B-1726(7)

FC6A SERIES

MICROSmart

Ladder Programming Manual

Automation Organizer
WindLDR

SAFETY PRECAUTIONS

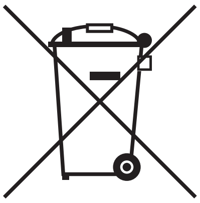
- Read the "FC6A Series MICROSmart Ladder Programming Manual" to ensure correct operation before starting installation, wiring, operation, maintenance, and inspection of the FC6A Series MICROSmart.
- All FC6A Series MICROSmart modules are manufactured under IDEC's rigorous quality control system, but users must add a backup or failsafe provision to the control system when using the FC6A Series MICROSmart in applications where heavy damage or personal injury may be caused, in case the FC6A Series MICROSmart should fail.
- In this manual, safety precautions are categorized in order of importance:

⚠ **Warning** **Warning notices are used to emphasize that improper operation may cause severe personal injury or death.**

- The FC6A Series MICROSmart is not designed for use in applications requiring a high degree of reliability and safety. The FC6A Series MICROSmart should not be used for such applications.
- When using the FC6A Series MICROSmart in applications (not described above) that require a high degree of reliability in terms of functionality and precision, appropriate measures such as failsafe mechanisms and redundant mechanisms must be taken for the system containing the FC6A Series MICROSmart. The following are specific examples.
  - Emergency stop and interlocking circuits must be configured outside the FC6A Series MICROSmart.
  - If relays or transistors in the FC6A Series MICROSmart output circuits should fail, outputs may remain at on or off state. For output signals which may cause serious accidents, configure monitor circuits outside the FC6A Series MICROSmart.
  - The FC6A Series MICROSmart self-diagnostic function may detect internal circuit or program errors, stop programs, and turn outputs off. Configure circuits so that the system containing the FC6A Series MICROSmart is not jeopardized when outputs turn off.
- Turn off power to the FC6A Series MICROSmart before installation, removal, wiring, maintenance, and inspection of the FC6A Series MICROSmart. Failure to turn power off may cause damage, electrical shocks or fire hazard.
- Special expertise is required to install, wire, program, and operate the FC6A Series MICROSmart. People without such expertise must not use the FC6A Series MICROSmart.
- Install the FC6A Series MICROSmart according to the instructions described in the "FC6A Series MICROSmart User's Manual". Improper installation will result in falling, failure, or malfunction of the FC6A Series MICROSmart.

⚠ **Caution** **Caution notices are used where inattention might cause personal injury or damage to equipment.**

- The FC6A Series MICROSmart is designed for installation in a cabinet. Do not install the FC6A Series MICROSmart outside a cabinet.
- Install the FC6A Series MICROSmart in environments described in the "FC6A Series MICROSmart User's Manual". If the FC6A Series MICROSmart is used in places where the FC6A Series MICROSmart is subjected to high-temperature, high-humidity, condensation, corrosive gases, excessive vibrations, or excessive shocks, then electrical shocks, fire hazard, or malfunction will result.
- The environment for using the FC6A Series MICROSmart is "Pollution degree 2." Use the FC6A Series MICROSmart in environments of pollution degree 2 (according to IEC 60664-1).
- Prevent the FC6A Series MICROSmart from falling while moving or transporting the FC6A Series MICROSmart, otherwise damage or malfunction of the FC6A Series MICROSmart will result.
- Wiring must use lead sizes that are appropriate for the applied voltage and current. Terminal screws must be tightened with the prescribed tightening torque.
- Prevent metal fragments and pieces of wire from dropping inside the FC6A Series MICROSmart housing. Put a cover on the FC6A Series MICROSmart modules during installation and wiring. Ingress of such fragments and chips may cause fire hazard, damage, or malfunction.
- Use a power supply of the rated value. Use of a wrong power supply may cause fire hazard.
- Use an IEC 60127-approved fuse on the power line outside the FC6A Series MICROSmart. This is required when equipment containing the FC6A Series MICROSmart is destined for Europe.
- Use an IEC 60127-approved fuse on the output circuit. This is required when equipment containing the FC6A Series MICROSmart is destined for Europe.
- Use an EU-approved circuit breaker. This is required when equipment containing the FC6A Series MICROSmart is destined for Europe.
- Make sure of safety before starting and stopping the FC6A Series MICROSmart or when operating the FC6A Series MICROSmart to force outputs on or off. Incorrect operation of the FC6A Series MICROSmart may cause machine damage or accidents.
- Do not connect the ground wire directly to the FC6A Series MICROSmart. Connect a protective ground to the cabinet containing the FC6A Series MICROSmart using an M4 or larger screw. This is required when equipment containing the FC6A Series MICROSmart is destined for Europe.
- Do not disassemble, repair, or modify the FC6A Series MICROSmart modules.
- The FC6A Series MICROSmart contains electronic parts and batteries. When disposing of the FC6A Series MICROSmart, do so in accordance with national and local regulations.



ABOUT THIS MANUAL

Thank you for purchasing the FC6A Series MICROSmart manufactured by IDEC Corporation.
This document describes the FC6A Series MICROSmart system configuration, specifications, and installation methods, and it provides descriptions of the various functions.
Read this manual to ensure the correct understanding of the entire functions of the FC6A Series MICROSmart.
IDEC Corporation makes the latest product manual PDFs available on our website at no additional cost.
Please download the latest product manual PDFs from our website.
Product manual PDF download page (www.idec.com/FC6Amanuals)

This manual describes device allocations, specifications of basic and advanced instructions, and operation basics of the FC6A Series MICROSmart.

Chapter 1: Operation Basics

General information about setting up the basic FC6A Series MICROSmart system for programming, starting and stopping the FC6A Series MICROSmart operation, and simple operating procedures. Everything from creating a user program using WindLDR on a computer to monitoring the FC6A Series MICROSmart operation.

Chapter 2: Devices

Descriptions of the allocations of devices such as inputs, outputs, internal relays, registers, timers, and counters that are used in the basic and advanced instructions, as well as details about the allocations of special internal relays and special data registers.

Chapter 3: Instructions Reference

List of basic and advanced instructions to program the FC6A Series MICROSmart and general rules of using advanced instructions.

Chapter 4: Basic Instructions

Programming of the basic instructions, available devices, and sample programs.

Chapter 5 through Chapter 28:

Detailed descriptions for advanced instructions.

Appendix

Additional information about execution times and byte sizes for instructions.

Index

Alphabetical listing of key words.

Publication history

December 2015:	First Edition
December 2016:	Second Edition
March 2017:	Third Edition
August 2017:	Fourth Edition
March 2018:	Fifth Edition
November 2018:	Sixth Edition
May 2019:	Seventh Edition
December 2019:	Eighth Edition

Trademarks

FC6A Series MICROSmart is a trademark of IDEC Corporation.

Regarding laws and compatible standards

This product adheres to the laws and compatible standards of all countries involved, as shown below.

European laws and standards

This product complies with the following EU directives.

- Low Voltage Directive
- RoHS Directive
- RE Directive (FC6A-PC4 only)

To comply with these directives, this product has been designed and evaluated on the basis of the following international and European standard.

- IEC/EN 61131-2: 2007
- EN50581:2012
- EN301 489-1 V2.1.1& EN301 489-17 V2.1.1 (FC6A-PC4 only)

For details on the compatible standards and EU Directives, contact the distributor from which you purchased this product or visit our web site.

North America laws and standards

This product complies with the following standards.

- UL508*1
- UL61010-1*1
- UL61010-2-201*1
- CSA C22.2 No.142*1
- CSA C22.2 No.61010-1*1
- CSA C22.2 No.61010-2-201*1
- ANSI/ISA 12.12.01
- CAN/CSA C22.2 No.213

*1 Certain FC6A Series MICROSmart models are not compatible. For details about applicable standards, please contact IDEC Corporation.

Chinese laws and standards

The FC6A-PC4 complies with the following certification.

- SRRC

Marine standards

This product has been certified by the following classification societies.
(Applications have been submitted for certain models.)

- ABS (American Bureau of Shipping)
- DNV GL (Det Norske Veritas·Germanischer Lloyd)
- LR (Lloyd's Register)
- NK (Nippon Kaiji Kyokai)

* This product has not been certified for use on the bridge or deck.

For details on applicable standards and EU directives, please contact the dealer where purchased or check the IDEC website.

IMPORTANT INFORMATION

Under no circumstances shall IDEC Corporation be held liable or responsible for indirect or consequential damages resulting from the use of or the application of IDEC PLC components, individually or in combination with other equipment.

All persons using these components must be willing to accept responsibility for choosing the correct component to suit their application and for choosing an application appropriate for the component, individually or in combination with other equipment.

All diagrams and examples in this manual are for illustrative purposes only. In no way does including these diagrams and examples in this manual constitute a guarantee as to their suitability for any specific application. To test and approve all programs, prior to installation, is the responsibility of the end user.

About the Warranty of the Products

1. **Warranty Period**

   The Products are warranted for 3 years from the date of purchase, or from the date of delivery completion.

   * Consumable/maintenance parts such as batteries and relays if the operation exceeds 100,000 times are excluded from the 3-year warranty.

2. **Extent of Warranty**

   IDEC CORPORATION is responsible for failures or defects of the Products during the above warranty period, either a replacement part will be provided or the defective parts of the Products will be repaired free of charge. If such failure or defects should occur, please offer them to the distributor, dealer or IDEC CORPORATION with the materials in which the date of purchase is specified.

   * The expenses for installation and construction at the time of repair will not be borne.

3. **Start**

   May 1, 2017. The Products which were produced after June 1, 2014 and purchased in last three years will also be warranted.

4. **Indemnification**

   IDEC CORPORATION will not be liable under this Warranty and be indemnified and held harmless from any and all demands, suits, expenses, claims, damages and liabilities in the following event that:

   1) The Products are used or operated beyond the conditions or environment range as described in catalog, specifications or instruction; or
   2) The failure or defects of the Products arise from the cause other than the Products; or
   3) The Products are improved, modified or altered by the party other than IDEC; or
   4) The failure or defects and damages of the Products arise from the usage of the Product in the way that is not intended; or
   5) The failure or defects and damages of the Products arise from the cause beyond IDEC's control including, but not limited to, fire, earthquake, flood, lightning, other natural disasters, and acts of God; or
   6) The failure or defects and damages of the Products arise from the relocation, transportation or drop after you purchase the Products; or
   7) The failure or defects and damages of the Products arise from improper installation; or
   8) Maintenance and inspection are not carried out in accordance with instruction.

   IDEC CORPORATION DISCLAIMS ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR USE OR PURPOSE, AS WELL AS LIABILITY FOR INCIDENTAL, SPECIAL, INDIRECT, CONSEQUENTIAL OR OTHER DAMAGES RELATING TO THE PRODUCTS

5. **Extent of Service**

   The price of the Products will not include the fee for any service such as sending technicians and engineers, IDEC CORPORATION will charge you the fee for the following:

   1) Instruction for installment and visiting for test operation, including, but not limited to creating application software and operation tests; and
   2) Maintenance and inspection, arrangement and repair; and
   3) Technical assistance and technical education; and
   4) Product test and inspection based on you request.

RELATED MANUALS

The following manuals related to the FC6A Series MICROSmart are available. Refer to them in conjunction with this manual.

Type No.	Manual Name	Description
FC9Y-B1722	FC6A Series MICROSmart User's Manual	Describes product specifications, installation and wiring instructions, instructions for basic programming operations and special functions, device and instruction lists, and troubleshooting procedures for the FC6A Series MICROSmart.
FC9Y-B1726	FC6A Series MICROSmart Ladder Programming Manual (this manual)	Describes basic operations for programming with ladders on the FC6A Series MICROSmart, monitoring methods, device and instruction lists, and details of each instruction.
FC9Y-B1730	FC6A Series MICROSmart Communication Manual	Describes specifications related to FC6A Series MICROSmart communication, descriptions of functions, configuration methods, and usage examples.
FC9Y-B1734	FC6A Series MICROSmart PID Module User's Manual	Describes PID module specifications and functions.
WindLDR Help		Describes usage instructions for WindLDR, programming software for the FC6A Series MICROSmart.

NAMES AND ABBREVIATIONS USED IN THIS MANUAL

Model Names

Name Used in This Manual			Type Number, Part Code, or Official Name
FC6A Series MICROSmart			FC6A Series MICROSmart
CPU module	All-in-One CPU module		FC6A-C16R1AE, FC6A-C16R1CE, FC6A-C16K1CE, FC6A-C16P1CE, FC6A-C24R1AE, FC6A-C24R1CE, FC6A-C24K1CE, FC6A-C24P1CE, FC6A-C40R1AE, FC6A-C40R1CE, FC6A-C40K1CE, FC6A-C40P1CE, FC6A-C40R1DE, FC6A-C40K1DE, FC6A-C40P1DE
	CAN J1939 All-in-One CPU module		FC6A-C40R1AEJ, FC6A-C40R1CEJ, FC6A-C40K1CEJ, FC6A-C40P1CEJ, FC6A-C40R1DEJ, FC6A-C40K1DEJ, FC6A-C40P1DEJ
	Plus CPU module		FC6A-D16R1CEE, FC6A-D16P1CEE, FC6A-D16K1CEE, FC6A-D32P3CEE, FC6A-D32K3CEE
	16-I/O type		FC6A-C16R1AE, FC6A-C16R1CE, FC6A-C16K1CE, FC6A-C16P1CE
	24-I/O type		FC6A-C24R1AE, FC6A-C24R1CE, FC6A-C24K1CE, FC6A-C24P1CE
	40-I/O type		FC6A-C40R1AE, FC6A-C40R1CE, FC6A-C40K1CE, FC6A-C40P1CE, FC6A-C40R1DE, FC6A-C40R1DE, FC6A-C40K1DE, FC6A-C40P1DE, FC6A-C40R1AEJ, FC6A-C40R1CEJ, FC6A-C40K1CEJ, FC6A-C40P1CEJ, FC6A-C40R1DEJ, FC6A-C40K1DEJ, FC6A-C40P1DEJ
	Plus 16-I/O type		FC6A-D16R1CEE, FC6A-D16P1CEE, FC6A-D16K1CEE
	Plus 32-I/O type		FC6A-D32P3CEE, FC6A-D32K3CEE
	AC power type		FC6A-C16R1AE, FC6A-C24R1AE, FC6A-C40R1AE, FC6A-C40R1AEJ
	DC power type	24V DC power type	FC6A-C16R1CE, FC6A-C24R1CE, FC6A-C40R1CE, FC6A-C16K1CE, FC6A-C24K1CE, FC6A-C40K1CE, FC6A-C16P1CE, FC6A-C24P1CE, FC6A-C40P1CE, FC6A-C40R1CEJ, FC6A-C40K1CEJ, FC6A-C40P1CEJ, FC6A-D16R1CEE, FC6A-D16P1CEE, FC6A-D16K1CEE, FC6A-D32P3CEE, FC6A-D32K3CEE
		12V DC power type	FC6A-C40R1DE, FC6A-C40K1DE, FC6A-C40P1DE, FC6A-C40R1DEJ, FC6A-C40K1DEJ, FC6A-C40P1DEJ
	Relay output type		FC6A-C16R1AE, FC6A-C16R1CE, FC6A-C24R1AE, FC6A-C24R1CE, FC6A-C40R1AE, FC6A-C40R1CE, FC6A-C40R1DE, FC6A-C40R1AEJ, FC6A-C40R1CEJ, FC6A-C40R1DEJ, FC6A-D16R1CEE
	Transistor output type	Transistor sink output type	FC6A-C16K1CE, FC6A-C24K1CE, FC6A-C40K1CE, FC6A-C40K1DE, FC6A-C40K1CEJ, FC6A-C40K1DEJ, FC6A-D16K1CEE, FC6A-D32K3CEE
		Transistor protection source output type	FC6A-C16P1CE, FC6A-C24P1CE, FC6A-C40P1CE, FC6A-C40P1DE, FC6A-C40P1CEJ, FC6A-C40P1DEJ, FC6A-D16P1CEE, FC6A-D32P3CEE
Expansion module	I/O module	Digital I/O module	Digital input module, digital output module, digital mixed I/O module
		Analog I/O module	Analog input module, analog output module, analog mixed I/O module
	Communication module		Serial communication module
			PID module
Expansion interface module			Expander, remote master, remote slave
Cartridge	I/O cartridge	Digital I/O cartridge	Digital input cartridge, digital output cartridge
		Analog I/O cartridge	Analog input cartridge, analog output cartridge
	Communication cartridge		RS232C communication cartridge, RS485 communication cartridge, Bluetooth communication cartridge
WindLDR			WindLDR application software
USB cable			USB maintenance cable (HG9Z-XCM42), USB Mini-B extension cable (HG9Z-XCE21)

Name Used in this Manual	WindLDR Operating Procedure
Function area settings	**Configuration** tab > **Function Area Settings** group
Monitors	Select **Online** > **Monitor** > **Start Monitor**.
PLC status	Select **Online** > **PLC** > **Status**.
Communication settings	Select **Online** > **Communication** > **Set Up**.
Modbus master request table	On the **Configuration** tab, in **Function Area Settings**, click **Communication Ports**, and in the displayed **Function Area Settings** dialog box, for **Communication Mode** under **Communication Ports**, select **Modbus RTU Master** or **Modbus TCP Client**
Application button	The button displayed on the left side of the menu bar. Click to display the menu with **New**, **Save**, and **Save As**, recent projects, **WindLDR Options**, and **Exit WindLDR**.

TABLE OF CONTENTS

CHAPTER 5: Move Instructions

CHAPTER 6: Data Comparison Instructions

CHAPTER 7: Binary Arithmetic Instructions

CHAPTER 8: Boolean Computation Instructions

CHAPTER 9: Shift / Rotate Instructions

CHAPTER 10: Data Conversion Instructions

1: Operation Basics

Introduction

The following version of WindLDR is required for FC6A Series MICROSmart programming and maintenance:

- When you use All-in-One CPU module/CAN J1939 All-in-One CPU module: WindLDR version 8.0.0 or later
- When you use Plus CPU module: WindLDR version 8.6.0 or later

For details on how to check the version, see "Checking the WindLDR Version Number" on page 1-16.

This chapter describes basic procedures for operating WindLDR, programming and maintenance software for the FC6A Series MICROSmart.

Start WindLDR

■ Windows 10

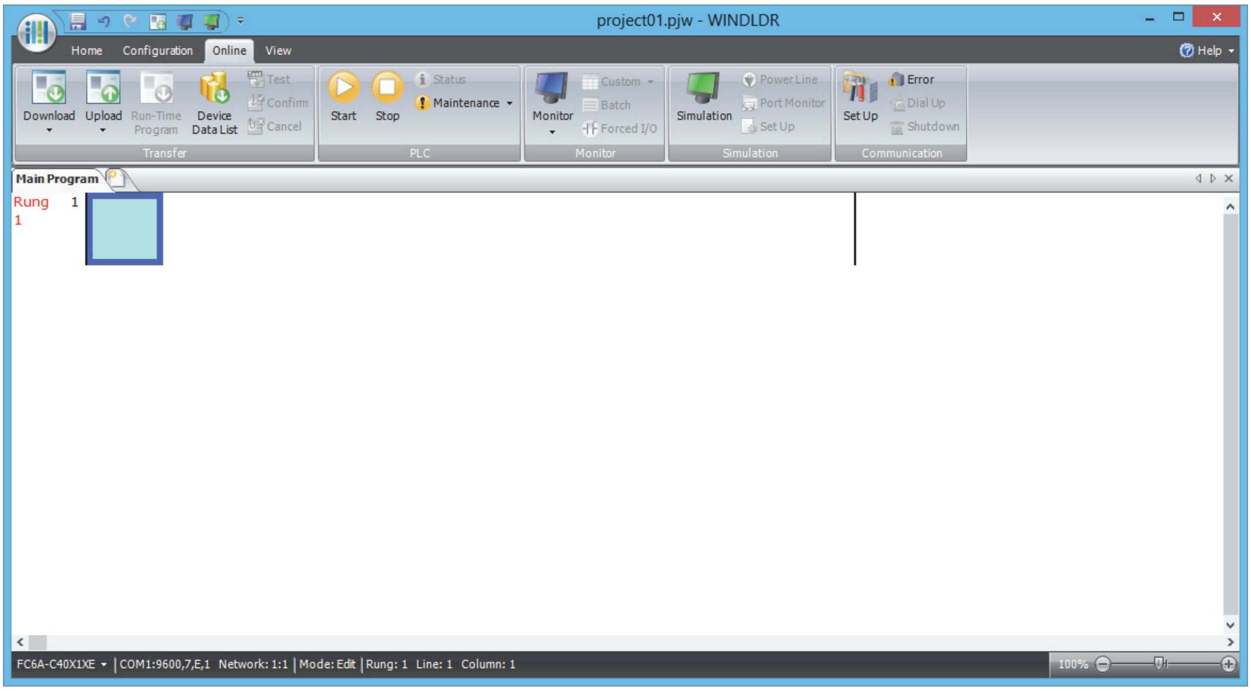
Click **Start** button and then > **All Apps** > **IDEC Automation Organizer** > **WindLDR**.

■ Windows 8

Click **WindLDR** in the tiles on the Start screen.

■ Windows 7

Click **Start** and then **Programs** > **IDEC Automation Organizer** > **WindLDR**.

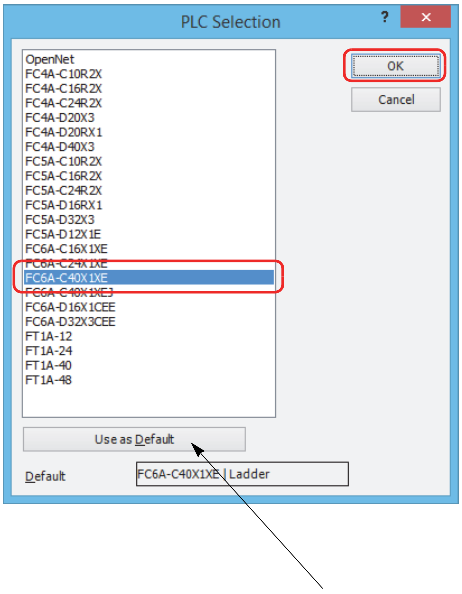


FC6A-C40X1XE | COM1:9600,7,E,1 Network: 1:1 | Mode: Edit | Rung: 1 Line: 1 Column: 1 100%

PLC Selection

Before programming a user program on WindLDR, select a PLC type.

1. Select **Configuration** from the WindLDR menu bar, then select **PLC Type**.
   The **PLC Selection** dialog box appears.
2. Select a PLC type in the selection box and the programming language to use.
3. Click **OK**.



Default FC6A-C40X1XE | Ladder

Select the PLC type from the list and click **Use as Default** to set the selected PLC type as the default type when WindLDR starts.

In WindLDR, the FC6A Series MICROSmart is categorized by the number of inputs and outputs and the PLC type names are displayed as follows.

PLC Selection Option	Type No.
FC6A-C16X1XE	FC6A-C16R1AE FC6A-C16R1CE FC6A-C16P1CE FC6A-C16K1CE
FC6A-C24X1XE	FC6A-C24R1AE FC6A-C24R1CE FC6A-C24P1CE FC6A-C24K1CE
FC6A-C40X1XE	FC6A-C40R1AE FC6A-C40R1CE FC6A-C40P1CE FC6A-C40K1CE FC6A-C40R1DE FC6A-C40P1DE FC6A-C40K1DE
FC6A-C40X1XEJ	FC6A-C40R1AEJ FC6A-C40R1CEJ FC6A-C40P1CEJ FC6A-C40K1CEJ FC6A-C40R1DEJ FC6A-C40P1DEJ FC6A-C40K1DEJ
FC6A-D16X1CEE	FC6A-D16R1CEE FC6A-D16P1CEE FC6A-D16K1CEE
FC6A-D32X3CEE	FC6A-D32P3CEE FC6A-D32K3CEE

PLC selection is now complete. Next, create a ladder program.

Create Program

Create Ladder Program

This section describes the operating procedure to create a ladder program in WindLDR.

Note: For details about devices, see "Devices" on page 2-1.

Sample User Program

Create a simple program using WindLDR. The sample program performs the following operation:

When only input I0 is turned on, output Q0 is turned on.
When only input I1 is turned on, output Q1 is turned on.
When both inputs I0 and I1 are turned on, output Q2 flashes in 1-s increments.

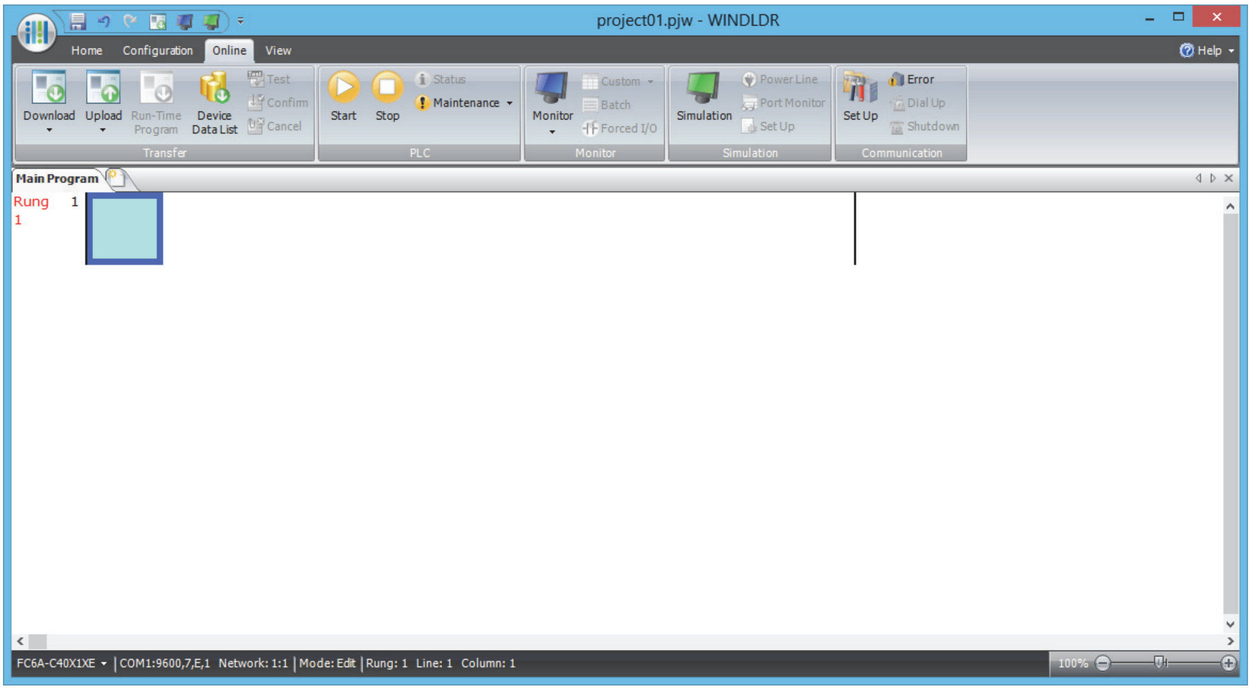
Rung No.	Input I0	Input I1	Output Operation
1	ON	OFF	Output Q0 is turned ON.
2	OFF	ON	Output Q1 is turned ON.
3	ON	ON	Output Q2 flashes in 1-s increments.

Note: One collection of a group of instructions that control output or advanced instructions is called a rung. WindLDR manages programs in rung units. Function descriptions can be configured as rung comments for individual rungs.

Start WindLDR

From the **Start** menu of Windows, select **Programs** > **Automation Organizer V2** > **WindLDR** > **WindLDR**.

WindLDR starts and a blank ladder editing screen appears with menus and tool bars shown on top of the screen.

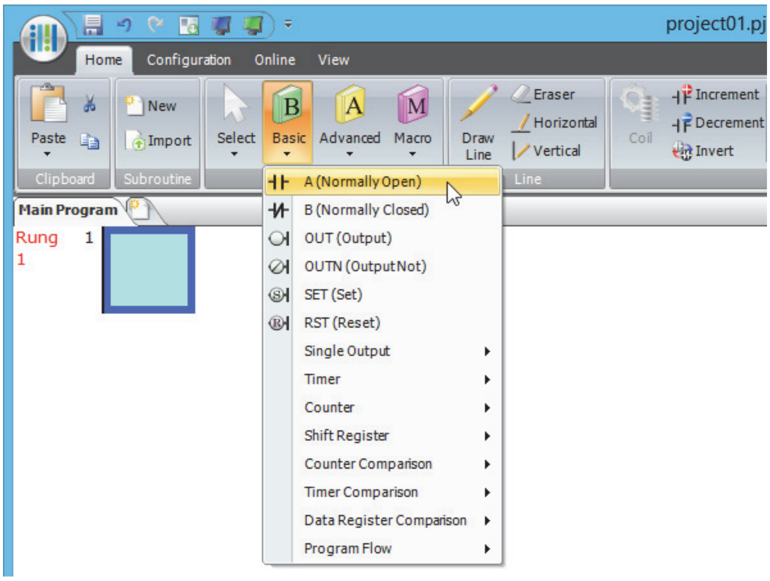


FC6A-C40X1XE | COM1:9600,7,E,1 Network: 1:1 | Mode: Edit | Rung: 1 Line: 1 Column: 1

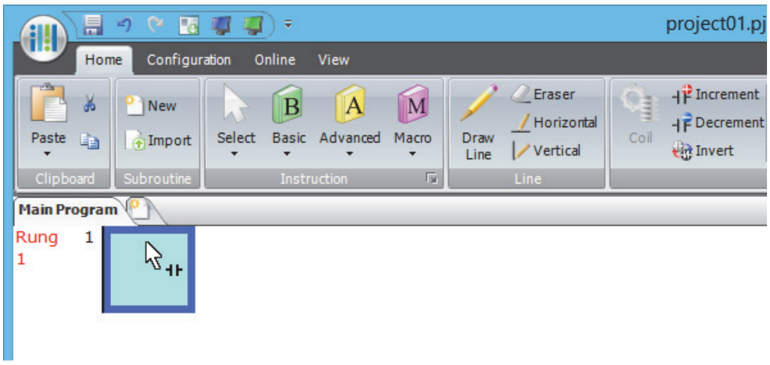
Edit User Program Rung by Rung

Start the user program with the LOD instruction by inserting a NO contact of input I0.

