(2) Execution Schedule for Scan Process 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.



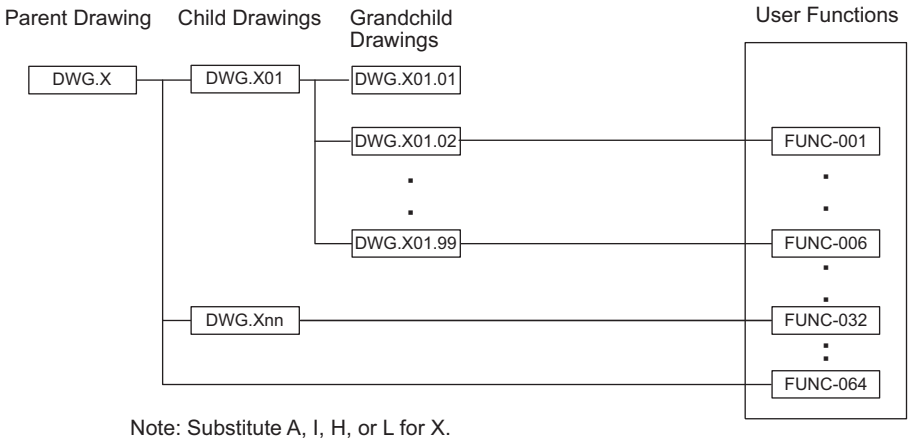
* Background processing is used to execute internal system processing, e.g., communication processing.

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.

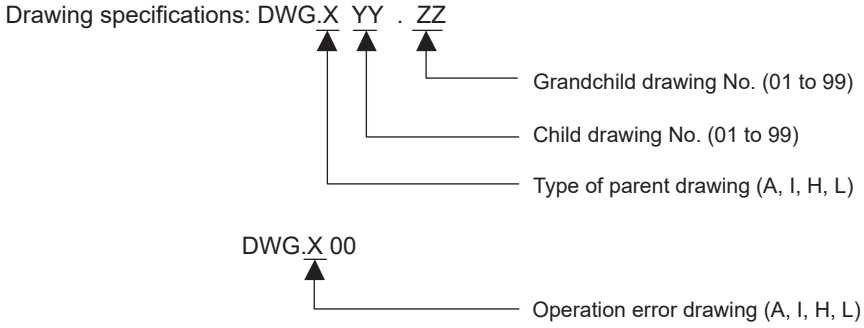
(3) Hierarchical Arrangement of Drawings

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.

Each processing program is prepared with the parent drawing, child drawing, grandchild drawing hierarchy, as shown below.

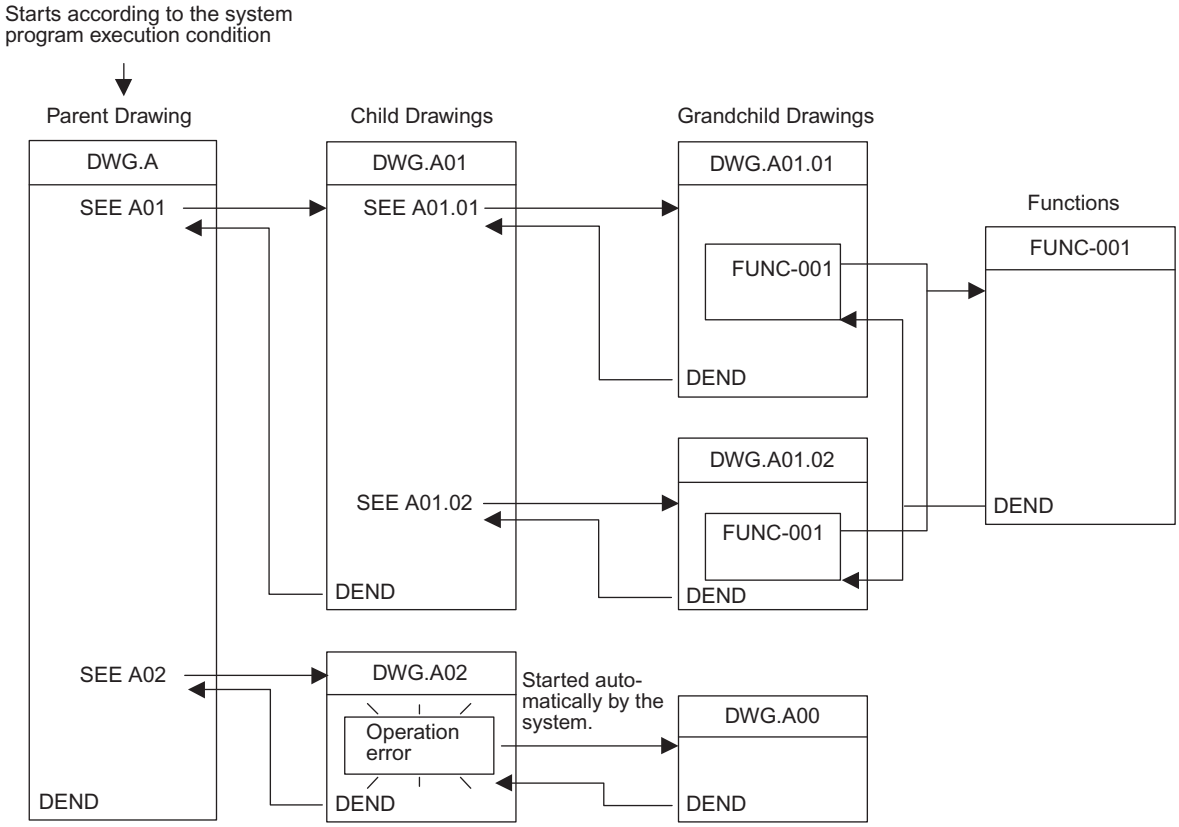


Note: Substitute A, I, H, or L for X.



(4) Execution Processing Method of Drawings

Drawings in the hierarchy are executed by the lower-level drawings being called from upper-level drawings. The execution method is shown below, using DWG.A as an example.



Note: 1. A parent drawing is automatically called by the system. The user can execute any child or grandchild drawing by programming an instruction that calls a drawing (the SEE instruction) in a parent or child drawing.
2. Functions can be called from any drawing. A function can also be called from a function.
3. If an operation error occurs, the operation error drawing corresponding to the drawing will be called.

5.2.3 Motion Program

Motion programs are written in a text-based language called motion language. Up to 256 motion programs can be created separate from ladder drawings.

The following table shows the two types of motion programs.

Classification	Designation Method	Feature	Number of Programs
Main Programs	MPM□□□ 1 to 256	Can be called from DWG.H.	A total of up to 256 main programs and subprograms can be created.
Subprograms	MPS□□□ 1 to 256	Can be called from the main programs.	

IMPORTANT

Each MPM□□□ and MPS□□□ program number must be unique.

With the MP2100, up to 16 motion programs can be executed at the same time. If 17 or more motion programs are executed, an alarm (no system work error) will occur.

- No system work error: Bit E in the first word of the MSEE work registers

To specify a motion program, use either of the following two methods: direct specification of the program number or indirect specification by specifying the register number where the program number is stored.

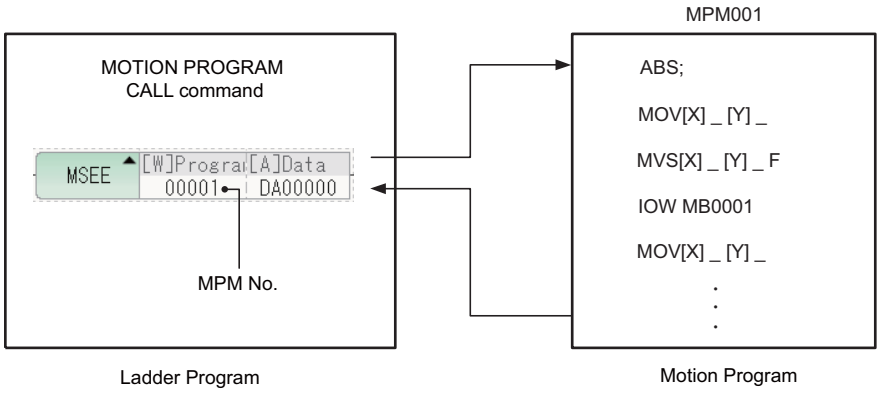


Fig. 5.1 Calling a Motion Program by Direct Specification

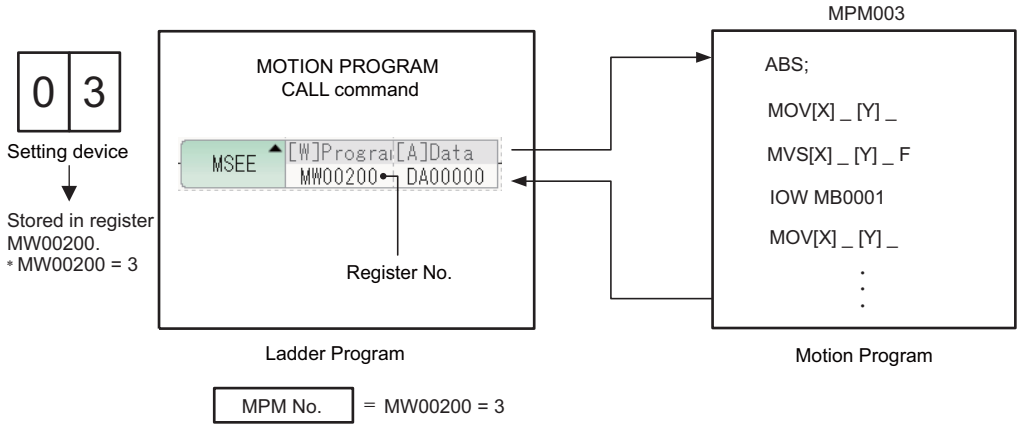


Fig. 5.2 Calling a Motion Program by Indirect Specification

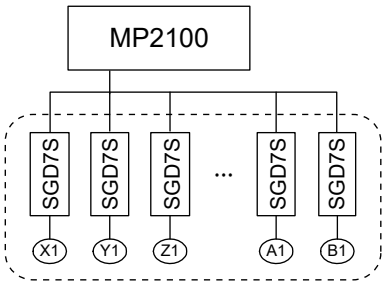
(1) Groups

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 MP2100/MP2100M to independently control multiple machines using group operation. Group operation can be single group operation or multiple group operation.

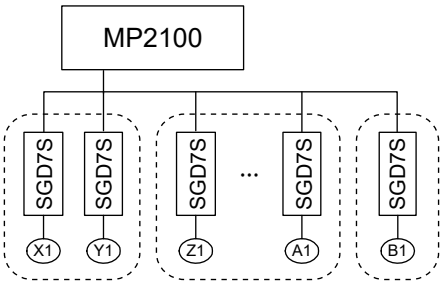
Definitions for axes to be grouped together are made under Group Definitions. For details on group definitions, refer to the following manuals.

System Integrated Engineering Tool MPE720 Ver.7 USER'S MANUAL (Manual No.: SIEP C880761 03)

<Operation as One Group>



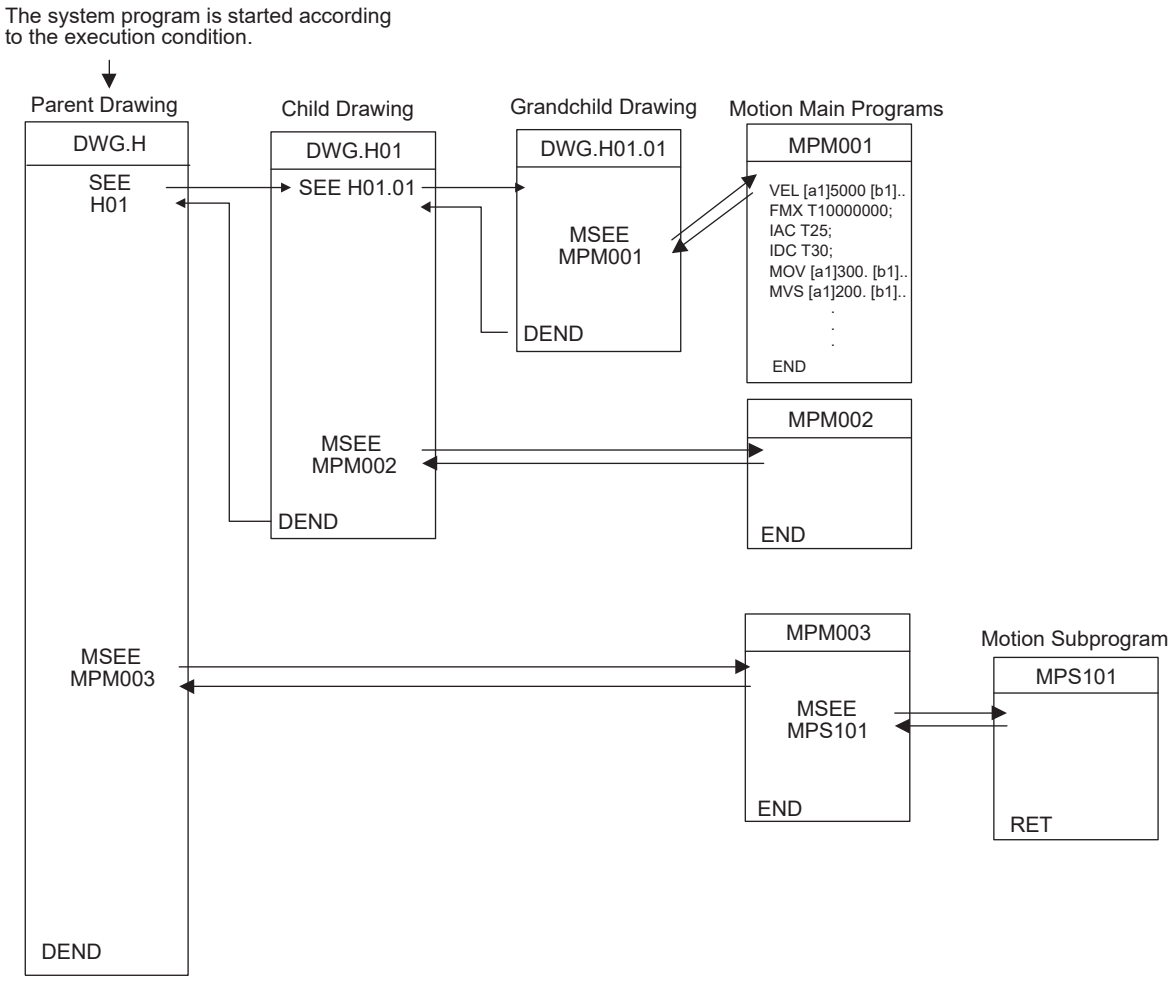
<Operation with Multiple Groups>



(2) Motion Program Execution Example

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.

A motion program execution example is shown below.



In each high-speed scan cycle, the ladder instructions for DWG.H are executed in the following hierarchical order: Parent drawing - child drawing - grandchild drawing.

Motion programs are called in the scan cycle, but as with ladder programs, all programs cannot necessarily be executed in one scan. Motion programs are executed and controlled by special system motion management functions.

INFO

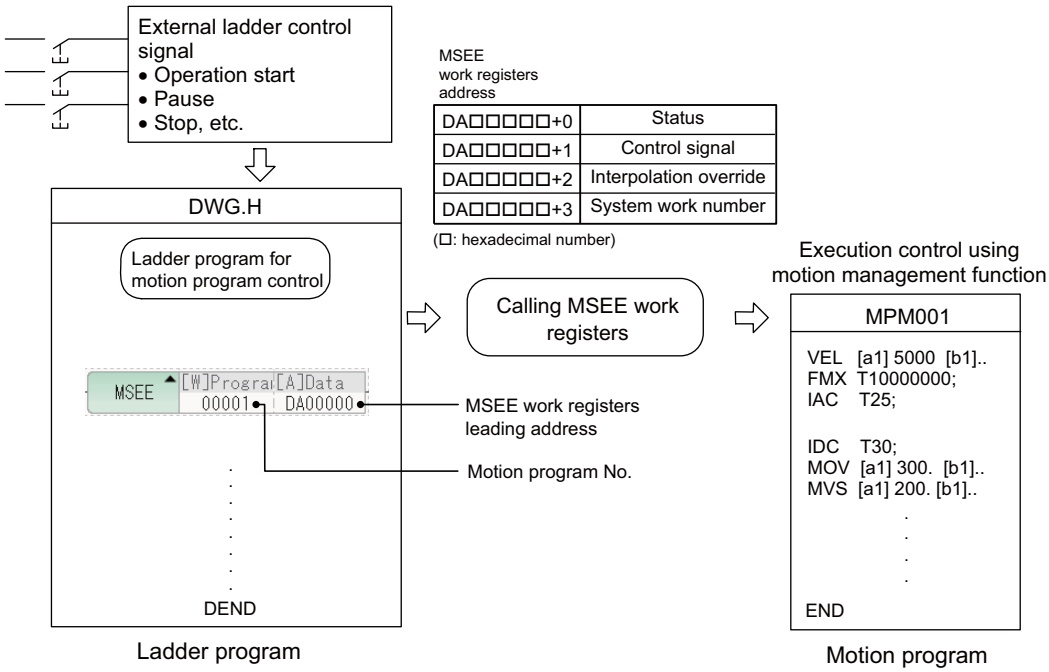
The following points apply to calling motion programs. Call motion program with care.

- More than one motion program with the same number cannot be called using the MSEE instruction.
- Subprograms (MPS□□□) cannot be called from the ladder program MSEE instruction. They can be called only from within motion programs (MPM□□□ and MPS□□□).
- The same subprogram cannot be called from two different locations at the same time.

5.2.4 Motion Programs and MSEE and S Registers

Motion program status, control signal, interpolation override, and system work number data is saved in four MSEE registers (4 words) with a DA□□□□□ (□: hexadecimal number) leading address. This data is called every time the MSEE command is executed in an H drawing. Motion program execution information can be monitored in the S registers.

The following figure shows the method for executing motion programs. MSEE register details and S register descriptions are also provided below from (1) onwards.



Note: For the meaning of register numbers and how to interpret them, refer to *5.3 Registers*.

(1) Motion Program Status Bits (DA□□□□□+0)

The leading word (DA□□□□□+0) in the MSEE work registers contains the motion program status bits for monitoring execution status of the motion program.

The following table shows details of status bit.

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

Note: Alarm details are saved in the S registers.

(2) Motion Program Control Signals (DA□□□□□+1)

Program control signals (e.g., program operation start requests and program stop requests) need to be entered to execute the motion program called from DWG.H using the MSEE command. The second word of the MSEE work registers (DA□□□□□+1) is the motion program control signal.

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
8	Skip 1 information	NO contact
9	Skip 2 information	NO contact
D	System work number setting	NO contact OFF: The system will use the automatically obtained system work number. The system work number may be different each time. ON: The work with the set system work number will be used.
E	Interpolation override setting	NO contact OFF: Interpolation override 100% fixed ON: Conforms to set interpolation override

These signals can perform run, stop, hold, and other controls for motion programs by entering from the ladder program to the work register specified by the MSEE command +1.

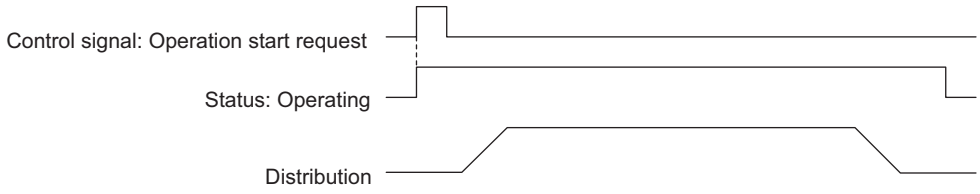
Use signals conforming to the above signal types when writing ladder programs.

Note: Motion programs are executed if the program operation start request signal is ON when the power is turned ON (e.g., when a start request signal is left in M register).

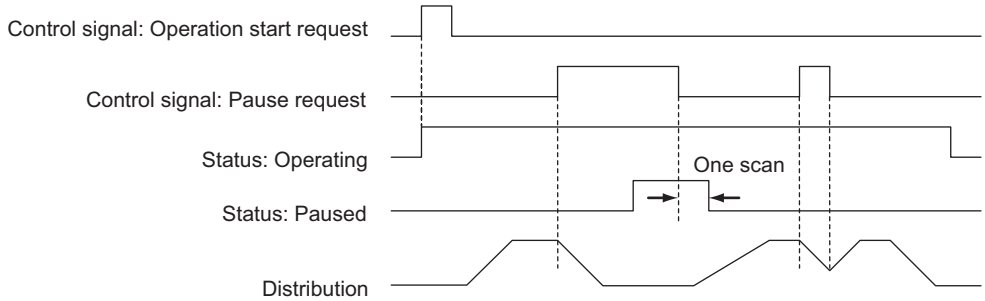
- Timing Chart for Motion Program Control Signals

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

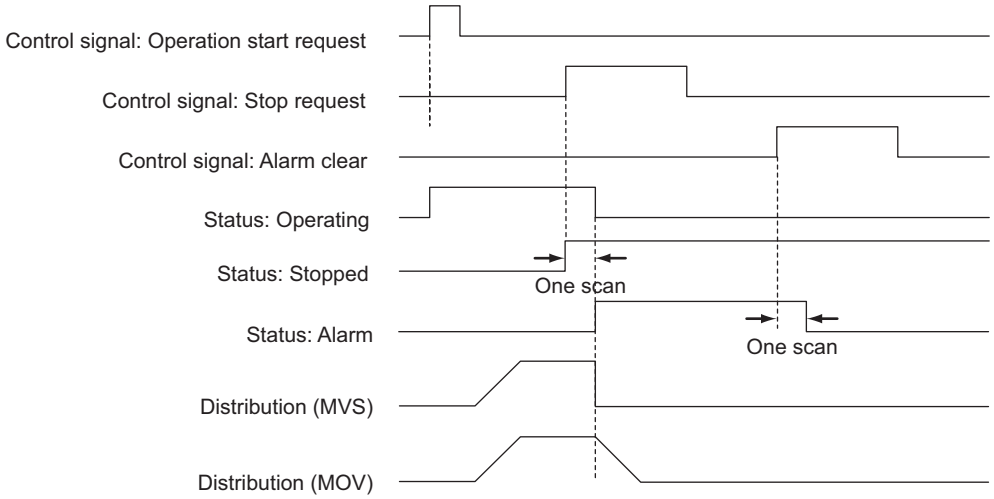
<Program Operation Start Request>



<Pause Request>



<Stop Request>



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

(3) Interpolation Override (DA□□□□□+2)

The override when executing interpolation travel commands (setting; unit: 1 = 0.01%) is set in the third word of the MSEE work registers (DA□□□□□+2).

This interpolation override is enabled only when the motion program control signal bit E (interpolation override setting) is ON.

(4) System Work Number (DA□□□□□+3)

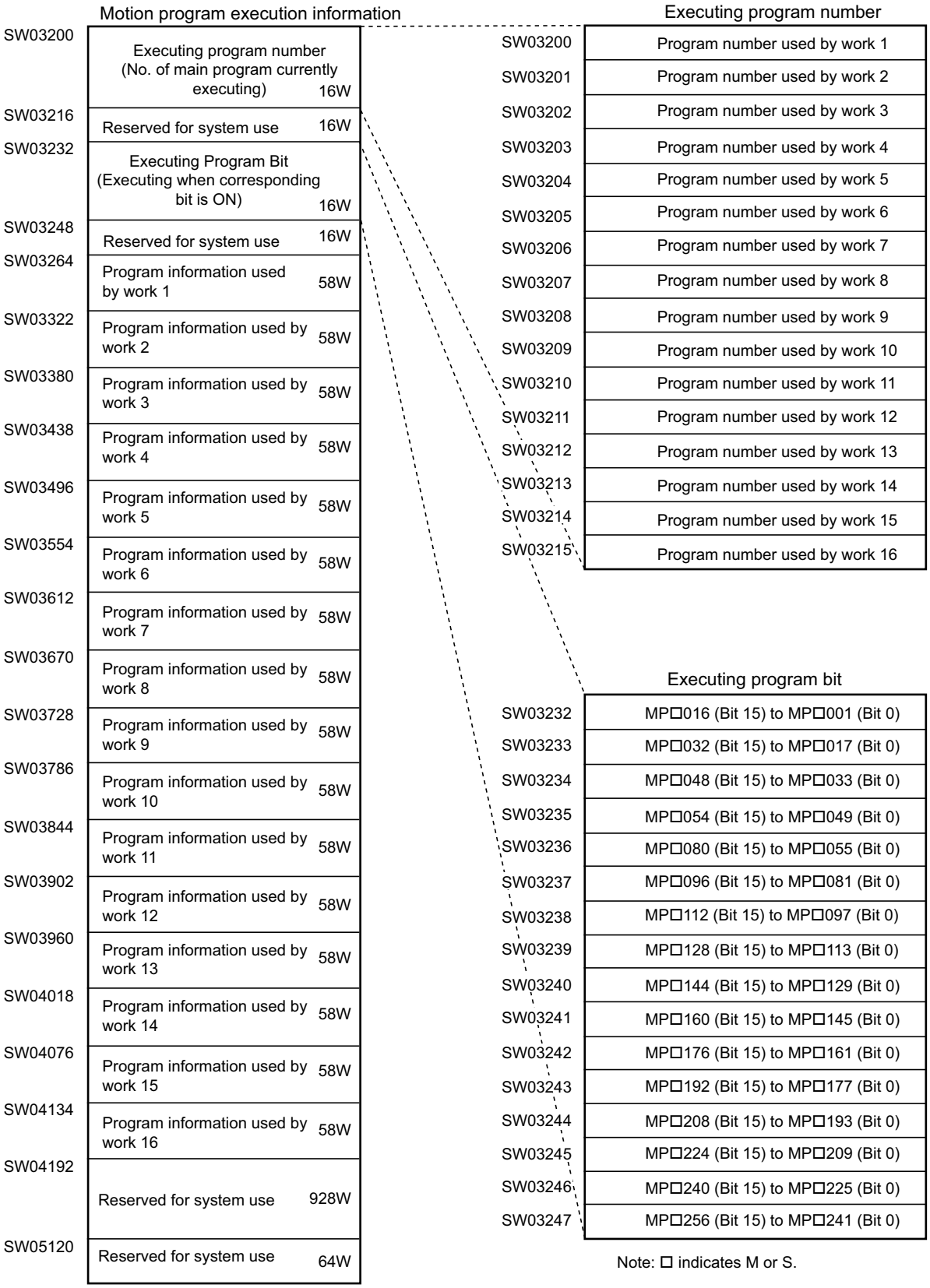
The system work number n (setting range: 1 to 16) used when executing motion programs is set by the fourth word of the MSEE work registers (DA□□□□□+3).

This system work number is enabled only when the motion program control signal bit D (system work number setting) is ON. The status bit, bit E (No system work error), will turn ON if the work number setting is outside the setting range or the specified system work is in use.

(5) Monitoring Motion Program Execution Information Using S Registers

The S registers (SW03200 to SW04191) can be used to monitor motion program execution information.

- Register Areas for Motion Program Execution Information



- Details of Program Information Used by Work n

Program information used by work n

+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

The monitoring method differs depending on the setting for bit D of the motion program control signal (system work number setting).

(a) When Bit D of Motion Program Control Signal (System Work Number Setting) is ON

The execution information is reported to the “Program information used by work n” registers (SW03264 to SW04133).

For example, when the system work number is 1, the motion program execution information can be monitored using SW03246 to SW03321 “Program information used by work 1.”

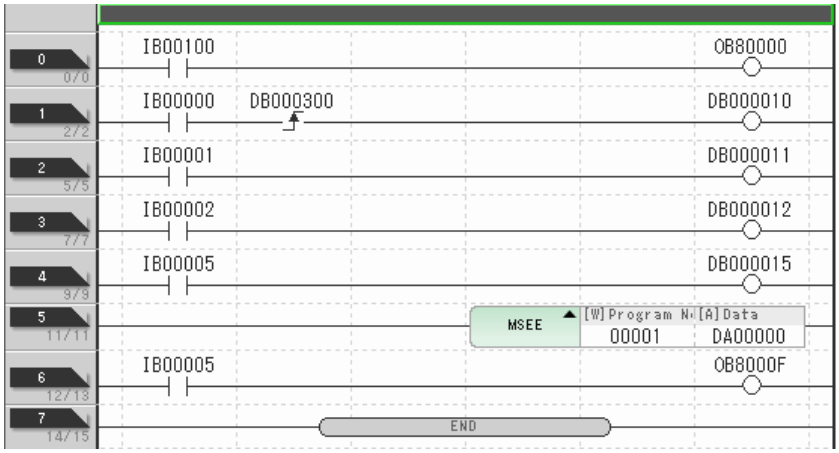
(b) When Bit D of Motion Program Control Signal (System Work Number Setting) is OFF

The system automatically determines the system work to be used. This means that the work being used can be checked under “Executing program number” (SW03200 to SW03215).

For example, if the motion program to be monitored is MPM001, and SW03202 is 001, the number of the work being used is 3. This means that the execution information for motion program MPM001 can be monitored using “Program information used by work 3” (SW03380 to SW03437).

5.2.5 Example of Ladder Programs for Motion Program Control

The following figure shows the minimum ladder programming required for controlling motion programs.



The following table shows the details of the above ladder program.

Step No.	Program Details
1	The servo ON signal (IB00100) sets the Servo ON motion settings parameter (OB80000) and turns ON the Servo.
2 to 10	The signals connected to the MP2100 external input signals are stored as the motion program control signals. IW0000 (external input signal) → DW00001 (Second word of MSEE work registers) • Start program operation • Pause program • Stop program • Reset alarm
11	Calls motion program MPM001 MSEE 00001 DA00000 (1) (2) (1) Motion program number (00001 = MPM001) (2) Leading MSEE work register address
12	Sets motion settings parameter Alarm Clear (OB0000F) using the alarm reset signal (IB00005) and clears the alarm.

If the ladder program is used to enter external input signals connected to the MP2100 (IB00000 to IB00005) to DW00001 (second word of MSEE work registers) as the motion program control signals, motion program operations such as run, pause, and stop can be performed using the system's motion management function.

The following tables show an example of the minimum external input signals required to create the ladder program.

External Signal Address	External Signal Name
IB00000:	Start program operation
IB00001:	Pause program
IB00002:	Stop program
IB00005:	Reset the alarm.

⇨

Bit No.	Motion Program Control Signal
0:	Program operation start request
1:	Program pause request
2:	Program stop request
5:	Alarm reset request

5.2.6 Functions

Functions are executed by calling them from parent, child, or grandchild drawings using the FSTART 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.

(1) 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

(2) 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.

- *Machine Controller MP2000 Series USER'S MANUAL LADDER PROGRAMMING* (Manual No.: SIE-C887-1.2)
- *Machine Controller MP2000 Series USER'S MANUAL USER'S MANUAL* (Manual No.: SIEP C880700 38)

5.3 Registers

This section explains the types of registers used in MP2100/MP2100M user programs (mainly ladder programs) and how to use them.

5.3.1 Types of Registers

(1) DWG Registers

DWG Registers are 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.	

* 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.
For details, refer to the *Machine Controller MP900/MP2000 Series MPE720 Software for Programming Device User's Manual* (Manual No.: SIEP C880700 05).

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.)
For details, refer to *5.3.2 Data Types and Register Specifications*.

IMPORTANT

S and M register data has a battery backup to ensure the data is held even if the MP2100/MP2100M power is turned OFF and ON. When data is to be held even when the power is turned OFF and back ON, Yaskawa recommends using the battery backup function to write the data to the M registers.

(2) Function Registers

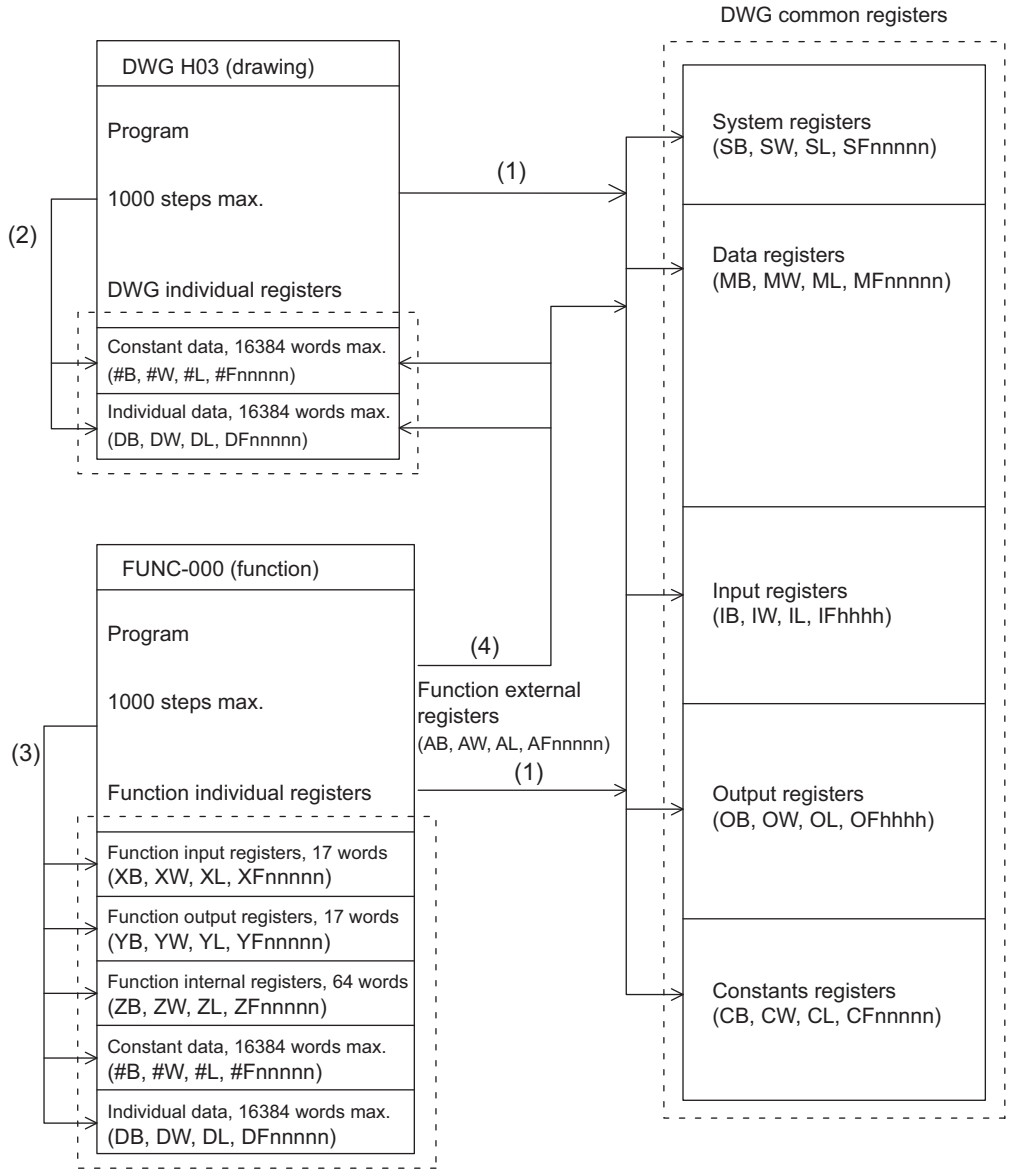
The following table shows the registers that can be used with each function.

Type	Name	Specification Method	Range	Details	Character-istics
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	Function internal registers	ZB, ZW, ZL, ZFnnnnn	ZW00000 to ZW00063	Internal registers unique to each function. Can be used for function internal processing.	
A	Function external registers	AB, AW, AL, AFnnnnn	AW00000 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. 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, OFnnnnn (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. For details, refer to *5.3.2 Data Types and Register Specifications.*
3. SA, MA, IA, OA, DA, #A, and CA registers can be used within functions.

(3) Register Ranges in Programs

The register ranges in the programs are shown below.



(1): DWG common registers can be called from any drawing or function.

(2): DWG individual registers can be called only from within the drawing.

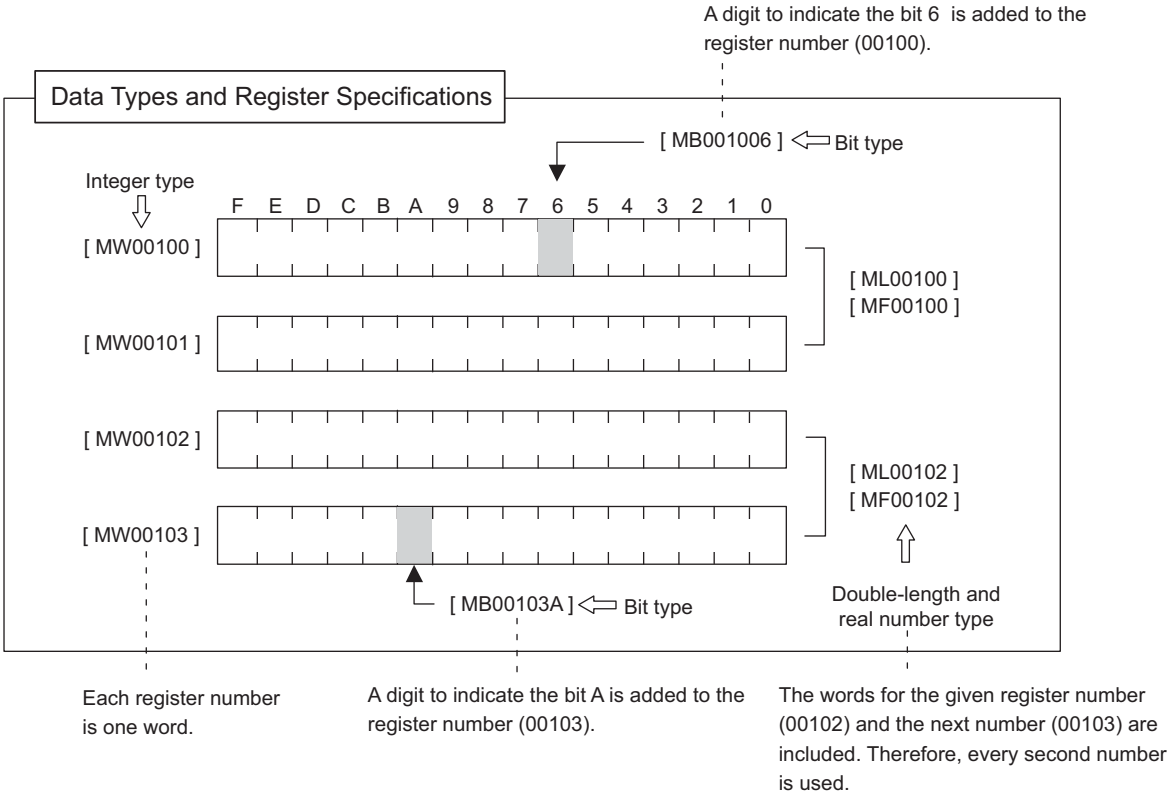
(3): Function individual registers can be called only from within the function.

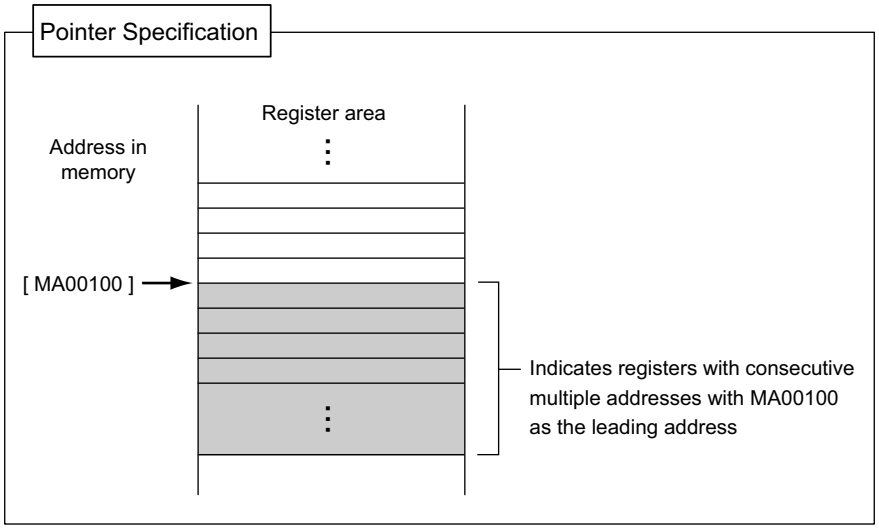
(4): DWG individual registers and DWG common registers can be called from functions using the external registers for the functions.

5.3.2 Data Types and Register Specifications

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	0, 1	Used by relay circuits.
W	Integer	–32768 to +32767 (8000H) (7FFFH)	Used for numeric value operations. The values in parentheses ( ) indicate 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 Using i and j Subscripts

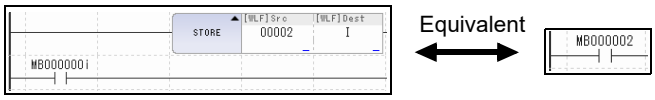
Two special register modifiers, i and j, can be used with relay and register numbers. The functions of i and j are identical. They are used for handling register numbers as variables.

Examples of each register data type are used to explain the use of i and j.

(1) Bit Registers with Subscripts

These are the same as when i or j values are added to relay numbers.

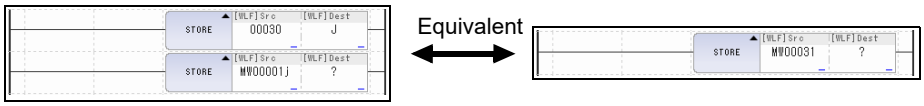
For example, when i = 2, MB000000i is the same as MB000002. And when j = 27, MB000000j is the same as MB00001B.



(2) Integer Registers with Subscripts

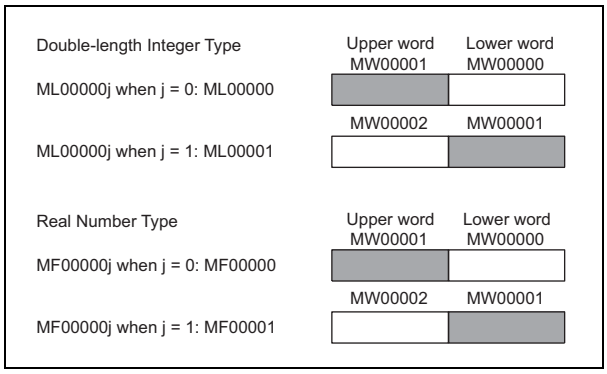
These are the same as when i or j values are added to register numbers.

For example, when i = 3, MW00010i is the same as MW00013. And when j = 30, MW00001j is the same as MW00031.



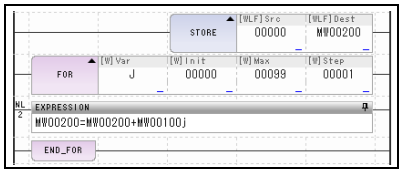
(3) Double-length Integers and Real Numbers with Subscripts

These are the same as when i or j values are added to register numbers. For example, when j = 1, ML00000j is the same as ML00001. And when j = 1, MF00000j is the same as MF00001. For double-length integers and real numbers, the word that indicates a register is the lower word. Caution is required because this means that ML00001 and MF00001 will be different when j = 0 and when j =1 because the upper and lower words will differ as shown in the following example.



<Example Program Using Subscripts>

The following program finds the total of 100 registers from MW00100 to MW00199, using subscript j, and writes the total to MW00200.



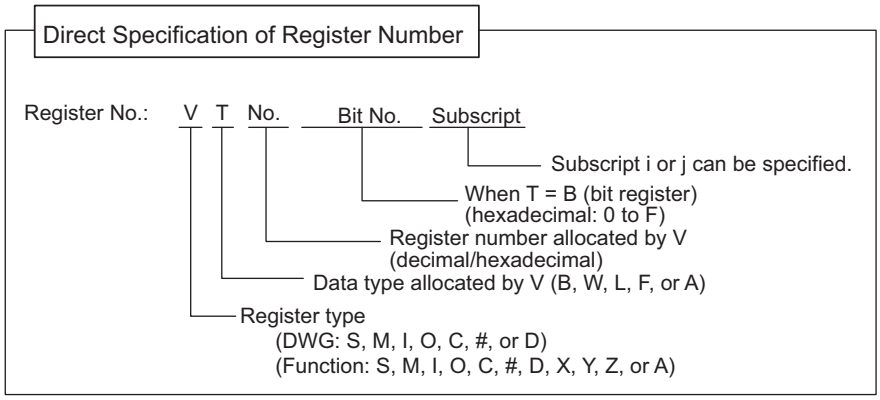
5.3.4 Register Specification Methods

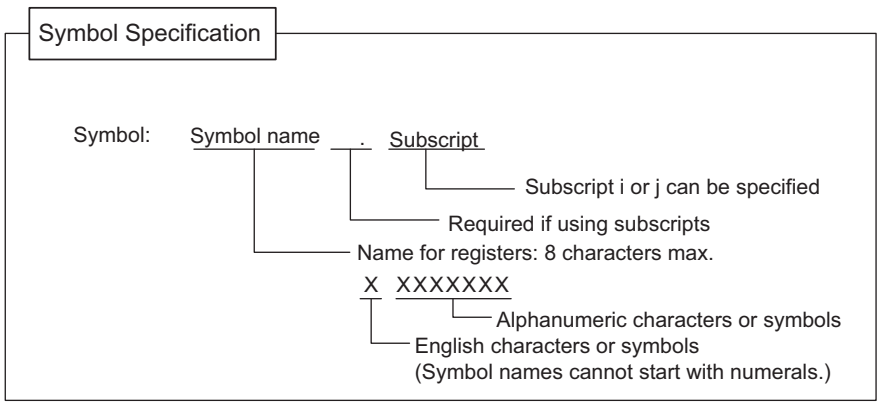
Registers can be specified directly by register number or by symbol (register name) specification. A combination of both of these register specification methods can be used in ladder programs.

When using the symbol specification method, the relationship between symbols and register numbers must be defined.

The following table shows the register specification methods.

Specification Method	Specification Example by Data Type
Register number direct specification	Bit register: MB00100AX Integer register: MW00100X Double-length integer register: ML00100X Real number register: MF00100X Address register: MA00100X X: When specifying subscripts, subscript i or j is added after the register number.
Symbol specification	Bit register: RESET1-A.X Integer register: STIME-H.X Double-length integer registers: POS-REF.X Real number registers: IN-DEF.X Address registers: 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.



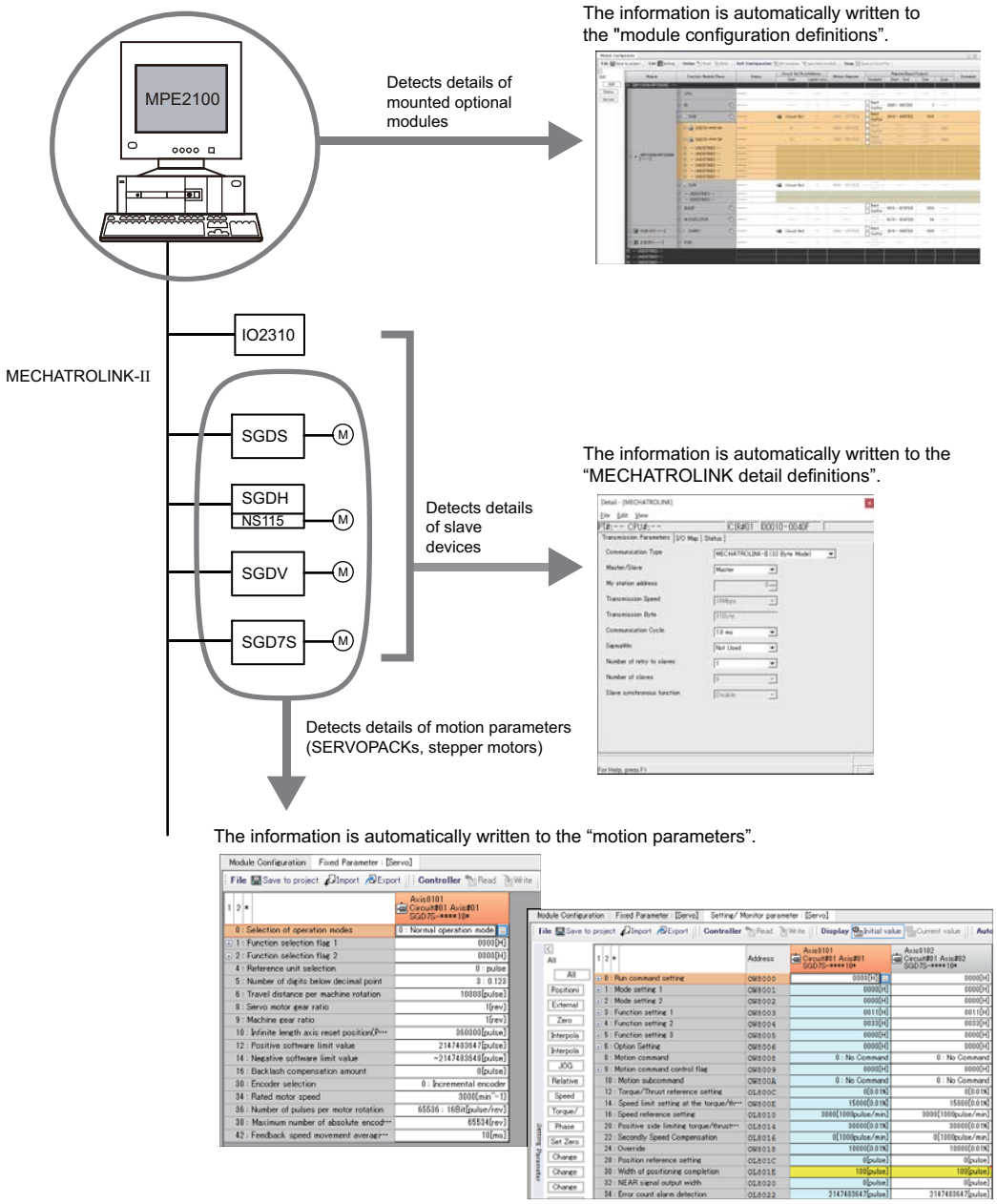


5.4 Self-configuration

When the self-configuration function is executed, the MP2100/MP2100M recognizes the mounted Optional Modules, and automatically creates the Module Configuration Definition, Module Details Definition files of each module. The self-configuration function greatly reduces the system startup time.

The following figure shows how the self-configuration function works.

<Execution of Self-configuration>



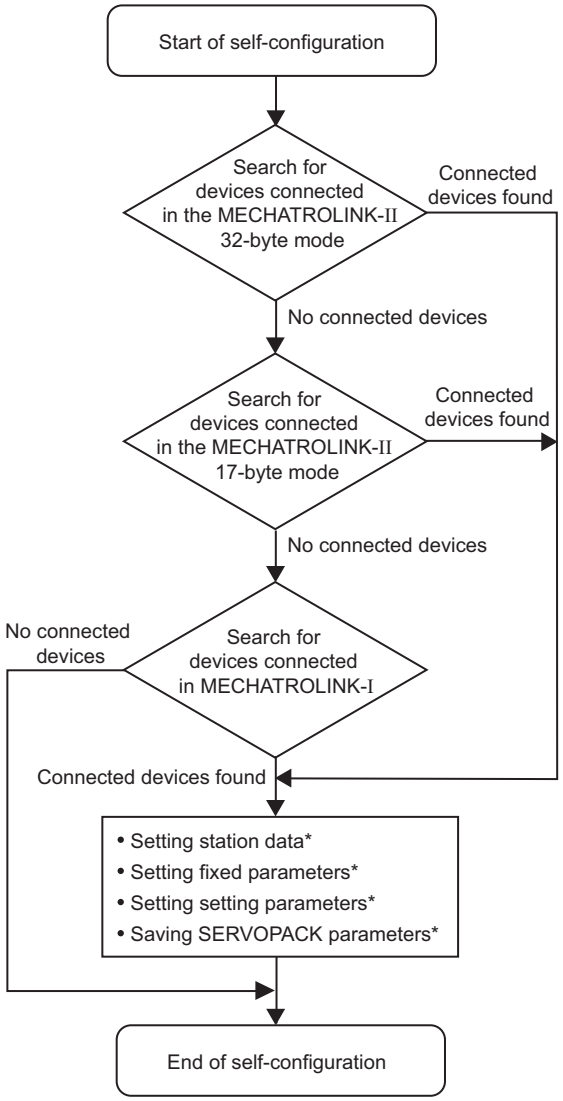
Self-configuration can be executed with the switch of the MP2100/MP2100M, or from the MPE720.

Note: For the procedure to execute the self-configuration, refer to *5.4.2 Procedure for Self-configuration Using Mode Switch* and *5.4.3 Procedure for Self-configuration Using MPE720*.

5.4.1 Details of Self-configuration

Self-configuration collects MECHATROLINK detail definition data and slave data according to the sequence shown below.