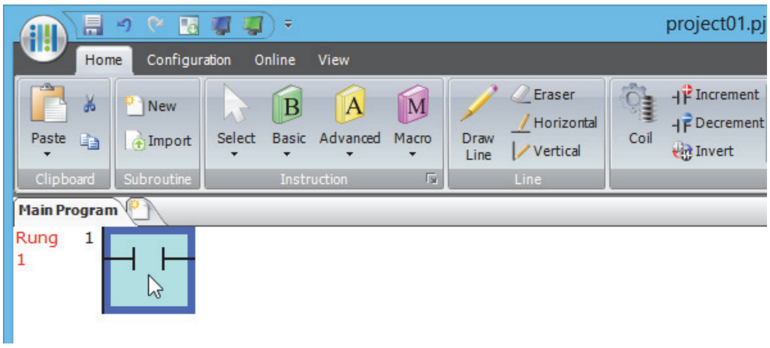
1. From the WindLDR menu bar, select **Home** > **Basic** > **A (Normally Open)**.



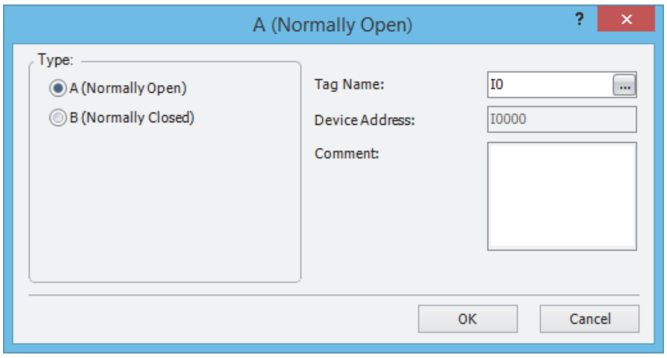
2. Move the mouse pointer to the first column of the first line where you want to insert a NO contact, and click the left mouse button.



3. Double-click **Normally Open**.
The **A (Normally Open)** dialog box is displayed.



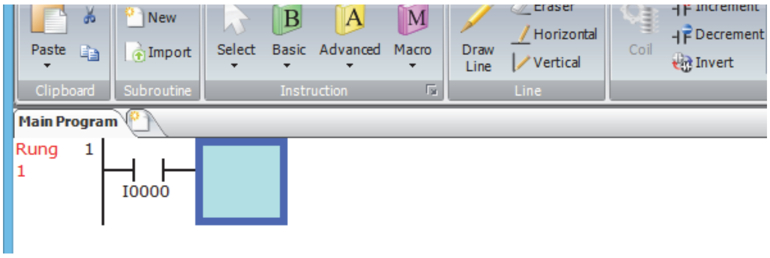
4. Enter **I0** in the **Tag Name** field, and click **OK**.



A (Normally Open)

Type:
● A (Normally Open)
○ B (Normally Closed)

Tag Name: I0
Device Address: I0000
Comment:



Notes:

- To enter an NO contact from the right-click menu, right-click at the location to insert the NO contact, and on the right-click menu, click **Basic Instructions (B)**, then **A (Normally Open)**.
- To enter an NO contact from the keyboard, press A (A) to display the **Coil Selection** dialog box. Select **A (Normally Open)**, and then press Enter (Enter).

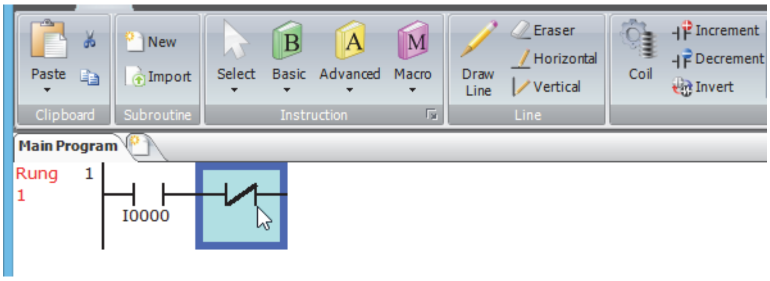
A NO contact of input I0 is programmed in the first column of the first ladder line.

Next, program the ANDN instruction by inserting a NC contact of input I1.

5. From the WindLDR menu bar, select **Home** > **Basic** > **B (Normally Closed)**.

6. Move the mouse pointer to the second column of the first ladder line where you want to insert a NC contact, and click the left mouse button.

7. Double-click **Normally Closed**.
   The **B (Normally Closed)** dialog box is displayed.



8. Enter **I1** in the Tag Name field, and click **OK**.

An NC contact of input I1 is programmed in the second column of the first ladder line.

At the end of the first ladder line, program the OUT instruction by inserting a NO coil of output Q0.

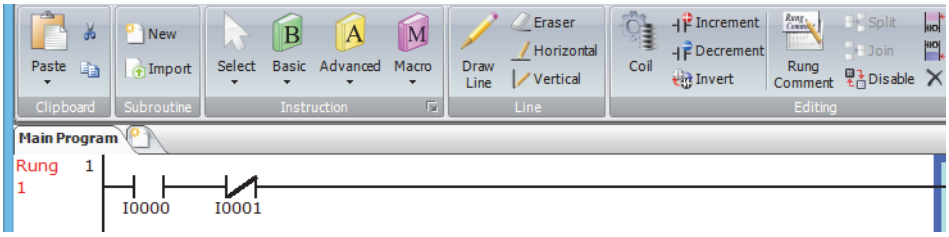
9. From the WindLDR menu bar, select **Home** > **Basic** > **OUT (Output)**.

10. Move the mouse pointer to the third column of the first ladder line where you want to insert an output coil, and click the left mouse button.

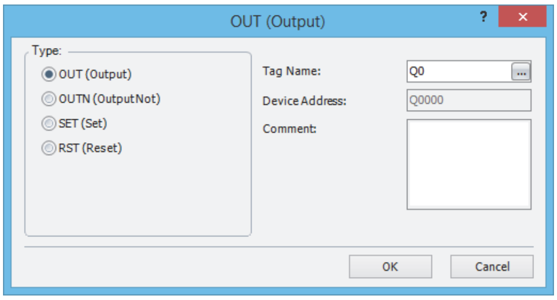
Note: Another method to insert an instruction (either basic or advanced) is to type the instruction symbol, OUT, where you want to insert the instruction.

11. Double-click **Output**.

The **Out (Output)** dialog box is displayed.



12. Enter **Q0** in the Tag Name field, and click **OK**.


OUT (Output)

Type:
- ● OUT (Output)
- ○ OUTN (OutputNot)
- ○ SET (Set)
- ○ RST (Reset)

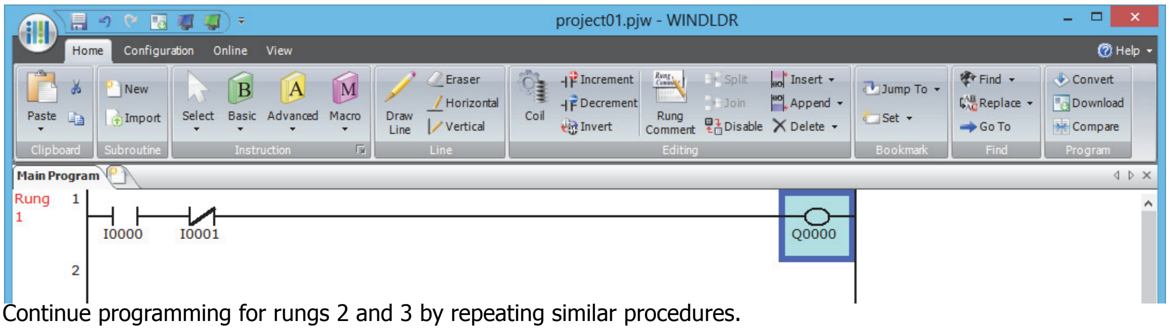
Tag Name: Q0

Device Address: Q0000

Comment:

OK Cancel

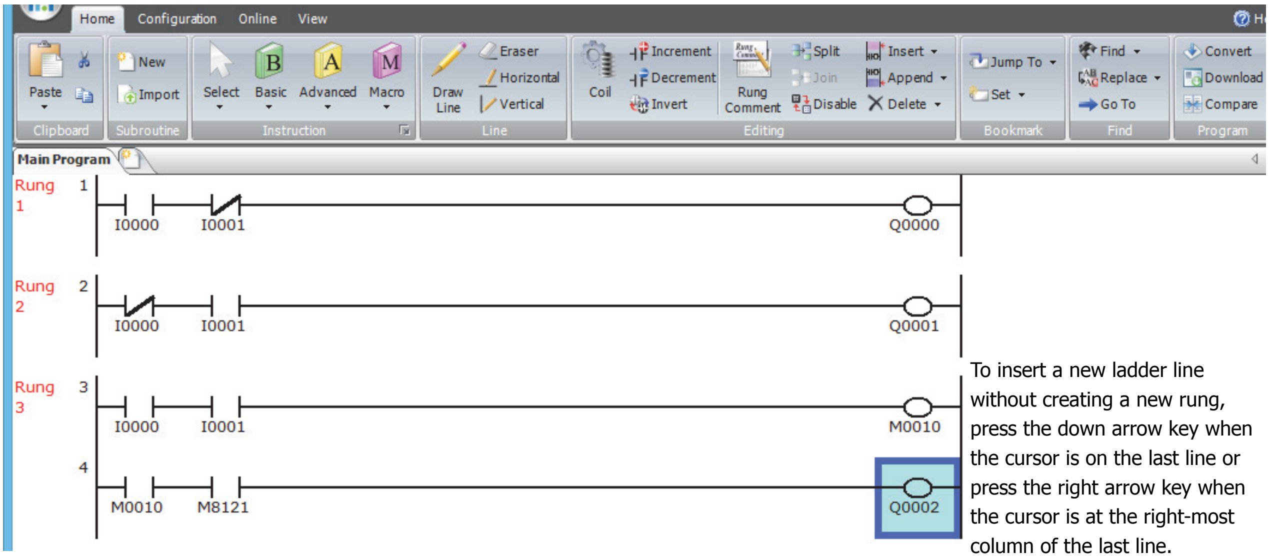
A NO output coil of output Q0 is programmed in the right-most column of the first ladder line. This completes programming for rung 1.



Continue programming for rungs 2 and 3 by repeating similar procedures.

A new rung is inserted by pressing the **Enter** key while the cursor is on the preceding rung. A new rung can also be inserted by selecting **Home** > **Append** > **Append a Rung**.

When completed, the ladder program looks like below.



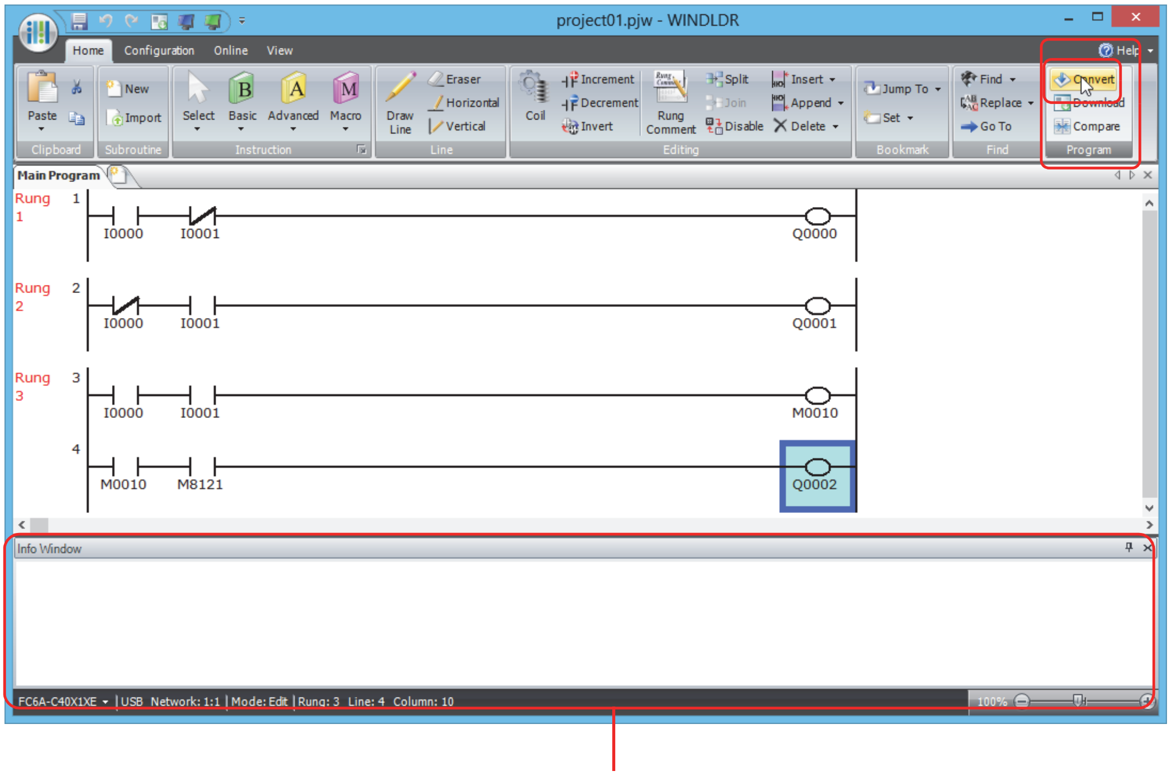
To insert a new ladder line without creating a new rung, press the down arrow key when the cursor is on the last line or press the right arrow key when the cursor is at the right-most column of the last line.

Convert Program

The program can be checked whether it contains any user program syntax error.

From the menu bar, select **Home** > **Convert** (**Program** group).

When the instruction/FB symbols are connected correctly, the program conversion is completed successfully. If any error is found, the errors are listed on the Info Window. Then, make corrections as necessary.



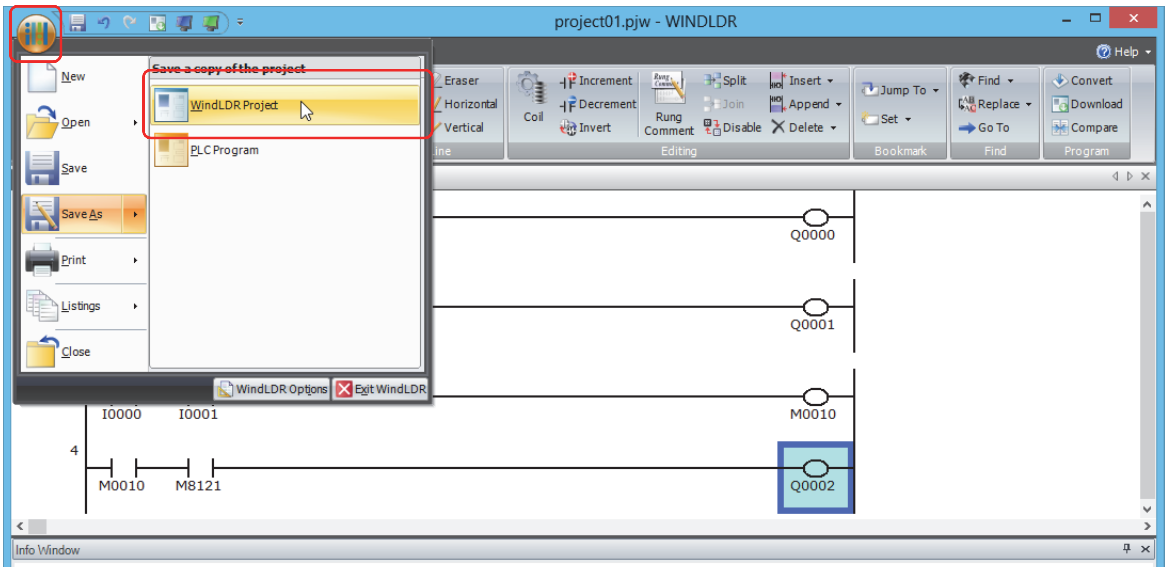
Info Window

Save Project

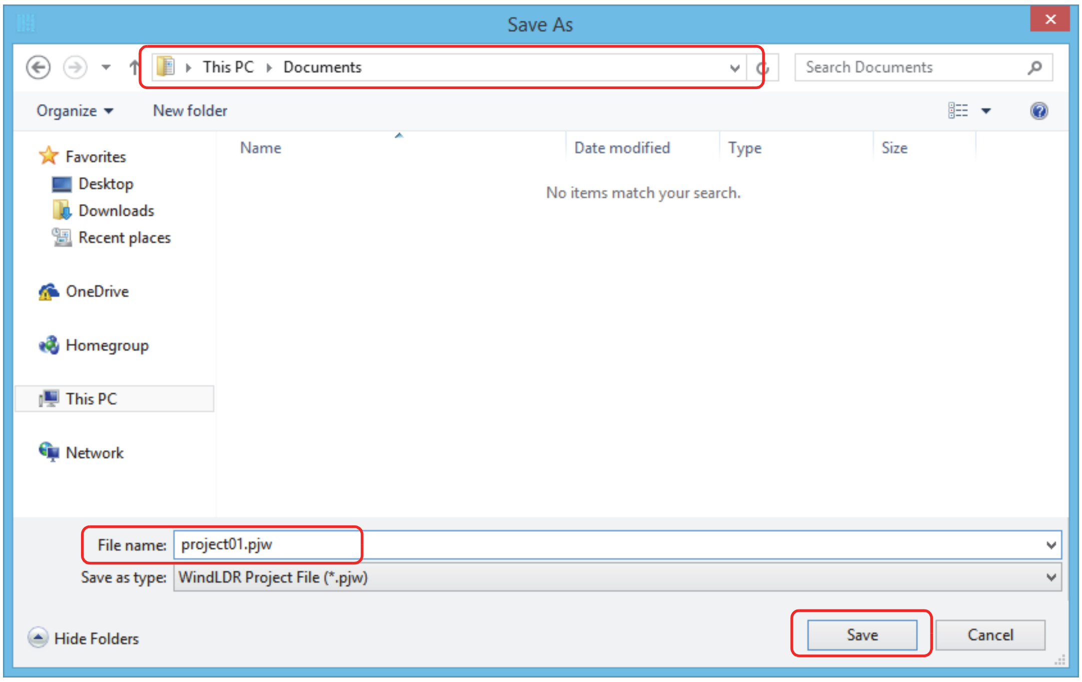
This section describes the operating procedure to save the created ladder program as a project file.

1. Save the current project with a new name.

   Click (application) button > **Save As** > **WindLDR Project**.



2. Enter the file name in **File name**, specify the folder to save to, and click **Save**.



File name: project01.pjw

Save as type: WindLDR Project File (*.pjw)

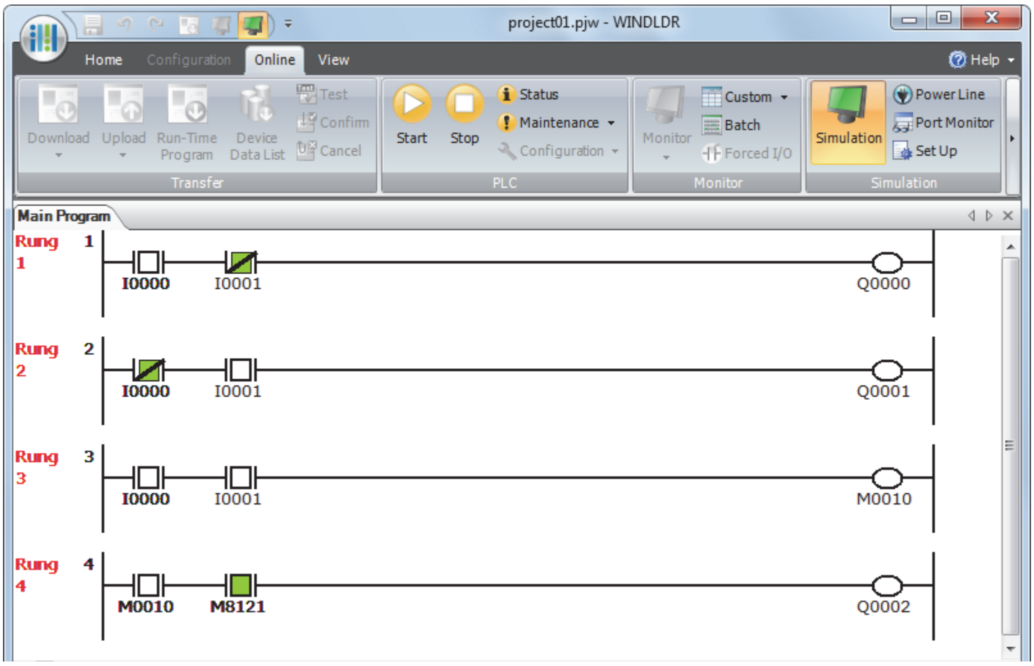
This completes the procedure to save a project to a file.

Note: The saved file is called a project file. The extension is ".pjw".

Simulate Operation

This section describes the operating procedure to check the operation of the user program before transferring it to the FC6A Series MICROSmart.

From the WindLDR menu bar, select **Online** > **Simulation**. The Simulation screen appears.



Select and right-click the input contact you want to change, and on the right-click menu, click **Set** or **Reset**.

To quit simulation, from the WindLDR menu bar, select **Online** > **Simulation**.

Notes:

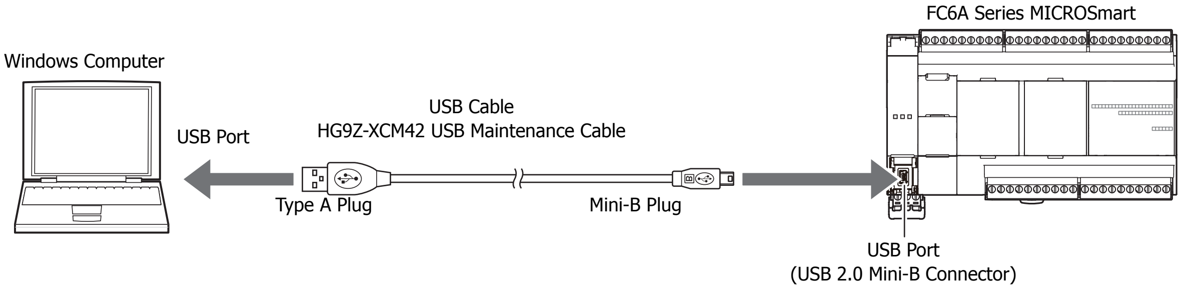
- You can also change the status of an input contact by double-clicking it.
- To quit simulation, from the WindLDR menu bar, once more select **Online** > **Simulation** > **Simulation**.

Download Program

While WindLDR is running on a PLC, you can download the user program to the FC6A Series MICROSmart.

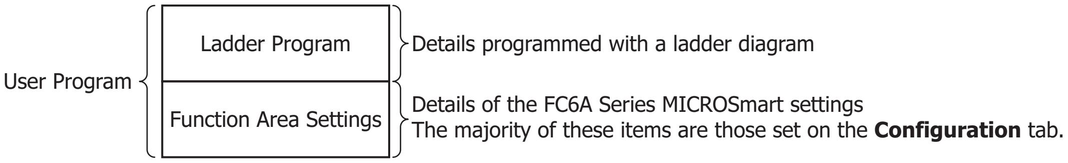
User programs can be downloaded to the FC6A Series MICROSmart from WindLDR using USB or Ethernet.

This section describes the operating procedure from configuring communication settings to downloading the user program using a USB connection as an example.



Notes:

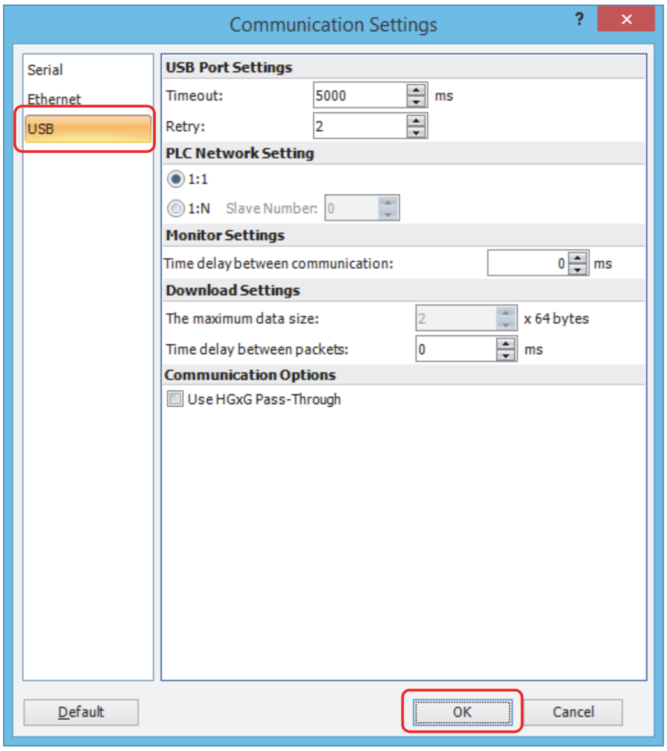
- In order for WindLDR to communicate with the FC6A Series MICROSmart via USB, a dedicated USB driver must be installed on the computer. See Appendix "USB Driver Installation Procedure" in the "FC6A Series MICROSmart User's Manual".
- A user program is a combination of a ladder program and the setting details (Function Area Settings).


User Program

Ladder Program: Details programmed with a ladder diagram

Function Area Settings: Details of the FC6A Series MICROSmart settings
The majority of these items are those set on the **Configuration** tab.

1. From the WindLDR menu bar, select **Online** > **Set Up**.
   The **Communication Settings** dialog box is displayed.
2. Click the **USB** tab and then click **OK**.


Communication Settings

Serial
Ethernet
USB

USB Port Settings

Timeout: 5000 ms

Retry: 2

PLC Network Setting

● 1:1

○ 1:N Slave Number: 0

Monitor Settings

Time delay between communication: 0 ms

Download Settings

The maximum data size: 2 x 64 bytes

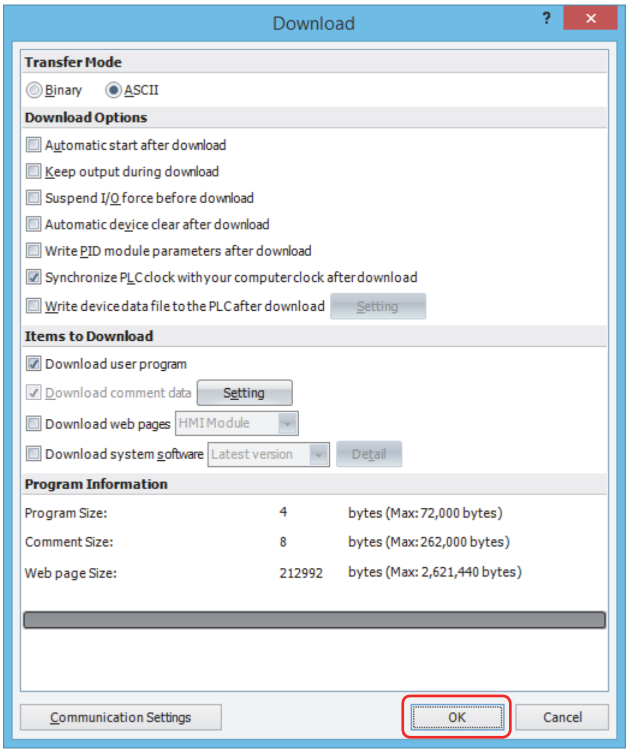
Time delay between packets: 0 ms

Communication Options

□ Use HGxG Pass-Through

The communication method is now set to USB. Next, download a user program.

3. From the WindLDR menu bar, select **Online** > **Download**. The **Download** dialog box appears, then click **OK**. The user program is downloaded to the FC6A Series MICROSmart.