The MP2100 decides the communication type at the point when a slave is detected, and after that point neither communication type selection nor slave detection is performed. If not even a single slave station has been detected, MECHATROLINK-I communication is continued.



Note: 1. For each communication, the slave devices are detected in the order SERVOPACKs, then I/O devices then inverters.

2. At stations where there is a communication error or no response, for example if the station number is duplicated or a cable is broken, the status is recognized as “no connection.”

3. For details on setting station data, setting fixed parameters, setting setting parameters and saving SERVOPACK parameters, refer to *4.4 System Startup Using Self-configuration* and *4.1.6 Executing the MP2100 Self-configuration*.

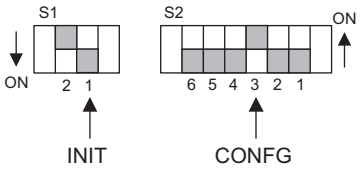
5.4.2 Procedure for Self-configuration Using Mode Switch

Self-configuration can be executed from the mode switch of MP2100/MP2100M.

(1) Executing Self-configuration for the First Time after Connecting Devices

Turn ON the power to the MP2100 and then use the procedure described below. With this operation, self-configuration will be executed for all modules and all new definition files will be created. In addition, ladder drawings, functions, and all register data will be cleared.

In the following procedure, it is assumed that the power supply to all SERVOPACKs are already turned ON.

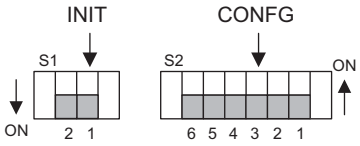


1. Turn OFF the power supply.
   Turn OFF the power supply to the MP2100.

2. Set the mode switch.
   Turn ON the INIT switch of mode switch 1 (S1),
   and the CNFG switch of mode switch 2 (S2), of the MP2100.

3. Turn ON the reset switch.

4. Check the LED indicators.
   Check that the LED indicators on MP2100 change as shown on the left.



5. Change the mode switch settings.
   Turn OFF the INIT switch of mode switch 1 (S1),
   and the CNFG switch of mode switch 2 (S2), of the MP2100.

(2) Self-configuration after Adding Devices Such as SERVOPACKs

For self-configurations after having added network devices such as SERVOPACKs, leave the INIT switch (S1) at OFF in step 2 of the above procedure.

Note: For network devices with existing definitions files, correctly connect and turn ON the power to the devices when executing self-configuration.

IMPORTANT

- If register allocations have been changed manually since the last time self-configuration was executed, the I/O addresses will be automatically changed when self-configuration is executed again. If the SVR is set to disabled (UNDEFINED), the setting will return to enabled. To keep the changed register allocations, do not use self-configuration, but manually make the register allocations for added devices and refresh the definitions file.

■ Power Restart and RAM Data Clear

If the INIT switch of the mode switch on the MP2100 is ON and the power is turned ON, RAM data will be cleared.

Also, flash memory data will be read and RAM data will be overwritten when the INIT switch is OFF and the power is turned ON. In either case, the RAM data will be cleared by turning the power ON. Therefore, always save data to the MP2100 flash memory before turning OFF the power when writing or editing programs.

For information on how to save data to flash memory, refer to *4.2.3 (1) Saving to Flash Memory*.

■ Turning OFF Power after Executing Self-configuration

Do not turn OFF the power supply to the MP2100 after executing self-configuration until the definitions data has been saved to flash memory in the MP2100. If the power is turned OFF somehow before the data is saved to flash memory, execute self-configuration again.

5.4.3 Procedure for Self-configuration Using MPE720

Executing self-configuration from the MPE720 allows self-configuration for individual modules in addition to self-configuration for all the modules.

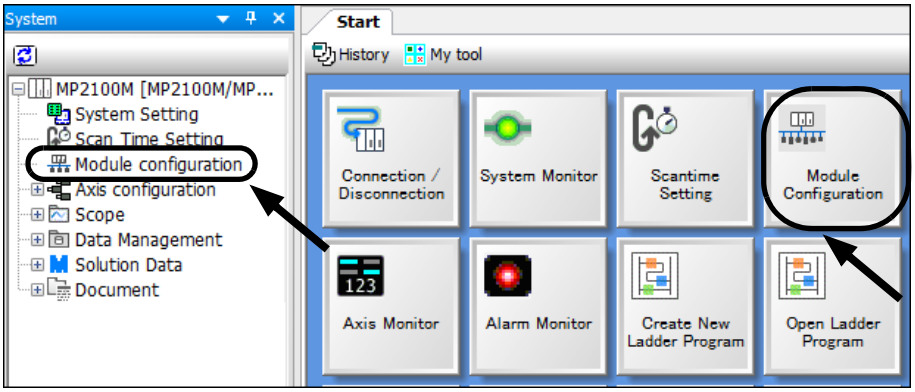
(1) Self-configuration for All the Modules

Execute self-configuration for all the Modules when executing self-configuration for the first time after connecting devices.

Note: After having added or removed modules or devices, use not the procedure shown here but the procedure described in *(2) Self-configuration of Each Module* to detect the configuration. If self-configuration for all the modules is executed, the parameter settings up to that point may be overwritten.

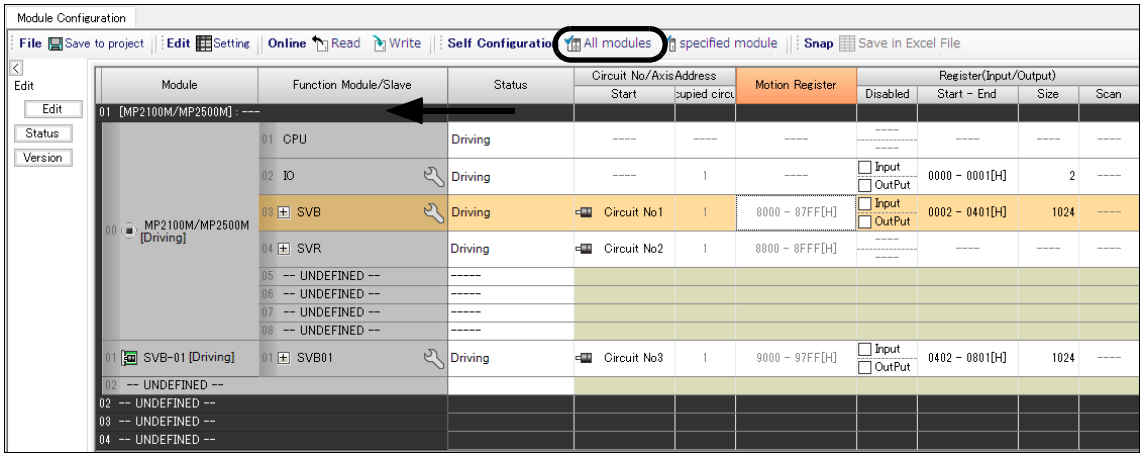
1. Open the Module Configuration Definition Window of MPE720.

   Start MPE720, and then open the target project file. Click **Module configuration** in the main window. Or, double-click **Module configuration** in the System sub programs.



   The Module Configuration Definition Window is displayed.

2. Click the **All modules** button to execute self-configuration.



3. On the Module Configuration Definition Window, the devices currently connected to the MP2100M and the station numbers of those devices are displayed.

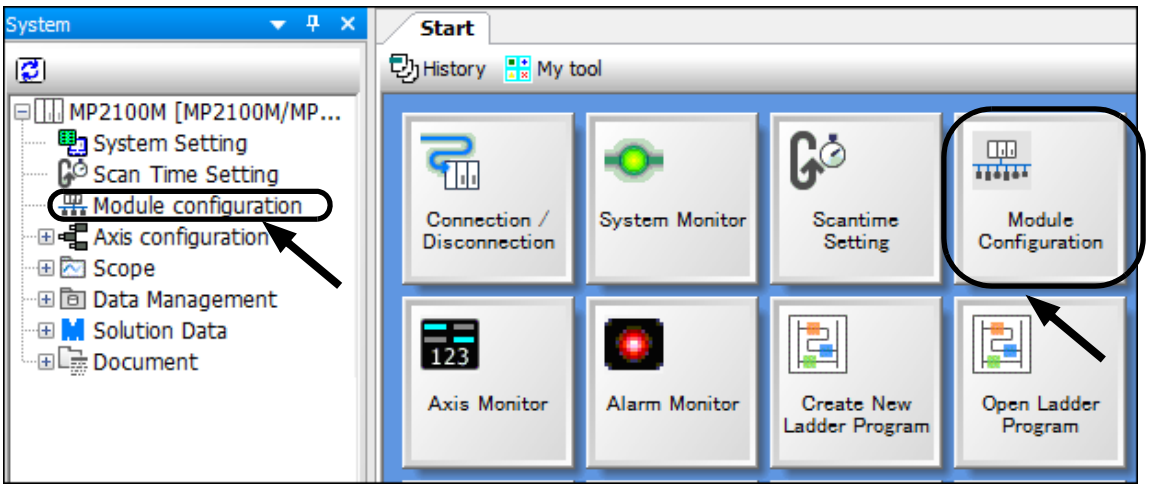
4. Select ***Online - Save to FLASH*** from the main menu to save the definitions data to the flash memory.

(2) Self-configuration of Each Module

If devices are added, self-configuration can be executed separately for the Module (port) that has been changed.

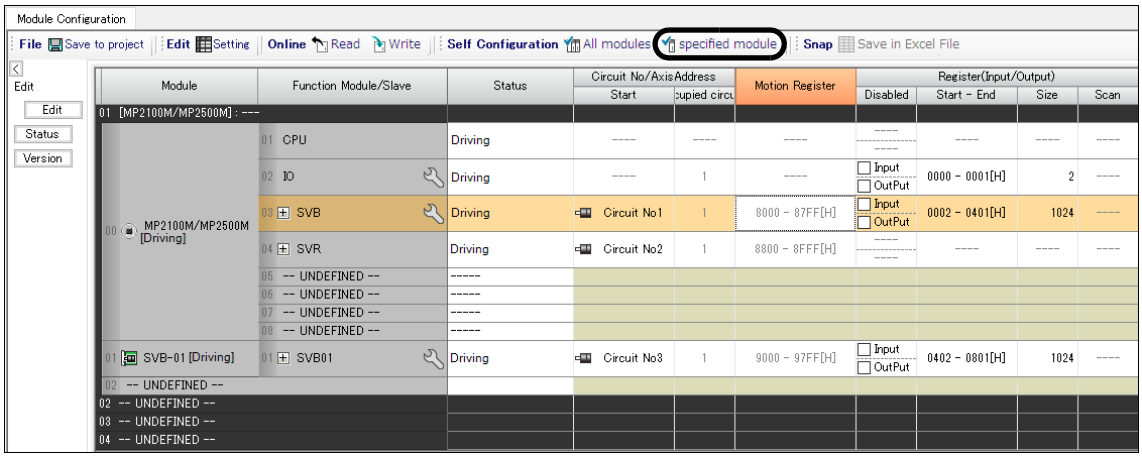
1. Open the Module Configuration Definition Window of MPE720.

   Start MPE720, and then open the target project file. Click **Module Configuration** in the main window. Or, double-click **Module configuration** in the System sub programs.



   The Module Configuration Definition Window is displayed.

2. Select the cell of the module which added the device, and click the **specified module** button to execute self-configuration.



   Note: If executing self-configuration for each module when MP2100M is selected as an individual module, all the modules will be self-configured.

3. Select ***Online - Save to FLASH*** from the main menu to save the definitions data to the flash memory.

5.4.4 Definition Data Updated by Self-configuration

This section shows the definition data of the MP2100/MP2100M Basic Module that is updated on executing self-configuration.

(1) I/O Allocation

Item	Allocation
Digital inputs (18 points)	IW0000
Digital output (4 points)	OW0001
MECHATROLINK	I/O leading registers: IW/OW0010 I/O end registers: IW/OW040F (Input registers: IW0010 to IW040F Output registers: OW0010 to OW040F)

(2) MECHATROLINK Detail Definition Data

The MECHATROLINK detail definition is automatically set as shown in the following table in accordance with the detected communication type and number of slaves.

Communication Type	MECHATROLINK-II (32 Bytes)		MECHATROLINK-II (17 Bytes)		MECHATROLINK-I
Transmission speed	10 Mbps		10 Mbps		4 Mbps
Number of transmission bytes	32		17		17
Communication cycle	1 ms*	2 ms*	1 ms		2 ms
Max. number of slave stations	*	*	14	15	14
Number of retry stations	*	*	1	0	–
SigmaWin	Not available		Not available		–

Note: The number of retry stations (communication cycle with MECHATROLINK-II (32-byte)) is selected as shown in the following table according to the highest station number among the detected slave stations.

Highest Slave Station Number	Communication Cycle (ms)	Number of Retry Stations
1 to 8	1	1
9	1	0
10 to 16	2	5
17 to 21	2	According to the following expression: 21 - (highest station number)

■ Wild Card I/O (*****I/O) and Wild Card Servo

If the module cannot be automatically detected with self-configuration, “Wild Card I/O (*****I/O)” or “Wild Card Servo” will be displayed in the **Function Module/Slave** column.

(3) Motion Parameters

As a result of executing self-configuration, the motion parameters are set based on the information of the SERVOPACK. Some of the parameters are written to the RAM of the SERVOPACK.

For details of the data written, refer to *11.6.5 Parameters Updated during Self-configuration* in the *Machine Controller MP2000 Series Built-in SVB/SVB-01 Motion Module User’s Manual* (Manual No.: SIEP C880700 33).

(4) SERVOPACK Parameters

As a result of executing self-configuration, the SERVOPACK parameters are written to the SERVOPACK's EEPROM or RAM. However, they are not written to the settings of the SERVOPACK parameters saved in the MP2100/MP2100M Basic Module and SVB-01 Module.

For details of the data written, refer to *11.6.5 Parameters Updated during Self-configuration* in the *Machine Controller MP2000 Series Built-in SVB/SVB-01 Motion Module User's Manual* (Manual No.: SIEP C880700 33).

Note: Operation at MPE720 is required to save SERVOPACK parameters in the MP2100/MP2100M Basic Module.

5.5 Precautions when Using the MP2100/MP2100M

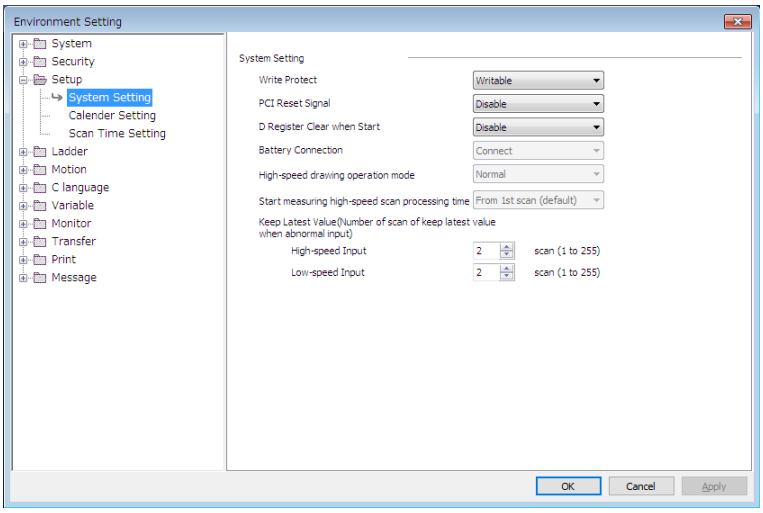
This section explains precautions for setting or changing user definition files and for setting the scan times.

5.5.1 Precautions when Setting or Changing User Definition Files

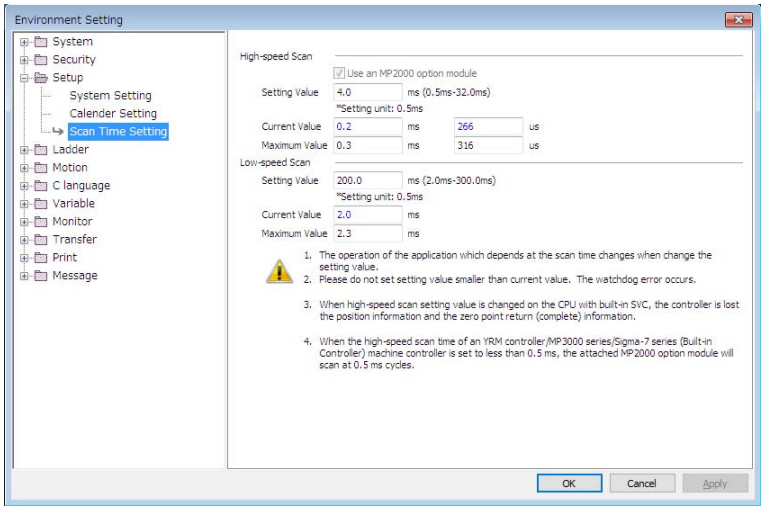
User definition files (system settings, scan time settings, and module configuration definitions) must be saved to flash memory.

Be sure to save the results to flash memory using the MPE720 after setting or changing these user definition files. If data is not saved to flash memory, the settings and changes will be lost when the power supply to the MP2100/MP2100M is turned OFF and back ON.

System Setting (Select ***File*** - ***Environment Setting*** - ***Setup*** - ***System Setting*** from the menu bar.)



Scan Time Setting (Select ***File*** - ***Environment Setting*** - ***Setup*** - ***Scan Time Setting*** from the menu bar.)



5.5.2 Precautions when 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 MP2100/MP2100M OFF and back ON.

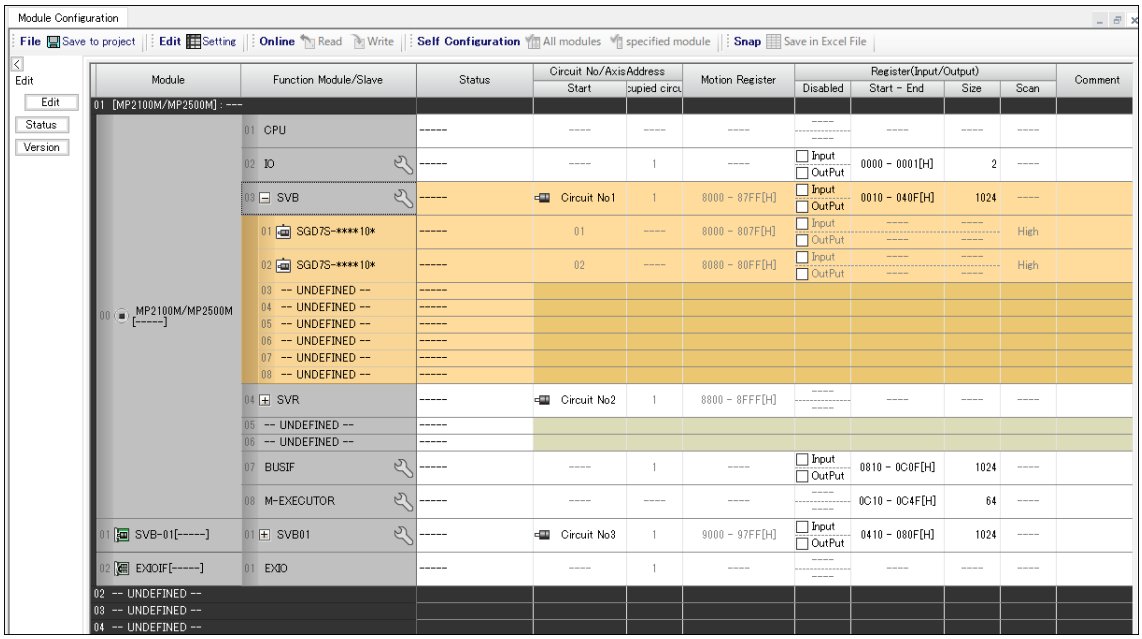
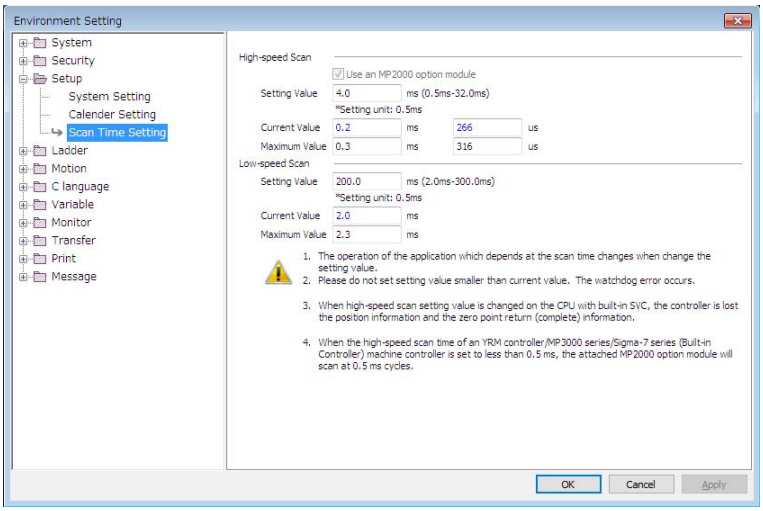


Fig. 5.3 Module Configuration Definition Window

5.5.3 Precautions when Setting or Changing the Scan Time

Scan time can be set or changed in the following window.

Scan Time Setting (Select ***File*** - ***Environment Setting*** - ***Setup*** - ***Scan Time Setting*** from the menu bar.)



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. Use the following formula to calculate the scan.

  (set value − maximum time to execute scan) ≥ (0.2 × set values of the scan time)

  The set values of the scan time should be to at least 1.25 times greater than the maximum time required for the scan.

  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 MP2100 system will stop.
- Set the set values of the high-speed (H) and low-speed (L) scan time to an integral multiple of the MECHATROLINK communication cycle (1 or 2 ms) set in the MP2100. Always check the set values of the scan time after changing the MECHATROLINK communication cycle.
- 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.

(1) Scan Time Setting Examples

- When the communication cycle is 1 ms (with MECHATROLINK-II only) and the maximum scan time value is 0.8 ms

  High-speed scan (or low-speed scan) set value ≥ 1.25 × 0.8 (= 1 ms)

  High-speed scan (or low-speed scan) set value = 1 ms, 2 ms, 3 ms, ... (an integer of 1 ms or higher)

- When the communication cycle is 1 ms (with MECHATROLINK-II only) and the maximum scan time value is 1.4 ms

  High-speed scan (or low-speed scan) set value ≥ 1.25 × 1.4 (= 1.75 ms)

  High-speed scan (or low-speed scan) set value = 2 ms, 3 ms, ... (an integer of 2 ms or higher)

- When the communication cycle is 2 ms (with MECHATROLINK-I or MECHATROLINK-II) and the maximum scan time value is 0.8 ms

  High-speed scan (or low-speed scan) set value ≥ 1.25 × 0.8 (= 1 ms)

  High-speed scan (or low-speed scan) set value = 1 ms, 2 ms, 4 ms, ... (1 ms or 2 ms or an integral multiple of 2 ms)

- When the communication cycle is 2 ms (with MECHATROLINK-I or MECHATROLINK-II) and the maximum scan time value is 1.4 ms

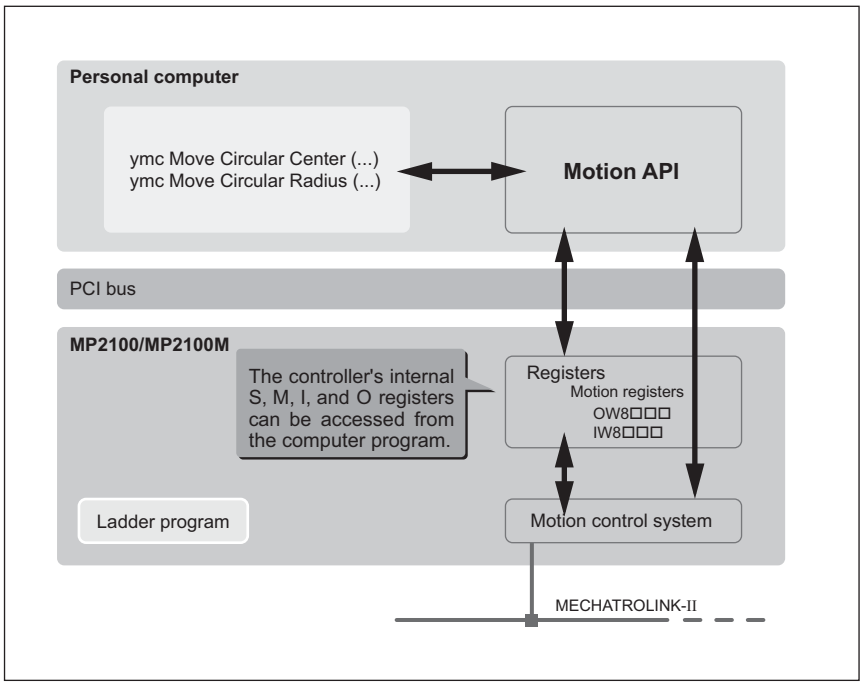
  High-speed scan (or low-speed scan) set value ≥ 1.25 × 1.4 (= 1.75 ms)

  High-speed scan (or low-speed scan) set value = 2 ms, 4 ms, ... (2 ms or an integral multiple of 2 ms)

5.6 Motion API

5.6.1 Overview of the Motion API

The Motion API is a group of C-language service functions that send commands to the MP2100/MP2100M from a user application running in the host computer. Commands such as motion commands can be sent easily from the host computer based on the mechanical characteristics of the system.



(1) Main Motion API Functions

The main motion API functions are listed below. Refer to *Appendix C Motion API* for more details.

(a) Common APIs

- Device
- Unit conversion function
- Parameter operations

(b) Sequential APIs

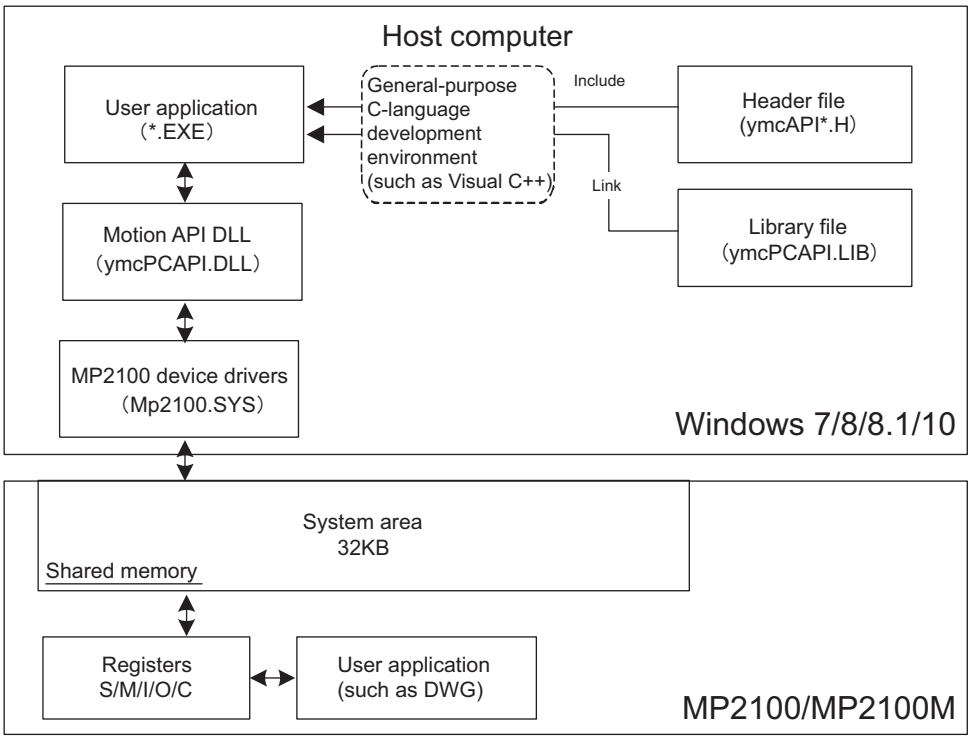
- Positioning function
- Interpolation function
- Torque reference function
- Synchronization function
- Compensation function
- Motion operations
- Driver operations

(c) System APIs

- Data operations
- System information
- System operations
- Calendar operations

5.6.2 Motion API Software

The following diagram shows the software configuration of the MP2100/MP2100M's Motion API.



The following table describes the files used in the diagram above.

File Name	Description
Mp2100.sys	This device driver transfers data to the MP2100/MP2100M. The driver is registered in the Windows system when the MP2100/MP2100M package is installed in the computer.
ymcPCAPI.LIB	This library file is needed to develop applications that use the Motion API.
ymcAPI∗.H	This header file is needed to execute applications that use the Motion API.
ymcPCAPI.DLL	This DLL is needed to execute the Motion API.

5.6.3 Installed Files List

The following table lists the files that are copied to the host computer when the MP2100/MP2100M's Motion API is installed.

File Name	Description	Destination Directory
Mp2100.sys	This device driver transfers data to the MP2100/MP2100M.	\WINNT\system32\drivers
ymcPCAPI.LIB	This library file is needed to develop application programs that use the Motion API. Link this file when linking the application program.	\YeTools\PCAPI\Lib
ymcAPI∗.H	This header file is needed to execute applications that use the Motion API.	\YeTools\PCAPI\Include
ymcPCAPI.DLL	This DLL is needed to execute the Motion API. When executing an application that uses the Motion API, either copy this file directly to the directory containing the application or copy it through a system directory with a path such as \WINNT\system32.	\YeTools\PCAPI\Dll

5.6.4 Installing MP2100/MP2100M

(1) Overview of the Software Package

The following table shows an overview of the MP2100/MP2100M's Motion API package.

Name	Model	Specifications	Package Overview
Motion API for the MP2100/MP2100M	CPMC-MPA700	Motion API for the MP2100/MP2100M (1 CD-ROM)	The following files are copied by the Installer program. • Dynamic link library (.dll) • Library (.lib) • Header file (.h) • Driver • Communication Platform • Help file • Sample program

Note: Use Motion API Ver. 1.05 or later for the MP2100M.

(2) MP2100/MP2100M Motion API Installation Procedure

Refer to the Motion API Manual (MotionAPI(PCAPI)_E.chm) on the CD-ROM for the Motion API (model: CPMC-MPA700).

6

Maintenance, Inspection, and Troubleshooting

This chapter explains items that must be inspected on a daily or regular basis as well as probable causes and troubleshooting measures for errors that might occur.

6.1 Inspection Items

This section explains daily and regular inspection items that must be performed by the user.

6.1.1 Daily Inspections

The following table lists the daily inspection items.

No.	Inspection Item		Inspection Details	Criteria	Action
1	Installation conditions of panel		Check the mounting screws for looseness.	The screws must be secured correctly.	Retighten the screws.
2	Connection conditions		Check the connectors for looseness.	The connectors must be tight.	Retighten the connector set screws.
3	LED Indicators	S1	Check whether the indicator is lit (green).	The indicator must be lit (green). (Lighting (red) indicates an abnormality.)	Refer to *6.3 Troubleshooting*.
		S2	Check whether the indicator is lit (green) while the system is in RUN state.	The indicator must be lit (green). (Lighting (red) indicates an abnormality.)	Refer to *6.3 Troubleshooting*.
		TX	Check whether the indicator is lit while the system is in communication state.	The indicator must be lit. It is abnormal if the indicator is not lit.	Refer to *6.3 Troubleshooting*.
		BAT	Check whether the indicator is not lit.	The indicator must be not lit. (The battery voltage is too low if the indicator is lit.)	Replace the battery.

6.1.2 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 system is relocated or modified or when the wiring is changed.

PROHIBITED

- Do not replace the built-in fuse.
  If the user replaces the built-in fuse, the MP2100/MP2100M may malfunction or break down.
  Contact your Yaskawa representative.

Periodic Inspection Items

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 MP2100/ MP2100M 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	PS Module	Measure the voltage between 24-VDC terminals.	19.2 to 28.8 VDC	Change the power supply as necessary.
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 LED indicator on the front panel of the MP2100/MP2100M.	The BAT LED indicator must be not lit.	If the BAT LED indicator is lit, replace the battery.

6.2 Replacing the Battery

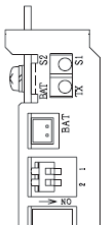
One battery can be connected to MP2100/MP2100M. This battery serves to back up the data recorded in the MP2100/MP2100M so that it is not lost when the power supply to the MP2100/MP2100M is cut off.

6.2.1 Installing the Battery

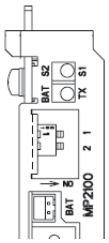
Fully insert the connector attached to the end of the battery leads for the JZSP-BA01 Battery specified by YASKAWA into the MP2100/MP2100M connector.

(1) JAPMC-MC2100, JAPMC-MC210□-E, or JAPMC-MC214□-E

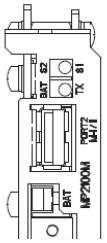
Connect the Battery to the BAT connector on the front panel.



JAPMC-MC2100



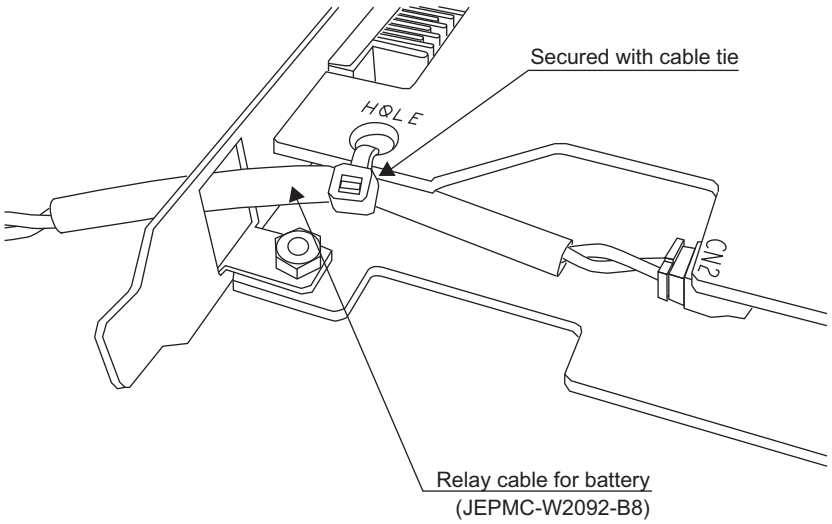
JAPMC-MC2100-E
JAPMC-MC2102-E



JAPMC-MC2140-E
JAPMC-MC2142-E

(2) JAPMC-MC2140

The battery connector on the MP2100M is the CN2 connector on the SVB Motion Module. The faceplate of the MP2100M has a hole provided for the battery cable. Extend the JEPMC-W2092-B8 Relay Cable for Battery provided with the MP2100M and secure with a cable tie as shown in the following diagram.



6.2.2 Battery Life

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 LED indicator on the MP2100/MP2100M lights, replace the battery with a new battery (JZSP-BA01) within two weeks. Any delay in battery replacement will result in the data stored in the memory being lost.

6.2.3 Battery Replacement

This section explains how to replace the battery.

(1) Preparations

(a) Saving the Memory Contents

Before replacing the battery, save the programs and data from the memory of the MP2100/MP2100M to floppy disks or a hard disk. The saved programs and data will be used if the programs and data are accidentally deleted during battery replacement. For the saved contents, refer to the *Machine Controller MP900/MP2000 Series MPE720 Software for Programming Device User's Manual* (Manual No.: SIEP C880700 05).

(b) Preparing a New Battery

Prepare a New battery (JZSP-BA01). This battery is not commercially available, and must be ordered from your nearest Yaskawa sales representative. The appearance of the battery is illustrated below.

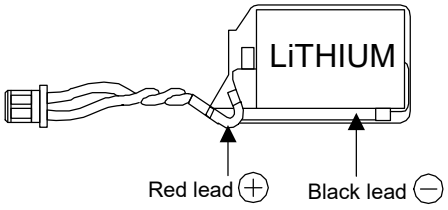


Fig. 6.1 JZSP-BA01 (Battery with Cable)

(2) Procedure

Use the following procedure to replace the battery.

1. Make sure that the S1 LED indicator on the MP2100/MP2100M is lit (green).
2. Disconnect the battery cable from the connector on the MP2100.
3. Firmly connect the replacement battery cable to the connector on the MP2100.
4. Make sure that the BAT LED indicator on the MP2100/MP2100M is not lit.

This completes the battery replacement procedure.

IMPORTANT

Be sure to replace the battery with the power supply to the MP2100/MP2100M turned ON.

Replacing the battery with the power supply to the MP2100/MP2100M turned OFF will result in the data stored in the MP2100/MP2100M being lost.

6.3 Troubleshooting

This section explains the basic troubleshooting procedure and provides a list of errors.

6.3.1 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 explained below.

Step 1	Visually confirm the following items.
• Machine movement (or status if stopped) • Presence/absence of power • I/O device status • Wiring status • Indicator status (LED indicators on each Module) • Switch settings (e.g., DIP switches) • Parameter settings and program contents	

↓

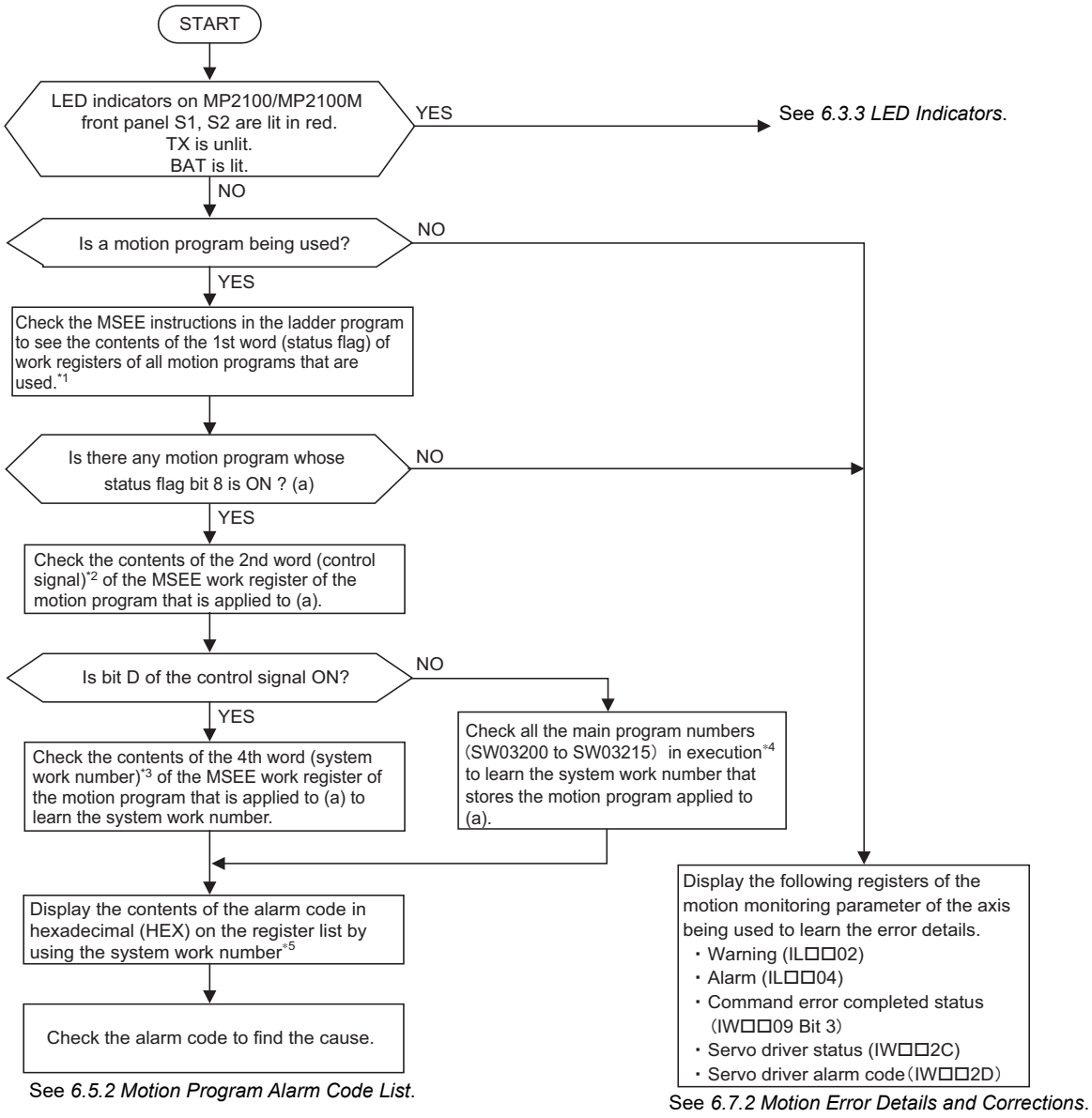
Step 2	Monitor the system to see if the problem changes for the following operations.
• Switching the Controller to STOP status • Resetting alarms • Turning the power supply OFF and ON	

↓

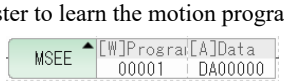
Step 3	Determine the location of the cause from the results of steps 1 and 2.
• Controller or external? • Sequence control or motion control? • Software or hardware?	

6.3.2 Error Check Flowchart

Find the correction to the problem using the following flowchart if the cause of the problem is thought to be the MP2100/MP2100M or SERVOPACK.



* 1. Under each MSEE instruction in ladder programs, the motion program number that is used and the MSEE work leading register number are displayed.
When bit 8 of the 1st word (status flag: refer to *5.2.4 (1) Motion Program Status Bits (DA□□□□□□+0)*) of MSEE work is ON, a motion program alarm is occurring (for details on how to refer to the register list, refer to *6.4.2 Accessing System Registers*).
<Example> In the figures below, when the MSEE instruction shown on the left is executed in a ladder program, DW00000 is the MSEE work leading register number. Therefore, if bit 8 of DW00000 is ON, it means that a motion program alarm is occurring.
When the motion program number is indicated by a register number as shown on the right, check the corresponding register to learn the motion program number.



* 2. For details on control signals, refer to *5.2.4 (2) Motion Program Control Signals (DA□□□□□□+1)*.
* 3. For details on system work number, refer to *5.2.4 (4) System Work Number (DA□□□□□□+3)*.
* 4. For details on the relationship between the registers from SW03200 to SW03215 and system work numbers, refer to *6.4.5 (9) Motion Program Execution Information*.

* 5. An alarm code of a motion program can be obtained from the program information used by work (58 words). Get the system work number first and then the alarm codes, referring to *6.4.5 (9) Motion Program Execution Information*. The alarm code is written in each set of parallel information. If no parallel execution instruction such as PFORK, JOINTO, and PJOINT is used, the alarm code is stored in the parallel 0 information.

6.3.3 LED Indicators

(1) Indicators

The status of the LED indicators on the front of the MP2100/MP2100M can be used to determine the error status and meaning.

Indicator Name	Indicator Color	Significance When Lit
S1	Green	Lights for normal operation.
	Red	Lights/flashes for failures.
S2	Not lit	User program stopping.
	Green	Lights for normal operation.
	Red	Lights/flashes for warning.
TX	Green	M-I/II data transmission in progress
BAT	Red	Battery alarm occurred.

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) Indicator Details

The following describes details and remedies for indicators showing operating status and errors in the MP2100/MP2100M.

Type	Indicator					Indicator Details	Remedy
	S1 (Green)	S2 (Green)	S1 (Red)	S2 (Red)	BAT		
Normal	Not lit	Not lit	Lit	Lit	Not lit	Hardware reset status	Normally the CPU activates within 10 seconds from power ON. If this status continues for longer than this, the problem is a user program error or hardware failure. Troubleshoot system errors.
	Not lit	Not lit	Not lit	Not lit	Not lit	Initial execution	
	Not lit	Lit	Not lit	Not lit	Not lit	Execute of drawing A	
	Lit	Not lit	Not lit	Not lit	Not lit	The user program is stopped (offline stop mode).	This status will occur for either of the following. • The program was stopped from the MPE720. • The RUN switch was turned OFF.
	Lit	Lit	Not lit	Not lit	Not lit	The user program is executing normally.	This status will occur during normal operation.
Error	Not lit	Not lit	Lit	Not lit	Not lit	A serious error has occurred.	Refer to *6.4.3 Troubleshooting when S1 Indicator Is Lit in Red.*
	Not lit	Not lit	Flashing	Not lit	Not lit	Number of flashes when software error occurs: 3: Address read error 4: Address write error 5: FPU error 6: General illegal command error 7: Slot illegal command error 8: General FPU suppression error 9: Slot FPU suppression error 10: TLB multi-bit error 11: LTB read error 12: LTB write error 13: LTB protection violation (read) 14: LTB protection violation (write) 15: First page write error	
	Not lit	Not lit	Flashing	Flashing	Not lit	Number of flashes when hardware error occurs: 2: RAM diagnosis error 3: ROM diagnosis error 4: CPU function diagnosis error 5: FPU function diagnosis error	A hardware error has occurred. Replace the Module.
Alarm	–*	–*	–*	–*	Lit	Battery alarm	Replace the battery. For details, refer to *6.2.3 Battery Replacement*.
	Lit	Not lit	Not lit	Lit	Not lit	Operation error	Refer to *6.4.5 (3) Ladder Program User Operation Error Status.*
	Lit	Not lit	Not lit	Lit	Not lit	I/O error	Refer to *6.4.5 (5) Alarm Counter and Alarm Clear.*

* A dash (–) means not specified.

6.4 Troubleshooting System Errors

This section explains troubleshooting information for system errors.

6.4.1 Outline of System Registers

The LED indicators on the front of the MP2100/MP2100M can be used to determine MP2100/MP2100M operating status and error status. To obtain more detailed information on errors, the system registers (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.

The following table shows the overall structure of the system registers. Refer to the sections given on the right for details.

SW00000	System Service Register	
SW00030	System Status	→ *6.4.5 (1) System Status*
SW00050	System Error Status	→ *6.4.5 (2) System Error Status*
SW00080	User Operation Error Status	→ *6.4.5 (3) Ladder Program User Operation Error Status*
SW00090	System Service Execution Status	→ *6.4.5 (4) System Service Execution Status*
SW00110	User Operation Error Status Details	→ *6.4.5 (3) Ladder Program User Operation Error Status*
SW00190	Alarm Counter and Alarm Clear	→ *6.4.5 (5) Alarm Counter and Alarm Clear*
SW00200	System I/O Error Status	→ *6.4.5 (6) System I/O Error Status*
SW00504	Reserved for system use	
SW00698	Interrupt Status	
SW00800	Module Information	→ *6.4.5 (8) MP210□/MP210□M Module Information*
SW01312	Reserved for system use	
SW02048	Reserved for system use	
SW03200	Motion Program Information	→ *6.4.5 System Registers and Error Statuses*
SW05200 to SW08191	Reserved for system use	

6.4.2 Accessing System Registers

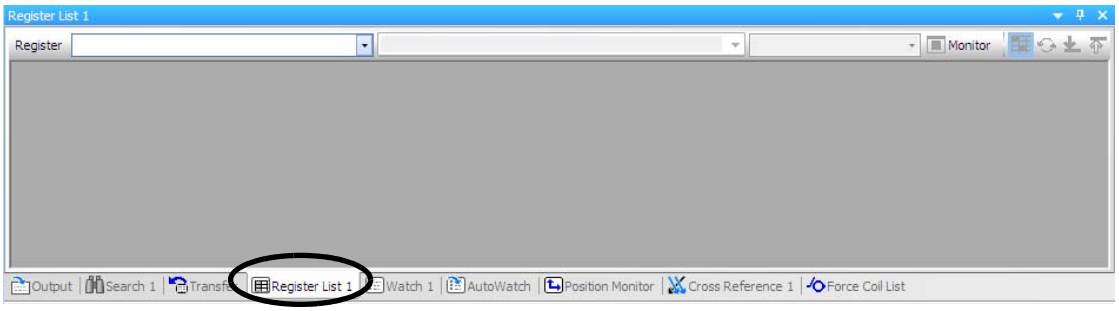
To access the contents of system registers, start the MPE720 Programming Tool and use the **Register List** or **Quick Reference** function.

(1) Displaying a Register List

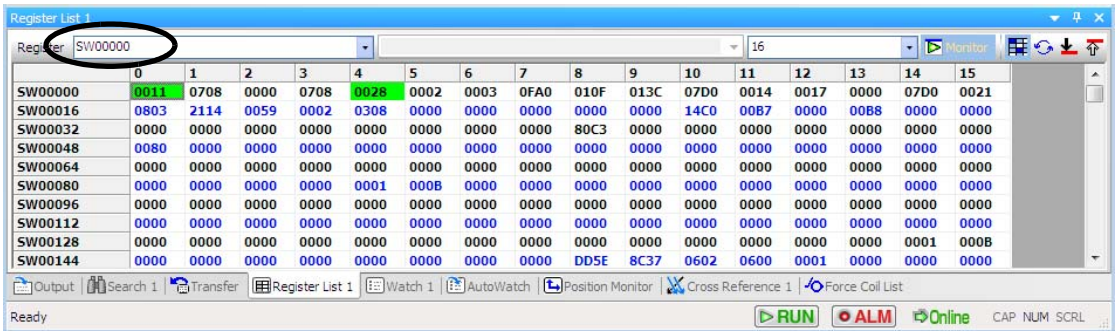
Use the following procedure to display a register list.

1. Open the **Register List** sub-window from the MPE720 Ver.7.

   The **Register List 1** tab is provided by default in the sub-window displayed on the bottom of the window.



2. Enter the leading register number of the system register "SW□□□□□" to be accessed in the **Register** input field.



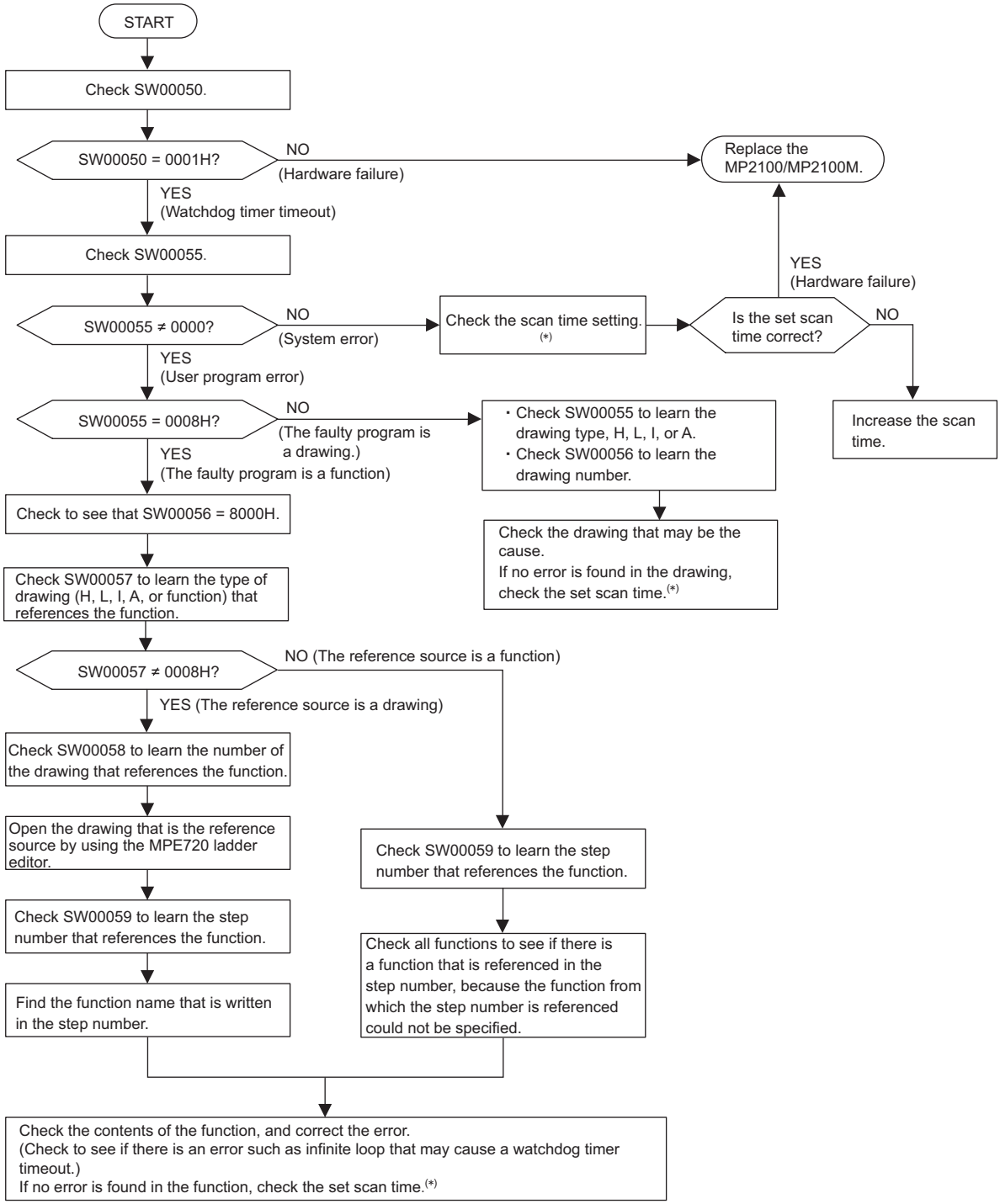
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
SW00000	0011	0708	0000	0708	0028	0002	0003	0FA0	010F	013C	07D0	0014	0017	0000	07D0	0021
SW00016	0803	2114	0059	0002	0308	0000	0000	0000	0000	0000	14C0	00B7	0000	00B8	0000	0000
SW00032	0000	0000	0000	0000	0000	0000	0000	0000	80C3	0000	0000	0000	0000	0000	0000	0000
SW00048	0080	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
SW00064	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
SW00080	0000	0000	0000	0000	0001	000B	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
SW00096	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
SW00112	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
SW00128	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0001	000B
SW00144	0000	0000	0000	0000	0000	0000	0000	0000	DD5E	8C37	0602	0600	0001	0000	0000	0000

The contents of the system register is displayed starting from the leading register number.