Download

Transfer Mode

○ Binary ● ASCII

Download Options

□ Automatic start after download
□ Keep output during download
□ Suspend I/O force before download
□ Automatic device clear after download
□ Write PID module parameters after download
☑ Synchronize PLC clock with your computer clock after download
□ Write device data file to the PLC after download [Setting]

Items to Download

☑ Download user program
☑ Download comment data [Setting]
□ Download web pages [HMI Module]
□ Download system software [Latest version] [Detail]

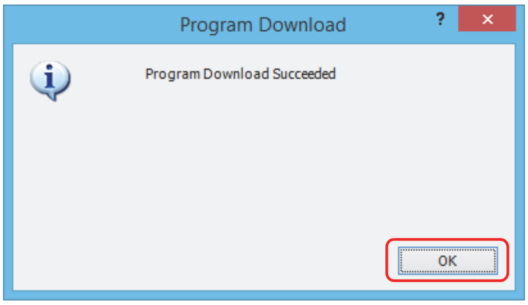
Program Information

Program Size:	4	bytes (Max: 72,000 bytes)
Comment Size:	8	bytes (Max: 262,000 bytes)
Web page Size:	212992	bytes (Max: 2,621,440 bytes)

Note: The Download dialog box can also be opened by selecting **Home** > **Download**.

Note: When downloading a user program, all values and selections in the **Function Area Settings** are also downloaded to the FC6A Series MICROSmart. For details on function settings, see Chapter 5 "Functions and Settings" in the "FC6A Series MICROSmart User’s Manual".

When the following message appears, the download has completed successfully. Click **OK**.



Program Download

Program Download Succeeded

Notes:

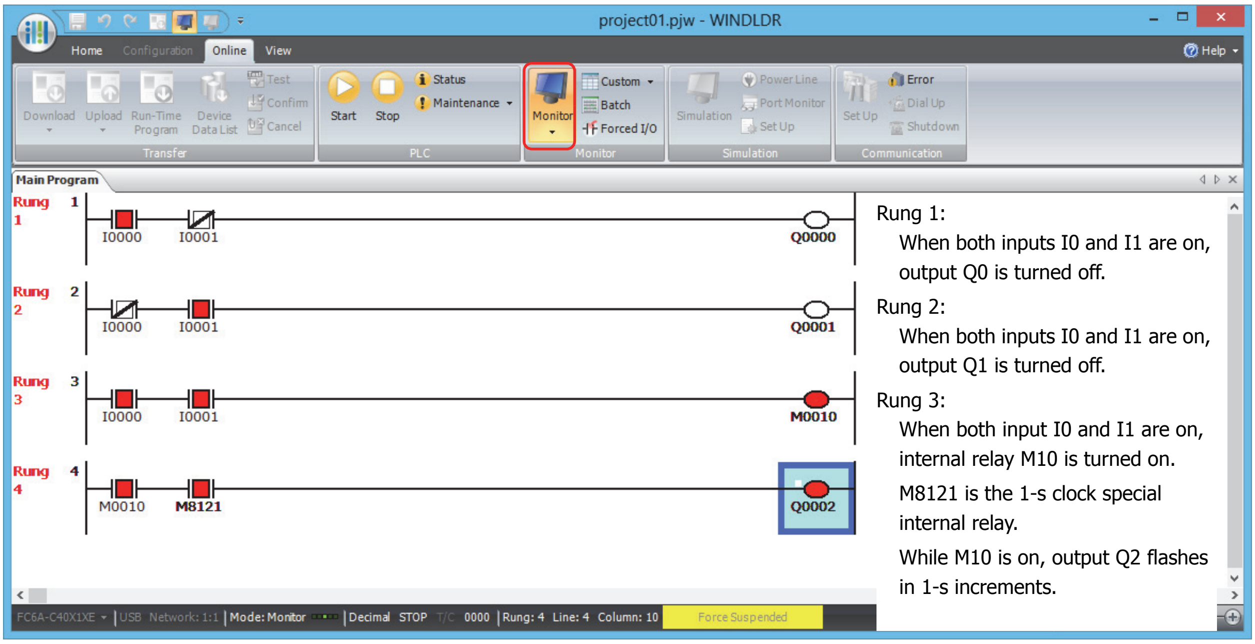
- The PLC will not run if the function switch is 0 when **Run/Stop PLC by Function Switch** is enabled in WindLDR and a program is downloaded with **Automatic start after download** enabled.
  To run the PLC, the function switch must be set to 1.
- Enabled is the default setting for **Run/Stop PLC by Function Switch** in WindLDR.

Monitor Operation

Another powerful function of WindLDR is to monitor the PLC operation on the computer. The input and output statuses of the sample program can be monitored in the ladder diagram.

From the WindLDR menu bar, select **Online** > **Monitor** > **Monitor**.

When both inputs I0 and I1 are on, the ladder diagram on the monitor screen looks as follows:



Rung 1:
When both inputs I0 and I1 are on, output Q0 is turned off.

Rung 2:
When both inputs I0 and I1 are on, output Q1 is turned off.

Rung 3:
When both input I0 and I1 are on, internal relay M10 is turned on.

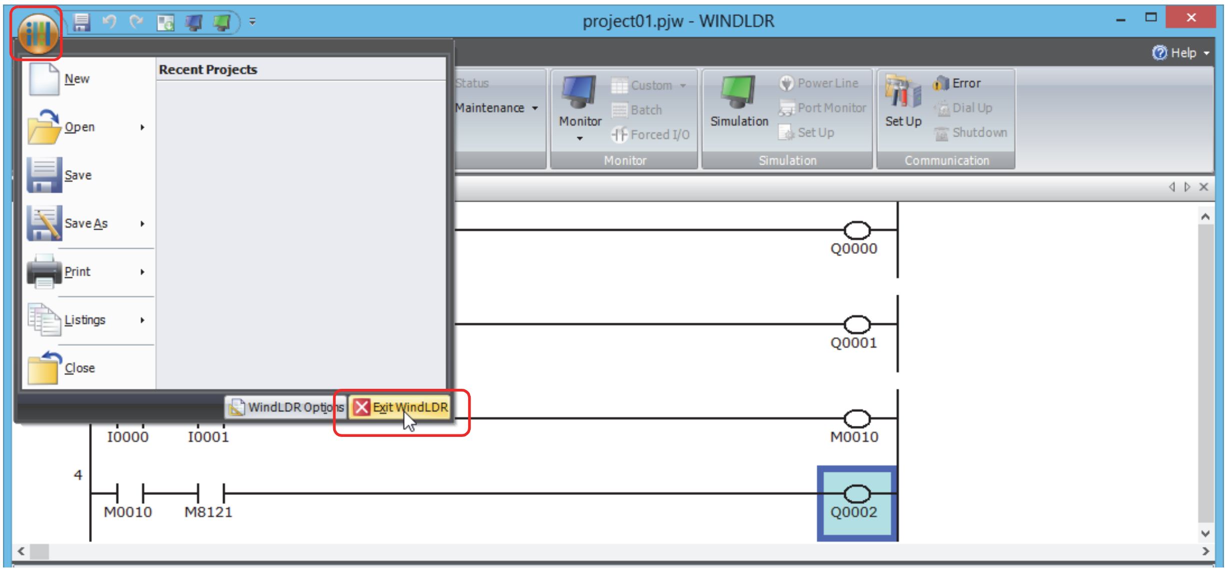
M8121 is the 1-s clock special internal relay.

While M10 is on, output Q2 flashes in 1-s increments.

Exit WindLDR

This section describes the operating procedure to exit WindLDR.

When you have completed monitoring, you can exit WindLDR either directly from the monitor screen or from the editing screen. In both cases, from the WindLDR application button, click **Exit WindLDR**.



Windows Displayed in the Workspace

This section describes how to change the position and display method of windows.

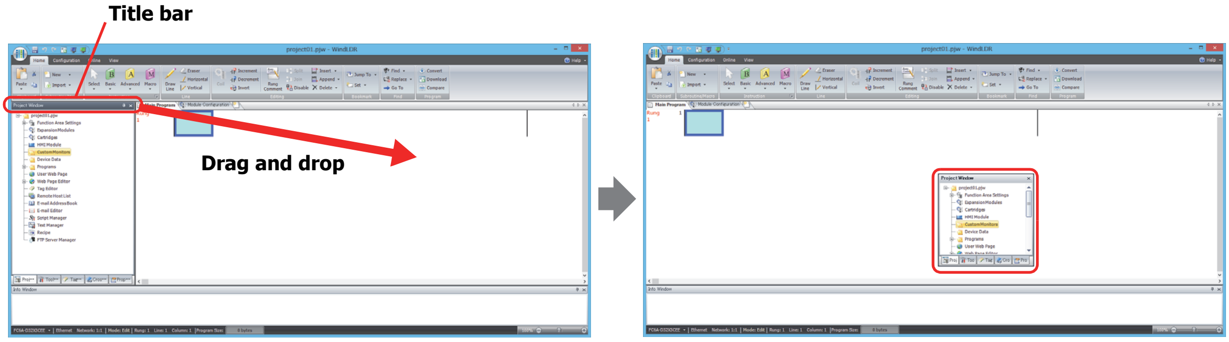
Changing the position of windows

You can change the display position of the window by dragging and dropping the title bar of the window or its tab to disable docking. Windows that are not docked are called floating windows.

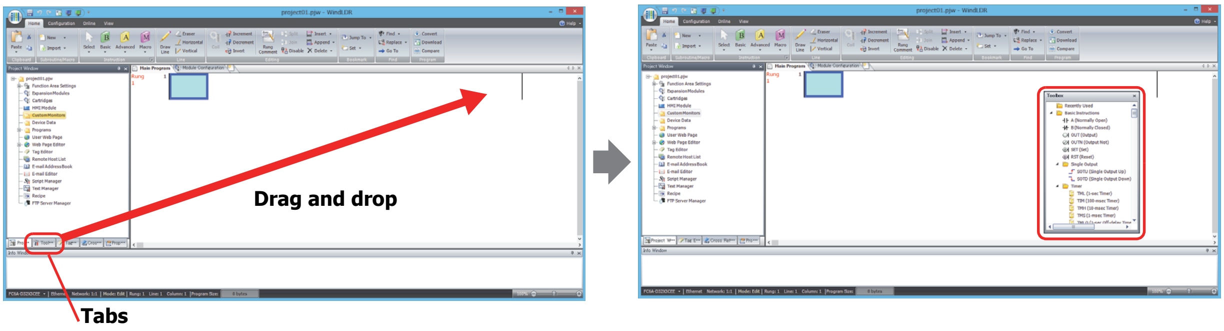
Disabling docking

This section describes how to disable the docking of a title bar or tab of a window.

- If you drag the title bar of the workspace window, you can move all the docked windows together.



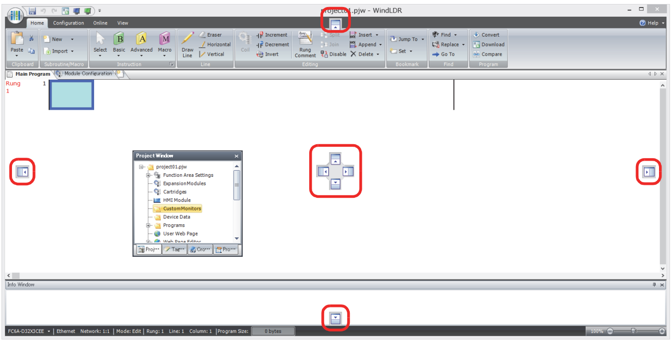
- If you drag the tab of a workspace window, you can move just the selected window.



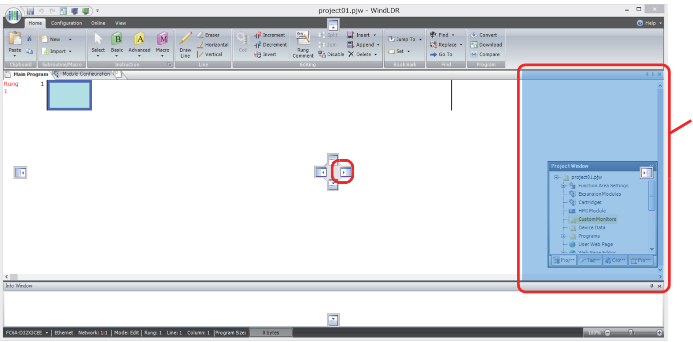
Docking windows

This section describes how to dock a floating window to another window or the frame at the top, bottom, left, and right sides of WindLDR.

1. Drag the window's title bar or tab.
The (Docking) icon is displayed.

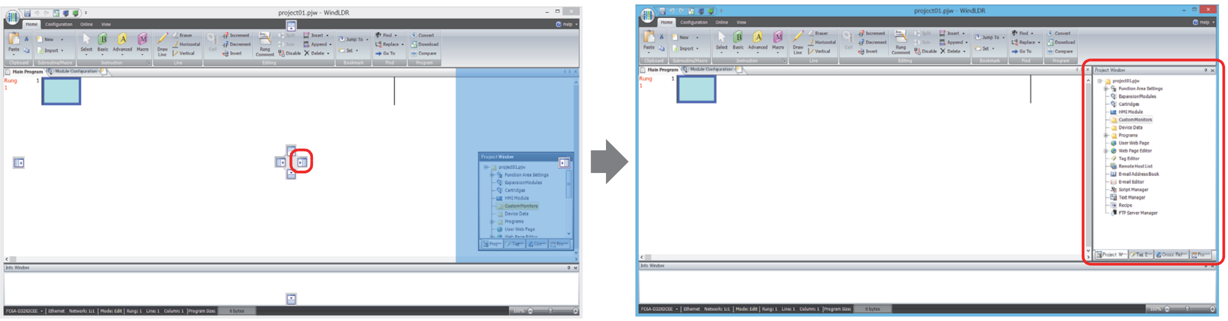


Note: When the mouse cursor gets close to a ▣ (Docking) icon while dragging the title bar or tab, the location to dock the window is displayed.

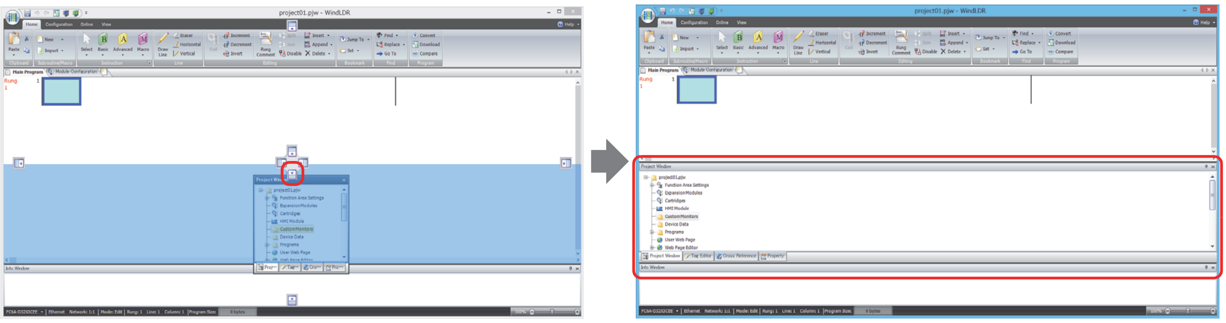


2. Drop the title bar or tab on the ▣ (Docking) icon to dock that window to WinidLDR's left, right, top, or bottom frame or a separate window.

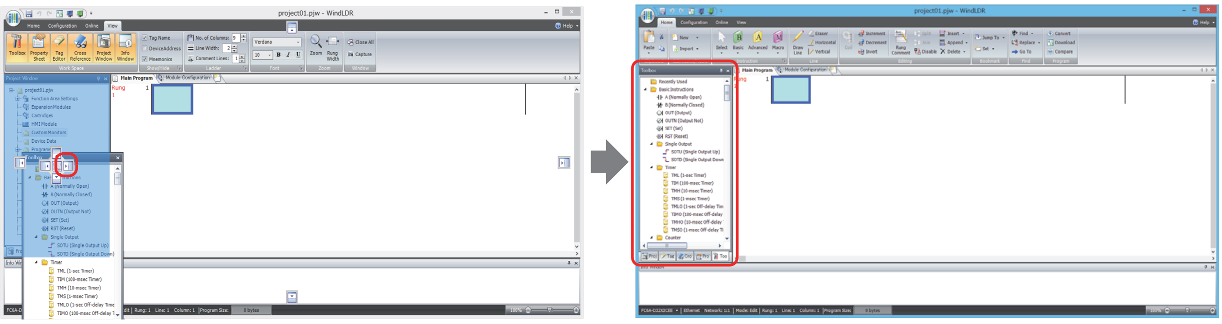
- If the workspace window is dropped on the ▣ (Docking) icon, it is docked to WindLDR's left, right, top, or bottom frame.



- If a floating window is dropped on the ▣ (Docking) icon, it is docked to WindLDR's left, right, top, or bottom frame or a docked window.



- If you put the mouse cursor on another window while dragging a floating windows title bar, the ▣ (Docking) icon is displayed. Drop the title bar on the ▣ (Docking) icon to dock the floating window to that window. Change the displayed window with the tabs.



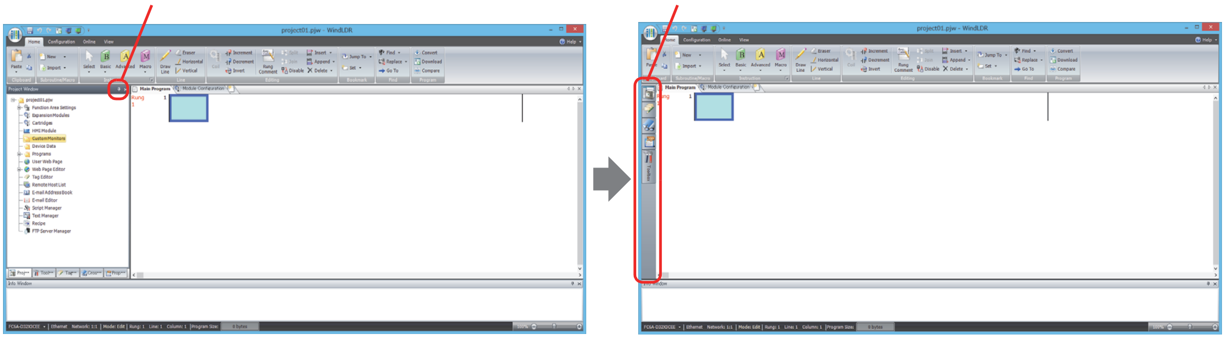
Changing the display method of windows

If the workspace window is docked, you can change the widow to automatically hide and show only its tabs.

- Click the (Auto Hide) icon to change the window to show only its tabs.

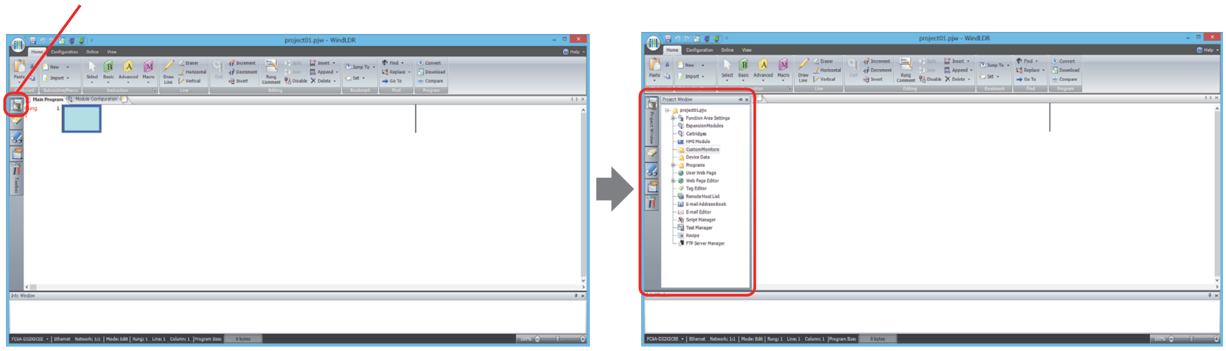
Auto Hide icon

Tabs



- The window is displayed when you bring the mouse cursor close to the tabs.

Mouse cursor



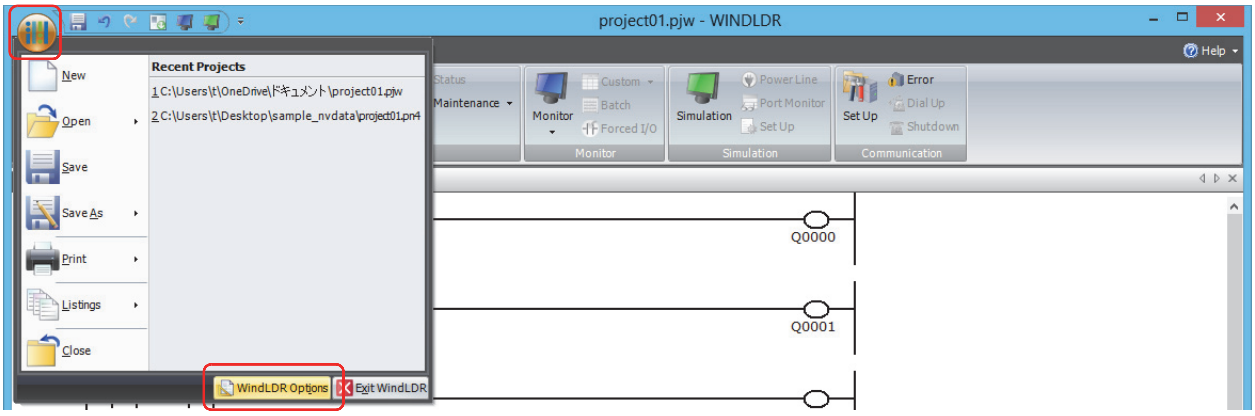
Notes:

- Click (Auto Hide) to secure the window in place.
- Click (Close) to close the window.

Checking the WindLDR Version Number

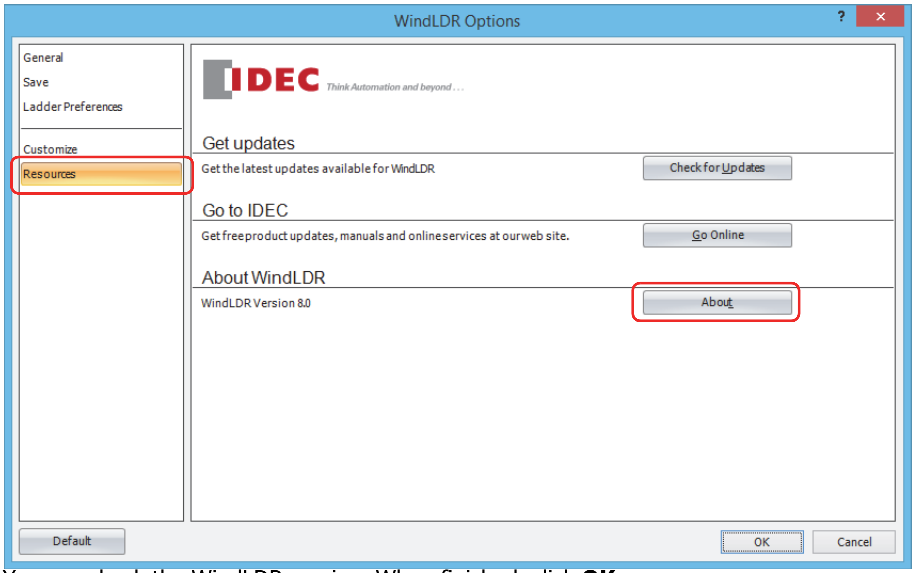
This section describes how to check the WindLDR version.

1. Click (application) button > **WindLDR Options**.
   The **WindLDR Options** dialog box appears.



1 C:\Users\t\OneDrive\ドキュメント\project01.pjw
2 C:\Users\t\Desktop\sample_nvdata\project01.pn4

2. Click the **Resources** tab, and then click **About**.
   The **About WindLDR** dialog box appears.



Get updates

Get the latest updates available for WindLDR

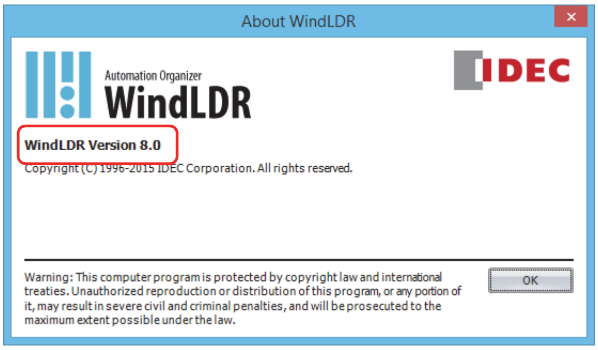
Go to IDEC

Get free product updates, manuals and online services at our web site.

About WindLDR

WindLDR Version 8.0

You can check the WindLDR version. When finished, click **OK**.



WindLDR Version 8.0

Copyright (C) 1996-2015 IDEC Corporation. All rights reserved.

Warning: This computer program is protected by copyright law and international treaties. Unauthorized reproduction or distribution of this program, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under the law.

Note: When using the FC6A Series MICROSmart, use the following version of WindLDR:

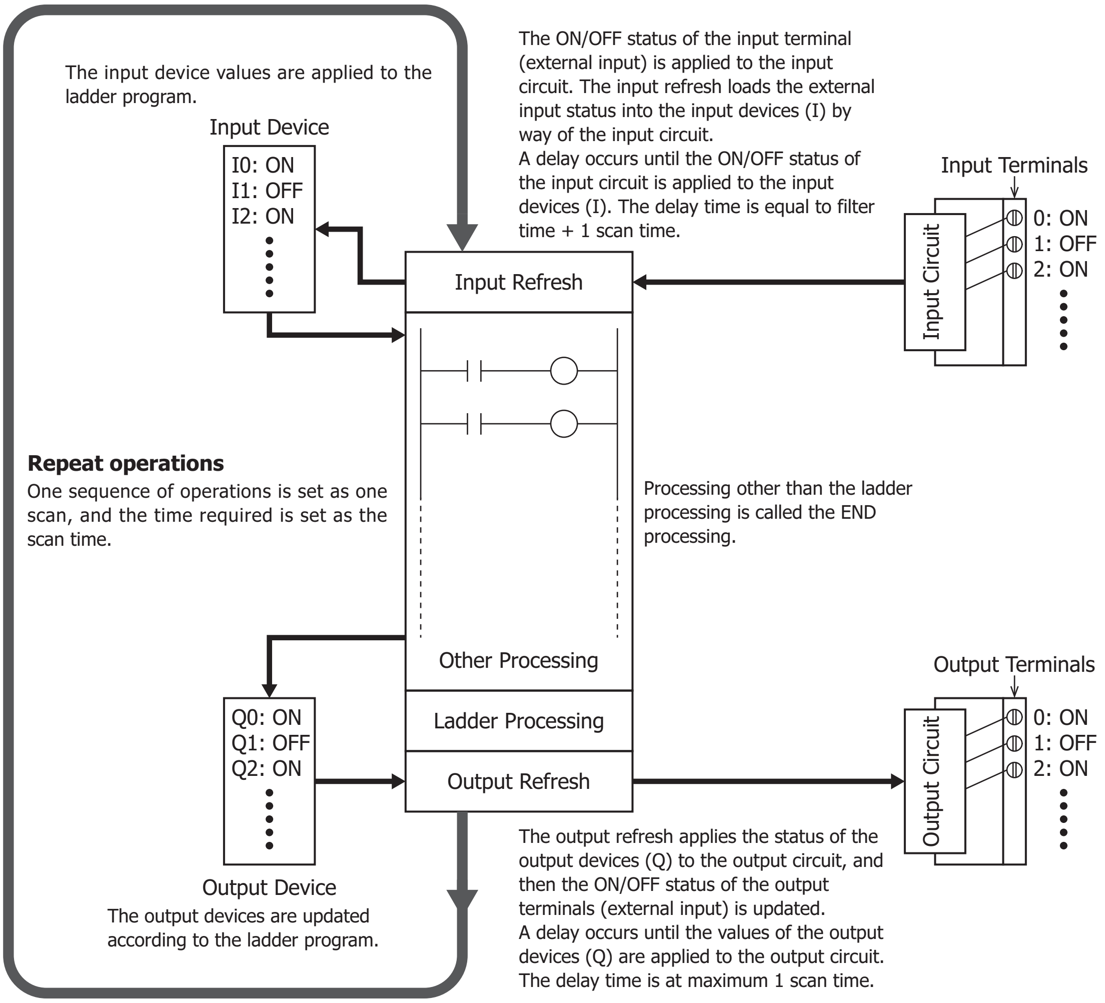
- When you use All-in-One CPU module/CAN J1939 All-in-One CPU module: WindLDR version 8.0.0 or later
- When you use Plus CPU module: WindLDR version 8.6.0 or later

If you are using a version of WindLDR that does not meet this condition, click **Check for Updates** on the **Resources** tab to obtain the latest version of WindLDR.

This concludes the procedure for checking the WindLDR version.

Ladder Program Operation

The FC6A Series MICROSmart performs the following operations to process ladder programs.



The input device values are applied to the ladder program.

The ON/OFF status of the input terminal (external input) is applied to the input circuit. The input refresh loads the external input status into the input devices (I) by way of the input circuit.
A delay occurs until the ON/OFF status of the input circuit is applied to the input devices (I). The delay time is equal to filter time + 1 scan time.

Repeat operations
One sequence of operations is set as one scan, and the time required is set as the scan time.

Processing other than the ladder processing is called the END processing.

The output devices are updated according to the ladder program.

The output refresh applies the status of the output devices (Q) to the output circuit, and then the ON/OFF status of the output terminals (external input) is updated.
A delay occurs until the values of the output devices (Q) are applied to the output circuit. The delay time is at maximum 1 scan time.

Notes:

- The delay time of the hardware is added to the actual input and output delay times.
  For details on the hardware input/output delay times, see Chapter 2 "Product Specifications" in the "FC6A Series MICROSmart User’s Manual".
- The filter time differs depending on the input used.
  - When the input is a CPU module input (I0 to I27)
    See the input filter time configured in the WindLDR **Function Area Settings**. The initial value is 3 ms.
  - When the input is an expansion input (I30 or greater)
    There is no filter time.

Start/Stop Operation

This section describes how to run and stop the FC6A Series MICROSmart.

> ⚠ **Caution** Make sure of safety before starting and stopping the FC6A Series MICROSmart. Incorrect operation of the FC6A Series MICROSmart may cause machine damage or accidents.

Start/Stop Schematic

You can start and stop FC6A Series MICROSmart operations by using WindLDR operations, FC6A Series MICROSmart operations, function switch operations, menu operations using an HMI module, by turning the FC6A Series MICROSmart on or off, or by using the stop/reset input function by way of external input.

- For operations using WindLDR, see "Start/Stop Operation Using WindLDR" on page 1-18.
- For operations using the FC6A Series MICROSmart power supply, see "Start/Stop Operation Using the Power Supply" on page 1-19.
- For operations using the function switch, see Chapter 5 "Function Switch Configuration" in the "FC6A Series MICROSmart User’s Manual".
- For operations using an HMI module, see Chapter 7 "HMI Function" in the "FC6A Series MICROSmart User’s Manual".
- For the stop input settings, see Chapter 5 "Stop Input" in the "FC6A Series MICROSmart User’s Manual".
- For the reset input settings, see Chapter 5 "Reset Input" in the "FC6A Series MICROSmart User’s Manual".

The switch from start to stop is performed during the END processing of the user program, and the following functions are also stopped when the user program is stopped. For details on END processing, see the "Breakdown of END Processing Time" on page A-5.

High-speed counter/frequency measurement/user interrupt/catch input/timer interrupt/user communication/pulse output

- You can use M8025 (Maintain Outputs While Stopped) to select whether to maintain or clear the outputs while the CPU is stopped.
  For details on M8025 (Maintain Outputs While Stopped), see "M8025: Maintain Outputs While Stopped" on page 2-10.
- The current value of the timer instruction is reset.
  For details on timer instructions, see "TML (1-s Timer)" on page 4-7".

Also, when you switch from stop to start, the device status is cleared or kept depending on **Memory Backup** in the **Function Area Settings**. For details, see Chapter 5 "Memory Backup" in the "FC6A Series MICROSmart User’s Manual".

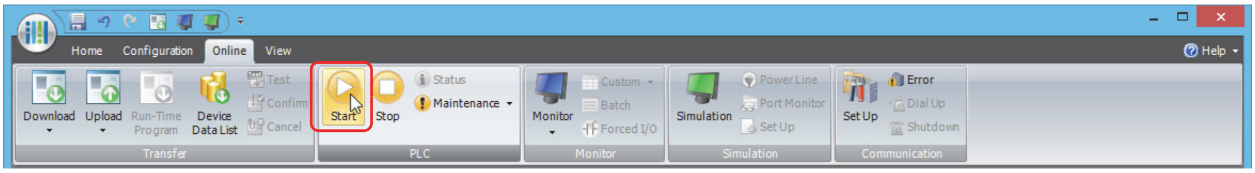
Start/Stop Operation Using WindLDR

The FC6A Series MICROSmart can be started and stopped using WindLDR run on a Windows computer connected to the FC6A Series MICROSmart. When the **Start** button is pressed in the menu bar shown below, start control special internal relay M8000 is turned on to start the FC6A Series MICROSmart. When the **Stop** button is pressed, M8000 is turned off to stop the FC6A Series MICROSmart.

1. Connect the computer to the FC6A Series MICROSmart, start WindLDR, and power up the FC6A Series MICROSmart.
2. Check that a stop input is not designated using **Configuration** > **Run/Stop Control** > **Stop and Reset Inputs**. See Chapter 5 "Stop Input and Reset Input" in the "FC6A Series MICROSmart User’s Manual".

Note: When a stop input is designated, the FC6A Series MICROSmart cannot be started or stopped by turning start control special internal relay M8000 on or off.

3. Select **Online** from the WindLDR menu bar.
   The Online tab appears.



4. Click the **Start** button to start operation, then the start control special internal relay M8000 is turned on.

5. Click the **Stop** button to stop operation, then the start control special internal relay M8000 is turned off.
   The PLC operation can also be started and stopped while WindLDR is in the monitor mode. Select **Online** > **Monitor** > **Monitor** and click the **Start** or **Stop** button.

Note: Special internal relay M8000 is a keep type internal relay and stores the status when power is turned off. M8000 retains its previous status when power is turned on again. However, when the backup battery is dead, M8000 loses the stored status, and can be turned on or off as programmed when the FC6A Series MICROSmart is powered up. The selection is made in **Configuration** > **Run/Stop Control** > **Run/Stop Selection at Memory Backup Error**. See Chapter 5 "Run/Stop Selection at Keep Data Error" in the "FC6A Series MICROSmart User's Manual".

Start/Stop Operation Using the Power Supply

The FC6A Series MICROSmart can be started and stopped by turning power on and off.

1. Power up the FC6A Series MICROSmart to start operation.

2. Turn power on and off to start and stop operation.

2: DEVICES

This chapter provides detailed descriptions of the allocations of devices such as inputs, outputs, internal relays, registers, timers, and counters that are used in the basic and advanced instructions, as well as details about the allocations of special internal relays and special data registers.

Please use this chapter as a reference for the devices.

Note: The programing and operation of FC6A Series MICROSmart user programs requires specialist knowledge.
Take the time to develop a thorough understanding of the contents and programs in this manual before using the FC6A Series MICROSmart.

Device Addresses

All-in-One CPU Module/J1939 All-in-One CPU Module

Device	Symbol	Unit	Range (Points)		
			16-I/O Type	24-I/O Type	40-I/O Type
Inputs[*1]	I	Bit	I0 - I10 (9 points)	I0 - I15 (14 points)	I0 - I27 (24 points)
Expansion Inputs[*1]	I	Bit	I30 - I187 (128 points) I190 - I507[*2] (256 points) I630 - I633[*4] (4 points)	I30 - I307 (224 points) I310 - I627[*3] (256 points) I630 - I633[*4] (4 points)	I30 - I307 (224 points) I310 - I627[*3] (256 points) I630 - I637[*4] (8 points)
Output[*1]	Q	Bit	Q0 - Q6 (7 points)	Q0 - Q11 (10 points)	Q0 - Q17 (16 points)
Expansion Outputs[*1]	Q	Bit	Q30 - Q187 (128 points) Q190 - Q507[*2] (256 points) Q630 - Q633[*4] (4 points)	Q30 - Q307 (224 points) Q310 - Q627[*3] (256 points) Q630 - Q633[*4] (4 points)	Q30 - Q307 (224 points) Q310 - Q627[*3] (256 points) Q630 - Q637[*4] (8 points)
Internal Relay[*1]	M	Bit	M0 - M7997 (6,400 points) M10000 - M17497 (6,000 points)		
Special Internal Relay[*1]	M	Bit	M8000 - M8317 (256 points)		
Shift Register	R	Bit	R0 - R255 (256 points)		
Timer	T	Bit/Word	T0 - T1023 (1,024 points)		
Counter	C	Bit/Word	C0 - C511 (512 points)		
Data Register	D	Bit/Word	D0000 - D7999 (8,000 points) D10000 to D55999 (46,000 points)		
Special Data Register	D	Bit/Word	D8000 - D8499 (500 points)		

*1 The least significant digit of the device address is an octal number (0 to 7).

*2 I190 to I507 and Q190 to Q507 are devices that can only be used when expansion modules are connected at the expansion interface side using the expansion interface module (expander).

*3 I310 to I627 and Q310 to Q627 are devices that can only be used when expansion modules are connected at the expansion interface side using the expansion interface module (expander).

*4 I630 to I637 and Q630 to Q637 are devices that can be used only when I/O cartridges are connected.

Plus CPU module

Device	Symbol	Unit	Range (Points)	
			Plus 16-I/O Type	Plus 32-I/O Type
Inputs*1	I	Bit	I0 - I7 (8 points)	I0 - I17 (16 points)
Expansion Inputs*1	I	Bit	I30 - I307 (224 points) I310 - I627*2 (256 points) I630 - I643*3 (12 points) I1000 - I10597*4 (2,016 points)	
Output*1	Q	Bit	Q0 - Q7 (8 points)	Q0 - Q17 (16 points)
Expansion Outputs*1	Q	Bit	Q30 - Q307 (224 points) Q310 - Q627*2 (256 points) Q630 - Q643*3 (12 points) Q1000 - Q10597*4 (2,016 points)	
Internal Relay*1	M	Bit	M0 - M7997 (6,400 points) M10000 - M21247 (9,000 points)	
Special Internal Relay*1	M	Bit	M8000 - M9997 (1,600 points)	
Shift Register	R	Bit	R0 - R255 (256 points)	
Timer	T	Bit/Word	T0 - T1999 (2,000 points)	
Counter	C	Bit/Word	C0 - C511 (512 points)	
Data Register	D	Bit/Word	D0000 - D7999 (8,000 points) D10000 - D61999 (52,000 points)	
Non-retentive Data Register	D	Bit*7/Word	D70000 - D269999*5 (200,000 points)	
Special Data Register	D	Bit/Word	D8000 - D8899 (900 points)	
Index Register*6	P	2 words	P0 - P15 (16 points)	

*1 The least significant digit of the device address is an octal number (0 to 7).
*2 I1310 to I627 and Q310 to Q627 are devices that can only be used when expansion modules are connected at the expansion interface side using the expansion interface module (expander). (Node 0)
*3 I630 to I643 and Q630 to Q643 are devices that can be used only when I/O cartridges are connected.
*4 I1000 to I10597 and Q1000 to Q10597 are devices that can be used only when expansion modules are connected using the expansion interface modules (remote master and slaves) and the expansion interface modules (expander). (Node 1 to 10)
*5 D70000 to D269999 cannot be designated as "Keep." Data register values are kept when you switch the PLC from Stop to Run, but those values are cleared at power up.
*6 The data type that can be used is L (Long) only.
*7 Usable only in scripts executed by the SCRPT instruction and as argument devices used in UMACRO instructions.

- **Inputs (I), Expansion Inputs (I)**
  Devices that input on/off information from external devices to the FC6A Series MICROSmart.

- **Outputs (Q), Expansion Outputs (Q)**
  Devices that output on/off information from the FC6A Series MICROSmart to external devices.

- **Internal Relays (M)**
  Bit devices used internally on the FC6A Series MICROSmart.

- **Special Internal Relays (M)**
  Bit devices used internally on the FC6A Series MICROSmart. Special functions are assigned to each bit.

- **Shift Registers (R)**
  Bit devices that are used with the SFR instruction and the SFRN instruction. The bit sequence of the data is shifted according to pulse input.

- **Timer (T)**
  Timers used internally in the FC6A Series MICROSmart. There are three devices: Timer bits (symbol: T, unit: bit), timer preset values (symbol: TP, unit: word), and timer current values (symbol: TC, unit: word).

  These can be used as an on-delay timer or an off-delay timer. For details on timers (T), see "Using Timer or Counter as Source Device" on page 3-6.

- **Counters (C)**
  Counters used internally in the FC6A Series MICROSmart. There are three devices: Counter bits (symbol: C, unit: bit), counter preset values (symbol: CP, unit: word), and counter current values (symbol: CC, unit: word). These can be used as an adding counter or a reversible counter. For details on counters (C), see "Using Timer or Counter as Source Device" on page 3-6.

- **Data Registers (D)**
  Word devices that are used for writing numerical data internally in the FC6A Series MICROSmart. These can also be used as bit devices.

- **Special Data Registers (D)**
  Word devices that are used for writing numerical data internally in the FC6A Series MICROSmart. Special functions are assigned to each data register. These can also be used as bit devices.

Notes:

- Although the device symbol for internal relays (M0000 to M7997, M10000 to M21247) and special internal relays (M8000 to M9997) is the same ("M"), the device characteristics are different. Special functions are assigned to each bit of the special internal relays.
- Although the device symbol for the data registers (D0000 to D7999, D10000 to D61999, D70000 to D269999) and special data registers (D8000 to D8899) is the same ("D"), the device characteristics are different. Special functions are assigned to each special data register.

Special Internal Relay

Special Internal Relay Device Addresses

⚠ **Warning**	Do not write to data in the area marked as reserved in the special internal relays list. Otherwise the system may not operate correctly.

Note: R/W is an abbreviation for read/write.
The notation for the R/W field is as follows.
R/W: The device can be both read from and written to
R: Read-only
W: Write-only

Device Address	Description		When Stopped	Power OFF	R/W
M8000	Start Control		Maintained	Maintained	R/W
M8001	1-s Clock Reset		Cleared	Cleared	R/W
M8002	All Outputs OFF		Cleared	Cleared	R/W
M8003	Carry (Cy) or Borrow (Bw)		Cleared	Cleared	R/W
M8004	User Program Execution Error		Cleared	Cleared	R/W
M8005	Communication Error		Maintained	Cleared	R/W
M8006	Communication Prohibited Flag (When Data Link Master)		Maintained	Maintained	R/W
M8007	Initialization Flag (When Data Link Master)/Stop Communication Flag (When Data Link Slave)		Cleared	Cleared	R/W
M8010	Status LED Operation		Operating	Cleared	R/W
M8011 M8012	**— Reserved —**		—	—	—
M8013	Calendar/Clock Data Write/Adjust Error Flag		Operating	Cleared	R/W
M8014	Calendar/Clock Data Read Error Flag		Operating	Cleared	R/W
M8015	**— Reserved —**		—	—	—
M8016	Calendar Data Write Flag		Operating	Cleared	R/W
M8017	Clock Data Write Flag		Operating	Cleared	R/W
M8020	Calendar/Clock Data Write Flag		Operating	Cleared	R/W
M8021	Clock Data Adjust Flag		Operating	Cleared	R/W
M8022	User Communication Receive Instruction Cancel Flag (Port 1)		Cleared	Cleared	R/W
M8023	User Communication Receive Instruction Cancel Flag (Port 2)		Cleared	Cleared	R/W
M8024	BMOV/WSFT Executing Flag		Maintained	Maintained	R/W
M8025	Maintain Outputs While Stopped		Maintained	Cleared	R/W
M8026	User Communication Receive Instruction Cancel Flag (Port 3)		Cleared	Cleared	R/W
M8027	High-speed Counter (Group 1/I0)	Count Direction Flag	Maintained	Cleared	R/W
M8030		Comparison Output Reset	Cleared	Cleared	R/W
M8031		Gate Input	Maintained	Cleared	R/W
M8032		Reset Input	Maintained	Cleared	R/W
M8033	User Communication Receive Instruction Cancel Flag (Port 4)		Cleared	Cleared	R/W
M8034	High-speed Counter (Group 3/I3)	Comparison Output Reset	Cleared	Cleared	R/W
M8035		Gate Input	Maintained	Cleared	R/W
M8036		Reset Input	Maintained	Cleared	R/W
M8037	**— Reserved —**		—	—	—
M8040	High-speed Counter (Group 4/I4)	Comparison Output Reset	Cleared	Cleared	R/W
M8041		Gate Input	Maintained	Cleared	R/W
M8042		Reset Input	Maintained	Cleared	R/W
M8043	High-speed Counter (Group 5/I6)	Count Direction Flag	Maintained	Cleared	R/W
M8044		Comparison Output Reset	Cleared	Cleared	R/W
M8045		Gate Input	Maintained	Cleared	R/W
M8046		Reset Input	Maintained	Cleared	R/W
M8047 M8050	**— Reserved —**		—	—	—

Device Address	Description		When Stopped	Power OFF	R/W
M8051	High-speed Counter (Group 2/I1)	Comparison Output Reset	Cleared	Cleared	R/W
M8052		Gate Input	Maintained	Cleared	R/W
M8053		Reset Input	Maintained	Cleared	R/W
M8054		Comparison ON Status	Maintained	Cleared	R
M8055		Overflow	Maintained	Cleared	R
M8056	**— Reserved —**		—	—	—
M8057	High-speed Counter (Group 6/I7)	Comparison Output Reset	Cleared	Cleared	R/W
M8060		Gate Input	Maintained	Cleared	R/W
M8061		Reset Input	Maintained	Cleared	R/W
M8062		Comparison ON Status	Maintained	Cleared	R
M8063		Overflow	Maintained	Cleared	R
M8064 to M8067	**— Reserved —**		—	—	—
M8070	SD Memory Card Mount Status		Maintained	Cleared	R
M8071	Accessing SD Memory Card		Maintained	Cleared	R
M8072	Unmount SD Memory Card		Operating	Cleared	R/W
M8073	Function Switch Status		Operating	Cleared	R
M8074	Battery Voltage Measurement Flag		Operating	Cleared	R/W
M8075 to M8077	**— Reserved —**		—	—	—
M8080	Data Link Slave 1 Communication Completed Relay (When Data Link Master)		Operating	Cleared	R
M8081	Data Link Slave 2 Communication Completed Relay		Operating	Cleared	R
M8082	Data Link Slave 3 Communication Completed Relay		Operating	Cleared	R
M8083	Data Link Slave 4 Communication Completed Relay		Operating	Cleared	R
M8084	Data Link Slave 5 Communication Completed Relay		Operating	Cleared	R
M8085	Data Link Slave 6 Communication Completed Relay		Operating	Cleared	R
M8086	Data Link Slave 7 Communication Completed Relay		Operating	Cleared	R
M8087	Data Link Slave 8 Communication Completed Relay		Operating	Cleared	R
M8090	Data Link Slave 9 Communication Completed Relay		Operating	Cleared	R
M8091	Data Link Slave 10 Communication Completed Relay		Operating	Cleared	R
M8092	Data Link Slave 11 Communication Completed Relay		Operating	Cleared	R
M8093	Data Link Slave 12 Communication Completed Relay		Operating	Cleared	R
M8094	Data Link Slave 13 Communication Completed Relay		Operating	Cleared	R
M8095	Data Link Slave 14 Communication Completed Relay		Operating	Cleared	R
M8096	Data Link Slave 15 Communication Completed Relay		Operating	Cleared	R
M8097	Data Link Slave 16 Communication Completed Relay		Operating	Cleared	R
M8100	Data Link Slave 17 Communication Completed Relay		Operating	Cleared	R
M8101	Data Link Slave 18 Communication Completed Relay		Operating	Cleared	R
M8102	Data Link Slave 19 Communication Completed Relay		Operating	Cleared	R
M8103	Data Link Slave 20 Communication Completed Relay		Operating	Cleared	R
M8104	Data Link Slave 21 Communication Completed Relay		Operating	Cleared	R
M8105	Data Link Slave 22 Communication Completed Relay		Operating	Cleared	R
M8106	Data Link Slave 23 Communication Completed Relay		Operating	Cleared	R
M8107	Data Link Slave 24 Communication Completed Relay		Operating	Cleared	R
M8110	Data Link Slave 25 Communication Completed Relay		Operating	Cleared	R
M8111	Data Link Slave 26 Communication Completed Relay		Operating	Cleared	R
M8112	Data Link Slave 27 Communication Completed Relay		Operating	Cleared	R
M8113	Data Link Slave 28 Communication Completed Relay		Operating	Cleared	R
M8114	Data Link Slave 29 Communication Completed Relay		Operating	Cleared	R
M8115	Data Link Slave 30 Communication Completed Relay		Operating	Cleared	R
M8116	Data Link Slave 31 Communication Completed Relay		Operating	Cleared	R
M8117	Data Link All Slaves Communication Completed Relay		Operating	Cleared	R
M8120	Initialize Pulse		Cleared	Cleared	R
M8121	1-s Clock		Operating	Cleared	R
M8122	100-ms Clock		Operating	Cleared	R

Device Address	Description		When Stopped	Power OFF	R/W
M8123	10-ms Clock		Operating	Cleared	R
M8124	Timer/Counter Preset Value Changed		Maintained	Cleared	R
M8125	In-operation Output		Cleared	Cleared	R
M8126	1 Scan ON After Run-Time Download Completes		Cleared	Cleared	R
M8127	**— Reserved —**		—	—	—
M8130	High-speed Counter (Group 1/I0)	Reset Status	Maintained	Cleared	R
M8131		Comparison ON Status	Maintained	Cleared	R
M8132	**— Reserved —**		—	—	—
M8133	High-speed Counter (Group 3/I3)	Comparison ON Status	Maintained	Cleared	R
M8134	High-speed Counter (Group 4/I4)	Comparison ON Status	Maintained	Cleared	R
M8135	High-speed Counter (Group 5/I6)	Reset Status	Maintained	Cleared	R
M8136		Comparison ON Status	Maintained	Cleared	R
M8137	Interrupt Input I0 Status (Group 1/I0)	(ON: Allowed, OFF: Prohibited)	Cleared	Cleared	R
M8140	Interrupt Input I1 Status (Group 2/I1)		Cleared	Cleared	R
M8141	Interrupt Input I3 Status (Group 3/I3)		Cleared	Cleared	R
M8142	Interrupt Input I4 Status (Group 4/I4)		Cleared	Cleared	R
M8143	Interrupt Input I6 Status (Group 5/I6)		Cleared	Cleared	R
M8144	Timer Interrupt Status		Cleared	Cleared	R
M8145	User Communication Receive Instruction Cancel Flag (Port 5)		Cleared	Cleared	R/W
M8146	User Communication Receive Instruction Cancel Flag (Port 6)		Cleared	Cleared	R/W
M8147	User Communication Receive Instruction Cancel Flag (Port 7)		Cleared	Cleared	R/W
M8150	Comparison Result 1		Maintained	Cleared	R
M8151	Comparison Result 2		Maintained	Cleared	R
M8152	Comparison Result 3		Maintained	Cleared	R
M8153	Catch Input ON/OFF Status	Group 1/I0	Maintained	Cleared	R
M8154		Group 2/I1	Maintained	Cleared	R
M8155		Group 3/I3	Maintained	Cleared	R
M8156		Group 4/I4	Maintained	Cleared	R
M8157		Group 5/I6	Maintained	Cleared	R
M8160		Group 6/I7	Maintained	Cleared	R
M8161	High-speed Counter (Group 1/I0)	Overflow	Maintained	Cleared	R
M8162		Underflow	Maintained	Cleared	R
M8163	High-speed Counter (Group 5/I6)	Overflow	Maintained	Cleared	R
M8164		Underflow	Maintained	Cleared	R
M8165	High-speed Counter (Group 3/I3)	Overflow	Maintained	Cleared	R
M8166	High-speed Counter (Group 4/I4)	Overflow	Maintained	Cleared	R
M8167	Interrupt Input I7 Status (Group 6/I7)	(ON: Allowed, OFF: Prohibited)	Maintained	Cleread	R
M8170	User Communication Receive Instruction Cancel Flag (Port 8)		Cleared	Cleared	R/W
M8171	**— Reserved —**		—	—	—
M8172	Transistor Source Output Overcurrent Detection	Group 1	Operating	Cleared	R
M8173		Group 2	Operating	Cleared	R
M8174		Group 3	Operating	Cleared	R
M8175		Group 4	Operating	Cleared	R
M8176	User Communication Receive Instruction Cancel Flag (Port 9)		Cleared	Cleared	R/W
M8177 to M8183	**— Reserved —**		—	—	—
M8184	Change HMI Module Network Settings Trigger		Operating	Cleared	R/W
M8185	In Daylight Saving Time Period		Operating	Cleared	R
M8186	Ethernet Port 1 Executing Auto Ping		Operating	Cleared	R
M8187	Ethernet Port 1 Auto Ping Stop Flag		Operating	Cleared	R/W
M8190	Change CPU Module Ethernet Port 1 Network Settings Trigger		Operating	Cleared	R/W
M8191	SNTP Acquisition Flag		Operating	Cleared	R/W

Device Address	Description		When Stopped	Power OFF	R/W
M8192	Interrupt Input I0 Edge	On: Rising Edge Off: Falling Edge	Cleared	Cleared	R
M8193	Interrupt Input I3 Edge		Cleared	Cleared	R
M8194	Interrupt Input I4 Edge		Cleared	Cleared	R
M8195	Interrupt Input I6 Edge		Cleared	Cleared	R
M8196	Interrupt Input I7 Edge		Cleared	Cleared	R
M8197	Interrupt Input I1 Edge		Cleared	Cleared	R
M8200	User Communication Receive Instruction Cancel Flag	Connection 1	Cleared	Cleared	R/W
M8201		Connection 2	Cleared	Cleared	R/W
M8202		Connection 3	Cleared	Cleared	R/W
M8203		Connection 4	Cleared	Cleared	R/W
M8204		Connection 5	Cleared	Cleared	R/W
M8205		Connection 6	Cleared	Cleared	R/W
M8206		Connection 7	Cleared	Cleared	R/W
M8207		Connection 8	Cleared	Cleared	R/W
M8210	**— Reserved —**		—	—	—
M8211	HMI Module Send E-mail Server Settings Initialization		Operating	Cleared	R/W
M8212	Connection Status (ON: Connected, OFF: Not Connected)	Connection 1	Operating	Cleared	R
M8213		Connection 2	Operating	Cleared	R
M8214		Connection 3	Operating	Cleared	R
M8215		Connection 4	Operating	Cleared	R
M8216		Connection 5	Operating	Cleared	R
M8217		Connection 6	Operating	Cleared	R
M8220		Connection 7	Operating	Cleared	R
M8221		Connection 8	Operating	Cleared	R
M8222	Disconnect User Communication Connection	Connection 1	Operating	Cleared	R/W
M8223		Connection 2	Operating	Cleared	R/W
M8224		Connection 3	Operating	Cleared	R/W
M8225		Connection 4	Operating	Cleared	R/W
M8226		Connection 5	Operating	Cleared	R/W
M8227		Connection 6	Operating	Cleared	R/W
M8230		Connection 7	Operating	Cleared	R/W
M8231		Connection 8	Operating	Cleared	R/W
M8232	HMI Module Connection Information Reference Connection Status		Operating	Cleared	R
M8233 to M8247	**— Reserved —**		—	—	—
M8250	Download from SD Memory Card Execution Flag		Operating	Cleared	R/W
M8251	Upload to SD Memory Card Execution Flag		Operating	Cleared	R/W
M8252	Executing SD Memory Card Download		Operating	Cleared	R
M8253	Executing SD Memory Card Upload		Operating	Cleared	R
M8254	SD Memory Card Download/Upload Execution Completion Output		Operating	Cleared	R
M8255	SD Memory Card Download/Upload Execution Error Output		Operating	Cleared	R
M8256 M8257	**— Reserved —**		—	—	—
M8260	Write Recipe Execution Flag		Operating	Cleared	R/W
M8261	Read Recipe Execution Flag		Operating	Cleared	R/W
M8262	Executing Write Recipe		Operating	Cleared	R/W
M8263	Executing Read Recipe		Operating	Cleared	R/W
M8264	Recipe Execution Completed Output		Operating	Cleared	R/W
M8265	Recipe Execution Error Output		Operating	Cleared	R/W
M8266 to M8297	**— Reserved —**		—	—	—
M8300	J1939 Communication Permitted Flag		Cleared	Cleared	R/W
M8301	J1939 Online Status		Cleared	Cleared	R
M8302	J1939 Local Station Address Confirmation Status		Cleared	Cleared	R
M8303	J1939 Communication Error Output		Cleared	Cleared	R