Note: The data type is set by default to decimal. Place the cursor anywhere on the list, and then right-click. Select **Hex** (hexadecimal) from the pop-up menu that appears. The data are displayed in hexadecimal.

6.4.3 Troubleshooting when S1 Indicator Is Lit in Red

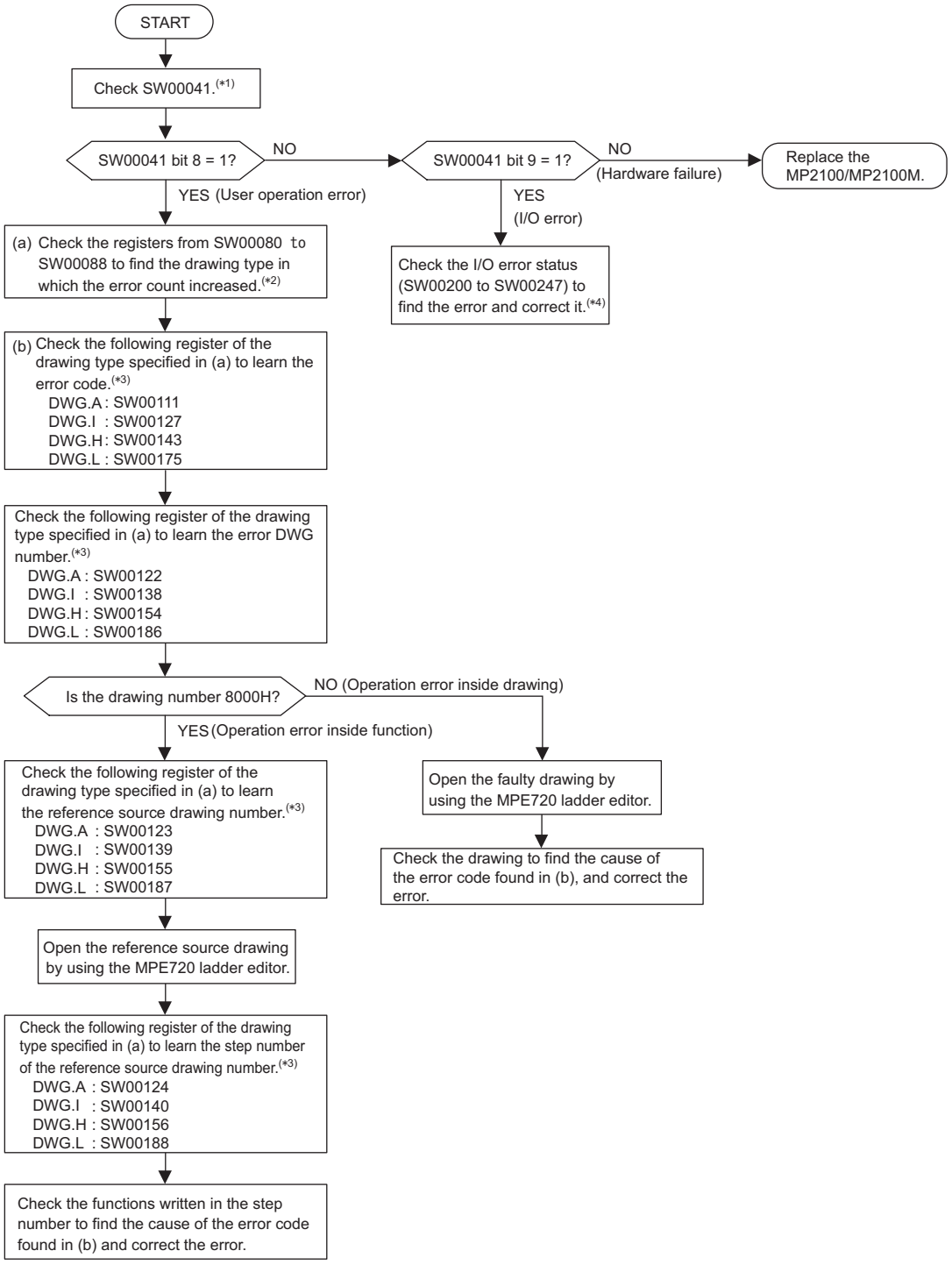
When the S1 LED indicator on the front panel of the MP2100/MP2100M lights/flashes in red, it is likely that a serious failure (software error or hardware error) is occurring. Set the MP2100/MP2100M in the stopped status (DIP SW 6: ON), and then investigate the cause using the following flowchart.



Note: 1. For details on the system registers from SW00050 to SW00059, refer to *6.4.5 (2) System Error Status*.
2. For precautions when setting the scan time, refer to *5.5.3 Precautions when Setting or Changing the Scan Time*.

6.4.4 Troubleshooting when S2 Indicator Is Lit in Red

When the S2 LED indicator on the front panel of the MP2100/MP2100M lights/flashes in red, it is likely that a hardware error (calculation error or I/O error) is occurring. Set the MP2100/MP2100M in the stopped status (DIP SW 6: ON), and then investigate the cause using the following flowchart.



Note: 1. For details on SW00041, refer to *6.4.5 (2) System Error Status*.
2. For details on the system registers from SW00080 to SW00088, refer to *6.4.5 (3) Ladder Program User Operation Error Status*.
3. For details on the system registers from SW00110 to SW00188, refer to *6.4.5 (3) Ladder Program User Operation Error Status*.
4. For details on the system registers from SW00200 to SW00247, refer to *6.4.5 (6) System I/O Error Status*.

6.4.5 System Registers and Error Statuses

(1) System Status

This is data that indicates the operating status of the system and error details, and it is stored in registers SW00040 to SW00048. Checking the system status details makes it possible to determine whether the hardware or software is the cause of an error.

Table 6.1 System Status List

Name	Register No.	Description		
Reserved for system use	SW00030 to SW00039	–		
CPU Status	SW00040	SW00040 Bit 0	READY	0: Failure, 1: Normal
		SW00040 Bit 1	RUN	0: Stopped, 1: Running
		SW00040 Bit 2	ALARM	0: Normal, 1: Alarm
		SW00040 Bit 3	ERROR	0: Normal, 1: Error
		SW00040 Bit 4	Reserved for system use	–
		SW00040 Bit 5	Reserved for system use	–
		SW00040 Bit 6	FLASH	1: Flash operation
		SW00040 Bit 7	WEN	0: Write-disabled, 1: Write-enabled
		SW00040 Bit 8 to SW00040 Bit D	Reserved for system use	–
		SW00040 Bit E	Operation Stop Request	0: RUN selection, 1: STOP selection
		SW00040 Bit F	Run Switch Status at Power ON	0: STOP, 1: RUN
CPU Error Status	SW00041	SW00041 Bit 0	Serious Failure	1: WDGE, undefined command See SW00050 for details.
		SW00041 Bit 1	Reserved for system use	–
		SW00041 Bit 2	Reserved for system use	–
		SW00041 Bit 3	Exception Error	–
		SW00041 Bit 4 to SW00041 Bit 7	Reserved for system use	–
		SW00041 Bit 8	User operation error	1: User operation error
		SW00041 Bit 9	I/O Error	1: I/O error
		SW00041 Bit A to SW00041 Bit F	Reserved for system use	–
H Scan Over Counter	SW00044	–		
L Scan Over Counter	SW00046	–		
Reserved for system use	SW00047	SW00047 Bit 0 to SW00047 Bit F	Reserved for system use	–

Table 6.1 System Status List

(cont'd)

Name	Register No.	Description		
Hardware Configuration Status	SW00048	SW00048 Bit 0	TEST	Mode switch alarms 0: OFF, 1: ON
		SW00048 Bit 1	MON	
		SW00048 Bit 2	CNFG	
		SW00048 Bit 3	INIT	
		SW00048 Bit 4	SUP	
		SW00048 Bit 5	STOP	
		SW00048 Bit 6	–	
		SW00048 Bit 7	Battery Alarm	–
		SW00048 Bit 8 to SW00048 Bit F	Reserved for system use	–
Reserved for system use	SW00049	SW00049 Bit 0 to SW00049 Bit F	Reserved for system use	–

(2) System Error Status

This data indicates the detailed information when a system error has occurred, and it is stored in registers SW00050 to SW00060.

Table 6.2 System Error Status List

Name	Register No.	Description		
32-bit Error Code	SW00050	0001H	Watchdog timer over error	
		0041H	ROM diagnosis error	
		0042H	RAM diagnosis error	
		0043H	CPU diagnosis error	
		0044H	FPU diagnosis error	
		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 inhibited exception error	
		0820H	Slot FPU inhibited exception error	
	SW00051	For system error analysis		
32-bit Addresses Generating Error	SW00052	For system error analysis		
	SW00053			
Ladder Program Error Task	SW00054	0000H: System 0001H: DWG.A	0002H: DWG.I 0003H: DWG.H	0005H: DWG.L
Ladder Program Type	SW00055	0000H: System 0001H: DWG.A	0002H: DWG.I 0003H: DWG.H 0005H: DWG.L	0008H: Function 000FH: Motion program
Ladder Program Error Drawing Number	SW00056	Ladder program parent drawing: FFFFH Ladder program function: 8000H Ladder program child drawing: □□00H (H□□: Child drawing number) Ladder program grandchild drawing: □□yyH (Hyy: Grandchild drawing number) Motion program: F0□□H (H□□: Program number)		
Ladder Program Function 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 0010H: Reserved for system use 0011H: Reserved for system use	

Table 6.2 System Error Status List (cont'd)

Name	Register No.	Description
Ladder Program Function Calling Drawing Number	SW00058	Number of drawing that calls the ladder program function in which an error occurred. Parent drawing: FFFFH Child drawing: □□00H (H□□: Child drawing number) Grandchild drawing: □□yyH (Hyy: Grandchild drawing number) Function: 0100H
	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 for system use
	SW00062 to SW00065	Name of Task Generating Error
	SW00066 and SW00067	Reserved for system use
	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 to SW00079	Reserved for system use

(3) Ladder Program User Operation Error Status

This data indicates the detailed information when a user operation error has occurred in the ladder program, and it is stored in registers SW00080 to SW00089 (Error Status 1) and in SW00110 to SW00189 (Error Status 2).

Table 6.3 Ladder Program User Operation Error Status 1

Name		Register No.	Description
DWG.A	Error Count	SW00080	Operation error code: Refer to *Ladder Program User Operation Error Status 3*. Error code when an index error occurs: Refer to *Ladder Program User Operation Error Status 4*.
	Error Code	SW00081	
DWG.I	Error Count	SW00082	
	Error Code	SW00083	
DWG.H	Error Count	SW00084	
	Error Code	SW00085	
Reserved for system use		SW00086	
		SW00087	
DWG.L	Error Count	SW00088	
	Error Code	SW00089	

Table 6.4 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: □□00H (H□□: Child drawing number) Grandchild drawing: □□yyH (Hyy: Grandchild drawing number) Function: 8000H Motion program/sequence program: F0□□H (H□□: Program number)
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	
Address Generating Error	SW00120	SW00136	SW00152	SW00184	• Reference Source Drawing Number Number of the drawing reference source in which an error occurred.
	SW00121	SW00137	SW00153	SW00185	
Error Drawing Number	SW00122	SW00138	SW00154	SW00186	
Reference Source Drawing Number	SW00123	SW00139	SW00155	SW00187	• Reference Source Drawing Step Number Step number of the drawing reference source in which an error occurred. 0 when there is an error in the parent drawing.
Reference Source Drawing Step Number	SW00124	SW00140	SW00156	SW00188	
Reserved for system use	SW00125	SW00141	SW00157	SW00189	

Table 6.5 Ladder Program User Operation Error Status 3

	Error Code	Error Contents		User*1	System Default Setting*2
Integer Operations	0001H	Integer operation - underflow		Yes	–32768 [–32768]
	0002H	Integer operation - overflow		Yes	32767 [32767]
	0003H	Integer operation - division error		Yes	The A register remains the same.
	0009H	Double-length integer operation - underflow		Yes	–2147483648 [–2147483648]
	000AH	Double-length integer operation - overflow		Yes	2147483647 [2147483647]
	000BH	Double-length integer operation - division error		Yes	The A register remains the same.
	010□H	Operation error drawing - integer operation error (□ = 1 to B)		No	Default indicated above.
Real Number Operation	0010H	Integer storage - non-numeric error		Yes	Store not executed. [00000]
	0011H	Integer storage - underflow		Yes	Store not executed. [–32768]
	0012H	Integer storage - overflow		Yes	Store not executed. [+32767]
	0021H	Real number storage - underflow		Yes	Store not executed. [–1.0E+38]
	0022H	Real number storage - overflow		Yes	Store not executed. [1.0E+38]
	0023H	Real number operation - division-by-zero error		Yes	Operation not executed. The F register remains the same.
	0030H	Real number operation - invalid operation (non-numeric)		No	Operation not executed.
	0031H	Real number operation - exponent underflow		No	0.0
	0032H	Real number operation - exponent overflow		No	Maximum value
	0033H	Real number operation - division error (non-numeric 0/0)		No	Operation not executed.
	0034H	Real number storage - exponent underflow		No	Stores 0.0.
	0035H	Real number operation - stack error		–	–
	0040H to 0059H	Standard System Functions Real number operation errors		No	Operation stopped and output = 0.0
		0040H: SQRT	0041H: SIN	0042H: COS	0043H: TAN
		0044H: ASIN	0045H: ACOS	0046H: ATAN	0047H: EXP
		0048H: LN	0049H: LOG	004AH: DZA	004BH: DZB
		004CH: LIM	004DH: PI	004EH: PD	004FH: PID
		0050H: LAG	0051H: LLAG	0052H: FGN	0053H: IFGN
		0054H: LAU	0055H: SLAU	0056H: REM	0057H: RCHK
		0058H: BSRCH	0059H: SQRT	–	–
		1000H or 2000H is added for an index error.			

* 1. Yes: Can be set to value other than system default from the user program.
No: System default value is fixed and the user cannot set another value.
* 2. The values in parentheses are the maximum allowable values (setting limits). When this limit is exceeded, an alarm occurs and the value will be changed to the default setting.

Table 6.6 Ladder Program User Operation Error Status 4

	Error Code	Error Contents		User*	System Default
Integer - Real Number Operations	1000H	Index error within drawing		No	Execute again when corresponding to i, j = 0. The i and j registers remain the same.
	2000H	Index error within function		No	Execute again when corresponding to i, j = 0. The i and j registers remain the same.
Integer Operation	□060H to □077H (□ = 1,2)	Integer system functions Index error		No	Operation stopped and output = input. The A register remains the same.
		□06DH: PI	□06DH: PD	□06FH: PID	□070H: LAG
		□071H: LLAG	□072H: FGN	□073H: IFGN	□074H: LAU
		□075H: SLAU	□076H: FGN	□077H: IFGN	–

(4) System Service Execution Status

(a) Data Trace Execution Status

Name	Register No.	Remarks
Reserved for system use	SW00090	–
Reserved for system use	SW00091	
Reserved for system use	SW00092	
Reserved for system use	SW00093	
Reserved for system use	SW00094 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

(b) Latest Data Trace Record Numbers

Table 6.7 Latest Data Trace Record Numbers

Name	Register No.	Remarks
Data Trace Group 1	SW00100	Latest record number
Data Trace Group 2	SW00101	
Data Trace Group 3	SW00102	
Data Trace Group 4	SW00103	

(5) Alarm Counter and Alarm Clear

Name	Register No.	Remarks
Number of Alarm Occurrences	SW00190	The number of alarm occurrences.
Number of Alarm History Records	SW00191	The number of alarms in the alarm history.
Clear Alarms	SW00192	1: Alarm cleared 2: Current alarms and alarm history cleared

(6) System I/O Error Status

Name	Register No.	Remarks
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 for system use	SW00205	Not used.
	SW00206	
	SW00207	
I/O Error Status	SW00208 to SW00215	Error status of MP2100/MP2100M
	SW00216 to SW00223	Reserved for system use
	SW00224 to SW00231	Error status of SVB-01 (MP2100M only)
	SW00232 to SW00239	Reserved for system use
	SW00240 to SW00247	Error status of rack 2, slot 1
	SW00248 to SW00255	Error status of rack 2, slot 2
	SW00256 to SW00263	Error status of rack 2, slot 3
	SW00264 to SW00271	Error status of rack 2, slot 4
	• • •	• • •
	SW00448 to SW00455	Error status of rack 2, slot 9

Note: Racks 2 to 4 apply only when MP2100MEX is used with MP2100M.

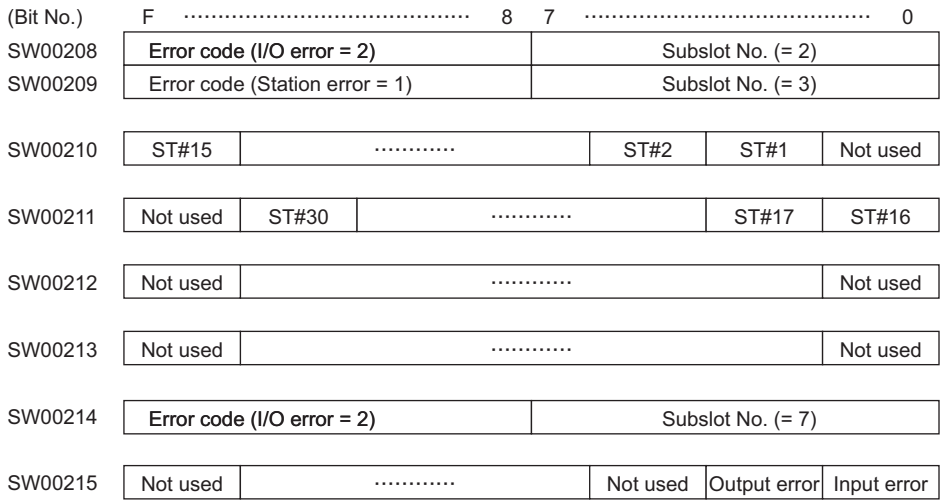
(7) Details on I/O Error Status

When a system I/O error occurs, the error status will be written in the system register.
The registers allocated for error statuses with MP2100/MP2100M are described below.

(a) Error Statuses of MP2100

Name	Register No.	Remarks
Slot 0 Error Status	SW00208 to SW00215	Error status of MP2100
Reserved for system use	SW00216 to SW00455	Not used.

- Register Allocations for MP2100 Error Statuses



(b) Error Statuses of MP2100M

Name	Register No.	Remarks
Slot 0 error status	SW00208 to SW00215	Error status of MP2100
Reserved for System Use	SW00216 to SW00223	Not used.
Slot 1 error status	SW00224 to SW00231	SVB-01 error status
Reserved for System Use	SW00232 to SW00239	Not used.
Information of rack 2, slot 1	SW00240 to SW00247	Differs depending on the installed module or error code.
Information of rack 2, slot 2	SW00248 to SW00255	
Information of rack 2, slot 3	SW00256 to SW00263	
Information of rack 2, slot 4	SW00264 to SW00271	
Information of rack 2, slot 5	SW00272 to SW00279	
Information of rack 2, slot 6	SW00280 to SW00287	
Information of rack 2, slot 7	SW00288 to SW00295	
Information of rack 2, slot 8	SW00296 to SW00303	
Information of rack 2, slot 9	SW00304 to SW00311	
Information of rack 3, slot 1	SW00312 to SW00319	
Information of rack 3, slot 2	SW00320 to SW00327	
Information of rack 3, slot 3	SW00328 to SW00335	
Information of rack 3, slot 4	SW00336 to SW00343	
Information of rack 3, slot 5	SW00344 to SW00351	
Information of rack 3, slot 6	SW00352 to SW00359	
Information of rack 3, slot 7	SW00360 to SW00367	
Information of rack 3, slot 8	SW00368 to SW00375	
Information of rack 3, slot 9	SW00376 to SW00383	
Information of rack 4, slot 1	SW00384 to SW00391	
Information of rack 4, slot 2	SW00392 to SW00399	
Information of rack 4, slot 3	SW00400 to SW00407	
Information of rack 4, slot 4	SW00408 to SW00415	
Information of rack 4, slot 5	SW00416 to SW00423	
Information of rack 4, slot 6	SW00424 to SW00431	
Information of rack 4, slot 7	SW00432 to SW00439	
Information of rack 4, slot 8	SW00440 to SW00447	
Information of rack 4, slot 9	SW00448 to SW00455	

Note: Racks 2 to 4 apply only when MP2100MEX is used.

- Register Allocations for MP2100M Error Statuses

(Bit No.)	F ····· 8	7 ····· 0
SW00208	Error code (I/O error = 2)	Subslot No. (= 2)
SW00209	Error code (Station error = 1)	Subslot No. (= 3)

SW00210	ST#15	·····	ST#2	ST#1	Not used
SW00211	Not used	ST#30 ·····		ST#17	ST#16
SW00212	Not used	·····			Not used
SW00213	Not used	·····			Not used
SW00214	Error code (I/O error = 2)		Subslot No. (= 7)		
SW00215	Not used	·····	Not used	Output error	Input error

- Register Allocations for SVB-01 Error Statuses

(Bit No.)	F ····· 8	7 ····· 0
SW00224	Error code (Station error = 1)	Subslot No. (= 1)

SW00225	ST#15	·····	ST#2	ST#1	Not used
SW00226	Not used	ST#30 ·····		ST#17	ST#16
SW00227	Not used	·····			Not used
SW00228	Not used	·····			Not used

(8) MP210□/MP210□M Module Information

(a) MP210□ Module Information

Name	Register No.	Remarks
Module Information	SW00800	MP2100 ID (C180H)/ MP2101 ID (C182H)
	SW00801	CPU hardware version (BCD)
	SW00802	CPU software version (BCD)
	SW00803	Number of sub-slots (0008H)
	SW00804	CPU Function Module ID (C110H)
	SW00805	CPU Function Module Status
	SW00806	I/O Function Module ID (8070H)
	SW00807	I/O Function Module Status
	SW00808	SVB Function Module ID (9112H)
	SW00809	SVB Function Module Status
	SW00810	SVR Function Module ID (9210H)
	SW00811	SVR Function Module Status
	SW00812 to SW00951	Reserved for system use

(b) MP210□M Module Information

Name	Register No.	Remarks
Module Information	SW00800	MP2100M ID (C181H)/ MP2101M ID (C183H)
	SW00801	CPU hardware version (BCD)
	SW00802	CPU software version (BCD)
	SW00803	Number of sub-slots (0008H)
	SW00804	CPU Function Module ID (C110H)
	SW00805	CPU Function Module Status
	SW00806	I/O Function Module ID (8070H)
	SW00807	I/O Function Module Status
	SW00808	SVB Function Module ID (9112H)
	SW00809	SVB Function Module Status
	SW00810	SVR Function Module ID (9210H)
	SW00811	SVR Function Module Status
	SW00812 to SW0815	Reserved for system use
	SW00816	SVB-01 Module (9195H)
	SW00817	CPU hardware version (BCD)
	SW00818	CPU software version (BCD)
	SW00819	Number of sub-slots (0001H)
	SW00820	SVB Function Module ID (9115H)
	SW00821	SVB Function Module Status
	SW00822 to SW00879	Reserved for system use
	SW00880 to SW00887	Information of rack 2, slot 1
	SW00888 to SW00895	Information of rack 2, slot 2
	・・・	・・・
	SW00944 to SW00951	Information of rack 4, slot 9

Note: Racks 2 to 4 apply only when MP2100MEX is used.

(9) Motion Program Execution Information

System Work	Main Program No. in Execution	Program Information Used by Work	Motion Program Alarm							
			Parallel 0	Parallel 1	Parallel 2	Parallel 3	Parallel 4	Parallel 5	Parallel 6	Parallel 7
			Offset* +4	Offset +7	Offset +10	Offset +13	Offset +16	Offset +19	Offset +22	Offset +25
1	SW3200	SW03264 to SW03321	SW03268	SW03271	SW03274	SW03277	SW03280	SW03283	SW03286	SW03289
2	SW3201	SW03322 to SW03379	SW03326	SW03329	SW03332	SW03335	SW03338	SW03341	SW03344	SW03347
3	SW3202	SW03380 to SW03437	SW03384	SW03387	SW03390	SW03393	SW03396	SW03399	SW03402	SW03405
4	SW3203	SW03438 to SW03495	SW03442	SW03445	SW03448	SW03451	SW03454	SW03457	SW03460	SW03463
5	SW3204	SW03496 to SW03553	SW03500	SW03503	SW03506	SW03509	SW03512	SW03515	SW03518	SW03521
6	SW3205	SW03554 to SW03611	SW03558	SW03561	SW03564	SW03567	SW03570	SW03573	SW03576	SW03579
7	SW3206	SW03612 to SW03669	SW03616	SW03619	SW03622	SW03625	SW03628	SW03631	SW03634	SW03637
8	SW3207	SW03670 to SW03727	SW03674	SW03677	SW03680	SW03683	SW03686	SW03689	SW03692	SW03695
9	SW3208	SW03728 to SW03785	SW03732	SW03735	SW03738	SW03741	SW03744	SW03747	SW03750	SW03753
10	SW3209	SW03786 to SW04843	SW03790	SW03793	SW03796	SW03799	SW03802	SW03805	SW03808	SW03811
11	SW3210	SW03844 to SW03901	SW03848	SW03851	SW03854	SW03857	SW03860	SW03863	SW03866	SW03869
12	SW3211	SW03902 to SW03959	SW03906	SW03909	SW03912	SW03915	SW03918	SW03921	SW03924	SW03927
13	SW3212	SW03960 to SW04017	SW03964	SW03967	SW03970	SW03973	SW03976	SW03979	SW03982	SW03985
14	SW3213	SW04018 to SW04075	SW04022	SW04025	SW04028	SW04031	SW04034	SW04037	SW04040	SW04043
15	SW3214	SW04076 to SW04133	SW04080	SW04083	SW04086	SW04089	SW04092	SW04095	SW04098	SW04101
16	SW3215	SW04134 to SW04191	SW04138	SW04141	SW04144	SW04147	SW04150	SW04153	SW04156	SW04159

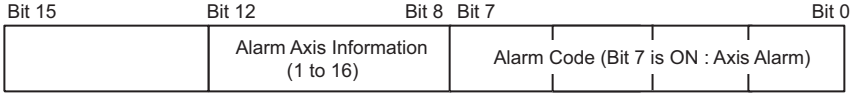
* Offset: Offset value from the first register number of Program Information Used by Work

6.5 Motion Program Alarms

If the result of investigation using *6.3.2 Error Check Flowchart* indicates that a motion program alarm has occurred, use the alarm code to determine the cause of the error.