Device Address	Description		When Stopped	Power OFF	R/W
M8304	J1939 Communication Bus Off Occurrence Output		Cleared	Cleared	R
M8305 to M8310	**— Reserved —**		—	—	—
M8311	ESC+Key Input (Up)	ESC+Key Input (▲)	Cleared	Cleared	R
M8312	ESC+Key Input (Down)	ESC+Key Input (▼)	Cleared	Cleared	R
M8313	ESC+Key Input (Left)	ESC+Key Input (◀)	Cleared	Cleared	R
M8314	ESC+Key Input (Right)	ESC+Key Input (▶)	Cleared	Cleared	R
M8315 to M8319	**— Reserved —**		—	—	—
M8320	Initialize Expansion Interface Remote Master/Slave		Operating	Cleared	R/W
M8321 to M8330	**— Reserved —**		—	—	—
M8331	Ethernet Port 2 Executing Auto Ping		Operating	Cleared	R
M8332	Ethernet Port 2 Auto Ping Stop Flag		Operating	Cleared	R/W
M8333	Change CPU Module Ethernet Port 2 Network Settings Trigger		Operating	Cleared	R/W
M8334	User Communication Receive Instruction Cancel Flag	Connection 9	Cleared	Cleared	R/W
M8335		Connection 10	Cleared	Cleared	R/W
M8336		Connection 11	Cleared	Cleared	R/W
M8337		Connection 12	Cleared	Cleared	R/W
M8340		Connection 13	Cleared	Cleared	R/W
M8341		Connection 14	Cleared	Cleared	R/W
M8342		Connection 15	Cleared	Cleared	R/W
M8343		Connection 16	Cleared	Cleared	R/W
M8344	Ethernet Port 1 Send E-mail Server Settings Initialization		Operating	Cleared	R/W
M8345	Connection Status (ON: Connected, OFF: Not Connected)	Connection 9	Operating	Cleared	R
M8346		Connection 10	Operating	Cleared	R
M8347		Connection 11	Operating	Cleared	R
M8350		Connection 12	Operating	Cleared	R
M8351		Connection 13	Operating	Cleared	R
M8352		Connection 14	Operating	Cleared	R
M8353		Connection 15	Operating	Cleared	R
M8354		Connection 16	Operating	Cleared	R
M8355	Disconnect User Communication Connection	Connection 9	Operating	Cleared	R/W
M8356		Connection 10	Operating	Cleared	R/W
M8357		Connection 11	Operating	Cleared	R/W
M8360		Connection 12	Operating	Cleared	R/W
M8361		Connection 13	Operating	Cleared	R/W
M8362		Connection 14	Operating	Cleared	R/W
M8363		Connection 15	Operating	Cleared	R/W
M8364		Connection 16	Operating	Cleared	R/W
M8365	User Communication Receive Instruction Cancel Flag (Port 10)		Cleared	Cleared	R/W
M8366	User Communication Receive Instruction Cancel Flag (Port 11)		Cleared	Cleared	R/W
M8367	User Communication Receive Instruction Cancel Flag (Port 12)		Cleared	Cleared	R/W
M8370	User Communication Receive Instruction Cancel Flag (Port 13)		Cleared	Cleared	R/W
M8371	User Communication Receive Instruction Cancel Flag (Port 14)		Cleared	Cleared	R/W
M8372	User Communication Receive Instruction Cancel Flag (Port 15)		Cleared	Cleared	R/W
M8373	User Communication Receive Instruction Cancel Flag (Port 16)		Cleared	Cleared	R/W
M8374	User Communication Receive Instruction Cancel Flag (Port 17)		Cleared	Cleared	R/W
M8375	User Communication Receive Instruction Cancel Flag (Port 18)		Cleared	Cleared	R/W
M8376	User Communication Receive Instruction Cancel Flag (Port 19)		Cleared	Cleared	R/W
M8377	User Communication Receive Instruction Cancel Flag (Port 20)		Cleared	Cleared	R/W
M8380	User Communication Receive Instruction Cancel Flag (Port 21)		Cleared	Cleared	R/W
M8381	User Communication Receive Instruction Cancel Flag (Port 22)		Cleared	Cleared	R/W
M8382	User Communication Receive Instruction Cancel Flag (Port 23)		Cleared	Cleared	R/W
M8383	User Communication Receive Instruction Cancel Flag (Port 24)		Cleared	Cleared	R/W

Device Address	Description		When Stopped	Power OFF	R/W
M8384	User Communication Receive Instruction Cancel Flag (Port 25)		Cleared	Cleared	R/W
M8385	User Communication Receive Instruction Cancel Flag (Port 26)		Cleared	Cleared	R/W
M8386	User Communication Receive Instruction Cancel Flag (Port 27)		Cleared	Cleared	R/W
M8387	User Communication Receive Instruction Cancel Flag (Port 28)		Cleared	Cleared	R/W
M8390	User Communication Receive Instruction Cancel Flag (Port 29)		Cleared	Cleared	R/W
M8391	User Communication Receive Instruction Cancel Flag (Port 30)		Cleared	Cleared	R/W
M8392	User Communication Receive Instruction Cancel Flag (Port 31)		Cleared	Cleared	R/W
M8393	User Communication Receive Instruction Cancel Flag (Port 32)		Cleared	Cleared	R/W
M8394	User Communication Receive Instruction Cancel Flag (Port 33)		Cleared	Cleared	R/W
M8395 to M8447	**— Reserved —**		—	—	—
M8450	BACnet Communication Bit		Operating	Cleared	R/W
M8451 to M8597	**— Reserved —**		—	—	—
M8600	High-speed Counter (Group 3/I3)	Reset Status	Maintained	Cleared	R
M8601		Underflow	Maintained	Cleared	R
M8602		Count Direction Flag	Maintained	Cleared	R
M8603 to M8997	**— Reserved —**		—	—	—

Supplementary Descriptions of the Special Internal Relays

■ M8000: Start Control

M8000 controls the run/stop status of the FC6A Series MICROSmart. The CPU is set to run when M8000 is turned on, and the CPU is set to off when M8000 is turned off. See "Start/Stop Operation" on page 1-18. However, the function switch, stop input, and reset input have precedence over start control. M8000 maintains its status when the CPU is powered down. When data to be maintained during a power failure disappears after the CPU has been off for a period longer than the battery backup duration, the CPU restarts operation as selected in Configuration > Run/Stop Control > Run/Stop Selection at Keep Data Error. For details, see Chapter 5 "Run/Stop Selection at Keep Data Error" in the "FC6A Series MICROSmart User's Manual".

■ M8001: 1-s Clock Reset

While M8001 is on, M8121 (1-s clock) is always off.

■ M8002: All Outputs OFF

While M8002 is on, all outputs are off. The self-holding circuit created in the ladder program is also off.

■ M8003: Carry/Borrow

When a carry (Cy) or borrow (Bw) results from executing an addition or subtraction instruction, M8003 is turned on. For details, see "Carry and Borrow" on page 3-10".

■ M8004: User Program Execution Error

When an error occurs while executing a user program, M8004 is turned on. For details on user program execution errors, see "User Program Execution Error" on page A-12.

■ M8005: Communication Error

When an error occurs during data link communication, M8005 is turned on. The state is retained even when the error is cleared.

■ M8006: Communication Prohibited Flag (When Data Link Master)

During data link communication, communication is stopped while M8006 is on.

■ M8007: Initialization Flag (When Data Link Master)/Stop Communication Flag (When Data Link Slave)

When data link master: When this flag is turned on in the run status, the data link is initialized just once to check the connection status. Use this when the slave configured in the data link is powered at a timing slower than the master.

When data link slave : This flag is turned on when communication from the master is interrupted for 10 s or longer. This flag is turned off when communication can be normally received.

■ M8010: Status LED Operation

While M8010 is on, the status LED [STAT] is turned on. While off, the status LED [STAT] is turned off.

- **M8013: Calendar/Clock Data Write/Adjust Error Flag**
  When the clock writing or clock adjustment processing could not be executed normally, M8013 is turned on. It is turned off when the processing completes normally.

- **M8014: Calendar/Clock Data Read Error Flag**
  When an error occurs while calendar/clock data is read from the internal clock to the special data registers (D8008 to D8021), M8014 is turned on. It is turned off when reading completes normally.

- **M8016: Calendar Data Write Flag**
  When M8016 is turned on after writing data to the calendar data (write-only) special data registers (D8015 to D8018), the calendar data (year, month, day, day of the week) is written to the internal clock.

- **M8017: Clock Data Write Flag**
  When M8017 is turned on after writing data to the clock data (write-only) special data registers (D8019 to D8021), the clock data (hours, minutes, seconds) is written to the internal clock.

- **M8020: Calendar/Clock Data Write Flag**
  When M8020 is turned on after writing data to the calendar/clock data (write-only) special data registers (D8015 to D8021), the calendar data (year, month, day, day of the week) and the clock data (hours, minutes, seconds) is written to the internal clock.

- **M8021: Clock Data Adjust Flag**
  When M8021 is turned on, the clock is adjusted with respect to seconds.
  - When M8021 is turned on, if the seconds are between 0 and 29, the seconds will be set to 0 and the minutes remain the same.
  - When M8021 is turned on, if the seconds are between 30 and 59, the seconds will be set to 0 and 1 will be added to the minutes.

- **M8022: User Communication Receive Instruction Cancel Flag (Port 1)**
  While M8022 is on, user communication (receive instruction) executing on Port 1 is canceled.

- **M8023: User Communication Receive Instruction Cancel Flag (Port 2)**
  While M8023 is on, user communication (receive instruction) executing on Port 2 is canceled.

- **M8024: BMOV/WSFT Executing Flag**
  While the WSFT (word shift) instruction or the BMOV (block move) instruction is executing, M8024 is turned on. When completed, M8024 is turned off.

- **M8025: Maintain Outputs While Stopped**
  When the FC6A Series MICROSmart is stopped while running with M8025 turned on, the outputs maintain their runtime status. When set to run again, M8025 is automatically turned off.

- **M8026: User Communication Receive Instruction Cancel Flag (Port 3)**
  While M8026 is on, user communication (receive instruction) executing on Port 3 is canceled.

- **M8033, M8145 to M8147, M8170, M8176, M8365 to M8394: User Communication Receive Instruction Cancel Flag (Port 4 to 33)**
  While these flags are on, user communication (receive instruction) being executed on the corresponding port is canceled.
  M8033 = User Communication Receive Instruction Cancel Flag (Port 4)
  M8145 = User Communication Receive Instruction Cancel Flag (Port 5)
  M8146 = User Communication Receive Instruction Cancel Flag (Port 6)
  M8147 = User Communication Receive Instruction Cancel Flag (Port 7)
  M8170 = User Communication Receive Instruction Cancel Flag (Port 8)
  M8176 = User Communication Receive Instruction Cancel Flag (Port 9)
  M8365 = User Communication Receive Instruction Cancel Flag (Port 10)
  M8366 = User Communication Receive Instruction Cancel Flag (Port 11)
  M8367 = User Communication Receive Instruction Cancel Flag (Port 12)
  M8370 = User Communication Receive Instruction Cancel Flag (Port 13)
  M8371 = User Communication Receive Instruction Cancel Flag (Port 14)
  M8372 = User Communication Receive Instruction Cancel Flag (Port 15)
  M8373 = User Communication Receive Instruction Cancel Flag (Port 16)
  M8374 = User Communication Receive Instruction Cancel Flag (Port 17)
  M8375 = User Communication Receive Instruction Cancel Flag (Port 18)
  M8376 = User Communication Receive Instruction Cancel Flag (Port 19)
  M8377 = User Communication Receive Instruction Cancel Flag (Port 20)

M8380 = User Communication Receive Instruction Cancel Flag (Port 21)
M8381 = User Communication Receive Instruction Cancel Flag (Port 22)
M8382 = User Communication Receive Instruction Cancel Flag (Port 23)
M8383 = User Communication Receive Instruction Cancel Flag (Port 24)
M8384 = User Communication Receive Instruction Cancel Flag (Port 25)
M8385 = User Communication Receive Instruction Cancel Flag (Port 26)
M8386 = User Communication Receive Instruction Cancel Flag (Port 27)
M8387 = User Communication Receive Instruction Cancel Flag (Port 28)
M8390 = User Communication Receive Instruction Cancel Flag (Port 29)
M8391 = User Communication Receive Instruction Cancel Flag (Port 30)
M8392 = User Communication Receive Instruction Cancel Flag (Port 31)
M8393 = User Communication Receive Instruction Cancel Flag (Port 32)
M8394 = User Communication Receive Instruction Cancel Flag (Port 33)

- **M8027 to M8032, M8034 to M8036, M8040 to M8046, M8051 to M8055, M8057 to M8063, M8130, M8131, M8133 to M8136, M8161 to M8167, M8600 to M8602: Special Internal Relays for High-speed Counter**

Special internal relays used for the high-speed counter. For details, see Chapter 5 "High-Speed Counter" in the "FC6A Series MICROSmart User's Manual".
M8027 to M8032, M8130, M8131, M8161, M8162 = High-speed counter (group 1/I0)
M8034 to M8036, M8133, M8165, M8600 to M8602 = High-speed counter (group 3/I3)
M8040 to M8042, M8134, M8166 = High-speed counter (group 4/I4)
M8043 to M8046, M8135, M8136, M8163, M8164 = High-speed counter (group 5/I6)
M8051 to M8055 = High-speed counter (group 2/I1)
M8057 to M8063 = High-speed counter (group 6/I7)

- **M8070: SD Memory Card Mount Status**

When an SD memory card is inserted in the FC6A Series MICROSmart and it has been recognized and can be used, M8070 is turned on. M8070 is turned off if no SD memory card has been inserted or if it is not recognized.

- **M8071: Accessing SD Memory Card**

M8071 is turned on while the SD memory card is being accessed. It is turned off when access has finished.

- **M8072: Unmount SD Memory Card**

When M8072 is turned on, access to the SD memory card is stopped. To make an SD memory card accessible that has had access to it stopped, insert the card once again.

- **M8073: Function Switch Status**

This relay indicates the status of the function switch on the front of the CPU module.
M8073 is on when the function switch is 1. M8073 is off when the function switch is 0.

- **M8074: Battery Voltage Measurement Flag**

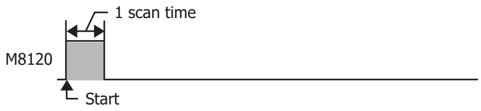
This relay indicates the battery voltage measurement status of the backup battery.
When M8074 is turned on, the battery voltage starts being measured, and it is turned off when the measurement has finished.

- **M8080 to M8117: Data Link Communication Completed Relay**

Special internal relays used for data link communication. For details, see Chapter 7 "Data Link Communication" in the "FC6A Series MICROSmart Communication Manual".

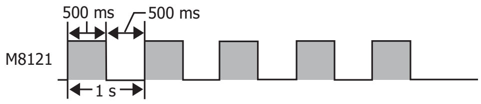
- **M8120: Initialize Pulse**

When operation (RUN) starts, M8120 is turned on for a period of one scan.



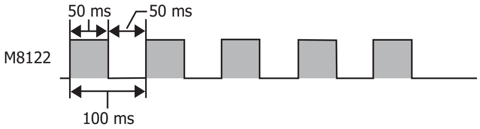
- **M8121: 1-s Clock**

While M8001 is off, M8121 generates clock pulses in a 1 s cycle, with a duty ratio of 1:1 (500 ms on and 500 ms off).



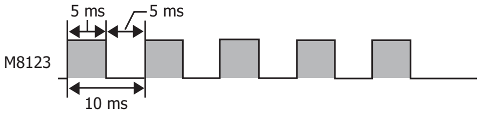
■ M8122: 100-ms Clock

M8122 generates clock pulses in a 100 ms cycle, with a duty ratio of 1:1 (50 ms on and 50 ms off).



■ M8123: 10-ms Clock

M8123 generates clock pulses in a 10 ms cycle, with a duty ratio of 1:1 (5 ms on and 5 ms off).



■ M8124: Timer/Counter Preset Value Changed

When timer or counter preset values are changed, M8124 is turned on. When a user program is transferred or when the changed data is cleared, M8124 is turned off.

■ M8125: In-operation Output

M8125 is on during the run status.

■ M8126: Scan ON After Run-Time Download Completes

When the user program is changed during the run status (run-time download), after the download completes, M8126 is turned on for one scan only when the user program starts operation.

■ M8137 to M8143, M8167: Interrupt Input Status

These relays are turned on when the corresponding user interrupt is allowed. These relays are turned off when the user interrupt is prohibited.

M8137 = Interrupt input I0 status
M8140 = Interrupt input I1 status
M8141 = Interrupt input I3 status
M8142 = Interrupt input I4 status
M8143 = Interrupt input I6 status
M8167 = Interrupt input I7 status

■ M8144: Timer Interrupt Status

When the timer interrupt is allowed, M8144 is turned on. When the timer interrupt is prohibited, M8144 is turned off.

■ M8150 to M8152: Comparison Result

M8150 to M8152 turn on according to the comparison results of the CMP= (Compare(=)) instruction and the ICMP>= (Interval Compare) instruction.

For details, see "Special Internal Relays M8150, M8151, and M8152 in ICMP>=" on page 6-6.

■ M8153 to M8160: Catch Input ON/OFF Status

When the rising or falling input edge of the input contact specified as the catch input is detected during a scan, the input contact status is captured. The detectable edges are once per scan.

M8153 = Group 1/I0 status
M8154 = Group 2/I1 status
M8155 = Group 3/I3 status
M8156 = Group 4/I4 status
M8157 = Group 5/I6 status
M8160 = Group 6/I7 status

■ M8172 to M8175: Transistor Source Output Overcurrent Detection

When overcurrent output occurs in a transistor protection source output on the CPU module, a special internal relay (M8172 to M8175) is turned on. The following special internal relays have been allocated with four outputs set as one group. If overcurrent output occurs in any of the special internal relays, it is turned on.

Even if the overcurrent output has been cleared, these special internal relays are not reset to off. To reset a special internal relay to off, create programming to do so in the ladder program.

M8172 = Group 1 (Q0 to Q3) status
M8173 = Group 2 (Q4 to Q7) status
M8174 = Group 3 (Q10 to Q13) status
M8175 = Group 4 (Q14 to Q17) status

- **M8184: Change HMI Module Network Settings Trigger**

When M8184 is turned on, the values stored in D8437 to D8456 are set as the HMI module IP address.
The IP address is not set just by changing the values of D8437 to D8456. For details on changing the HMI module network settings, see Chapter 3 "Network settings by HMI module special data registers" in the "FC6A Series MICROSmart Communication Manual".

- **M8185: In Daylight Saving Time Period**

When the daylight saving time function is enabled, M8185 is on during the daylight saving time period. When outside of the daylight saving time period, M8185 is off.

When the daylight saving time function is disabled, M8185 is off.

- **M8186: Ethernet Port 1 Executing Auto Ping**

M8186 is on when the auto ping on Ethernet port 1 is operating. M8186 is off when auto ping is stopped. For details on auto ping, see Chapter 3 "Auto Ping Function" in the "FC6A Series MICROSmart Communication Manual".

- **M8187: Ethernet Port 1 Auto Ping Stop Flag**

While M8187 is on, the auto ping on Ethernet port 1 is stopped. While M8187 is off, auto ping is executed. At that time, auto ping is executed from the smallest remote host number specified in the remote host list, regardless of the previous end status.

- **M8190: Change CPU Module Ethernet Port 1 Network Settings Trigger**

When M8190 is turned on, the values stored in D8303 to D8323 are set as the IP Settings/DNS Settings of Ethernet port 1 on the CPU module.
The IP address is not set just by changing the values of D8303 to D8323. For details on changing the IP Settings/DNS Settings of Ethernet port 1 on the CPU module, see Chapter 3 "Network settings by special data registers" in the "FC6A Series MICROSmart Communication Manual".

- **M8191: SNTP Acquisition Flag**

When M8191 is turned on, the time information is acquired from the SNTP server.

- **M8192 to M8197: Interrupt Input Edge**

These relays turn on when an interrupt occurs with the rising edge of an interrupt input. They turn off when an interrupt occurs with the falling edge of an interrupt input.
M8192 = Interrupt input I0 edge
M8193 = Interrupt input I3 edge
M8194 = Interrupt input I4 edge
M8195 = Interrupt input I6 edge
M8196 = Interrupt input I7 edge
M8197 = Interrupt input I1 edge

- **M8200 to M8207, M8334 to M8343: User Communication Receive Instruction Cancel Flag**

When M8200 to M8207 or M8334 to M8343 are turned on, the user communication receive instruction being executed is stopped.
M8200 = User communication receive instruction being executed on client connection 1
M8201 = User communication receive instruction being executed on client connection 2
M8202 = User communication receive instruction being executed on client connection 3
M8203 = User communication receive instruction being executed on client connection 4
M8204 = User communication receive instruction being executed on client connection 5
M8205 = User communication receive instruction being executed on client connection 6
M8206 = User communication receive instruction being executed on client connection 7
M8207 = User communication receive instruction being executed on client connection 8
M8334 = User communication receive instruction being executed on client connection 9
M8335 = User communication receive instruction being executed on client connection 10
M8336 = User communication receive instruction being executed on client connection 11
M8337 = User communication receive instruction being executed on client connection 12
M8340 = User communication receive instruction being executed on client connection 13
M8341 = User communication receive instruction being executed on client connection 14
M8342 = User communication receive instruction being executed on client connection 15
M8343 = User communication receive instruction being executed on client connection 16

- **M8211: HMI Module Send E-mail Server Settings Initialization**

When M8211 is turned on, the settings for the send E-mail server on the HMI-Ethernet port are initialized.

- **M8212 to M8221, M8345 to M8354: Connection Status**

While connected to a network device via the maintenance communication server, user communication server/client, or Modbus TCP server/client, the connection status is turned on. While not connected to a network device, the connection status is turned off.

M8212 = Connection 1
M8213 = Connection 2
M8214 = Connection 3
M8215 = Connection 4
M8216 = Connection 5
M8217 = Connection 6
M8220 = Connection 7
M8221 = Connection 8
M8345 = Connection 9
M8346 = Connection 10
M8347 = Connection 11
M8350 = Connection 12
M8351 = Connection 13
M8352 = Connection 14
M8353 = Connection 15
M8354 = Connection 16

- **M8222 to M8231, M8355 to M8364: Disconnect User Communication Connection**

When connected to a remote host via user communication, the corresponding connection is disconnected when M8222 to M8231 is turned on.

M8222 = Connection 1
M8223 = Connection 2
M8224 = Connection 3
M8225 = Connection 4
M8226 = Connection 5
M8227 = Connection 6
M8230 = Connection 7
M8231 = Connection 8
M8355 = Connection 9
M8356 = Connection 10
M8357 = Connection 11
M8360 = Connection 12
M8361 = Connection 13
M8362 = Connection 14
M8363 = Connection 15
M8364 = Connection 16

These relays are enabled only when a user communication client is used.

- **M8232: HMI Module Connection Information Reference Connection Status**

M8232 is turned on when there is a connection with the connection number specified by D8429. M8232 is turned off when there is no connection.

- **M8250: Download from SD Memory Card Execution Flag**

When M8250 is turned on, a ZLD file is downloaded from the SD memory card. The file that will be downloaded is the ZLD file specified in the autoexec.ini file.

- **M8251: Upload to SD Memory Card Execution Flag**

When M8251 is turned on, a ZLD file is uploaded to the SD memory card.
A ZLD file is created with the file name specified in the autoexec.ini file.

- **M8252: Executing SD Memory Card Download**

M8252 is turned on when starting execution of the download from the SD memory card, and when the download has completed, it is turned off.

- **M8253: Executing SD Memory Card Upload**

M8253 is turned on when starting execution of the upload to the SD memory card, and when the upload has completed, it is turned off.

■ **M8254: SD Memory Card Download/Upload Execution Completed Output**

M8254 is turned off when starting execution of the download from the SD memory card or the upload to the SD memory card, and when the download or upload has completed, it is turned on.

■ **M8255: SD Memory Card Download/Upload Execution Error Output**

M8255 is updated when execution of the download from the SD memory card or the upload to the SD memory card has completed. M8255 is turned on when D8255 (Download/Upload Execution Status) is a value other than 0.

■ **M8260: Write Recipe Execution Flag**

When M8260 is turned on, write recipe is executed.

■ **M8261: Read Recipe Execution Flag**

When M8261 is turned on, read recipe is executed.

■ **M8262: Executing Write Recipe**

M8262 is turned on when the write recipe processing starts, and it is turned off when the processing has completed. M8262 also is turned off when read recipe processing starts.

■ **M8263: Executing Read Recipe**

M8263 is turned on when the read recipe processing starts, and it is turned off when the processing has completed. M8263 also is turned off when write recipe processing starts.

■ **M8264: Recipe Execution Completed Output**

M8264 is turned off when starting to read or write a recipe, and it is turned on when reading or writing a recipe has completed.

■ **M8265: Recipe Execution Error Output**

M8265 is turned on when recipe execution has completed and D8264 (Recipe Execution Status) is a value other than 0. For details on recipes, see Chapter 11 "Recipe Function" in the "FC6A Series MICROSmart User's Manual".

■ **M8300 to M8304: J1939 Communication**

Special data registers used in J1939 communication. For details, see Chapter 8 "J1939 Communication" in the "FC6A Series MICROSmart Communication Manual".

■ **M8311 to M8314: Key Input Status**

While the ESC button and direction buttons on the HMI module are simultaneously pressed, the corresponding special internal relays are turned on. When the keys are not pressed, the relays are turned off.

M8311 = ESC key + Up (▲) key
M8312 = ESC key + Down (▼) key
M8313 = ESC key + Left (◀) key
M8314 = ESC key + Right (▶) key

■ **M8320: Initialize Expansion Interface Remote Master/Slave**

When M8320 is turned on, expansion interface remote master and slave modules along with the expansion modules that are connected to the expansion interface remote slave modules are initialized. One the initialization is finished, M8320 is automatically turned off and the I/O refresh at the expansion interface remote slave modules will resume.

■ **M8331: Ethernet Port 2 Executing Auto Ping**

M8331 is on when the auto ping on Ethernet port 2 is operating. M8331 is off when auto ping is stopped. For details on auto ping, see Chapter 3 "Auto Ping Function" in the "FC6A Series MICROSmart Communication Manual".

■ **M8332: Ethernet Port 2 Auto Ping Stop Flag**

While M8332 is on, the auto ping on Ethernet port 2 is stopped. While M8332 is off, auto ping is executed. At that time, auto ping is executed from the smallest remote host number specified in the remote host list, regardless of the previous end status.

■ **M8333: Change CPU Module Ethernet Port 2 Network Settings Trigger**

When M8333 is turned on, the values stored in D8630 to D8650 are set as the IP Settings/DNS Settings of Ethernet port 2 on the CPU module.

The IP address is not set just by changing the values of D8630 to D8650. For details on changing the IP Settings/DNS Settings of Ethernet port 2 on the CPU module, see Chapter 3 "Network settings by special data registers" in the "FC6A Series MICROSmart Communication Manual".

■ **M8344: Ethernet Port 1 Send E-mail Server Settings Initialization**

When M8344 is turned on, the settings for the send E-mail server on the Ethernet port 1 are initialized.

■ **M8450: BACnet Communication Bit**

While M8450 is on, BACnet communication will be performed.

Special Data Register

Special Data Register Device Addresses

⚠ **Warning**	Do not write to data in the area marked as reserved in the special data registers list. Otherwise the system may not operate correctly.