6.5.1 Motion Program Alarm Configuration

Motion program alarms stored in the alarm output register (default: SW03268) are displayed as shown in the following diagram.



Note: Refer to the Controller user's manual for information on finding the alarm output register.

6.5.2 Motion Program Alarm Code List

The motion program alarm codes are listed in the following table.

Note: When displaying these on the register list, set the view mode to hexadecimal.

Alarm Code		Name	Description	Corrective Actions
Program Alarm	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.

(cont'd)

Alarm Code		Name	Description	Corrective Actions
Program Alarm	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.	Perform 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.
	21H	PFORK execution error	A motion command was instructed simultaneously at the second line in the PFORK of both a main motion program and a subprogram.	Review the main 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.
Axis Alarm*	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 is issuing a command to the same axis at the same time. If this is the case, 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 is issuing a command to the same axis at the same time. If this is the case, 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 incomplete	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.

* The axis number is stored in bits 8 to 12 for axis alarms.

6.6 List of Causes for Command Error Occurrence

The Command Error Completed Status (Command Error Occurrence) bit (IW□□09, bit 3) turns ON when the set motion command cannot be executed or when the execution of a motion command ends error. The triggers for which this bit turns ON depend on the motion command.

The following table explains the causes of Command Error Occurrence for each motion command.

Motion Command		Cause of Command Error Occurrence	Warning (W:) and Alarm (A:) That Occur at Command Error Occurrence
1	Position Mode (Positioning) (POSING)	The positioning movement exceeds the allowable range.	A: Excessive Positioning Moving Amount
		The axis is ABS infinite-length, and the zero point return setting is not completed.	A: Zero Point Unsetting
		In servo OFF status	A: Servo OFF
		Alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
2	Latch Torque Positioning (External positioning) (EX_POSING)	The positioning movement exceeds the allowable range.	A: Excessive Positioning Moving Amount
		The axis is ABS infinite-length, and the zero point return setting is not completed.	A: Zero Point Unsetting
		In servo OFF status	A: Servo OFF
		Alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
		SERVOPACK parameter writing was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Warning A.94 or A.95 has occurred at the SERVOPACK.	W: Servo Driver Error
		The external signal selection is out of the setting range.	W: Set 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 Communications 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 has occurred at the SERVOPACK.	W: Servo Driver Error
		The selected zero point return method is outside the setting range.	W: Set Parameter Error
		POT method is selected for zero point return, but the approach speed is a negative value.	W: Set Parameter Error
		NOT method is selected for zero point return, but the approach speed is a positive value.	W: Set Parameter Error
		During zero point return using DEC1 + Phase-C, ZERO signal, DEC1 + 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		Cause of Command Error Occurrence	Warning (W:) and Alarm (A:) That Occur at Command Error Occurrence
4 and 5	Interpolation (INTERPOLATE) Last Interpolation Segment (ENDOF_INTERPOLATE)	The travel specified for one scan has exceeded the segment that can be specified for a MECHATROLINK SERVOPACK, or the speed feed forward value has exceeded the maximum speed that can be specified.	A: Excessive Speed
		The axis is ABS infinite length, and the zero point return (setting) is not completed.	A: Zero Point Unsetting
		In servo OFF status	A: Servo OFF
		An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
6	Interpolation Mode with Latch Input (LATCH)	The travel specified for one scan has exceeded the segment that can be specified for a MECHATROLINK SERVOPACK, or the speed feed forward value has exceeded the maximum speed that can be specified.	A: Excessive Speed
		The axis is ABS infinite length, and the zero point return (setting) is not completed.	A: Zero Point Unsetting
		In servo OFF status	A: Servo OFF
		An alarm is occurring.	–
		The latch signal is out of the setting range.	W: Set Parameter Error
7	JOG Mode (FEED)	In machine lock status	–
		In servo OFF status	A: Servo OFF
		An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
8	Relative Position Mode (Step mode) (STEP)	Positioning movement 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 Communications Error
9	Set Zero Point (ZSET)	An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
10 and 11	Change Acceleration Time (ACC) Change Deceleration Time (DCC)	An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
		Executed before distribution has been completed (DEN = OFF)	–
		SERVOPACK parameter writing was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Warning A.94 or A.95 has occurred at the SERVOPACK.	W: Servo Driver Error
12	Change Filter Time Constant (SCC)	An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
		Executed before distribution has 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 has occurred at the SERVOPACK.	W: Servo Driver Error

(cont'd)

Motion Command		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 Communications Error
		Executed before distribution has been completed (DEN = OFF).	A: Filter Time Constant Change Error
		The selected filter type is out of the setting range.	W: Set Parameter Error
14, 15, and 16	Change Speed Loop Gain (KVS) Change Position Loop Gain (KPS) Change Feed Forward (KFS)	An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
		SERVOPACK parameter writing was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Warning A.94 or A.95 has occurred at the SERVOPACK.	W: Servo Driver Error
17 and 18	Read User Constant (PRM_RD) Write User Constant (PRM_WR)	An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
		SERVOPACK parameter reading was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Warning A.94 or A.95 has occurred at the SERVOPACK.	W: Servo Driver Error
		The servo user constant No. or user constant size is outside the setting range.	W: Set Parameter Error
19 and 20	Alarm Monitor (ALM_MON) Alarm History Monitor (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 outside setting range.	W: Set Parameter Error
21	Clear Alarm History (ALMHIST_CLR)	The command to the SERVOPACK was not completed within the specified time.	A: Servo Driver Command Timeout Error
22	Absolute Encoder Reset (ABS_RST)	The command was issued with respect to a Σ-I type SERVOPACK.	–
		The command was issued in the servo ON status.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
		The command to the SERVOPACK was not completed within the specified time.	A: Servo Driver Command Timeout Error
23	Speed Reference (VELO)	The command was issued in a MECHATROLINK-I connection.	–
		An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
24	Torque/Thrust Reference (TRQ)	The command was issued in a MECHATROLINK-I connection.	–
		An alarm is occurring	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
25	Phase Reference (PHASE)	The axis is ABS infinite length, and the zero point return (setting) is not completed.	A: Zero Point Unsetting
		In servo OFF status	A: Servo OFF
		An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error

(cont'd)

Motion Command		Cause of Command Error Occurrence	Warning (W:) and Alarm (A:) That Occur at Command Error Occurrence
26	Change Position Loop Integral Time Constant (KIS)	An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
		SERVOPACK parameter writing was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Warning A.94 or A.95 has occurred at the SERVOPACK.	W: Servo Driver Error
	Others Automatic Parameter Updating when Move Command Starts *	An alarm is occurring.	–
		Asynchronous communication status	A: Servo Driver Synchronization Communications Error
		SERVOPACK parameter writing was not completed within the specified time.	A: Servo Driver Command Timeout Error
		Warning A.94 or A.95 has occurred at the SERVOPACK.	W: Servo Driver Error
		Distribution had not been completed (DEN = OFF).	–

* When Automatic Updating of Parameter has been enabled for fixed parameters, and the setting of Filter Time Constant, Acceleration/Acceleration Time Constant or Deceleration/Deceleration Time Constant was changed at the same time as a move command was set.

6.7 Troubleshooting Motion Errors

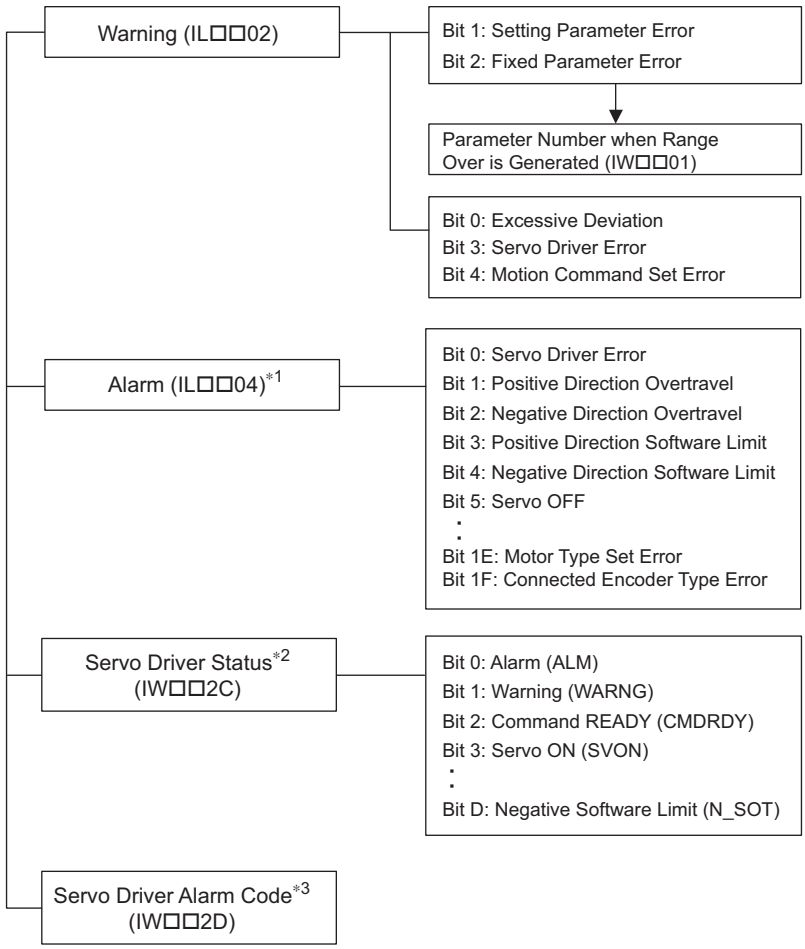
This section explains the details and corrective actions for errors that occur in motion control functions.

6.7.1 Overview of 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.

The motion alarms for the Basic Module's MECHATROLINK-I/MECHATROLINK-II functionality are shown below.



* 1. Refer to *6.7.2 Motion Error Details and Corrections*.
* 2. Refer to *6.7.3 (1) Servo Driver Status (IW□□2C)*.
* 3. Refer to *6.7.3 (2) Servo Driver Alarm Code (IW□□2D)*.

6.7.2 Motion Error Details and Corrections

The following tables show the contents of the axis alarms (IL□□04) and axis alarm details.

(1) Alarm IL□□04 List

IL□□04	Alarm Contents	IL□□04	Alarm Contents
Bit 0	Servo Driver Error	Bit 10	Servo Driver Synchronization Communications Error
Bit 1	Positive Direction Overtravel	Bit 11	Servo Driver Communication Error
Bit 2	Negative Direction Overtravel	Bit 12	Servo Driver Command Timeout Error
Bit 3	Positive Direction Software Limit	Bit 13	Excessive ABS Encoder Rotations
Bit 4	Negative Direction Software Limit	Bit 14	Reserved for system use
Bit 5	Servo OFF	Bit 15	Reserved for system use
Bit 6	Positioning Time Over	Bit 16	Not used
Bit 7	Excessive Positioning Moving Amount	Bit 17	Not used
Bit 8	Excessive Speed	Bit 18	Not used
Bit 9	Excessive Deviation	Bit 19	Not used
Bit A	Filter Type Change Error	Bit 1A	Not used
Bit B	Filter Time Constant Change Error	Bit 1B	Not used
Bit C	Not used	Bit 1C	Not used
Bit D	Zero Point Unsetting	Bit 1D	Not used
Bit E	Not used	Bit 1E	Motor Type Set Error
Bit F	Not used	Bit 1F	Connected Encoder Type Error

(2) Bit 0: Servo Driver Error

Detection Timing	• SERVOPACK alarms are continuously monitored by the alarm management section.
Processing when Alarm Occurs	• The current command will be aborted. If a SERVOPACK error is detected during execution of a POSING command, the positioning will be aborted and the axis will decelerate to a stop. • The Command Error Completed Status in the Motion Command Status (IW□□09,bit 3) will turn ON.
Error and Cause	• The cause of the error depends on the type of alarm. The contents of an alarm is monitored in IW□□2D. For details, refer to the list of SERVOPACK alarms in *6.7.3 (2) Servo Driver Alarm Code (IW□□2D)*.
Correction	• Confirm the SERVOPACK alarm and remove the cause. • Reset the alarm.

Note: The above status bit will turn ON for any of the SERVOPACK alarm for alarms classified as SERVOPACK alarms.

(3) Bit 1: Positive Direction Overtravel and Bit 2: Negative Direction Overtravel

Detection Timing	• Overtravel is continuously monitored by the position management section during execution of a motion command. • Overtravel is detected when the overtravel signal in the direction of movement turns OFF.
Processing when Alarm Occurs	• Stop processing of the SERVOPACK The stop method and processing after stopping depends on the SERVOPACK parameter settings. • The Command Error Completed Status in the Motion Command Status (IW□□09, bit 3) will turn ON. • Processing of MP2100/MP2100M The command is canceled and the axis decelerates to a stop. Follow-up processing (each scan the current position of the machine is adjusted to the reference position) is executed.
Error and Cause	One of the following is possible. • A move command that exceeded the travel limit of the machine was executed as follows: A user program command exceeded the travel limit. The software limit was exceeded in manual operation. • Overtravel signal malfunction.
Correction	• Check the following. Check the overtravel signal. Check the program or manual operation. • Then, after clearing the motion command and resetting the alarm, use a return operation to eliminate the overtravel status. Commands in the overtravel direction will be disabled and an alarm will occur again if one is executed.

IMPORTANT

- For a vertical axis, the following should be set at the SERVOPACK to avoid dropping and vibration at the overtravel limit.
  - An emergency deceleration stop
  - Zero clamp status after the deceleration stop

(4) Bit 3: Positive Direction Software Limit and Bit 4: Negative Direction Software Limit

Detection Timing	• Enabled when using a motion command and detected by the position management section. • The software limits are valid after a ZRET or ZSET command has been completed.
Processing when Alarm Occurs	• The axis decelerates to a stop at the software limit. • The Command Error Completed Status in the Motion Command Status (IW□□09, bit 3) will turn ON.
Error and Cause	• A move command that exceeded a software limit of the machine was executed as follows: A user program command exceeded the software limit. The software limit was exceeded in manual operation.
Correction	• Check the program or manual operation. • Then, after clearing the motion command and resetting the alarm, use a return operation to eliminate the software limit status. Commands in the direction of the software limit will be disabled and an alarm will occur again if one is executed.

(5) Bit 5: Servo OFF

Detection Timing	• Servo OFF status is detected when a move command is executed.
Processing when Alarm Occurs	• The specified movement command will not be executed. • The Command Error Completed Status in the Motion Command Status (IW□□09, bit 3) will turn ON.
Error and Cause	• A move command (commands for positioning, external positioning, STEP operation, JOG operation, etc.) was executed when the SERVOPACK was Servo OFF status.
Correction	• After clearing the motion command and resetting the alarm, turn the SERVOPACK to the Servo ON status.

(6) Bit 6: Positioning Time Over

Detection Timing	• Positioning was not completed within the time specified in OW□□26 (Positioning Completion Check Time) after completing pulse distribution.
Processing when Alarm Occurs	• The current command was ended forcibly. • The Command Error Completed Status in the Motion Command Status (IW□□0,9 bit 3) will turn ON.
Error and Cause	One of the following is possible. • The position loop gain and speed loop gain are not set correctly, creating poor response. Or, there is oscillation. • The value of the Positioning Completion Check Time (OW□□26) is too short. • The capacity of the motor is insufficient for the machine load. • Connections are not correct between the SERVOPACK and the motor.
Correction	Check the following. • Check the SERVOPACK gain parameters. • Check connections between the SERVOPACK and the motor. • Check the motor capacity. • Check the value of the Positioning Completion Check Time (OW□□26).

Note: If the positioning Completion Check Time (OW□□26) is set to 0, the Positioning Time Over error will not occur and these items will not be checked.

(7) Bit 7: Excessive Positioning Moving Amount

Detection Timing	• Positioning command is executed.
Processing when Alarm Occurs	• The move command is not executed. • The Command Error Completed Status in the Motion Command Status (IW□□09, bit 3) will turn ON.
Error and Cause	• A move command (commands for positioning, external positioning, or STEP operation) was executed that exceeded the limit of the positioning moving amount.
Correction	• Check the moving amount for the axis being positioned.

(8) Bit 8: Excessive Speed

Detection Timing	• A move command is executed.
Processing when Alarm Occurs	• The move command is not executed. • The Command Error Completed Status in the Motion Command Status (IW□□09, bit 3) will turn ON.
Error and Cause	• The speed (movement output for one scan in case of interpolation) commanded to MECHATROLINK servo exceeds the upper limit.
Correction	• Check the settings for speed reference, interpolation command movement per scan, and speed compensation.

(9) Bit 9: Excessive Deviation

Detection Timing	• At all times other than during speed control or torque control
Processing when Alarm Occurs	• The move command is not executed. • The Command Error Completed Status in the Motion Command Status (IW□□09, bit 3) will turn ON.
Error and Cause	One of the following is possible. • The position loop gain and speed loop gain are not set correctly, creating poor response. • The value of the Error Count Alarm Detection (OL□□22) is too small. • The capacity of the motor is insufficient for the machine load. • SERVOPACK failure
Correction	Check the following and correct the problem. If the problem persists, contact the maintenance department. • Check the position loop gain and speed loop gain. • Check the value of the Error Count Alarm Detection (OL□□22). • Check the motor capacity.

Note: The above check is not performed if the Error Count Alarm Detection (OL□□22) is set to 0.

(10) Bit A: Filter Type Change Error

Detection Timing	• Continuously monitored by the motion command processing section.
Processing when Alarm Occurs	• The Change Filter Type command will not be executed. • The Command Error Completed Status in the Motion Command Status (IW□□09, bit 3) will turn ON.
Error and Cause	• An error occurs if the Change Filter Type command is executed before the specified pulse distribution has not been completed (i.e., when IW□□0C, bit 0 was OFF).
Correction	• Correct the program to execute the Change Filter Type command after Distribution Completed status (i.e., that IW□□0C, bit 0 is ON) is checked.

Note: The command running will not stop even if the above error occurs. The stop processing from the user program is needed to stop running commands when necessary.

(11) Bit B: Filter Time Constant Change Error

Detection Timing	• Continuously monitored by the motion command processing section.
Processing when Alarm Occurs	• The SCC (Change Filter Time Constant) command will not be executed. • The Command Error Completed Status in the Motion Command Status (IW□□09, bit 3) will turn ON.
Error and Cause	• An error occurs if the SCC command is executed before the specified pulse distribution has not been completed (i.e., when IW□□0C, bit 0 was OFF).
Correction	• Correct the program to execute the SCC command after Distribution Completed status (i.e., that IW□□0C, bit 0 is ON) is checked.

Note: The command running will not stop even if the above error occurs. The stop processing from the user program is needed to stop running commands when necessary.

(12) Bit D: Zero Point Unsetting

Detection Timing	• Enabled only when an absolute encoder is used for an infinite length axis and detected when the next command is set in the Motion Command (OW□□08). Commands: Positioning, External Positioning, Interpolation, Interpolation with position detection function, phase reference
Processing when Alarm Occurs	• The set command will not be executed. • The Command Error Completed Status in the Motion Command Status (IW□□09, bit 3) will turn ON.
Error and Cause	• A move command was set without executing the ZSET command (IW□□0C, bit 5 is OFF).
Correction	• After clearing the motion command and resetting the alarm, execute a Zero Point Setting operation.

(13) Bit 10: Servo Driver Synchronization Communications Error

Detection Timing	• Detected by the communication control section when the MP2100/MP2100M and SERVOPACK are linked.
Processing when Alarm Occurs	• The current command will be aborted.
Error and Cause	• Data of either MP2100/MP2100M or servo was not correctly updated.
Correction	• Check the MECHATROLINK cable and reset the alarm.

(14) Bit 11: Servo Driver Communication Error

Detection Timing	• Detected by the communication control section when communication is not synchronized between the MP2100/MP2100M and SERVOPACK.
Processing when Alarm Occurs	• The current command will be aborted. • The SERVOPACK will be Servo OFF status.
Error and Cause	• MECHATROLINK communication stopped because the cable was disconnected, there is noise interference to the communication line or the power supply to the SERVOPACK was turned OFF.
Correction	• Check the MECHATROLINK cable and reset the alarm. • If this error occurs frequently, refer to the *MECHATROLINK-II Installation Manual* (Manual No.: SIEP S800000 30) to correct the wiring and eliminate noise interference.

(15) Bit 12: Servo Driver Command Timeout Error

Detection Timing	• Detected during execution of each motion commands. • Detected by the MECHATROLINK communication control section when the Servo command responses are checked for each process.
Processing when Alarm Occurs	• The current command will be aborted.
Error and Cause	• The MECHATROLINK Servo command did not complete within the specified time (5 s).
Correction	• Check for alarms in the SERVOPACK for MECHATROLINK communication.

Note: The above error occurs when Module allocations of SERVOPACK for MECHATROLINK communication have been completed and the power is not being supplied to the SERVOPACK.

(16) Bit 13: Excessive ABS Encoder Rotations

Detection Timing	• Enabled only when an absolute encoder is used for a finite length axis, and the electronic gear used. Detected by the position management section when power is turned ON.
Processing when Alarm Occurs	• The absolute position information read from the absolute encoder when the SEN signal turned ON is ignored.
Error and Cause	• An operation error occurred when the absolute position information read from the absolute encoder is converted from pulses to reference units at power ON.
Correction	• Check the gear ratio, number of encoder pulses for other motion fixed parameters.

(17) Bit 1E: Motor Type Set Error

Detection Timing	• Detected when communications with the SERVOPACK are established.
Processing when Alarm Occurs	• None
Error and Cause	• The motor type setting (rotary/linear) of the fixed parameters does not agree with the SERVOPACK setting (for SGDH: startup selection setting Pn000.3, for SGDS: for rotary/for linear).
Correction	• Check the setting and model of the SERVOPACK.

(18) Bit 1F: Connected Encoder Type Error

Detection Timing	• Detected when communications with the SERVOPACK are established.
Processing when Alarm Occurs	• None
Error and Cause	• The motor type setting (rotary/linear) of the fixed parameter does not agree with the actual motor connected to the SERVOPACK.
Correction	• Check the motor.

6.7.3 Servo Driver Status and Servo Driver Error Codes

(1) Servo Driver Status (IW□□2C)

The status of a SERVOPACK for MECHATROLINK communication can be monitored in Servo Driver Status monitoring parameter IW□□2C.

The Servo alarm codes are listed in the following table.

Bit No.	Status	Description
Bit 0	Alarm (ALARM)	0: No alarm occurred. 1: Alarm occurred.
Bit 1	Warning (WARNG)	0: No warning occurred. 1: Warning occurred.
Bit 2	Command Ready (CMDRDY)	0: Command reception not possible (busy). 1: Command reception possible (ready).
Bit 3	Servo ON (SVON)	0: Servo OFF (baseblock) 1: Servo ON (baseblock cleared)
Bit 4	Main Power Supply ON (PON)	0: Main power OFF 1: Main power ON
Bit 5	Machine Lock (MLOCK)	0: Machine lock released 1: Machine locked
Bit 6	Zero Position (ZPOINT)	0: The APOS (absolute position) is not in the zero point. 1: The APOS (absolute position) is in the zero point range.
Bit 7	Positioning Complete (PSET)	0: Pulse distribution is not completed or the APOS is not in the positioning completed width. 1: Pulse distribution is completed and the APOS is within the positioning completed width.
Bit 8	Commanded Profile Complete (DEN)	0: Pulse distribution is being performed for positioning command. 1: Pulse distribution for positioning commands has been completed
Bit 9	Torque Restriction (T_LIM)	0: A torque limit is not being applied. 1: A torque limit is being applied.
Bit A	Latch Complete (L_CMP)	0: Latch not completed. 1: Latch completed.
Bit B	Locating Neighborhood (NEAR)	0: The APOS is outside the NEAR Signal Output Width. 1: The APOS is inside the NEAR Signal Output Width.
Bit C	Positive Software Limit (P-SOT)	0: The positive software limit has not been exceeded. 1: The positive software limit has been exceeded.
Bit D	Negative Software Limit (N-SOT)	0: The negative software limit has not been exceeded. 1: The negative software limit has been exceeded.
Bit E	Reserved	–
Bit F	Reserved	–

(2) Servo Driver Alarm Code (IW□□2D)

When the Servo Driver Error (IL□□04, bit 0) turns ON, a SERVOPACK alarm will exist. The content of the alarm can be confirmed using the Servo Driver Alarm Code (monitoring parameter IW□□2D).
The Servo alarm codes are listed in the following table.