Note: R/W is an abbreviation for read/write.
The notation for the R/W field is as follows.
R/W: The device can be both read from and written to
R: Read-only
W: Write-only

Device Address	Description		Update Timing	R/W
D8000	Quantity of Inputs		When I/O initialized	R
D8001	Quantity of Outputs		When I/O initialized	R
D8002	CPU Module Type Information		Power-up	R
D8003 D8004	**— Reserved —**		—	-
D8005	General Error Code		When error occurred	R/W
D8006	User Program Execution Error Code		When error occurred	R
D8007	**— Reserved —**		—	-
D8008	Calendar/Clock Current Data (Read only)	Year	Every 500 ms	R
D8009		Month	Every 500 ms	R
D8010		Day	Every 500 ms	R
D8011		Day of the Week	Every 500 ms	R
D8012		Hour	Every 500 ms	R
D8013		Minute	Every 500 ms	R
D8014		Second	Every 500 ms	R
D8015	Calendar/Clock New Data (Write only)	Year	—	W
D8016		Month	—	W
D8017		Day	—	W
D8018		Day of the Week	—	W
D8019		Hour	—	W
D8020		Minute	—	W
D8021		Second	—	W
D8022	Scan Time Data	Constant Scan Time Preset Value (1 to 1,000 ms)	—	W
D8023		Scan Time Current Value (ms)	Every scan	R
D8024		Scan Time Maximum Value (ms)	At occurrence	R
D8025		Scan Time Minimum Value (ms)	At occurrence	R
D8026	Communication Mode Information (Port 1 to Port 3)		Every scan	R
D8027 D8028	**— Reserved —**		—	-
D8029	System Software Version		Power-up	R
D8030	Communication Cartridge Information		Power-up	R
D8031	Optional Device Connection Information		Power-up	R
D8032	Interrupt Input Jump Destination Label No. (I1)		—	R/W
D8033	Interrupt Input Jump Destination Label No. (I3)		—	R/W
D8034	Interrupt Input Jump Destination Label No. (I4)		—	R/W
D8035	Interrupt Input Jump Destination Label No. (I6)		—	R/W
D8036	Timer Interrupt Jump Destination Label No.		—	R/W
D8037	Number of Connected I/O Modules		When I/O initialized	R
D8038 D8039	**— Reserved —**		—	—
D8040	Slave Number (Port 4)		—	R/W
D8041	Slave Number (Port 5)		—	R/W
D8042	Slave Number (Port 6)		—	R/W
D8043	Slave Number (Port 7)		—	R/W

Device Address	Description	Update Timing	R/W
D8044	Slave Number (Port 8)	—	R/W
D8045	Slave Number (Port 9)	—	R/W
D8046 to D8051	**— Reserved —**	—	—
D8052	J1939 Communication Error Code	Every scan	R/W
D8053 to D8055	**— Reserved —**	—	—
D8056	Battery Voltage	—	R
D8057	Analog Volume (AI0)	Every scan	R
D8058	Built-in Analog Input (AI1)	Every scan	R
D8059	Analog Input Status AI0	Every scan	R
D8060	Analog Input Status AI1	Every scan	R
D8061 to D8066	**— Reserved —**	—	—
D8067	Backlight ON Time	—	R/W
D8068	**— Reserved —**	—	—
D8069	Slave 1 Communication Status/Error (When Data Link Master Mode) Slave Communication Status/Error (When Data Link Slave Mode)	When error occurred	R
D8070	Slave 2 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8071	Slave 3 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8072	Slave 4 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8073	Slave 5 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8074	Slave 6 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8075	Slave 7 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8076	Slave 8 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8077	Slave 9 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8078	Slave 10 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8079	Slave 11 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8080	Slave 12 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8081	Slave 13 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8082	Slave 14 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8083	Slave 15 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8084	Slave 16 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8085	Slave 17 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8086	Slave 18 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8087	Slave 19 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8088	Slave 20 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8089	Slave 21 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8090	Slave 22 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8091	Slave 23 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8092	Slave 24 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8093	Slave 25 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8094	Slave 26 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8095	Slave 27 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8096	Slave 28 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8097	Slave 29 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8098	Slave 30 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8099	Slave 31 Communication Status/Error (When Data Link Master Mode)	When error occurred	R
D8100	Slave Number (Port 1)	—	R/W
D8101	**— Reserved —**	—	—
D8102	Slave Number (Port 2)	—	R/W
D8103	Slave Number (Port 3)	—	R/W
D8104	Control Signal Status (Port 1 to 5)	Every scan	R
D8105	RS232C DSR Input Control Signal Option (Port 1 to 5)	Every scan	R/W
D8106	RS232C DTR Output Control Signal Option (Port 1 to 5)	Every scan	R/W

Device Address	Description			Update Timing	R/W
D8107 to D8119	**— Reserved —**			**—**	**—**
D8120	HMI Module Information	Type ID/Status		—	R
D8121		System Software Version		—	R
D8122	Cartridge Slot 1 Information	Type ID/Status		—	R
D8123		System Software Version		—	R
D8124	Cartridge Slot 2 Information	Type ID/Status		—	R
D8125		System Software Version		—	R
D8126	Cartridge Slot 3 Information	Type ID/Status		—	R
D8127		System Software Version		—	R
D8128 to D8169	**— Reserved —**			**—**	**—**
D8170	Analog I/O Cartridge I/O AI2/AQ2			Every scan	R
D8171	Analog I/O Cartridge I/O AI3/AQ3			Every scan	R
D8172	Analog I/O Cartridge Status AI2/AQ2			Every scan	R
D8173	Analog I/O Cartridge Status AI3/AQ3			Every scan	R
D8174	Analog I/O Cartridge I/O AI4/AQ4			Every scan	R
D8175	Analog I/O Cartridge I/O AI5/AQ5			Every scan	R
D8176	Analog I/O Cartridge Status AI4/AQ4			Every scan	R
D8177	Analog I/O Cartridge Status AI5/AQ5			Every scan	R
D8178	Analog I/O Cartridge I/O AI6/AQ6			Every scan	R
D8179	Analog I/O Cartridge I/O AI7/AQ7			Every scan	R
D8180	Analog I/O Cartridge Status AI6/AQ6			Every scan	R
D8181	Analog I/O Cartridge Status AI7/AQ7			Every scan	R
D8182 to D8191	**— Reserved —**			**—**	**—**
D8192	High-speed Counter (Group 2/I1)	High Word	Current Value/Frequency Measurement (I1) Current Value	Every scan	R
D8193		Low Word		Every scan	R
D8194		High Word	Preset Value	—	R/W
D8195		Low Word		—	R/W
D8196		High Word	Reset Value	—	R/W
D8197		Low Word		—	R/W
D8198	High-speed Counter (Group 6/I7)	High Word	Current Value/Frequency Measurement (I7) Current Value	Every scan	R
D8199		Low Word		Every scan	R
D8200		High Word	Preset Value	—	R/W
D8201		Low Word		—	R/W
D8202		High Word	Reset Value	—	R/W
D8203		Low Word		—	R/W
D8204	Control Signal Status (Port 6 to 9)			Every scan	R
D8205	RS232C DSR Input Control Signal Option (Port 6 to 9)			Every scan	R/W
D8206	RS232C DTR Output Control Signal Option (Port 6 to 9)			Every scan	R/W
D8207 to D8209	**— Reserved —**			**—**	**—**
D8210	High-speed Counter (Group 1/I0)	High Word	Current Value/Frequency Measurement (I0) Current Value	Every scan	R
D8211		Low Word		Every scan	R
D8212		High Word	Preset Value	—	R/W
D8213		Low Word		—	R/W
D8214	Interrupt Input Jump Destination Label No. (I7)			—	R/W
D8215	Interrupt Input Jump Destination Label No. (I0)			—	R/W
D8216	High-speed Counter (Group 1/I0)	High Word	Reset Value	—	R/W
D8217		Low Word		—	R/W
D8218	High-speed Counter (Group 3/I3)	High Word	Current Value/Frequency Measurement (I3) Current Value	Every scan	R
D8219		Low Word		Every scan	R
D8220		High Word	Preset Value	—	R/W
D8221		Low Word		—	R/W

Device Address	Description			Update Timing	R/W
D8222	High-speed Counter (Group 4/I4)	High Word	Current Value/Frequency Measurement (I4) Current Value	Every scan	R
D8223		Low Word		Every scan	R
D8224		High Word	Preset Value	—	R/W
D8225		Low Word		—	R/W
D8226	High-speed Counter (Group 5/I6)	High Word	Current Value/Frequency Measurement (I6) Current Value	Every scan	R
D8227		Low Word		Every scan	R
D8228		High Word	Preset Value	—	R/W
D8229		Low Word		—	R/W
D8230 D8231	**— Reserved —**			**—**	**—**
D8232	High-speed Counter (Group 5/I6)	High Word	Reset Value	—	R/W
D8233		Low Word		—	R/W
D8234	High-speed Counter (Group 3/I3)	High Word	Reset Value	—	R/W
D8235		Low Word		—	R/W
D8236	High-speed Counter (Group 4/I4)	High Word	Reset Value	—	R/W
D8237		Low Word		—	R/W
D8238	**— Reserved —**			**—**	**—**
D8239	Absolute Position Control Status			Every scan	R
D8240	Absolute Position Counter 1	High Word	Absolute Position	Every scan	R
D8241		Low Word		Every scan	R
D8242	Absolute Position Counter 2	High Word	Absolute Position	Every scan	R
D8243		Low Word		Every scan	R
D8244	Absolute Position Counter 3	High Word	Absolute Position	Every scan	R
D8245		Low Word		Every scan	R
D8246	Absolute Position Counter 4	High Word	Absolute Position	Every scan	R
D8247		Low Word		Every scan	R
D8248 D8249	**— Reserved —**			**—**	**—**
D8250	Read SD Memory Card Capacity			Every scan	R
D8251	Read SD Memory Card Free Capacity			Every scan	R
D8252 D8253	**— Reserved —**			**—**	**—**
D8254	SD Memory Card Download/Upload Execution Information			When processing has completed	R
D8255	SD Memory Card Download/Upload Execution Status			When processing has completed	R
D8256 to D8359	**— Reserved —**			**—**	**—**
D8260	Recipe Block Number			—	R/W
D8261	Recipe Execution Block Number			When recipe execution has completed	R
D8262	Recipe Execution Channel No.			When recipe execution has completed	R
D8263	Recipe Execution Operation			When recipe execution has completed	R
D8264	Recipe Execution Status			When recipe execution has completed	R
D8265	Recipe Execution Error Information			When recipe execution has completed	R
D8266 D8267	**— Reserved —**			**—**	**—**
D8268	Remote Host Number of Connection 1 (1 to 255)			—	R/W
D8269	Remote Host Number of Connection 2 (1 to 255)			—	R/W
D8270	Remote Host Number of Connection 3 (1 to 255)			—	R/W

Device Address	Description		Update Timing	R/W
D8271	Remote Host Number of Connection 4 (1 to 255)		—	R/W
D8272	Remote Host Number of Connection 5 (1 to 255)		—	R/W
D8273	Remote Host Number of Connection 6 (1 to 255)		—	R/W
D8274	Remote Host Number of Connection 7 (1 to 255)		—	R/W
D8275	Remote Host Number of Connection 8 (1 to 255)		—	R/W
D8276 D8277	**— Reserved —**		—	—
D8278	Communication Mode Information (Client Connection)	Connection 1 to 4	—	R
D8279		Connection 5 to 8	—	R
D8280 to D8283	**— Reserved —**		—	—
D8284	Communication Mode Information (HMI Connection)	HMI Connection 1 to 4	—	R
D8285		HMI Connection 5 to 8	—	R
D8286 to D8302	**— Reserved —**		—	—
D8303	CPU Module Ethernet Port 1 IP Settings/DNS Settings Switching		—	R/W
D8304	CPU Module Ethernet Port 1 IP Address (Write-only)		—	W
D8305			—	W
D8306			—	W
D8307			—	W
D8308	CPU Module Ethernet Port 1 Subnet Mask (Write-only)		—	W
D8309			—	W
D8310			—	W
D8311			—	W
D8312	CPU Module Ethernet Port 1 Default Gateway (Write-only)		—	W
D8313			—	W
D8314			—	W
D8315			—	W
D8316	CPU Module Ethernet Port 1 Preferred DNS Server (Write-only)		—	W
D8317			—	W
D8318			—	W
D8319			—	W
D8320	CPU Module Ethernet Port 1 Alternate DNS Server (Write-only)		—	W
D8321			—	W
D8322			—	W
D8323			—	W
D8324	CPU Module Ethernet Port 1 MAC Address (Current Value Read-only)		Every 1 s	R
D8325			Every 1 s	R
D8326			Every 1 s	R
D8327			Every 1 s	R
D8328			Every 1 s	R
D8329			Every 1 s	R
D8330	CPU Module Ethernet Port 1 IP Address (Current Value Read-only)		Every 1 s	R
D8331			Every 1 s	R
D8332			Every 1 s	R
D8333			Every 1 s	R
D8334	CPU Module Ethernet Port 1 Subnet Mask (Current Value Read-only)		Every 1 s	R
D8335			Every 1 s	R
D8336			Every 1 s	R
D8337			Every 1 s	R
D8338	CPU Module Ethernet Port 1 Default Gateway (Current Value Read-only)		Every 1 s	R
D8339			Every 1 s	R
D8340			Every 1 s	R
D8341			Every 1 s	R

Device Address	Description	Update Timing	R/W
D8342	CPU Module Ethernet Port 1 Preferred DNS Server (Current Value Read-only)	Every 1 s	R
D8343		Every 1 s	R
D8344		Every 1 s	R
D8345		Every 1 s	R
D8346	CPU Module Ethernet Port 1 Alternate DNS Server (Current Value Read-only)	Every 1 s	R
D8347		Every 1 s	R
D8348		Every 1 s	R
D8349		Every 1 s	R
D8350	Connection 1 Connected IP Address	Every 1 s	R
D8351		Every 1 s	R
D8352		Every 1 s	R
D8353		Every 1 s	R
D8354	Connection 2 Connected IP Address	Every 1 s	R
D8355		Every 1 s	R
D8356		Every 1 s	R
D8357		Every 1 s	R
D8358	Connection 3 Connected IP Address	Every 1 s	R
D8359		Every 1 s	R
D8360		Every 1 s	R
D8361		Every 1 s	R
D8362	Connection 4 Connected IP Address	Every 1 s	R
D8363		Every 1 s	R
D8364		Every 1 s	R
D8365		Every 1 s	R
D8366	Connection 5 Connected IP Address	Every 1 s	R
D8367		Every 1 s	R
D8368		Every 1 s	R
D8369		Every 1 s	R
D8370	Connection 6 Connected IP Address	Every 1 s	R
D8371		Every 1 s	R
D8372		Every 1 s	R
D8373		Every 1 s	R
D8374	Connection 7 Connected IP Address	Every 1 s	R
D8375		Every 1 s	R
D8376		Every 1 s	R
D8377		Every 1 s	R
D8378	Connection 8 Connected IP Address	Every 1 s	R
D8379		Every 1 s	R
D8380		Every 1 s	R
D8381		Every 1 s	R
D8382	HMI Module MAC Address (Current Value Read-only)	Every 1 s	R
D8383		Every 1 s	R
D8384		Every 1 s	R
D8385		Every 1 s	R
D8386		Every 1 s	R
D8387		Every 1 s	R
D8388	HMI Module IP Address (Current Value Read-only)	Every 1 s	R
D8389		Every 1 s	R
D8390		Every 1 s	R
D8391		Every 1 s	R
D8392	HMI Module Subnet Mask (Current Value Read-only)	Every 1 s	R
D8393		Every 1 s	R
D8394		Every 1 s	R
D8395		Every 1 s	R

Device Address	Description		Update Timing	R/W
D8396	HMI Module Default Gateway (Current Value Read-only)		Every 1 s	R
D8397			Every 1 s	R
D8398			Every 1 s	R
D8399			Every 1 s	R
D8400	HMI Module Preferred DNS Server (Current Value Read-only)		Every 1 s	R
D8401			Every 1 s	R
D8402			Every 1 s	R
D8403			Every 1 s	R
D8404	HMI Module Alternate DNS Server (Current Value Read-only)		Every 1 s	R
D8405			Every 1 s	R
D8406			Every 1 s	R
D8407			Every 1 s	R
D8408 to D8412	**— Reserved —**		**—**	**—**
D8413	Time Zone Offset		—	R/W
D8414	SNTP Operation Status		—	R
D8415	SNTP Access Elapsed Time		—	R
D8416 to D8428	**— Reserved —**		**—**	**—**
D8429	HMI Module Connection Information Reference	Connection No.	—	R/W
D8430		Connected IP Address	Every 1 s	R
D8431			Every 1 s	R
D8432			Every 1 s	R
D8433			Every 1 s	R
D8434		Connected Port No.	Every 1 s	R
D8435 D8436	**— Reserved —**		**—**	**—**
D8437	HMI Module IP Address (Write-only)		—	W
D8438			—	W
D8439			—	W
D8440			—	W
D8441	HMI Module Subnet Mask (Write-only)		—	W
D8442			—	W
D8443			—	W
D8444			—	W
D8445	HMI Module Default Gateway (Write-only)		—	W
D8446			—	W
D8447			—	W
D8448			—	W
D8449	HMI Module Preferred DNS Server (Write-only)		—	W
D8450			—	W
D8451			—	W
D8452			—	W
D8453	HMI Module Alternate DNS Server (Write-only)		—	W
D8454			—	W
D8455			—	W
D8456			—	W
D8457	EMAIL Instruction Detailed Error Information (HMI-Ethernet Port)		—	R
D8458 to D8469	**— Reserved —**		**—**	**—**
D8470	Expansion Module Slot 1 Information	Type ID/Status	—	R
D8471		System Software Version/Position Information	—	R
D8472	Expansion Module Slot 2 Information	Type ID/Status	—	R
D8473		System Software Version/Position Information	—	R

Device Address	Description		Update Timing	R/W
D8474	Expansion Module Slot 3 Information	Type ID/Status	—	R
D8475		System Software Version/Position Information	—	R
D8476	Expansion Module Slot 4 Information	Type ID/Status	—	R
D8477		System Software Version/Position Information	—	R
D8478	Expansion Module Slot 5 Information	Type ID/Status	—	R
D8479		System Software Version/Position Information	—	R
D8480	Expansion Module Slot 6 Information	Type ID/Status	—	R
D8481		System Software Version/Position Information	—	R
D8482	Expansion Module Slot 7 Information	Type ID/Status	—	R
D8483		System Software Version/Position Information	—	R
D8484	Expansion Module Slot 8 Information	Type ID/Status	—	R
D8485		System Software Version/Position Information	—	R
D8486	Expansion Module Slot 9 Information	Type ID/Status	—	R
D8487		System Software Version/Position Information	—	R
D8488	Expansion Module Slot 10 Information	Type ID/Status	—	R
D8489		System Software Version/Position Information	—	R
D8490	Expansion Module Slot 11 Information	Type ID/Status	—	R
D8491		System Software Version/Position Information	—	R
D8492	Expansion Module Slot 12 Information	Type ID/Status	—	R
D8493		System Software Version/Position Information	—	R
D8494	Expansion Module Slot 13 Information	Type ID/Status	—	R
D8495		System Software Version/Position Information	—	R
D8496	Expansion Module Slot 14 Information	Type ID/Status	—	R
D8497		System Software Version/Position Information	—	R
D8498	Expansion Module Slot 15 Information	Type ID/Status	—	R
D8499		System Software Version/Position Information	—	R
D8500	Expansion Module Slot 16 Information	Type ID/Status	—	R
D8501		System Software Version/Position Information	—	R
D8502	Expansion Module Slot 17 Information	Type ID/Status	—	R
D8503		System Software Version/Position Information	—	R
D8504	Expansion Module Slot 18 Information	Type ID/Status	—	R
D8505		System Software Version/Position Information	—	R
D8506	Expansion Module Slot 19 Information	Type ID/Status	—	R
D8507		System Software Version/Position Information	—	R
D8508	Expansion Module Slot 20 Information	Type ID/Status	—	R
D8509		System Software Version/Position Information	—	R
D8510	Expansion Module Slot 21 Information	Type ID/Status	—	R
D8511		System Software Version/Position Information	—	R
D8512	Expansion Module Slot 22 Information	Type ID/Status	—	R
D8513		System Software Version/Position Information	—	R
D8514	Expansion Module Slot 23 Information	Type ID/Status	—	R
D8515		System Software Version/Position Information	—	R
D8516	Expansion Module Slot 24 Information	Type ID/Status	—	R
D8517		System Software Version/Position Information	—	R
D8518	Expansion Module Slot 25 Information	Type ID/Status	—	R
D8519		System Software Version/Position Information	—	R
D8520	Expansion Module Slot 26 Information	Type ID/Status	—	R
D8521		System Software Version/Position Information	—	R
D8522	Expansion Module Slot 27 Information	Type ID/Status	—	R
D8523		System Software Version/Position Information	—	R
D8524	Expansion Module Slot 28 Information	Type ID/Status	—	R
D8525		System Software Version/Position Information	—	R
D8526	Expansion Module Slot 29 Information	Type ID/Status	—	R
D8527		System Software Version/Position Information	—	R

Device Address	Description		Update Timing	R/W
D8528	Expansion Module Slot 30 Information	Type ID/Status	—	R
D8529		System Software Version/Position Information	—	R
D8530	Expansion Module Slot 31 Information	Type ID/Status	—	R
D8531		System Software Version/Position Information	—	R
D8532	Expansion Module Slot 32 Information	Type ID/Status	—	R
D8533		System Software Version/Position Information	—	R
D8534	Expansion Module Slot 33 Information	Type ID/Status	—	R
D8535		System Software Version/Position Information	—	R
D8536	Expansion Module Slot 34 Information	Type ID/Status	—	R
D8537		System Software Version/Position Information	—	R
D8538	Expansion Module Slot 35 Information	Type ID/Status	—	R
D8539		System Software Version/Position Information	—	R
D8530	Expansion Module Slot 36 Information	Type ID/Status	—	R
D8541		System Software Version/Position Information	—	R
D8542	Expansion Module Slot 37 Information	Type ID/Status	—	R
D8543		System Software Version/Position Information	—	R
D8544	Expansion Module Slot 38 Information	Type ID/Status	—	R
D8545		System Software Version/Position Information	—	R
D8546	Expansion Module Slot 39 Information	Type ID/Status	—	R
D8547		System Software Version/Position Information	—	R
D8548	Expansion Module Slot 40 Information	Type ID/Status	—	R
D8549		System Software Version/Position Information	—	R
D8550	Expansion Module Slot 41 Information	Type ID/Status	—	R
D8551		System Software Version/Position Information	—	R
D8552	Expansion Module Slot 42 Information	Type ID/Status	—	R
D8553		System Software Version/Position Information	—	R
D8554	Expansion Module Slot 43 Information	Type ID/Status	—	R
D8555		System Software Version/Position Information	—	R
D8556	Expansion Module Slot 44 Information	Type ID/Status	—	R
D8557		System Software Version/Position Information	—	R
D8558	Expansion Module Slot 45 Information	Type ID/Status	—	R
D8559		System Software Version/Position Information	—	R
D8560	Expansion Module Slot 46 Information	Type ID/Status	—	R
D8561		System Software Version/Position Information	—	R
D8562	Expansion Module Slot 47 Information	Type ID/Status	—	R
D8563		System Software Version/Position Information	—	R
D8564	Expansion Module Slot 48 Information	Type ID/Status	—	R
D8565		System Software Version/Position Information	—	R
D8566	Expansion Module Slot 49 Information	Type ID/Status	—	R
D8567		System Software Version/Position Information	—	R
D8568	Expansion Module Slot 50 Information	Type ID/Status	—	R
D8569		System Software Version/Position Information	—	R
D8570	Expansion Module Slot 51 Information	Type ID/Status	—	R
D8571		System Software Version/Position Information	—	R
D8572	Expansion Module Slot 52 Information	Type ID/Status	—	R
D8573		System Software Version/Position Information	—	R
D8574	Expansion Module Slot 53 Information	Type ID/Status	—	R
D8575		System Software Version/Position Information	—	R
D8576	Expansion Module Slot 54 Information	Type ID/Status	—	R
D8577		System Software Version/Position Information	—	R
D8578	Expansion Module Slot 55 Information	Type ID/Status	—	R
D8579		System Software Version/Position Information	—	R
D8580	Expansion Module Slot 56 Information	Type ID/Status	—	R
D8581		System Software Version/Position Information	—	R

Device Address	Description		Update Timing	R/W
D8582	Expansion Module Slot 57 Information	Type ID/Status	—	R
D8583		System Software Version/Position Information	—	R
D8584	Expansion Module Slot 58 Information	Type ID/Status	—	R
D8585		System Software Version/Position Information	—	R
D8586	Expansion Module Slot 59 Information	Type ID/Status	—	R
D8587		System Software Version/Position Information	—	R
D8588	Expansion Module Slot 60 Information	Type ID/Status	—	R
D8589		System Software Version/Position Information	—	R
D8590	Expansion Module Slot 61 Information	Type ID/Status	—	R
D8591		System Software Version/Position Information	—	R
D8592	Expansion Module Slot 62 Information	Type ID/Status	—	R
D8593		System Software Version/Position Information	—	R
D8594	Expansion Module Slot 63 Information	Type ID/Status	—	R
D8595		System Software Version/Position Information	—	R
D8596	Expansion Interface Remote Master Module Slot Information	Type ID/Status	—	R
D8597		System Software Version/ Number of Connected Expansion Interface Remote Slave Modules	—	R
D8598	Expansion Interface Remote Slave Module (Node 1) Slot Information	Type ID/Status	—	R
D8599		System Software Version/ Expansion Module Connection Information	—	R
D8600	Expansion Interface Remote Slave Module (Node 2) Slot Information	Type ID/Status	—	R
D8601		System Software Version/ Expansion Module Connection Information	—	R
D8602	Expansion Interface Remote Slave Module (Node 3) Slot Information	Type ID/Status	—	R
D8603		System Software Version/ Expansion Module Connection Information	—	R
D8604	Expansion Interface Remote Slave Module (Node 4) Slot Information	Type ID/Status	—	R
D8605		System Software Version/ Expansion Module Connection Information	—	R
D8606	Expansion Interface Remote Slave Module (Node 5) Slot Information	Type ID/Status	—	R
D8607		System Software Version/ Expansion Module Connection Information	—	R
D8608	Expansion Interface Remote Slave Module (Node 6) Slot Information	Type ID/Status	—	R
D8609		System Software Version/ Expansion Module Connection Information	—	R
D8610	Expansion Interface Remote Slave Module (Node 7) Slot Information	Type ID/Status	—	R
D8611		System Software Version/ Expansion Module Connection Information	—	R
D8612	Expansion Interface Remote Slave Module (Node 8) Slot Information	Type ID/Status	—	R
D8613		System Software Version/ Expansion Module Connection Information	—	R
D8614	Expansion Interface Remote Slave Module (Node 9) Slot Information	Type ID/Status	—	R
D8615		System Software Version/ Expansion Module Connection Information	—	R
D8616	Expansion Interface Remote Slave Module (Node 10) Slot Information	Type ID/Status	—	R
D8617		System Software Version/ Expansion Module Connection Information	—	R
D8618	Refresh Time of Expansion Interface Remote Master/Slave Modules Current Value		Every scan	R
D8619	Refresh Time of Expansion Interface Remote Master/Slave Modules Maximum Value		Every scan	R
D8620 to D8629	— **Reserved** —		—	—
D8630	CPU Module Ethernet Port 2 IP Settings/DNS Settings Switching		—	R/W
D8631	CPU Module Ethernet Port 2 IP Address (Write-only)		—	W
D8632			—	W
D8633			—	W
D8634			—	W

Device Address	Description	Update Timing	R/W
D8635	CPU Module Ethernet Port 2 Subnet Mask (Write-only)	—	W
D8636		—	W
D8637		—	W
D8638		—	W
D8639	CPU Module Ethernet Port 2 Default Gateway (Write-only)	—	W
D8640		—	W
D8641		—	W
D8642		—	W
D8643	CPU Module Ethernet Port 2 Preferred DNS Server (Write-only)	—	W
D8644		—	W
D8645		—	W
D8646		—	W
D8647	CPU Module Ethernet Port 2 Alternate DNS Server (Write-only)	—	W
D8648		—	W
D8649		—	W
D8650		—	W
D8651	CPU Module Ethernet Port 2 MAC Address (Current Value Read-only)	Every 1 s	R
D8652		Every 1 s	R
D8653		Every 1 s	R
D8654		Every 1 s	R
D8655		Every 1 s	R
D8656		Every 1 s	R
D8657	CPU Module Ethernet Port 2 IP Address (Current Value Read-only)	Every 1 s	R
D8658		Every 1 s	R
D8659		Every 1 s	R
D8660		Every 1 s	R
D8661	CPU Module Ethernet Port 2 Subnet Mask (Current Value Read-only)	Every 1 s	R
D8662		Every 1 s	R
D8663		Every 1 s	R
D8664		Every 1 s	R
D8665	CPU Module Ethernet Port 2 Default Gateway (Current Value Read-only)	Every 1 s	R
D8666		Every 1 s	R
D8667		Every 1 s	R
D8668		Every 1 s	R
D8669	CPU Module Ethernet Port 2 Preferred DNS Server (Current Value Read-only)	Every 1 s	R
D8670		Every 1 s	R
D8671		Every 1 s	R
D8672		Every 1 s	R
D8673	CPU Module Ethernet Port 2 Alternate DNS Server (Current Value Read-only)	Every 1 s	R
D8674		Every 1 s	R
D8675		Every 1 s	R
D8676		Every 1 s	R
D8677	Connection 9 Connected IP Address	Every 1 s	R
D8678		Every 1 s	R
D8679		Every 1 s	R
D8680		Every 1 s	R
D8681	Connection 10 Connected IP Address	Every 1 s	R
D8682		Every 1 s	R
D8683		Every 1 s	R
D8684		Every 1 s	R
D8685	Connection 11 Connected IP Address	Every 1 s	R
D8686		Every 1 s	R
D8687		Every 1 s	R
D8688		Every 1 s	R

Device Address	Description	Update Timing	R/W
D8689	Connection 12 Connected IP Address	Every 1 s	R
D8690		Every 1 s	R
D8691		Every 1 s	R
D8692		Every 1 s	R
D8693	Connection 13 Connected IP Address	Every 1 s	R
D8694		Every 1 s	R
D8695		Every 1 s	R
D8696		Every 1 s	R
D8697	Connection 14 Connected IP Address	Every 1 s	R
D8698		Every 1 s	R
D8699		Every 1 s	R
D8700		Every 1 s	R
D8701	Connection 15 Connected IP Address	Every 1 s	R
D8702		Every 1 s	R
D8703		Every 1 s	R
D8704		Every 1 s	R
D8705	Connection 16 Connected IP Address	Every 1 s	R
D8706		Every 1 s	R
D8707		Every 1 s	R
D8708		Every 1 s	R
D8709 to D8716	**— Reserved —**	—	—
D8717	Control Signal Status (Port 10 to 13)	Every scan	R
D8718	RS232C DSR Input Control Signal Option (Port 10 to 13)	Every scan	R/W
D8719	RS232C DTR Output Control Signal Option (Port 10 to 13)	Every scan	R/W
D8720	Control Signal Status (Port 14 to 17)	Every scan	R
D8721	RS232C DSR Input Control Signal Option (Port 14 to 17)	Every scan	R/W
D8722	RS232C DTR Output Control Signal Option (Port 14 to 17)	Every scan	R/W
D8723	Control Signal Status (Port 18 to 21)	Every scan	R
D8724	RS232C DSR Input Control Signal Option (Port 18 to 21)	Every scan	R/W
D8725	RS232C DTR Output Control Signal Option (Port 18 to 21)	Every scan	R/W
D8726	Control Signal Status (Port 22 to 25)	Every scan	R
D8727	RS232C DSR Input Control Signal Option (Port 22 to 25)	Every scan	R/W
D8728	RS232C DTR Output Control Signal Option (Port 22 to 25)	Every scan	R/W
D8729	Control Signal Status (Port 26 to 29)	Every scan	R
D8730	RS232C DSR Input Control Signal Option (Port 26 to 29)	Every scan	R/W
D8731	RS232C DTR Output Control Signal Option (Port 26 to 29)	Every scan	R/W
D8732	Control Signal Status (Port 30 to 33)	Every scan	R
D8733	RS232C DSR Input Control Signal Option (Port 30 to 33)	Every scan	R/W
D8734	RS232C DTR Output Control Signal Option (Port 30 to 33)	Every scan	R/W
D8735	Slave Number (Port 10)	—	R/W
D8736	Slave Number (Port 11)	—	R/W
D8737	Slave Number (Port 12)	—	R/W
D8738	Slave Number (Port 13)	—	R/W
D8739	Slave Number (Port 14)	—	R/W
D8740	Slave Number (Port 15)	—	R/W
D8741	Slave Number (Port 16)	—	R/W
D8742	Slave Number (Port 17)	—	R/W
D8743	Slave Number (Port 18)	—	R/W
D8744	Slave Number (Port 19)	—	R/W
D8745	Slave Number (Port 20)	—	R/W
D8746	Slave Number (Port 21)	—	R/W
D8747	Slave Number (Port 22)	—	R/W
D8748	Slave Number (Port 23)	—	R/W
D8749	Slave Number (Port 24)	—	R/W

Device Address	Description		Update Timing	R/W
D8750	Slave Number (Port 25)		—	R/W
D8751	Slave Number (Port 26)		—	R/W
D8752	Slave Number (Port 27)		—	R/W
D8753	Slave Number (Port 28)		—	R/W
D8754	Slave Number (Port 29)		—	R/W
D8755	Slave Number (Port 30)		—	R/W
D8756	Slave Number (Port 31)		—	R/W
D8757	Slave Number (Port 32)		—	R/W
D8758	Slave Number (Port 33)		—	R/W
D8759	EMAIL Instruction Detailed Error Information (Ethernet Port 1)		—	R
D8760	Communication Mode Information (Client Connection)	Connection 9 to 12	—	R
D8761		Connection 13 to 16	—	R
D8762 to D8773	**— Reserved —**		**—**	**—**
D8774	Remote Host Number of Connection 9 (1 to 255)		—	R/W
D8775	Remote Host Number of Connection 10 (1 to 255)		—	R/W
D8776	Remote Host Number of Connection 11 (1 to 255)		—	R/W
D8777	Remote Host Number of Connection 12 (1 to 255)		—	R/W
D8778	Remote Host Number of Connection 13 (1 to 255)		—	R/W
D8779	Remote Host Number of Connection 14 (1 to 255)		—	R/W
D8780	Remote Host Number of Connection 15 (1 to 255)		—	R/W
D8781	Remote Host Number of Connection 16 (1 to 255)		—	R/W
D8782	BACnet Operation Status		—	R
D8783	BACnet Error Information		—	R
D8784 to D8899	**— Reserved —**		**—**	**—**

Supplementary Descriptions of the Special Data Registers

■ D8000: Quantity of Inputs

The number of FC6A Series MICROSmart inputs is written to this register.
The total number of inputs for the CPU module and the expansion modules is written to this register.