(a) Σ-I Series

Name	Register No.	Code	Meaning
Servo Driver Alarm Code	IW□□2D	99	Normal
		94	Parameter Setting Warning
		95	MECHATROLINK Command Warning
		96	MECHATROLINK Communication Error Warning
		00	Absolute Value Data Error
		02	Parameter Corrupted
		10	Overcurrent
		11	Ground Fault
		40	Overvoltage
		41	Undervoltage
		51	Overspeed
		71	Overload (Instantaneous)
		72	Overload (Continuous)
		7A	Heat Sink Heating
		80	Absolute Encoder Error
		81	Absolute Encoder Backup Error
		82	Absolute Encoder Checksum Error
		83	Absolute Encoder Battery Error
		84	Absolute Encoder Data Error
		85	Absolute Encoder Overspeed
		B1	Gate Array 1 Error
		B2	Gate Array 2 Error
		B3	Current Feedback Phase-U Error
		B4	Current Feedback Phase-V Error
		B5	Watchdog Detector Error
		C1	Servo Run-away
		C2	Encoder Phase Error Detected
		C3	Encoder Phase-A or -B Broken
		C4	Encoder Phase-C Broken
		C5	Incremental Encoder Initial Pulses Error
		D0	Position Error Exceeded
		E5	MECHATROLINK Sync Error
		E6	MECHATROLINK Communication Error
		F1	Broken Phase in Power Line
		F3	Momentary Power Loss

(b) Σ-II Series

Name	Register No.	Code	Meaning
Servo Driver Alarm Code	IW□□2D	99	Normal
		90	Excessive Position Error Warning
		91	Overload Warning
		92	Regeneration Overload Warning
		93	Absolute Encoder Battery Error
		94	Data Setting Warning
		95	Command Warning
		96	Communication Warning
		02	Parameter Corrupted
		03	Main Circuit Detector Error
		04	Parameter Setting Error
		05	Combination Error
		09	Division Setting Error
		0A	Encoder Type Unmatch
		10	Overcurrent or Heat Sink Overheat
		30	Regeneration Error
		32	Regeneration Overload
		33	Main Circuit Wiring Error
		40	Overvoltage
		41	Undervoltage
		51	Overspeed
		71	Overload (Instantaneous Maximum Load)
		72	Overload (Continuous Maximum Load)
		73	DB Overload
		74	Inrush Resistance Overload
		7A	Heat Sink Overheat
		81	Encoder Backup Alarm
		82	Encoder Checksum Alarm
		83	Encoder Battery Alarm
		84	Encoder Data Alarm
		85	Encoder Overspeed
		86	Encoder Overheat
		B1	Speed Reference A/D Error
		B2	Torque Reference A/D Error
		B3	Current Detection Error
		B6	Gate Array Error
		BF	System Alarm
		C1	Servo Run-away
		C6	Full-closed Loop Phase-A or -B Broken
		C7	Full-closed Loop Phase-C Broken
		C8	Encoder Clear Error Multiturn Limit Setting Error
		C9	Encoder Communication Error
		CA	Encoder Parameter Error
		CB	Encoder Echoback Error
		CC	Multiturn Limit Unmatch
		D0	Excessive Position Error
		D1	Excessive Error Between Motor Load Positions
		E0	No Option
		E1	Option Timeout

Name	Register No.	Code	Meaning
Servo Driver Alarm Code (cont'd)	IW□□2D (cont'd)	E2	Option WDC Error
		E5	WDT Error
		E6	Communication Error
		E7	Failed to Detect Application Module
		E9	Bus OFF Error
		EA	SERVOPACK Failure
		EB	SERVOPACK Initial Access Error
		EC	SERVOPACK WDC Error
		ED	Command Execution Not Completed
		EF	Application Module Alarm
		F1	Broken Phase in Power Line
		F5	Motor Disconnection (when Control Power Turned ON)
		F6	Motor Disconnection (at Servo ON)

(c) Σ-III Series

Name	Register No.	Code	Meaning
Servo Driver Alarm Code	IW□□2D	000	Normal
		900	Excessive Position Error
		901	Excessive Position Error at Servo ON
		910	Overload
		911	Vibration
		920	Regeneration Overload
		930	Absolute Encoder Battery Error
		941	Parameter Change Requiring Power Recycling
		94A	Data Setting Warning 1 (Parameter Number)
		94B	Data Setting Warning 2 (Outside Data Range)
		94C	Data Setting Warning 3 (Calculation Error)
		94D	Data Setting Warning 4 (Parameter Size)
		95A	Command Warning 1 (Command Conditions Not Met)
		95B	Command Warning 2 (Unsupported Command)
		95C	Command Warning 3
		95D	Command Warning 4
		95E	Command Warning 5
		960	MECHATROLINK Communication Warning
		020	Parameter Checksum Error 1
		021	Parameter Format Error 1
		022	System Constant Checksum Error 1
		023	Parameter Password Error 1
		02A	Parameter Checksum Error 2
		02B	System Constant Checksum Error 2
		030	Main Circuit Detector Error
		040	Parameter Setting Error 1
		04A	Parameter Setting Error 2
		041	Divided Pulse Output Setting Error
		042	Parameter Combination Error
		050	Combination Error
		051	Product Not Supported Alarm
		0B0	Servo ON Command Invalid Alarm
		100	Overcurrent or Heat Sink Overheat
		300	Regeneration Error
		320	Regeneration Overload
		330	Main Circuit Wiring Error
		400	Overvoltage
		410	Undervoltage
		510	Overspeed
		511	Divided Pulse Output Overspeed
		520	Vibration Alarm
		710	Overload (Instantaneous Maximum Load)
		720	Overload (Continuous Maximum Load)
		730, 731	DB Overload
		740	Inrush Resistance Overload
		7A0	Heat Sink Overheat
		810	Encoder Backup Alarm
		820	Encoder Checksum Alarm
		830	Encoder Battery Alarm

Name	Register No.	Code	Meaning
Servo Driver Alarm Code (cont'd)	IW□□2D (cont'd)	840	Encoder Data Alarm
		850	Encoder Overspeed
		860	Encoder Overheat
		870	Full-closed Serial Encoder Checksum Alarm
		880	Full-closed Serial Encoder Data Alarm
		8A0	Full-closed Serial Encoder Scale Error
		8A1	Full-closed Serial Encoder Module Error
		8A2	Full-closed Serial Encoder Sensor Error (Incremental)
		8A3	Full-closed Serial Encoder Position Error (Absolute Value)
		B31	Current Detection Error 1
		B32	Current Detection Error 2
		B33	Current Detection Error 3
		B6A	MECHATROLINK Communication ASIC Error 1
		B6B	MECHATROLINK Communication ASIC Error 2
		BF0	System Alarm 0
		BF1	System Alarm 1
		BF2	System Alarm 2
		BF3	System Alarm 3
		BF4	System Alarm 4
		C10	Servo Run-away
		C80	Encoder Clear Error Multiturn Limit Setting Error
		C90	Encoder Communication Error
		C91	Encoder Communication Position Data Acceleration Error
		C92	Encoder Communication Timer Error
		CA0	Encoder Parameter Error
		CB0	Encoder Echoback Error
		CC0	Multiturn Limit Mismatch
		CF1	Full-closed Serial Conversion Unit Communication Error (Reception Failure)
		CF2	Full-closed Serial Conversion Unit Communication Error (Timer Stopped)
		D00	Excessive Position Error
		D01	Excessive Position Error Alarm at Servo ON
		D02	Excessive Position Error Alarm for Speed Limit at Servo ON
		D10	Excessive Error between Motor Load and Position
		E00	COM Alarm 0
		E01	COM Alarm 1
		E02	COM Alarm 2
		E07	COM Alarm 7
		E08	COM Alarm 8
		E09	COM Alarm 9
		E40	MECHATROLINK-II Transmission Cycle Setting Error
		E50	MECHATROLINK-II Sync Error
		E51	MECHATROLINK-II Sync Failure
		E60	MECHATROLINK-II Communication Error
		E61	MECHATROLINK-II Transmission Cycle Error
		EA0	DRV Alarm 0
		EA1	DRV Alarm 1
		EA2	DRV Alarm 2

Note: Alarm codes normally comprise two digits, but a three-digit code is stored when alarm monitoring has been performed on motion commands.

(d) Σ-V Series

Name	Register No.	Code	Meaning
Servo Driver Alarm Code	IW□□2D	000	Normal
		900	Position Error Overflow
		901	Excessive Position Deviation for Servo ON
		910	Overload
		911	Vibration
		920	Regeneration Overload
		921	Dynamic Brake Overload
		930	Absolute Encoder Battery Error
		94A	Data Setting Warning 1 (Parameter Number)
		94B	Data Setting Warning 2 (Data Out of Range)
		94C	Data Setting Warning 3 (Calculation Error)
		94D	Data Setting Warning 4 (Parameter Size)
		94E	Data Setting Warning 5 (Latch Mode Error)
		95A	Command Warning 1 (Command Conditions Not Met)
		95B	Command Warning 2 (Unsupported Command)
		95D	Command Warning 4 (Command Conflict)
		95E	Command Warning 5 (Subcommand Cannot Be Executed)
		95F	Command Warning 6 (Undefined Command)
		960	MECHATROLINK Communications Warning
		971	Undervoltage
		9A0	Overtravel
		020	Parameter Checksum Error 1
		021	Parameter Format Error 1
		022	System Checksum Error 1
		023	Parameter Password Error 1
		030	Main Circuit Detector Error 1
		040	Parameter Setting Error 1
		041	Encoder Output Pulse Setting Error
		042	Parameter Combination Error
		044	Semi-closed/Fully-closed Loop Control Parameter Setting Error
		050	Combination Error
		051	Unsupported Device Alarm
		0b0	Cancelled Servo ON Command Alarm
		100	Overcurrent or Heat Sink Overheated
		300	Regeneration Error
		320	Regenerative Overload
		330	Main Circuit Power Supply Wiring Error
		400	Overvoltage
		410	Undervoltage
		510	Overspeed
		511	Overspeed of Encoder Output Pulse Rate
		520	Vibration Alarm
		521	Autotuning Alarm
		710	Overload: High Load
		720	Overload: Low Load
		730 731	Dynamic Brake Overload
		740	Overload of Surge Current Limit Resistor
		7A0	Heat Sink Overheated
		7AB	Built-in Fan in SERVOPACK Stopped

Name	Register No.	Code	Meaning
Servo Driver Alarm Code (cont'd)	IW□□2D (cont'd)	810	Encoder Backup Error
		820	Encoder Checksum Error
		830	Absolute Encoder Battery Error
		840	Encoder Data Error
		850	Encoder Overspeed
		860	Encoder Overheated
		891	Encoder Module Error
		8A0	External Encoder Error of Scale
		8A1	External Encoder Error of Module
		8A2	External Encoder Error of Sensor (Incremental)
		8A3	External Encoder Error of Position (Absolute)
		b10	Speed Command A/D Error
		b11	Speed Command A/D Conversion Error
		b20	Torque Command A/D Error
		b31	Current Detection Error 1 (Phase-U)
		b32	Current Detection Error 2 (Phase-V)
		b33	Current Detection Error 3 (Current detector)
		b6A	MECHATROLINK Communications ASIC Error 1
		bF0	System Alarm 0
		bF1	System Alarm 1
		bF2	System Alarm 2
		bF3	System Alarm 3
		bF4	System Alarm 4
		C10	Servo Overrun Detected
		C20	Phase Detection Error *1
		C21	Hall Sensor Error*1
		C22	Phase Information Disagreement*1
		C50	Polarity Detection Error*1
		C51	Overtravel Detection at Polarity Detection*1
		C52	Polarity Detection Uncompleted*1
		C53	Out of Range for Polarity Detection
		C54	Polarity Detection Error 2
		C80	Absolute Encoder Clear Error and Multi-turn Limit Setting Error
		C90	Encoder Communications Error
		C91	Encoder Communications Position Data Error
		C92	Encoder Communications Timer Error
		CA0	Encoder Parameter Error
		Cb0	Encoder Echoback Error
		CC0	Multi-turn Limit Disagreement
		CF1	Feedback Option Module Communications Error (Reception error) *1
		CF2	Feedback Option Module Communications Error (Timer stop)*1
		d00	Position Error Pulse Overflow
		d01	Position Error Pulse Overflow Alarm at Servo ON
		d02	Position Error Pulse Overflow Alarm by Speed Limit at Servo ON
		d10	Motor-load Position Error Pulse Overflow
		EB0	Safety Function Drv Monitor Circuit Error*2
		EB1	Safety Function Signal Input Timing Error
		EB2	Safety Function Drv Internal Signal Error *2
		EB3	Safety Function Drv Communication Error 1 *2

Name	Register No.	Code	Meaning
Servo Driver Alarm Code (cont'd)	IW□□2D (cont'd)	EB4	Safety Function Drv Communication Error 2 *2
		EB5	Safety Function Drv Communication Error 3 *2
		EB6	Safety Function Drv Communication Data Error *2
		EC7	Safety Option Card Stop Command Error *2
		ED1	Command Execution Timeout
		F10	Main Circuit Cable Open Phase
		CPF00	Digital Operator Transmission Error 1
		CPF01	Digital Operator Transmission Error 2
		–	Not an error

* 1. Only when using the Fully-closed Module.
* 2. Only when using the safety function.

(e) Σ-7 Series

Name	Register No.	Code	Meaning
Servo Driver Alarm Code	IW□□2D	000	Normal
		900	Position Error Overflow
		901	Excessive Position Deviation for Servo ON
		910	Overload
		911	Vibration
		912	Internal Temperature Warning 1 (Control Board Temperature Error)
		913	Internal Temperature Warning 2 (Power Board Temperature Error)
		920	Regeneration Overload
		921	Dynamic Brake Overload
		923	Internal-in Fan in SERVOPACK Stopped
		930	Absolute Encoder Battery Error
		93B	Overheat Warnings
		942	Speed Ripple Compensation Information Disagreement
		94A	Data Setting Warning 1 (Parameter Number)
		94b	Data Setting Warning 2 (Data Out of Range)
		94C	Data Setting Warning 3 (Calculation Error)
		94d	Data Setting Warning 4 (Parameter Size)
		94E	Data Setting Warning 5 (Latch Mode Error)
		95A	Command Warning 1 (Command Conditions Not Met)
		95b	Command Warning 2 (Unsupported Command)
		95d	Command Warning 4 (Command Conflict)
		95E	Command Warning 5 (Subcommand Cannot Be Executed)
		95F	Command Warning 6 (Undefined Command)
		960	MECHATROLINK Communications Warning
		971	Undervoltage
		9A0	Overtravel
		9b0	Preventive Maintenance Warnings
		020	Parameter Checksum Error
		021	Parameter Format Error
		022	System Checksum Error
		024	System Alarm
		025	System Alarm
		030	Main Circuit Detector Error
		040	Parameter Setting Error
		041	Encoder Output Pulse Setting Error
		042	Parameter Combination Error

Name	Register No.	Code	Meaning
Servo Driver Alarm Code (cont'd)	IW□□2D (cont'd)	044	Semi-closed/Fully-closed Loop Control Parameter Setting Error
		050	Combination Error
		051	Unsupported Device Alarm
		070	Detected Motor Type Change
		080	Linear Encoder Scale Pitch Setting Error
		0B0	Canceled Servo ON Command Alarm
		100	Overcurrent Detected
		300	Regeneration Error
		320	Regeneration Overload
		330	Main Circuit Power Supply Wiring Error
		331	Power Monitor Input Signal Error
		400	Overvoltage
		410	Undervoltage
		450	Main Circuit Capacitor Overvoltage
		510	Overspeed
		511	Overspeed of Encoder Output Pulse Rate
		520	Vibration Alarm
		521	Autotuning Alarm
		550	Maximum Speed Setting Error
		710	Maximum Momentary Overload
		720	Maximum Continuous Overload
		730, 731	Dynamic Brake Overload
		740	Overload of Surge Current Limit Resistor
		7A1	Internal Temperature Error 1 (Control Board Temperature Error)
		7A2	Internal Temperature Error 2 (Power Board Temperature Error)
		7A3	Internal Temperature Detector Error
		7AB	Built-in Fan in SERVOPACK Stopped
		810	Encoder Backup Alarm
		820	Encoder Checksum Alarm
		830	Encoder Battery Alarm
		840	Encoder Data Alarm
		850	Encoder Overspeed
		860	Encoder Overheated
		861	Overheat
		890	Encoder Scale Error
		891	Encoder Module Error
		8A0	External Encoder Error
		8A1	External Encoder Module Error
		8A2	External Incremental Encoder Sensor Error
		8A3	External Absolute Encoder Position Error
		8A5	External Encoder Overspeed Error
		8A6	External Encoder Overheat Error
		B10	Speed Reference A/D Error
		B11	Speed Reference A/D Conversion Data Error
		B20	Torque Reference A/D Error
		B33	Current Detection Error 3
		BF0	System Alarm 0
		BF1	System Alarm 1
		BF2	System Alarm 2
		BF3	System Alarm 3
		BF4	System Alarm 4

Name	Register No.	Code	Meaning
Servo Driver Alarm Code (cont'd)	IW□□2D (cont'd)	C10	Runaway Detected
		C20	Phase Detection Error
		C21	Hole Sensor Error
		C22	Phase Information Disagreement
		C50	Magnetic Pole Detection Failed
		C51	Overtravel Detected during Magnetic Pole Detection
		C52	Magnetic Pole Detection Incomplete
		C53	Magnetic Pole Detection Variable Range Exceeded
		C54	Magnetic Pole Detection Failed 2
		C80	Absolute Encoder Clear Error and Multiturn Limit Setting Error
		C90	Encoder Communications Error
		C91	Encoder Communications Position Data Acceleration Rate Error
		C92	Encoder Communications Timer Error
		CA0	Encoder Parameter Error
		CB0	Encoder Echoback Error
		CC0	Multiturn Limit Disagreement
		CF1	Feedback Optional Module Communications Error, Reception Failed
		CF2	Feedback Optional Module Communications Error, Timer Stopped
		D00	Position Error Overflow
		D01	Position Error Overflow Alarm at Servo ON
		D02	Position Error Overflow Alarm by Speed Limit at Servo ON
		D10	Motor-load Position Error Overflow
		D30	Position Data Overflow
		E72	Feedback Optional Module Detection Failure Alarm
		EB1	Safety Function Signal Input Timing Error
		F10	Main Circuit Cable Open Phase
		F50	Motor Main Circuit Cable Disconnection

Appendix A

System Registers Lists

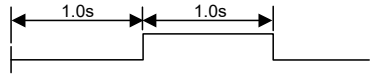
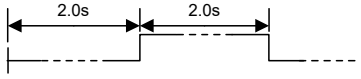
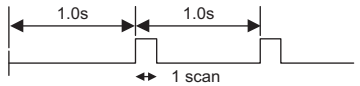
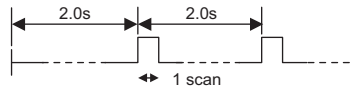
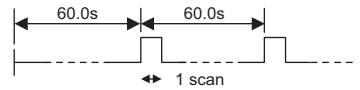
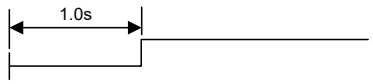
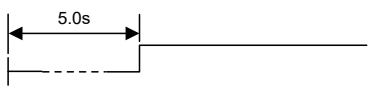
A.1 System Service Registers

A.1.1 Shared by All Drawings

Name	Register No.	Remarks
Reserved (Reserved for system use)	SB000000	Not used.
First High-speed Scan	SB000001	ON for only the first scan after high-speed scan is started.
First Low-speed Scan	SB000003	ON for only the first scan after low-speed scan is started.
Always ON	SB000004	Always ON (= 1)
Reserved (Reserved for system use)	SB000005 to SB00000F	Not used.

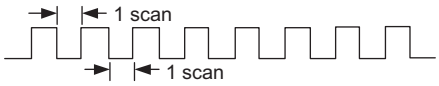
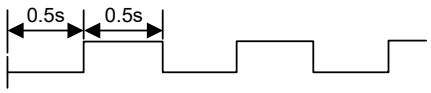
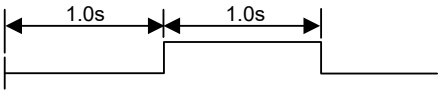
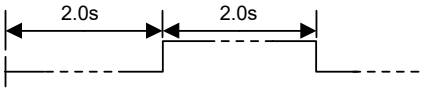
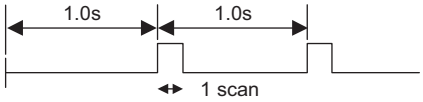
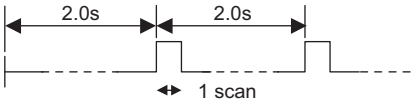
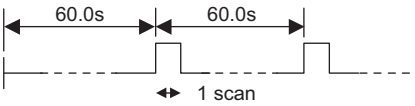
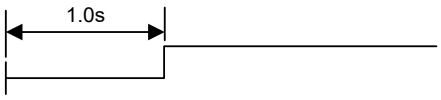
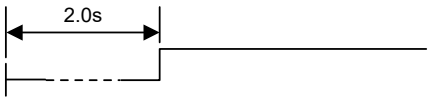
A.1.2 DWG.H Only

The following relays begin operation 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	

A.1.3 DWG.L Only

The following relays begin operation at the start of the low-speed scan.

Name	Register No.	Remarks
1-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	

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 for System Use	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 for System Use	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)
System Number	SW00021 to SW00025	Not used.
Remaining Program Memory Capacity	SL00026	Unit: Bytes
Total Memory Capacity	SL00028	Unit: Bytes

Appendix B

Motion Parameter Lists

B.1 Motion Parameter Lists

This chapter shows the lists of the motion parameters (fixed parameters, setting parameters, and monitoring parameters) for SVB and SVR Modules.
For information on how to use each motion parameter, refer to *Machine Controller MP2000 Series Built-in SVB/ SVB-01 Motion Module User's Manual* (Manual No.: SIEP C880700 33).

Note: 1. The Yes in the SVB or SVR column indicates that the motion parameter is supported by the corresponding module.
2. Entries marked with "*" are applicable when a Σ-V is used with MP2100CPU Ver. 2.61 or later.

B.1.1 Fixed Parameter List

No.	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 for system use	–	–
1	Function Selection Flag 1	Bit 0: Axis Selection (0: Finite length axis/1: Infinite length axis) Note: Set to 0 for linear type.	Yes	Yes
		Bit 1: Soft Limit (Positive Direction) Enable/Disable (0: Disabled/1: Enabled)	Yes	–
		Bit 2: Soft Limit (Negative Direction) Enable/Disable (0: Disabled/1: Enabled)	Yes	–
		Bit 3: Overtravel Positive Direction Enable/Disable (0: Disabled/1: Enabled)	Yes	–
		Bit 4: Overtravel Negative Direction Enable/Disable (0: Disabled/1: Enabled)	Yes	–
		Bits 5 to 7: Reserved for system use	–	–
		Bit 8: Interpolation Segment Distribution Processing	Yes	–
		Bit 9: Simple ABS Rotary Pos. Mode (Simple absolute infinite axis position control) (0: Disabled/1: Enabled) Note: Set to 0 for linear type.	Yes	–
		Bit A: User Constants Self-writing Function	Yes	–
		Bits B to F: Reserved for system use	–	–
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 for system use	–	–
4	Reference Unit Selection	0: pulse 3: inch 1: mm 4: μm 2: deg Note: For linear type, 0 (pulse), 1 (mm), and 4 (μm) can be used. If 2 (deg) or 3 (inch) is selected, the selected unit will be converted to mm.	Yes	Yes
5	Number of Digits below Decimal Point	1 = 1 digit	Yes	Yes
6	Travel Distance per Machine Rotation (rotary motor)	1 = 1 reference unit	Yes	Yes
	Linear Scale Pitch (linear motor)	1 = 1 reference unit	Yes	Yes
8	Servo Motor Gear Ratio	1 = 1 rotation Note: Invalid for linear type	Yes	Yes

(cont'd)

No.	Name	Contents	SVB	SVR
9	Machine Gear Ratio	1 = 1 rotation Note: Invalid for linear type	Yes	Yes
10	Infinite Length Axis Reset Position (POSMAX)	1 = 1 reference unit Note: 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 28	–	Reserved for system use	–	–
29	Motor Type	0: Rotary motor 1: Linear motor	Yes	–
30	Encoder Selection	0: Incremental encoder 1: Absolute encoder 2: Absolute encoder (Incremental encoder is used.) 3: Reserved for system use	Yes	–
31 to 33	–	Reserved for system use	–	–
34	Rated Motor Speed (Rotary Motor)	1 = 1 min^{-1}	Yes	Yes
	Rated Speed (Linear Motor)	1 = 0.1 m/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 Motor)	1 = 1 pulse/scale pitch	Yes	Yes
38	Maximum Number of Absolute Encoder Turns Rotation	1 = 1 rotation Note: 1. Set to 0 when a direct drive motor is being used. 2. Invalid for linear type	Yes	–
40 to 41	–	Reserved for system use	–	–
42	Feedback Speed Movement Averaging Time Constant	1 = 1 ms	Yes	Yes

B.1.2 Setting Parameter List

The register number "OW□□00" indicates the leading output register number + 00.

Note: The Yes in the SVB or SVR column indicates that the motion parameter is supported by the corresponding module.