■ D8001: Quantity of Outputs

The number of FC6A Series MICROSmart outputs is written to this register.
The total number of outputs for the CPU module and the expansion modules is written to this register.

■ D8002: CPU Module Type Information

CPU module type information is written this register.
0 (00h): 16-I/O type
1 (01h): 24-I/O type
2 (02h): 40-I/O type (All-in-One CPU module)
18 (12h): 40-I/O type (CAN J1939 All-in-One CPU module)
32 (20h): Plus 16-I/O type
33 (21h): Plus 32-I/O type

■ D8005: General Error Code

FC6A Series MICROSmart general error information is written to this register. When a general error occurs, the bit corresponding to the error occurred turns on. The general error and user program execution error can be cleared by writing "1" to the most significant bit of D8005. For details on general error codes, see Chapter 13 "General Error Codes" in the "FC6A Series MICROSmart User's Manual".

■ D8006: User Program Execution Error Code

FC6A Series MICROSmartuser program execution error information is written to this register. When a user program execution error occurs, the error code corresponding to the error that occurred is written to this register. For details on user program execution errors, see "User Program Execution Error" on page A-12.

■ D8008 to D8021: Calendar/Clock Data

D8008 through D8021 are used for reading calendar/clock data from the internal clock and for writing calendar/clock data to the internal clock. For details on the calendar/clock data, see Chapter 5 "Clock Function" in the "FC6A Series MICROSmart User's Manual".

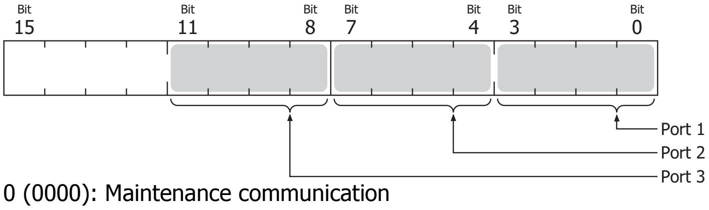
■ D8022 to D8025: Scan Time Data

D8022 through D8025 are special data registers for checking the scan time and configuring the constant scan time. For details on the scan time, see Chapter 5 "Constant Scan Time" in the "FC6A Series MICROSmart User's Manual".

■ D8026: Communication Mode Information (Port 1 to 3)

This register indicates communication mode information for Port 1 to Port 3.

The allocation of communication ports in the device (bit assignment) is as follows.



0 (0000): Maintenance communication
1 (0001): User communication
2 (0010): Modbus RTU master
3 (0011): Modbus RTU slave
4 (0100): Data link communication

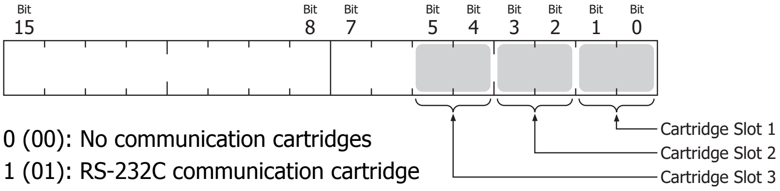
■ D8029: System Software Version

The CPU module system software version number is written to this register.

■ D8030: Communication Cartridge Information

This register indicates information about the communication cartridges in Port 2 and Port 3.

The allocation of communication ports in the device (bit assignment) is as follows.

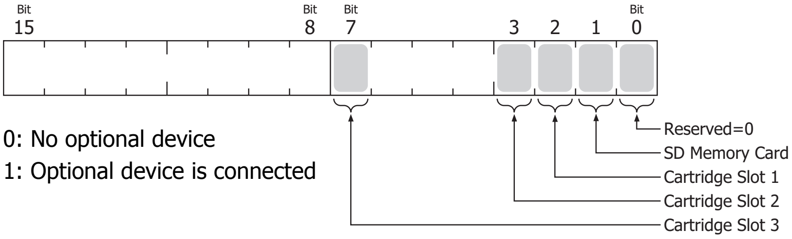


0 (00): No communication cartridges
1 (01): RS-232C communication cartridge
2 (10): RS-485 communication cartridge
3 (11): Bluetooth communication cartridge

■ D8031: Optional Device Connection Information

Optional device connection information is written to this register.

The allocation of bits in the device (bit assignment) is as follows.



0: No optional device
1: Optional device is connected

■ D8032 to D8035, D8214, D8215: Interrupt Input Jump Destination Label No.

Jump destination label numbers for interrupt inputs are written to these registers. When using interrupt inputs, write the label number that corresponds to the special data register that has been allocated to the interrupt input. For details on interrupt inputs, see Chapter 5 "Interrupt Input" in the "FC6A Series MICROSmart User's Manual".

D8032 = I1
D8033 = I3
D8034 = I4
D8035 = I6
D8214 = I7
D8215 = I0

■ D8036: Timer Interrupt Jump Destination Label No.

The jump destination label number when the timer interrupt occurs is written to this register. When using the timer interrupt, store the corresponding label number.

For details on the timer interrupt, see Chapter 5 "Timer Interrupt" in the "FC6A Series MICROSmart User's Manual".

■ D8037: Number of Connected Expansion Modules

The number of expansion modules connected to the CPU module (I/O modules, PID modules, and communication modules) is written to this register.

■ D8052: J1939 Communication Error Code

When an error occurs in J1939 communication, the error code is written to this register. For details on J1939 communication error codes, see Chapter 8 "J1939 Communication Error Code (D8052)" in the "FC6A Series MICROSmart Communication Manual".

■ D8056: Battery Voltage

The measurement result of the backup battery voltage is written to this register in 1 mV units.
0: Measurement error

■ D8057: Analog Volume (AI0)

The analog volume value is converted to a digital value and written to this register. For details, see Chapter 5 "Analog Potentiometer" in the "FC6A Series MICROSmart User’s Manual".

■ D8058: Built-in Analog Input (AI1)

The analog input's analog value (0 to 10V DC) is converted to a digital value and written to this register. For details, see Chapter 5 "Analog Voltage Input" in the "FC6A Series MICROSmart User’s Manual".

■ D8059: Analog Input Status AI0

The analog volume's analog input status is written to this register. For details, see Chapter 5 "Analog Potentiometer" in the "FC6A Series MICROSmart User’s Manual".

■ D8060: Analog Input Status AI1

The analog input's analog input status is written to this register. For details, see Chapter 5 "Analog Voltage Input" in the "FC6A Series MICROSmart User’s Manual".

■ D8067: Backlight ON Time

The backlight on time for the HMI module is written to this register. The backlight on time can be set by modifying the value of D8067 in a range between 1 and 65,535 s. When D8067 is 0 s, the backlight is always on. The backlight on time can also be changed with the HMI environment settings. For details, see Chapter 7 "Setting the LCD Backlight ON Time" in the "FC6A Series MICROSmart User’s Manual".

■ D8069 to D8099: Slave (1 to 31) Communication Status/Error

Special data registers used in data link communication. For details, see Chapter 7 "Master Station" in the "FC6A Series MICROSmart Communication Manual".

■ D8040 to D8045, D8100, D8102, D8103, D8735 to D8758: Slave Number (Port 1 to 33)

The slave number is written to these registers when the Port 1 to 33 communication mode is maintenance communication, Modbus RTU slave, or data link communication. The slave number can be changed by changing the value of the corresponding device in **Function Area Settings**.

D8100 = Port 1 Slave No.
D8102 = Port 2 Slave No.
D8103 = Port 3 Slave No.

D8040 = Port 4 Slave No.
D8041 = Port 5 Slave No.
D8042 = Port 6 Slave No.
D8043 = Port 7 Slave No.
D8044 = Port 8 Slave No.
D8045 = Port 9 Slave No.
D8735 = Port 10 Slave No.
D8736 = Port 11 Slave No.
D8737 = Port 12 Slave No.
D8738 = Port 13 Slave No.
D8739 = Port 14 Slave No.
D8740 = Port 15 Slave No.
D8741 = Port 16 Slave No.

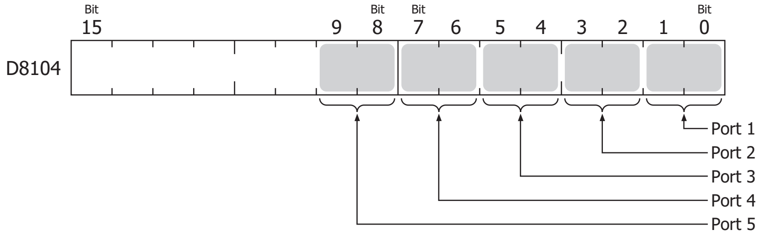
D8742 = Port 17 Slave No.
D8743 = Port 18 Slave No.
D8744 = Port 19 Slave No.
D8745 = Port 20 Slave No.
D8746 = Port 21 Slave No.
D8747 = Port 22 Slave No.
D8748 = Port 23 Slave No.
D8749 = Port 24 Slave No.
D8750 = Port 25 Slave No.
D8751 = Port 26 Slave No.
D8752 = Port 27 Slave No.
D8753 = Port 28 Slave No.
D8754 = Port 29 Slave No.
D8755 = Port 30 Slave No.
D8756 = Port 31 Slave No.
D8757 = Port 32 Slave No.
D8758 = Port 33 Slave No.

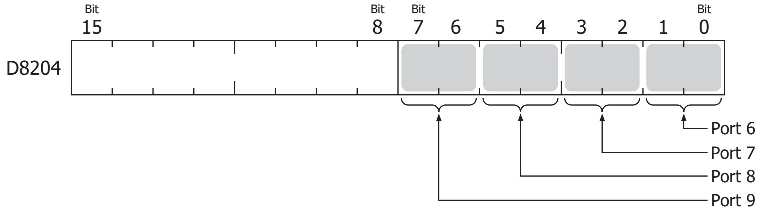
For details on communication modes, see the following in the "FC6A Series MICROSmart Communication Manual".

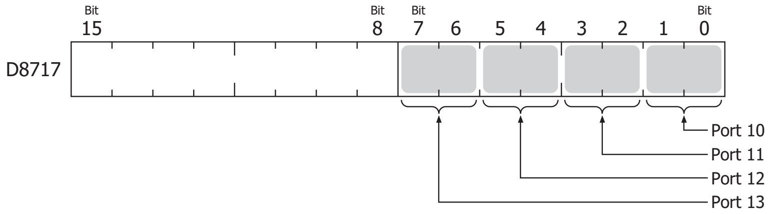
- Maintenance communication: Chapter 4 "Maintenance Communication"
- Modbus RTU slave: Chapter 6 "Modbus Communication"
- Data link communication: Chapter 7 "Data Link Communication"

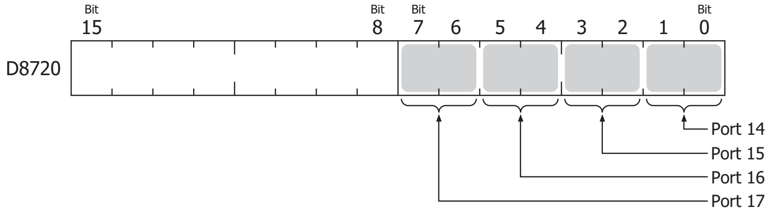
■ D8104, D8204, D8717, D8720, D8723, D8726, D8729, D8732: Control Signal Status (Port 1 to 33)

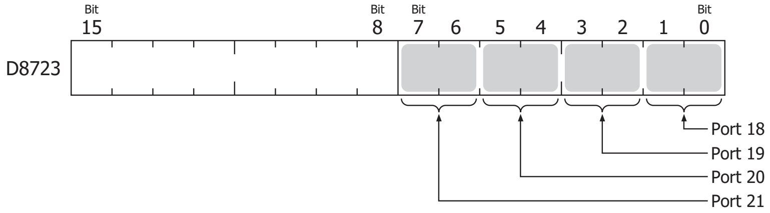
The signal statuses of the DSR and DTR controls lines are written to this register. This register is updated in END processing when stopped and while running. The allocation of communication ports in the device (bit assignment) is as follows.

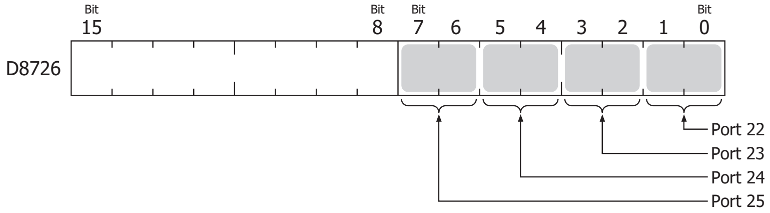


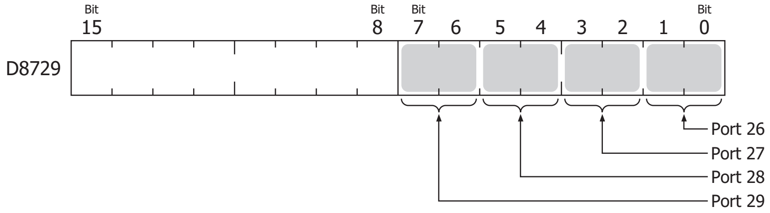


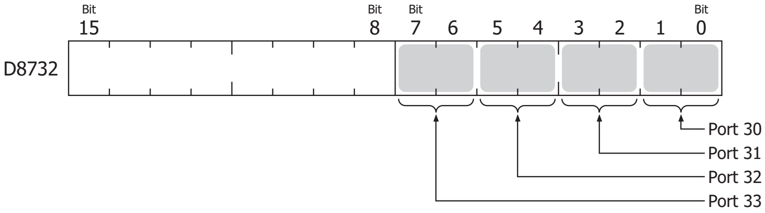








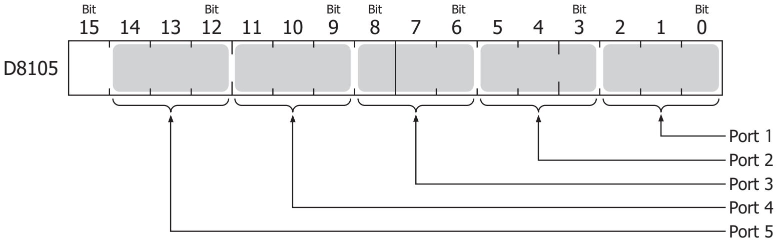


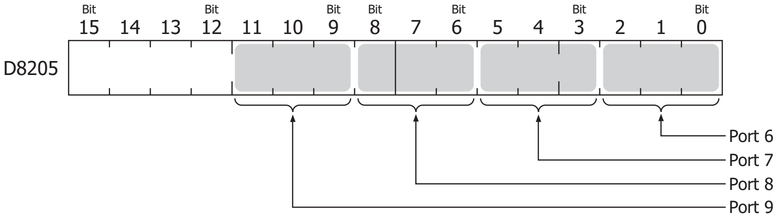


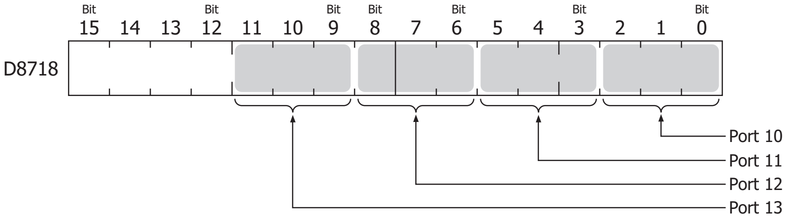
0 (00): DTR and DSR are both off.
1 (01): DTR is off and DSR is on.
2 (10): DTR is on and DSR is off.
3 (11): DTR and DSR are both on.

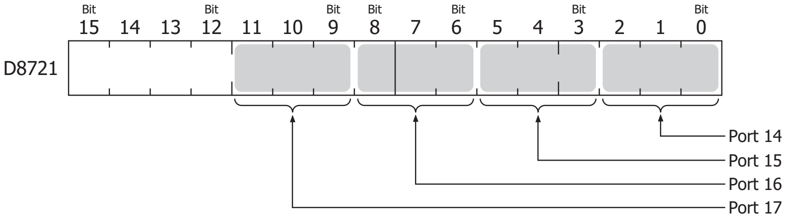
■ D8105, D8205, D8718, D8721, D8724, D8727, D8730, D8733: RS232C DSR Input Control Signal Option (Port 1 to 33)

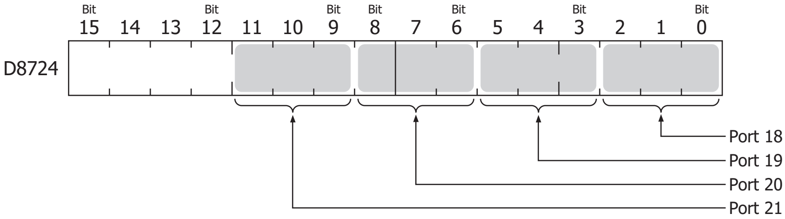
The signal statuses of the DSR and DTR controls lines are written to this register. This register is updated in END processing when stopped and while running. The allocation of communication ports in the device (bit assignment) is as follows.

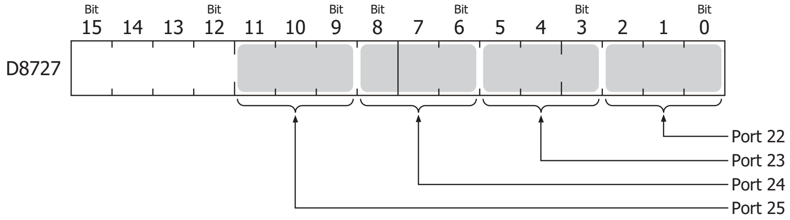


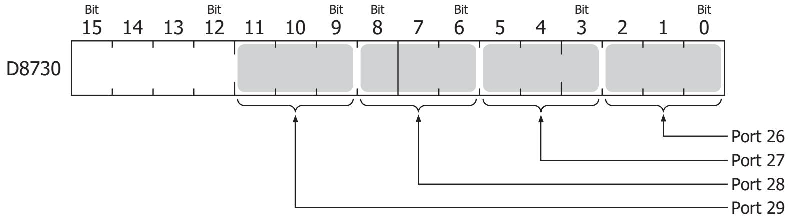


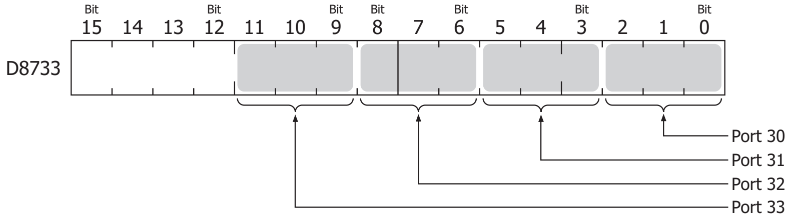






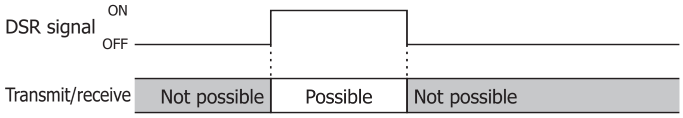




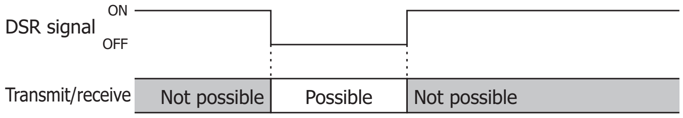


0 (000): The DSR signal status is not used for FC6A Series MICROSmart transmission control. Use this status when DSR signal control is not required.

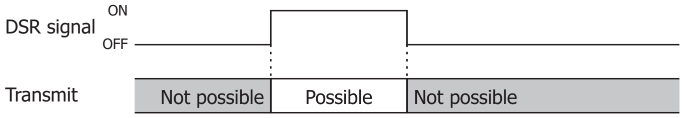
1 (001): When the DSR signal is on, the FC6A Series MICROSmart can transmit and receive.



2 (010): When the DSR signal is off, the FC6A Series MICROSmart can transmit and receive.



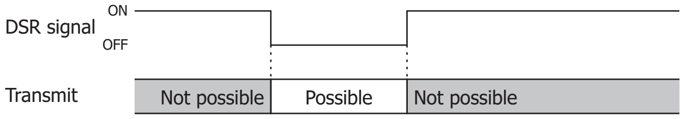
3 (011): When the DSR signal is on, transmission is possible (reception is always possible).



This is normally called "busy control" and is used for transmission control for devices with a slow processing speed such as printers.

(As viewed from the connected device, there is a limit on the data that can be input.)

4 (100): When the DSR signal is off, transmission is possible.

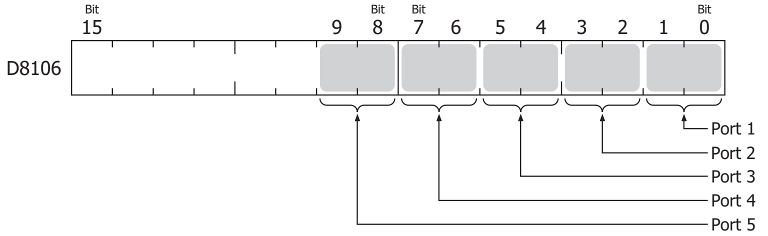


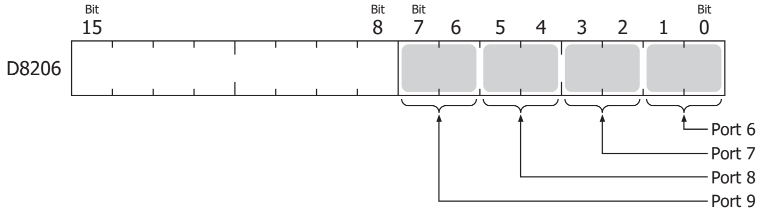
5 or higher: The operation is the same as the setting "000".

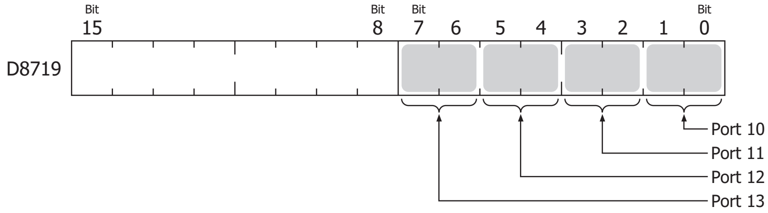
■ D8106, D8206, D8719, D8722, D8725, D8728, D8731, D8734: RS232C DTR Output Control Signal Option (Port 1 to 33)

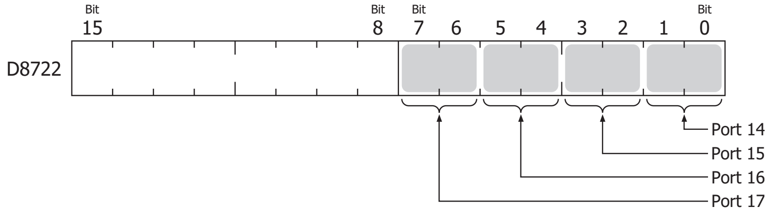
This register is used when indicating the FC6A Series MICROSmart control status and the transmit/receive status to the connected device. This control line is an output signal from the FC6A Series MICROSmart to the connected device. This register is only valid during user communication.

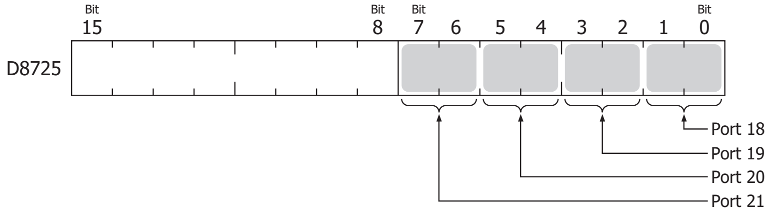
The allocation of communication ports in the device (bit assignment) is as follows.

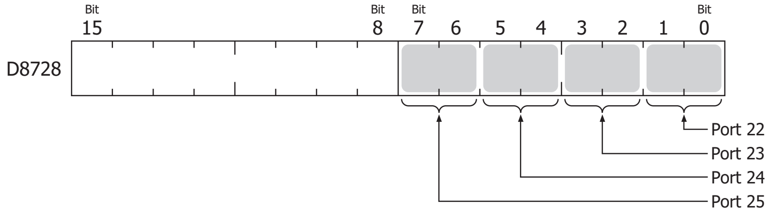


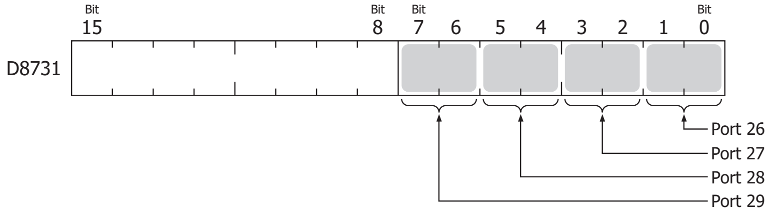


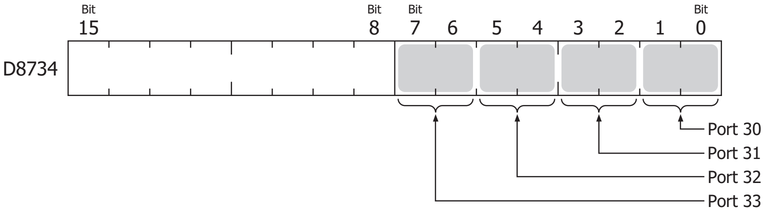




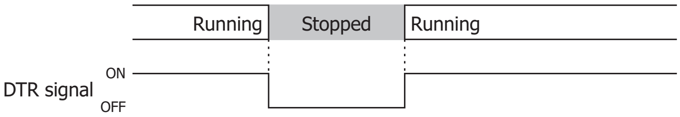






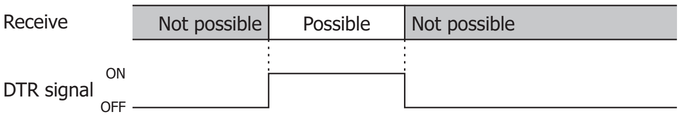


0 (00): The signal is on when the FC6A Series MICROSmart is set to run and off when stopped.
While running, the signal is always on regardless of transmitting or receiving data. Set this value when it is necessary to indicate the run status.