Register No.	Name	Contents	SVB	SVR
OW□□00	RUN Command Setting	Bit 0: Servo ON (0: OFF/1: ON)	Yes	Yes
		Bit 1: Machine Lock (0: Normal Operation/1: Machine Lock)	Yes	
		Bits 2 to 3: Reserved for system use	–	–
		Bit 4: Latch Detection Demand (0: OFF/1: ON)	Yes	
		Bit 5: Reserved for system use	–	–
		Bit 6: POSMAX Turn Number Presetting Demand (0: OFF/1:ON) • Set to 0 for linear type.	Yes	Yes
		Bit 7: Request ABS Rotary Pos. Load (Absolute system infinite length position information load request) (0: OFF/1:ON) • Set to 0 for linear type.	Yes	
		Bit 8: Forward Outside Limiting Torque/Thrust Input (0: OFF/1: ON)	Yes	
		Bit 9: Reverse Outside Limiting Torque/Thrust Input (0: OFF/1: ON)	Yes	
		Bit A: Reserved for system use	–	–
		Bit B: Integration Reset (0: OFF/1: ON)	Yes	
		Bit C: Reserved for system use	–	–
		Bit D*: Latch Completion Status Clear Request (0: OFF/1: ON)	Yes	
		Bit E: Communication Reset (0: OFF/1: ON)	Yes	
		Bit F: Alarm Clear (0: OFF/1: ON)	Yes	Yes
OW□□01	Mode Setting 1	Bit 0: Excessive Deviation Error Level Setting (0: Alarm/1: Warning)	Yes	
		Bits 1 to 2: Reserved for system use	–	–
		Bit 3: Speed Loop P/PI Switch	Yes	
		Bit 4: Gain Switch	Yes	
		Bit 5: Gain Switch 2	Yes	
		Bit 6*: Latch Operation Selection (0: Regular latch/1: Continuous latch)	Yes	
		Bits 6 to F: Reserved for system use	–	–
OW□□02	Mode Setting 2	Bit 0: Monitor 2 Enabled (0: Disabled/1: Enabled)	Yes	
		Bits 1 to F: Reserved for system use	–	–
OW□□03	Function Setting 1	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%)	Yes	Yes
		Bits 4 to 7: Acceleration/Deceleration Degree Unit Selection 0: Reference unit/s^2 1: ms	Yes	Yes
		Bits 8 to B: Filter Type Selection 0: None 1: Exponential acceleration/deceleration filter 2: Moving average filter	Yes	Yes
		Bits C to F: Torque Unit Selection 0: Percentage of rated toque (1 = 0.01%) 1: Percentage of rated toque (1 = 0.0001%)	Yes	Yes

(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 for system use	–	–
		Bits C to F: Bank Selector	Yes	
OW□□05	Function Setting 3	Bit 0: Reserved for system use	–	–
		Bit 1: Phase Reference Creation Calculation Disable (0: Enabled/1: Disabled)	Yes	
		Bits 2 to A: Reserved for system use	–	–
		Bit B: Zero Point Return Input Signal (0: OFF/1: ON)	Yes	
		Bits C to F: Reserved for system use	–	–
OW□□06 to OW□□07	–	Reserved for system use	–	–

(cont'd)

Register No.	Name	Contents	SVB	SVR
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) (Reserved for system use) 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 Integration Time Constant) 27: PPRM_WR (Stored Parameter Write)	Yes	Yes
OW□□09	Motion Command Control Flag	Bit 0: Holds a Command (0: OFF/1: ON)	Yes	Yes
		Bit 1: Interrupt a Command (0: OFF/1: ON)	Yes	Yes
		Bit 2: Moving Direction (JOG/STEP) (0: Forward rotation/1: Reverse rotation)	Yes	Yes
		Bit 3: Zero Point Return Direction Selection (0: Reverse rotation/1: Forward rotation)	Yes	
		Bit 4: Latch Zone Effective Selection (0: Disabled/1: Enabled)	Yes	
		Bit 5: Position Reference Type (0: Incremental Value Add Method/1: Absolute Value Set Method)	Yes	Yes
		Bit 6: Phase Compensation Type (0: Incremental Value Add Method/1: Absolute Value Set Method)	Yes	
		Bits 7 to F: Reserved for system use	–	–
OW□□0A	Motion Subcommand	0: NOP (No Command)	Yes	Yes
		1: PRM_RD (Read User constant) (Read SERVOPACK parameter) 2: PRM_WR (Write User constant) (Write SERVOPACK parameter) 3: Reserved 4: SMON (Status Monitor)	Yes	
		5: FIXPRM_RD (Read Fixed Parameters)	Yes	Yes
OW□□0B	–	Reserved for system use		
OL□□0C	Torque/Thrust Reference Setting	Unit is according to OW□□03, bits 12 to 15 (Torque Unit Selection).	Yes	Yes

(cont'd)

Register No.	Name	Contents	SVB	SVR
OW□□0E	Speed Limit Setting at the Torque/Thrust Reference	1 = 0.01% (percentage of rated speed)	Yes	
OW□□0F	–	Reserved for system use	–	–
OL□□10	Speed Reference Setting	Unit is according to OW□□03, bits 0 to 3 (Speed Unit Selection).	Yes	Yes
OW□□12 to OW□□13	–	Reserved for system use	–	–
OL□□14	Limiting Torque/Thrust Setting at the Speed Reference	Unit is according to OW□□03, bits C to F (Torque Unit Selection).	Yes	
OL□□16	Secondly Speed Compensation	Unit is according to OW□□03, bits 0 to 3 (Speed Unit Selection).	Yes	Yes
OW□□18	Override	1=0.01%	Yes	
OW□□19 to OW□□1B	–	Reserved for system use	–	–
OL□□1C	Position Reference Setting	1 = 1 reference unit	Yes	Yes
OL□□1E	Width of Positioning Completion	1 = 1 reference unit	Yes	
OL□□20	NEAR Signal Output Width	1 = 1 reference unit	Yes	
OL□□22	Error Count Alarm Detection	1 = 1 reference unit	Yes	
OL□□24	–	Reserved for system use	–	–
OW□□26	Positioning Completion Check Time	1=1 ms	Yes	
OW□□27	–	Reserved for system use	–	–
OL□□28	Phase Correction Setting	1 = 1 reference unit	Yes	
OL□□2A	Latch Zone Lower Limit Setting	1 = 1 reference unit	Yes	
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 Feed Forward	1 = 0.01% (percentage of distribution segment)	Yes	
OW□□31	Speed Compensation	1 = 0.01% (percentage of rated speed)	Yes	Yes
OW□□32	Position Loop Integration Time Constant	1=1 ms	Yes	
OW□□33	–	Reserved for system use	–	–
OW□□34	Speed Loop Integration Time Constant	1=0.01 ms	Yes	
OW□□35	–	Reserved for system use	–	–
OL□□36	Straight Line Acceleration/ Acceleration Time Constant	Unit is according to OW□□03, bits 4 to 7 (Acceleration/Deceleration Degree Unit Selection).	Yes	Yes
OL□□38	Straight Line Deceleration/ Deceleration Time Constant	Unit is according to OW□□03, bits 4 to 7 (Acceleration/Deceleration Degree Unit Selection).	Yes	Yes
OW□□3A	Filter Time Constant	1=0.1 ms	Yes	Yes

(cont'd)

Register No.	Name	Contents	SVB	SVR
OW□□3B	Bias Speed for Exponential Acceleration/ Deceleration Filter	Unit is according to OW□□03, bits 0 to 3 (Speed Unit Selection).		Yes
OW□□3C	Zero Point Return Method	0: DEC1 + C (DEC1 and C-phase) 1: ZERO (Zero signal) 2: DEC1 + ZERO (DEC1 and ZERO Signal) 3: C (C-phase)	Yes	
		4 to 10: Reserved for system use	–	–
		11: C Pulse Only 12: P-OT +C-phase 13: P-OT Only 14: HOME LS +C-phase 15: HOME LS Only 16: N-OT +C-phase 17: N-OT Only 18: INPUT +C-phase 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 Selection).	Yes	
OL□□40	Creep Speed	Unit is according to OW□□03, bits 0 to 3 (Speed Unit Selection).	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 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 liner type	Yes	Yes
OW□□4E	Servo User Monitor Setting	Bits 0 to 3: Monitor 1 (Cannot be set.) Bits 4 to 7: Monitor 2 Bits 8 to B: Monitor 3 (Cannot be set.) Bits C to F: Monitor 4	Yes	
OW□□4F	Servo Driver Alarm Monitor No.	Sets the number of the alarm to monitor.	Yes	
OW□□50	Servo Driver User Constant No. (SERVOPACK parameter No. for motion command)	Set the number of the SERVOPACK parameter.	Yes	
OW□□51	Servo Driver User Constant Size	Sets the number of words in the SERVOPACK parameter.	Yes	
OL□□52	Servo Driver User Constant Set Point (SERVOPACK parameter setting value for motion command)	Sets the setting for the SERVOPACK parameter.	Yes	

(cont'd)

Register No.	Name	Contents	SVB	SVR
OW□□54	Servo Driver for Assistance User Constant No. (SERVOPACK parameter No. for motion subcommand)	Set the number of the SERVOPACK parameter.	Yes	
OW□□55	Servo Driver for Assistance User Constant Size (SERVOPACK parameter size for motion subcommand)	Sets the number of words in the SERVOPACK parameter.	Yes	
OL□□56	Servo Driver for Assistance User Constant Set Point (SERVOPACK parameter setting value for motion subcommand)	Sets the setting for the SERVOPACK parameter.	Yes	
OW□□58 to OW□□5B	–	Reserved for system use	–	–
OW□□5C	Fixed Parameter Number	Sets the number of the fixed parameter to read with the FIXPRM_RD motion subcommand.	Yes	Yes
OW□□5D	–	Reserved for system use	–	–
OL□□5E	Encoder Position When Power is OFF (Lower 2 words)	1=1 pulse • For linear type, do not set this register.	Yes	
OL□□60	Encoder Position When Power is OFF (Upper 2 words)	1=1 pulse • For linear type, do not set this register.	Yes	
OL□□62	Pulse Position When Power is OFF (Lower 2 words)	1=1 pulse • For linear type, do not set this register.	Yes	
OL□□64	Pulse Position When Power is OFF (Upper 2 words)	1=1 pulse • For linear type, do not set this register.	Yes	
OL□□66 to OL□□6E	–	Reserved for system use	–	–
OW□□70 to OW□□7F	Command Buffer for Servo Driver Transmission Reference Mode	This area is used for command data when MECHATROLINK servo commands are specified directly.	Yes	

B.1.3 Monitoring Parameter List

Register number "IW□□00" indicates the leading input register number + 00.

Note: The Yes in the SVB or SVR column indicates that the motion parameter is supported by the corresponding module.

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 for system use	–	–
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 for system use	–	–
		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	–
		Bit A*: Servo Driver Stop Signal Input	Yes	–
		Bits B to 1F: Reserved for system use	–	–
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 for system use	–	–
		Bit D: Zero Point Unsetting Note: Invalid for linear type.	Yes	–
		Bit E and F: Reserved for system use	–	–
		Bit 10: Servo Driver Synchronization Communications Error	Yes	–
		Bit 11: Servo Driver Communication Error	Yes	–
		Bit 12: Servo Driver Command Timeout Error	Yes	–
		Bit 13: Excessive ABS Encoder Rotations Note: Invalid for linear type	Yes	–
		Bits 14 to 1D: Reserved for system use	–	–
		Bit 1E: Motor type set error	Yes	–
		Bit 1F: Connected Encoder Type Error	Yes	–
IL□□06	–	Reserved for system use	–	–

(cont'd)

Register No.	Name	Contents	SVB	SVR
IW□□08	Motion Command Response Code	Same as OW□□08 (Motion Command).	Yes	Yes
IW□□09	Motion Command Status	Bit 0: Command Execution Flag (BUSY)	Yes	Yes
		Bit 1: Command Hold Completed (HOLDL)	Yes	Yes
		Bit 2: Reserved for system use		
		Bit 3: Command Error Completed Status (FAIL) (Command Encoder Type Error)	Yes	Yes
		Bits 4 to 6: Reserved for system use	–	–
		Bit 7: Reset Absolute Encoder Completed	Yes	–
		Bit 8: Command Execution Completed (COMPLETE)	Yes	Yes
		Bits 9 to F: Reserved for system use	–	–
IW□□0A	Subcommand Response Code	Same as OW□□0A (Motion Subcommand).	Yes	Yes
IW□□0B	Subcommand Status	Bit 0: Command Executing Flag	Yes	Yes
		Bits 1 to 2: Reserved for system use	–	–
		Bit 3: Command Error Completed Status (Command Error Occurrence)	Yes	Yes
		Bits 4 to 7: Reserved for system use	–	–
		Bit 8: Command Execution Completed	Yes	Yes
		Bits 9 to F: Reserved for system use	–	–
IW□□0C	Position Management Status	Bit 0: Discharging Completed (DEN)	Yes	Yes
		Bit 1: Positioning Completed (POSCOMP)	Yes	Yes
		Bit 2: Latch Complete (LCOMP)	Yes	–
		Bit 3: NEAR Position (NEAR)	Yes	Yes
		Bit 4: Zero Point Position (ZERO)	Yes	Yes
		Bit 5: Zero Point Return (Setting) Completed (ZRNC)	Yes	Yes
		Bit 6: During Machine Lock (MLKL)	Yes	–
		Bit 7: Reserved for system use	–	–
		Bit 8: ABS Rotary Pos. LOAD Complete (ABS System Infinite Length Position Control Information Load Completed) (ABSLDE) Note: Invalid for linear type	Yes	–
		Bit 9: POSMAX Turn Preset Complete (TPRSE) Note: Invalid for linear type	Yes	Yes
		Bits A to F: Reserved for system use		
IW□□0D	–	Reserved for system use	–	–
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 (DPOS)	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	–

(cont'd)

Register No.	Name	Contents	SVB	SVR
IL□□1C	Reference Position Incrementing Value Monitor	1 = 1 reference unit		Yes
IL□□1E	Number of POSMAX Turns	1 = 1 turn Note: Invalid for linear type	Yes	Yes
IL□□20	Speed Reference Output Monitor	pulse/s	Yes	–
IL□□22 to IL□□2A	–	Reserved for system use	–	–
IW□□2C	Servo Driver Status	Bit 0: Alarm Occurrence (ALM) Bit 1: Warning Occurrence (WARNING) 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: Positioning Complete (PSET)/Speed Coincidence (V-CMP) Bit 8: Commanded Profile Complete (DEN)/Zero Speed (ZSPD) Bit 9: Torque Restriction (T_LIM) Bit A: Latch Complete (L_CMP) Bit B: Locating Neighborhood (NEAR)/Speed Limit (V_LIM) Bit C: Position Software Limit (P-SOT) Bit D: Negative Software Limit (N-SOT)	Yes	
		Bits E to F: Reserved for system use	–	–
IW□□2D	Servo Driver Alarm Code	Stores the alarm codes from the servo driver.	Yes	–
IW□□2E	Network Servo I/O Monitor	Bit 0: Positive Drive Prohibited Input (P_OT) Bit 1: Negative Drive Prohibited Input (N_OT) Bit 2: Zero Point Return Deceleration Limit Switch Input (DEC) Bit 3: Encoder Phase-A Input (PA) Bit 4: Encoder Phase-B Input (PB) Bit 5: Encoder Phase-C Input (PC) Bit 6: First External Latch Input (EXT1) Bit 7: Second External Latch Input (EXT2) Bit 8: Third External Latch Input (EXT3) Bit 9: Brake Output (BRK) Bit A*: Stop Signal (HWBB) Bit B: Reserved for system use Bit C to Bit F: For SGDH + NS115, SGDS, SGDV: CN1 input signal (IO12 to IO15) For other SERVOPACKs: Reserved for system use	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 for system use	–	–
IL□□34	Servo Driver User Monitor 4	Stores the result of the selected monitor.	Yes	–

(cont'd)

Register No.	Name	Contents	SVB	SVR
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	–
IL□□38	Servo Driver User Constant Reading Data (SERVOPACK parameter reading area 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 area 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: Rotary type motor 1: Linear type motor	Yes	–
IL□□40	Feedback Speed	Unit is according to OW□□03, bits 0 to 3 (Speed Unit Selection).	Yes	Yes
IL□□42	Feedback Torque/ Thrust	Unit is according to OW□□03, bits 12 to 15 (Torque Unit Selection).	Yes	Yes
IW□□44*	Completed Latching Sequence Number	1 = Once	Yes	
IW□□45*	Continuous Latching Sequence Completed Cycle Number	1 = 1 cycle	Yes	
IW□□46 to IW□□55	–	Reserved for system use	–	–
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 for system use	–	–
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	–

(cont'd)

Register No.	Name	Contents	SVB	SVR
IL□□64	Pulse Position When the Power is OFF (Upper 2 Words)	1 = 1 pulse	Yes	–
IW□□66 to IW□□6F	–	Reserved for system use	–	–
IW□□70 to IW□□7F	Response Buffer for Transparent Command Mode	This area is used for the response data area when MECHATROLINK servo commands are specified directly.	Yes	

Appendix C

Motion API

This chapter shows the list of motion API. For details, refer to the MP2100/MP2100M Motion API reference file (PCAPI.chm) which is installed in the CD-ROM (CPMC-MPA700) for MP2100/MP2100M Motion API.

C.1 Motion API

C.1.1 Common APIs

The following tables show the list of common APIs.

(1) Device

Function Name	Summary
ymcClearAllAxes ( )	Deletes all axis definitions that were defined by the MP2100/MP2100M.
ymcClearAxis ( )	Deletes the axis definition specified by the axis handle.
ymcClearDevice ( )	Deletes the device specified by the device handle.
ymcDeclareDevice ( )	Specifies the axis number and axes being used and gets a device handle.
ymcDeclareAxis ( )	Defines an axis and gets an axis handle.
ymcGetAxisHandle ( )	Specifies the physical axis and axis name and gets the registered axis handle.

(2) Unit Conversion Function

Function Name	Summary
ymcConvertFloat2Fix ( )	Converts floating-point data to fixed-point data with the specified number of digits below the decimal point.
ymcConvertFix2Float ( )	Converts fixed-point data with the specified number of digits below the decimal point to floating-point data.

(3) Parameter Operations

Function Name	Summary
ymcGetMotionParameter Value ( )	Gets the motion parameter data of the axis specified by the axis handle.
ymcSetMotionParameter Value ( )	Sets the data to the motion parameter of the axis specified by the axis handle.
ymcDefinePosition ( )	Sets the current position of the device specified by the device handle.

C.1.2 Sequential APIs

The following tables show the list of Sequential APIs.

(1) Positioning Function

Function Name	Summary
ymcMovePositioning ( )	Executes PTP positioning in the axis specified by the device handle at the specified acceleration/deceleration rates.
ymcMoveJOG ( )	Executes JOG operation in the axis specified by the device handle.
ymcStopJOG ( )	Stops JOG operation in the axis specified by the device handle.
ymcMoveHomePosition ( )	Performs home position return in the specified axis.
ymcMoveIntimePositioning ( )	Clamps the feed speed so that moving is completed in the specified time and executes PTP positioning.
ymcMoveExternalPositioning ( )	If an external positioning signal is input during positioning, the specified axis will move only the “external positioning moving amount” (incremental value).
ymcMoveDriverPositioning ( )	Executes PTP positioning in the axis specified by the device handle.

(2) Interpolation Function

Function Name	Summary
ymcMoveLinear ( )	Executes linear interpolation in the axis specified by the device handle.
ymcMoveCircularCenter ( )	Executes circular interpolation in the specified direction at the specified tangential speed along an arc with the center point coordinate (u, v) specified by lpCenter.
ymcMoveCircularRadius ( )	Executes circular interpolation in the specified direction at the specified tangential speed along an arc with the specified radius.
ymcMoveHelicalCenter ( )	Executes circular interpolation in the specified direction at the specified tangential speed along an arc with the center point coordinates (u, v) specified by lpCenter and simultaneously executes linear interpolation for three axes outside of the circular interpolation plane.
ymcMoveHelicalRadius ( )	Executes circular interpolation in the specified direction at the specified tangential speed along an arc with the specified radius and simultaneously executes linear interpolation in a third axis outside of the circular interpolation plane.

(3) Torque Reference Function

Function Name	Summary
ymcMoveTorque ( )	Specifies the torque for a torque control axis.

(4) Synchronization Function

Function Name	Summary
ymcDisableGear ( )	Turns the Gear OFF by the specified method. Multiple slave axes can be selected for a single master axis.
ymcEnableGear ( )	Accelerates with speed control and turns the Gear ON when speed coincidence is detected. Multiple slave axes can be selected for a single master axis.
ymcSetGearRatio ( )	Sets the gear ratio. Multiple axes can be selected.

(5) Compensation Function

Function Name	Summary
ymcPositionOffset ( )	Executes position compensation in the axis specified by the device handle at the specified time or position.

(6) Motion Operations

Function Name	Summary
ymcChangeDynamics ( )	Changes the motion data (acceleration, deceleration, position, and speed) for the specified axis.
ymcStopMotion ( )	Stops the axis specified by the device handle by the specified stop method (immediate or decelerating to stop).

(7) Driver Operations

Function Name	Summary
ymcServoControl ( )	Turns the specified axis' servo ON or OFF. Verifies that the servo has gone ON or OFF within the specified timeout time.

(8) Others

Function Name	Summary
ymcDisableLatch ( )	Disables the Latch function in the specified axis.
ymcEnableLatch ( )	Enables the Latch function in the specified axis.
ymcWaitTime ( )	Waits for the specified timeout set time.

C.1.3 System APIs

The following tables show the list of system APIs.

(1) Data Operations

Function Name	Summary
ymcSetIoDataBit ( )	Sets the specified device type bit.
ymcGetIoDataBit ( )	Gets the specified device type bit.
ymcSetIoDataValue ( )	Sets the specified device type data.
ymcGetIoDataValue ( )	Gets the specified device type data.
ymcSetRegisterData ( )	Sets the number of register data elements starting from the register that corresponds to the RegisterData handle.
ymcGetRegisterData ( )	Gets the number of register data elements starting from the register address that corresponds to the RegisterData handle.
ymcGetRegisterDataHandle ( )	Gets the RegisterData handle from a register name.
ymSetGroupRegisterData ( )	Sets the number of register data elements starting from the register that corresponds to multiple RegisterData handles.
ymcGetGroupRegisterData ( )	Gets the number of register data elements starting from the register that corresponds to multiple RegisterData handles.
ymcGetBusIFInfo ( )	Gets information relating to the bus type interface.
ymcSetBusIFData ( )	Writes to the bus type interface.
ymcGetBusIFData ( )	Reads from the bus type interface.

(2) System Information

Function Name	Summary
ymcGetAlarm ( )	Reads the specified number of alarm entries from the MP2100/MP2100M beginning with the most recent alarm entry. Clears the alarm specified by the alarm handle.
ymcClearAlarm ( )	Clears the alarm specified by the alarm handle.
ymcClearServoAlarm ( )	Clears the servo alarm in the axis specified by the axis handle.

(3) System Operations

Function Name	Summary
ymcOpenController ( )	Specifies the communication medium for the target MP2100/MP2100M and gets a controller handle.
ymcCloseController ( )	Resets the target MP2100/MP2100M specification and opens the controller handle.
ymcSetController ( )	Specifies a controller handle and switches the target MP2100/MP2100M.
ymcGetController ( )	Gets the controller handle that is currently targeted.
ymcGetLastError ( )	Gets the error information for the function that was executed last.
ymcResetController ( )	Resets the applicable controller.

(4) Calendar Operations

Function Name	Summary
ymcSetCalendar ( )	Sets the MP2100/MP2100M's calendar.
ymcGetCalendar ( )	Gets the MP2100/MP2100M's calendar information.

(5) Other

Function Name	Summary
ymcSetAPITimeoutValue ( )	Sets the timeout detection time to be used when the API is executed.

Index

Symbols

A

B

C

D

E

F

H

I

L

M

N

O

P

R

S

T

U

V

W

Z

Revision History

The revision dates and numbers of the revised manuals are given on the bottom of the back cover.

MANUAL NO. SIEP C880700 01D <5>-1
- WEB revision number
- Revision number

Published in Japan April 2011
- Date of publication

Date of Publication	Rev. No.	WEB Rev. No.	Section	Revised Contents
August 2022	<7>	0	All chapters	Revision: MPE720 windows changed to Ver.7
			Back cover	Revision: Address
May 2020	<6>	2	1.6, 3.2.2, 3.2.3, 4.1.2, 4.1.6, 6.4.5 (1), F.1.1, F.2.1 (1), F.4.1	Partly revised.
			Back cover	Revision: Address
November 2018		1	6.4.5 (8) (a) (b)	Revision: Description of the Machine Controller model and Register No.SW00800
			Back cover	Revision: Address
January 2018		0	2.1.2	Revision: Conformity Standard
			3.1.1 (2)	Revision: Software Specifications
			Back cover	Revision: Address
January 2017	<5>	3	5.3.1 (1)	Revision: Notes for DWG Registers
			Back cover	Revision: Address
April 2016		2	Front cover	Revision: Format
			1.3	Revision: MP2100/MP2100M Modules
			1.5	Revision: Devices Connectable to MECHATROLINK
			1.6	Revision: Cables, Included Accessories, and Options
			2.1.1	Revision: Model number
			6.2.1	Addition: Installing the battery for JAPMC-MC2100, JAPMC-MC210□-E, or JAPMC-MC214□-E
			Back cover	Revision: Address and format
April 2011		1	3.2.3 (2)	Revision: Connector model
December 2010		0	–	Based on Japanese user's manual, SIJPC88070001E <9>-1, available on the Web in July 2010.
			Front cover	Revision: Format
			All chapters	Revision: All revised
			Preface	Addition: Warranty
			Back cover	Revision: Address, format
May 2006	<4>	0	10.2.2 (1)	Revision: Number of eccoder resolution
November 2005	<3>	0	7.2.2 (2), (19)	Revision: Description of Deviation Abnormal Detection Error Level (OB□□010)
			Back cover	Revision: Address
January 2005	<2>	0	All chapters	Addition: MP2100M
			1.2	Addition: Outline of MP2100M
			2.2	Addition: MP2100M system configuration
			3.6	Addition: MP2100M Startup
			4.4	Addition: MP2100M LED indicators and switch settings
			Chapter 7	Revision: Parameter descriptions
			8.2.22	Addition: Reset Absolute Encoder (ABS_RST)
			Chapter 9	Addition: Control block diagram
			Chapter 11	Addition: Virtual motion module (SVR)
			12.2.1	Addition: Battery installation
December 2003	<1>	0	2.2.2	Addition: I/O module models (JEPMC-AN2900 and JEPMC-AN2910)
			2.3.2 and 10.2	Revision: Battery model (BA000517 to ZZK000064)
			6.5.2 (2)-(a)	Addition: Distributed I/O module models (JEPMC-AN2900 and JEPMC-AN2910)
			7.2.2 and 7.2.3	Deletion: Parameter size Addition: Parameter number
			11.1.4	Addition: Motion program alarm
			Back cover	Revision: Address
April 2003	–	–	–	First edition

Machine Controller

MP210□/MP210□M

USER'S MANUAL
Design and Maintenance

IRUMA BUSINESS CENTER (SOLUTION CENTER)
480, Kamifujisawa, Iruma, Saitama, 358-8555, Japan
Phone: +81-4-2962-5151 Fax: +81-4-2962-6138
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
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
www.yaskawa.com.br

YASKAWA EUROPE GmbH
Hauptstraße 185, 65760 Eschborn, Germany
Phone: +49-6196-569-300 Fax: +49-6196-569-398
www.yaskawa.eu.com E-mail: info@yaskawa.eu.com

YASKAWA ELECTRIC KOREA CORPORATION
18F, Hi Investment & Securities Building, 66 Yeoui-daero, Yeongdeungpo-gu, Seoul, 07325, Korea
Phone: +82-2-784-7844 Fax: +82-2-784-8495
www.yaskawa.co.kr

YASKAWA ASIA PACIFIC PTE. LTD.
30A, Kallang Place, #06-01, 339213, Singapore
Phone: +65-6282-3003 Fax: +65-6289-3003
www.yaskawa.com.sg

YASKAWA ELECTRIC (THAILAND) CO., LTD.
59, 1F-5F, Flourish Building, Soi Ratchadapisek 18, Ratchadapisek Road, Huaykwang, Bangkok, 10310, Thailand
Phone: +66-2-017-0099 Fax: +66-2-017-0799
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
www.yaskawa.com.cn

YASKAWA ELECTRIC (CHINA) CO., LTD. BEIJING OFFICE
Room 1011, Tower W3 Oriental Plaza, No.1, East Chang An Avenue,
Dong Cheng District, Beijing, 100738, China
Phone: +86-10-8518-4086 Fax: +86-10-8518-4082

YASKAWA ELECTRIC TAIWAN CORPORATION
12F, No. 207, Section 3, Beishin Road, Shindian District, New Taipei City 23143, Taiwan
Phone: +886-2-8913-1333 Fax: +886-2-8913-1513 or +886-2-8913-1519
www.yaskawa.com.tw

YASKAWA

YASKAWA ELECTRIC CORPORATION

MANUAL NO. SIEP C880700 01E <7>-0

Published in Japan August 2022
22-4-18
Original instructions