1 (01): Always off.

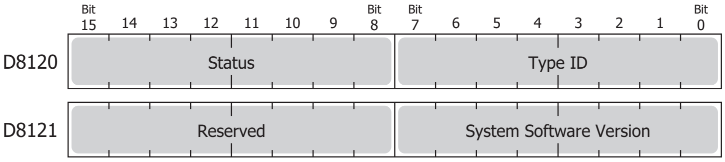
2 (10): Set this value when performing flow control for received data. The signal is on when data from the connected device can be received.
The signal is off when data cannot be received.



3 (11): The operation is the same as the setting "0".

■ D8120, D8121: HMI Module Information

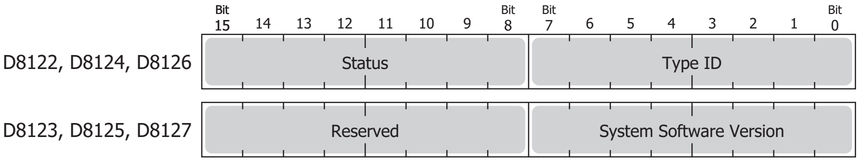
HMI module type information is written to these registers.
The allocation of information (bit assignment) is as follows.



The system software version displays the version of the software written to the device.
For details on the type IDs and statuses, see "List of Type IDs and Status" on page 2-46.

■ D8122 to D8127: Cartridge Slot Information

Cartridge Slot 1 to 3 information is written to these registers.
The allocation of information (bit assignment) is as follows.



D8122, D8123 = Cartridge Slot 1 Information
D8124, D8125 = Cartridge Slot 2 Information
D8126, D8127 = Cartridge Slot 3 Information
For details on the type IDs and statuses, see "List of Type IDs and Status" on page 2-46.

■ D8170, D8171, D8174, D8175, D8178, D8179: Analog I/O Cartridge I/O

Analog I/O values for the analog I/O cartridges are written to these registers.

For the analog input type : The analog values input to the analog I/O cartridge are converted to digital values and written to the registers.

For the analog output type : The digital values stored as digital values are converted to analog values and output from the analog I/O cartridges.

For details, see Chapter 10 "Analog I/O Cartridge" in the "FC6A Series MICROSmart User's Manual".

D8170 = AI2/AQ2
D8171 = AI3/AQ3
D8174 = AI4/AQ4
D8175 = AI5/AQ5
D8178 = AI6/AQ6
D8179 = AI7/AQ7

■ D8172, D8173, D8176, D8177, D8180, D8181: Analog I/O Cartridge Status

The analog status of analog I/O cartridges is written to these registers.
For details, see Chapter 10 "Analog I/O Cartridge" in the "FC6A Series MICROSmart User's Manual".

D8172 = AI2/AQ2
D8173 = AI3/AQ3
D8176 = AI4/AQ4
D8177 = AI5/AQ5
D8180 = AI6/AQ6
D8181 = AI7/AQ7

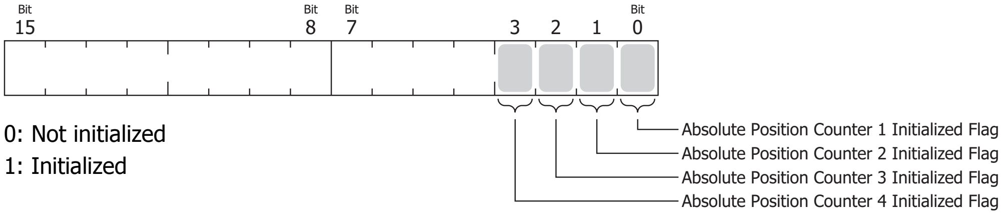
■ D8192 to D8203, D8210 to D8213, D8216 to D8229, D8232 to D8237: High-speed Counter

These special data registers are used with the high-speed counter function and the frequency measurement function.
For details on high-speed counters, see Chapter 5 "High-Speed Counter" in the "FC6A Series MICROSmart User's Manual".

D8210 to D8213, D8216, D8217 = High-speed counter (group 1/I0)
D8218 to D8221, D8234, D8235 = High-speed counter (group 3/I3)
D8222 to D8225, D8236, D8237 = High-speed counter (group 4/I4)
D8226 to D8229, D8232, D8233 = High-speed counter (group 5/I6)
D8192 to D8197 = High-speed counter (group 2/I1)
D8198 to D8203 = High-speed counter (group 6/I7)

■ D8239: Absolute Position Control Status

This register indicates the status of absolute position control.
The allocation of the absolute position counter initialized flags in the device (bit assignment) is as follows. For details on the absolute position control status, see "ABS (Set Absolute Position)" on page 18-68.


Bit 15
Bit 8
Bit 7
3
2
1
Bit 0
0: Not initialized
1: Initialized
Absolute Position Counter 1 Initialized Flag
Absolute Position Counter 2 Initialized Flag
Absolute Position Counter 3 Initialized Flag
Absolute Position Counter 4 Initialized Flag

■ D8240 to D8247: Absolute Position Counter 1 to 4

The absolute position is written to these registers according to pulse output. The absolute position can be initialized with the ABS instruction.
When reversible control is set in the pulse output instructions (RAMP/ARAMP/ZRN/JOG), the absolute position is incremented or decremented according to the output result. For details on absolute position counters, see "ABS (Set Absolute Position)" on page 18-68.

D8240, D8241 = Absolute position counter 1
D8242, D8243 = Absolute position counter 2
D8244, D8245 = Absolute position counter 3
D8246, D8247 = Absolute position counter 4

■ D8250: Read SD Memory Card Capacity

This register indicates the capacity of the recognized SD or SDHC (maximum size 32 GB) memory card in megabytes.

It is 0 when no SD memory card has been inserted or if it is not recognized.

- **D8251: Read SD Memory Card Free Capacity**

This register indicates the free capacity of the SD memory card in megabytes.
It is 0 when no SD memory card has been inserted or if it is not recognized.

- **D8254: SD Memory Card Download/Upload Execution Information**

Information about the SD memory card download/upload that was executed.
For details, see Chapter 11 "SD Memory Card" in the "FC6A Series MICROSmart User's Manual".

- **D8255: SD Memory Card Download/Upload Execution Status**

The status of the SD memory card download/upload that was executed.
For details, see Chapter 11 "SD Memory Card" in the "FC6A Series MICROSmart User's Manual".

- **D8260: Recipe Block Number**

The recipe block number to read or write using special internal relays.
All channels for the specified block number are applicable.

- **D8261: Recipe Execution Block Number**

The recipe block number of the recipe that was executed.
This is updated when the recipe block starts, and the status is retained when completed.

- **D8262: Recipe Execution Channel No.**

The channel number of the recipe that was executed.
This is updated when the channel processing starts, and the status is retained when completed.

- **D8263: Recipe Execution Operation**

Information about the operation of the recipe that was executed.
For details on recipes, see Chapter 11 "Recipe Function" in the "FC6A Series MICROSmart User's Manual".

- **D8264: Recipe Execution Status**

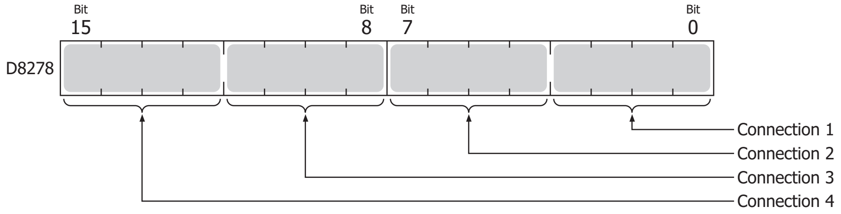
The status of the recipe that was executed.
For details on recipes, see Chapter 11 "Recipe Function" in the "FC6A Series MICROSmart User's Manual".

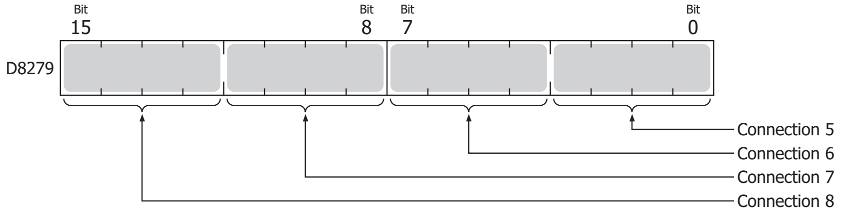
- **D8265: Recipe Execution Error Information**

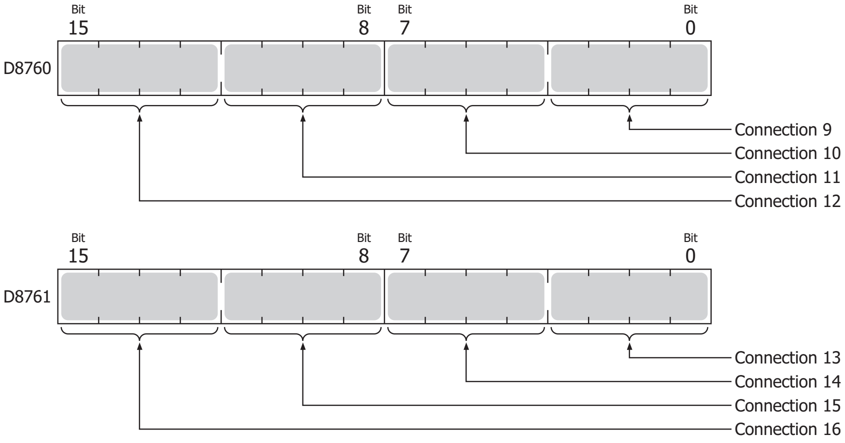
Error information about the recipe that was executed.
For details on recipes, see Chapter 11 "Recipe Function" in the "FC6A Series MICROSmart User's Manual".

- **D8278, D8279, D8760, D8761: Communication Mode Information (Client Connection) (Connection 1 to 16)**

D8278 = Indicates the communication mode of connections 1 through 4.
D8279 = Indicates the communication mode of connections 5 through 8.
D8760 = Indicates the communication mode of connections 9 through 12.
D8761 = Indicates the communication mode of connections 13 through 16.
The allocation of connections in the device (bit assignment) is as follows.



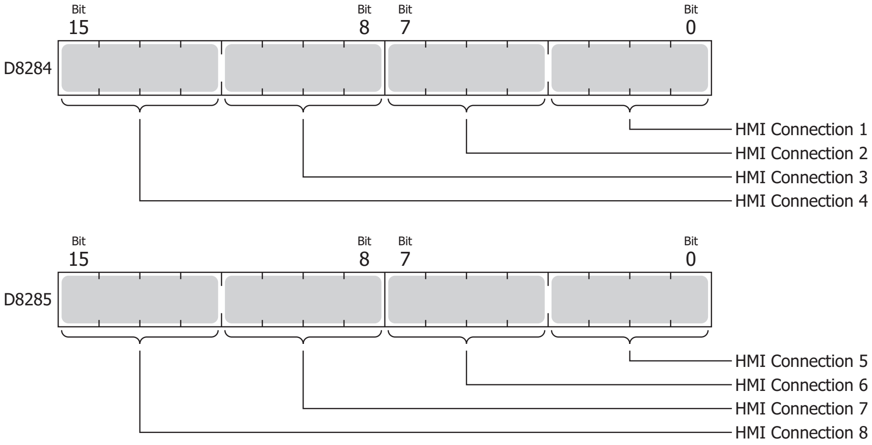




- Client connection (most significant bit = 0)
  0000: Unused
  0001: User Communication
  0010: Modbus TCP client
  0100: User communication UDP
- Server connection (most significant bit = 1)
  1000: Maintenance Communication
  1001: User Communication
  1010: Modbus TCP server

■ D8284, D8285: Communication Mode Information (HMI Connection) (HMI Connection 1 to 8)

D8284: Indicates the communication mode of HMI connections 1 through 4.
D8285: Indicates the communication mode of HMI connections 5 through 8.
The allocation of connections in the device (bit assignment) is as follows.



- Client connection (most significant bit = 0)
  0000: Unused
- Server connection (most significant bit = 1)
  1000: Maintenance Communication

■ D8303: CPU Module Ethernet Port 1 IP Settings/DNS Settings Switching

The IP settings/DNS settings for Ethernet Port 1 can be changed by writing one of the setting values in the table on the next page to D8303 and then turning on M8190.

To use this function, enable the following items in **Function Area Settings** of WindLDR.

- All-in-One CPU module/CAN J1939 All-in-One CPU module
  Enable D8303 (IP Settings / DNS Settings switching) on **Network Settings** in **Function Area Settings**
- Plus CPU module
  Enable D8303 (IP Settings / DNS Settings switching) on **Ethernet Port 1** in **Function Area Settings**

The meanings of the setting values are as follows.

Setting Value	IP Settings/DNS Settings
0	Conform to function area settings.
1	Enable DHCP.
2	Conform to special data register (D8304 to D8323) settings.

■ D8304 to D8307: CPU Module Ethernet Port 1 IP Address (Write-only)

These registers are used to write the CPU module's IP address.
IP address: To set as aaa.bbb.ccc.ddd, write the following.
D8304=aaa, D8305=bbb, D8306=ccc, D8307=ddd

■ D8308 to D8311: CPU Module Ethernet Port 1 Subnet Mask (Write-only)

These registers are used to write the CPU module's subnet mask.
Subnet mask: For aaa.bbb.ccc.ddd, write the following.
D8308=aaa, D8309=bbb, D8310=ccc, D8311=ddd

■ D8312 to D8315: CPU Module Ethernet Port 1 Default Gateway (Write-only)

These registers are used to write the CPU module's default gateway.
Default gateway: For aaa.bbb.ccc.ddd, write the following.
D8312=aaa, D8313=bbb, D8314=ccc, D8315=ddd

■ D8316 to D8319: CPU Module Ethernet Port 1 Preferred DNS Server (Write-only)

These registers are used to write the CPU module's preferred DNS server.
Preferred DNS server: For aaa.bbb.ccc.ddd, write the following.
D8316=aaa, D8317=bbb, D8318=ccc, D8319=ddd

■ D8320 to D8323: CPU Module Ethernet Port 1 Alternate DNS Server (Write-only)

These registers are used to write the CPU module's alternate DNS server.
Alternate DNS server: For aaa.bbb.ccc.ddd, write the following.
D8320=aaa, D8321=bbb, D8322=ccc, D8323=ddd

■ D8324 to D8329: CPU Module Ethernet Port 1 MAC Address (Current Value Read-only)

The CPU module's MAC address is written to the special data registers as follows.
Example: MAC address: AA-BB-CC-DD-EE-FF
D8324=AA, D8325=BB, D8326=CC, D8327=DD, D8328=EE, D8329=FF

■ D8330 to D8333: CPU Module Ethernet Port 1 IP Address (Current Value Read-only)

The CPU module's IP address is written to the special data registers as follows.
Example: The CPU module's own IP address aaa.bbb.ccc.ddd
D8330=aaa, D8331=bbb, D8332=ccc, D8333=ddd

■ D8334 to D8337: CPU Module Ethernet Port 1 Subnet Mask (Current Value Read-only)

The CPU module's subnet mask value is written to the special data registers as follows.
Example: Subnet mask: aaa.bbb.ccc.ddd
D8334=aaa, D8335=bbb, D8336=ccc, D8337=ddd

■ D8338 to D8341: CPU Module Ethernet Port 1 Default Gateway (Current Value Read-only)

The CPU module's default gateway address is written to the special data registers as follows.
Example: Default gateway: aaa.bbb.ccc.ddd
D8338=aaa, D8339=bbb, D8340=ccc, D8341=ddd

■ D8342 to D8345: CPU Module Ethernet Port 1 Preferred DNS Server (Current Value Read-only)

The CPU module's preferred DNS server address is written to the special data registers as follows.
Example: Preferred DNS server: aaa.bbb.ccc.ddd
D8342=aaa, D8343=bbb, D8344=ccc, D8345=ddd

■ D8346 to D8349: CPU Module Ethernet Port 1 Alternate DNS Server (Current Value Read-only)

The CPU module's alternate DNS server address is written to the special data registers as follows.
Example: Alternate DNS server: aaa.bbb.ccc.ddd
D8346=aaa, D8347=bbb, D8348=ccc, D8349=ddd

■ D8350 to D8381, D8677 to D8708: Connection Connected IP Address

The IP address of the connected device that is being accessed through a connection is written as follows.

Connection 1 Connected IP Address: For aaa.bbb.ccc.ddd
D8350=aaa, D8351=bbb, D8352=ccc, D8353=ddd

Connection 2 Connected IP Address: For aaa.bbb.ccc.ddd
D8354=aaa, D8355=bbb, D8356=ccc, D8357=ddd

Connection 3 Connected IP Address: For aaa.bbb.ccc.ddd
D8358=aaa, D8359=bbb, D8360=ccc, D8361=ddd

Connection 4 Connected IP Address: For aaa.bbb.ccc.ddd
D8362=aaa, D8363=bbb, D8364=ccc, D8365=ddd

Connection 5 Connected IP Address: For aaa.bbb.ccc.ddd
D8366=aaa, D8367=bbb, D8368=ccc, D8369=ddd

Connection 6 Connected IP Address: For aaa.bbb.ccc.ddd
D8370=aaa, D8371=bbb, D8372=ccc, D8373=ddd

Connection 7 Connected IP Address: For aaa.bbb.ccc.ddd
D8374=aaa, D8375=bbb, D8376=ccc, D8377=ddd

Connection 8 Connected IP Address: For aaa.bbb.ccc.ddd
D8378=aaa, D8379=bbb, D8380=ccc, D8381=ddd
Connection 9 Connected IP Address: For aaa.bbb.ccc.ddd
D8677=aaa, D8678=bbb, D8679=ccc, D8680=ddd
Connection 10 Connected IP Address: For aaa.bbb.ccc.ddd
D8681=aaa, D8682=bbb, D8683=ccc, D8684=ddd
Connection 11 Connected IP Address: For aaa.bbb.ccc.ddd
D8685=aaa, D8686=bbb, D8687=ccc, D8688=ddd
Connection 12 Connected IP Address: For aaa.bbb.ccc.ddd
D8689=aaa, D8690=bbb, D8691=ccc, D8692=ddd
Connection 13 Connected IP Address: For aaa.bbb.ccc.ddd
D8693=aaa, D8694=bbb, D8695=ccc, D8696=ddd
Connection 14 Connected IP Address: For aaa.bbb.ccc.ddd
D8697=aaa, D8698=bbb, D8699=ccc, D8700=ddd
Connection 15 Connected IP Address: For aaa.bbb.ccc.ddd
D8701=aaa, D8702=bbb, D8703=ccc, D8704=ddd
Connection 16 Connected IP Address: For aaa.bbb.ccc.ddd
D8705=aaa, D8706=bbb, D8707=ccc, D8708=ddd

■ D8382 to D8387: HMI Module MAC Address (Current Value Read-only)

The MAC address is written to the special data registers as follows.
Example: MAC address: AA-BB-CC-DD-EE-FF
D8382=AA, D8383=BB, D8384=CC, D8385=DD, D8386=EE, D8387=FF

■ D8388 to D8391: HMI Module IP Address (Current Value Read-only)

The HMI module's IP address is written to the special data registers as follows.
Example: HMI module IP address: aaa.bbb.ccc.ddd
D8388=aaa, D8389=bbb, D8390=ccc, D8391=ddd

■ D8392 to D8395: HMI Module Subnet Mask (Current Value Read-only)

The HMI module's subnet mask value is written to the special data registers as follows.
Example: HMI module subnet mask: aaa.bbb.ccc.ddd
D8392=aaa, D8393=bbb, D8394=ccc, D8395=ddd

■ D8396 to D8399: HMI Module Default Gateway (Current Value Read-only)

The HMI module's default gateway address is written to the special data registers as follows.
Example: HMI module default gateway: aaa.bbb.ccc.ddd
D8396=aaa, D8397=bbb, D8398=ccc, D8399=ddd

■ D8400 to D8403: HMI Module Preferred DNS Server (Current Value Read-only)

The HMI module's preferred DNS server address is written to the special data registers as follows.
Example: HMI module preferred DNS server: aaa.bbb.ccc.ddd
D8400=aaa, D8401=bbb, D8402=ccc, D8403=ddd

- **D8404 to D8407: HMI Module Alternate DNS Server (Current Value Read-only)**

The HMI module's alternate DNS server address is written to the special data registers as follows.
Example: HMI module alternate DNS server: aaa.bbb.ccc.ddd
D8404=aaa, D8405=bbb, D8406=ccc, D8407=ddd

- **D8413: Time Zone Offset**

The time zone configured in the function area settings can be finely adjusted in 15-minute increments.
For details, see Chapter 3 "SNTP Settings" in the "FC6A Series MICROSmart Communication Manual".

- **D8414: SNTP Operation Status**

The operation status is written to this register when the time information is acquired through operation of M8191 (SNTP Time Acquisition Flag) or when time information acquisition was executed by automatic acquisition.
For details, see Chapter 3 "SNTP Settings" in the "FC6A Series MICROSmart Communication Manual".

- **D8415: SNTP Access Elapsed Time**

The elapsed time in minutes from when the time information was last acquired from the SNTP server.
For details, see Chapter 3 "SNTP Settings" in the "FC6A Series MICROSmart Communication Manual".

- **D8429: HMI Module Connection Information Reference Connection Number**

The connection information for the specified connection number is reflected in D8430 to D8434. When 0 is written, 0 is written to D8430 to D8434. If a connection number that does not exist is specified, 0 is written to D8430 to D8434.

- **D8430 to D8433: HMI Module Connection Information Reference Connected IP Address**

The IP address of the terminal being accessed through the connection is written to the special data registers as follows.
Example: IP address to read: aaa.bbb.ccc.ddd
D8430=aaa, D8431=bbb, D8432=ccc, D8433=ddd

- **D8434: HMI Module Connection Information Reference Connected Port No.**

The port number of the terminal being accessed through the connection is written to this register.

- **D8437 to D8440: HMI Module IP Address (Write-only)**

These registers are used to write the HMI module's IP address.
HMI module IP address: To set as aaa.bbb.ccc.ddd, write the following.
D8437=aaa, D8438=bbb, D8439=ccc, D8440=ddd

- **D8441 to D8444: HMI Module Subnet Mask (Write-only)**

These registers are used to write the HMI module's subnet mask.
HMI module subnet mask: For aaa.bbb.ccc.ddd, write the following.
D8441=aaa, D8442=bbb, D8443=ccc, D8444=ddd

- **D8445 to D8448: HMI Module Default Gateway (Write-only)**

These registers are used to write the HMI module's default gateway.
HMI module default gateway: For aaa.bbb.ccc.ddd, write the following.
D8445=aaa, D8446=bbb, D8447=ccc, D8448=ddd

- **D8449 to D8452: HMI Module Preferred DNS Server (Write-only)**

These registers are used to write the HMI module's preferred DNS server.
HMI module preferred DNS server: For aaa.bbb.ccc.ddd, write the following.
D8449=aaa, D8450=bbb, D8451=ccc, D8452=ddd

- **D8453 to D8456: HMI Module Alternate DNS Server (Write-only)**

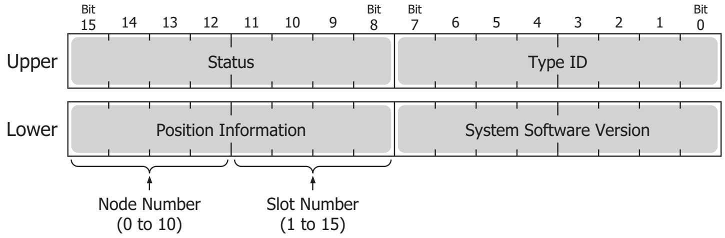
These registers are used to write the HMI module's alternate DNS server.
HMI module alternate DNS server: For aaa.bbb.ccc.ddd, write the following.
D8453=aaa, D8454=bbb, D8455=ccc, D8456=ddd

- **D8457: EMAIL Instruction Detailed Error Information (HMI-Ethernet Port)**

Detailed error information for the EMAIL instruction is written to this register.
For details, see Chapter 12 "Confirm the error detail of EMAIL instruction" in the "FC6A Series MICROSmart Communication Manual".

- **D8470 to D8595: Expansion Module Slot Information**

Expansion module type information is written to these registers.
The allocation of information (bit assignment) is as follows.



Upper	Lower	
D8470,	D8471	= Expansion Module Slot 1 Information
D8472,	D8473	= Expansion Module Slot 2 Information
D8474,	D8475	= Expansion Module Slot 3 Information
D8476,	D8477	= Expansion Module Slot 4 Information
D8478,	D8479	= Expansion Module Slot 5 Information
D8480,	D8481	= Expansion Module Slot 6 Information
D8482,	D8483	= Expansion Module Slot 7 Information
D8484,	D8485	= Expansion Module Slot 8 Information
D8486,	D8487	= Expansion Module Slot 9 Information
D8488,	D8489	= Expansion Module Slot 10 Information
D8490,	D8491	= Expansion Module Slot 11 Information
D8492,	D8493	= Expansion Module Slot 12 Information
D8494,	D8495	= Expansion Module Slot 13 Information
D8496,	D8497	= Expansion Module Slot 14 Information
D8498,	D8499	= Expansion Module Slot 15 Information
D8500,	D8501	= Expansion Module Slot 16 Information
D8502,	D8503	= Expansion Module Slot 17 Information
D8504,	D8505	= Expansion Module Slot 18 Information
D8506,	D8507	= Expansion Module Slot 19 Information
D8508,	D8509	= Expansion Module Slot 20 Information
D8510,	D8511	= Expansion Module Slot 21 Information
D8512,	D8513	= Expansion Module Slot 22 Information
D8514,	D8515	= Expansion Module Slot 23 Information
D8516,	D8517	= Expansion Module Slot 24 Information
D8518,	D8519	= Expansion Module Slot 25 Information
D8520,	D8521	= Expansion Module Slot 26 Information
D8522,	D8523	= Expansion Module Slot 27 Information
D8524,	D8525	= Expansion Module Slot 28 Information
D8526,	D8527	= Expansion Module Slot 29 Information
D8528,	D8529	= Expansion Module Slot 30 Information
D8530,	D8531	= Expansion Module Slot 31 Information
D8532,	D8533	= Expansion Module Slot 32 Information
D8534,	D8535	= Expansion Module Slot 33 Information
D8536,	D8537	= Expansion Module Slot 34 Information
D8538,	D8539	= Expansion Module Slot 35 Information
D8540,	D8541	= Expansion Module Slot 36 Information
D8542,	D8543	= Expansion Module Slot 37 Information
D8544,	D8545	= Expansion Module Slot 38 Information
D8546,	D8547	= Expansion Module Slot 39 Information

Upper	Lower	
D8548,	D8549	= Expansion Module Slot 40 Information
D8550,	D8551	= Expansion Module Slot 41 Information
D8552,	D8553	= Expansion Module Slot 42 Information
D8554,	D8555	= Expansion Module Slot 43 Information
D8556,	D8557	= Expansion Module Slot 44 Information
D8558,	D8559	= Expansion Module Slot 45 Information
D8560,	D8561	= Expansion Module Slot 46 Information
D8562,	D8563	= Expansion Module Slot 47 Information
D8564,	D8565	= Expansion Module Slot 48 Information
D8566,	D8567	= Expansion Module Slot 49 Information
D8568,	D8569	= Expansion Module Slot 50 Information
D8570,	D8571	= Expansion Module Slot 51 Information
D8572,	D8573	= Expansion Module Slot 52 Information
D8574,	D8575	= Expansion Module Slot 53 Information
D8576,	D8577	= Expansion Module Slot 54 Information
D8578,	D8579	= Expansion Module Slot 55 Information
D8580,	D8581	= Expansion Module Slot 56 Information
D8582,	D8583	= Expansion Module Slot 57 Information
D8584,	D8585	= Expansion Module Slot 58 Information
D8586,	D8587	= Expansion Module Slot 59 Information
D8588,	D8589	= Expansion Module Slot 60 Information
D8590,	D8591	= Expansion Module Slot 61 Information
D8592,	D8593	= Expansion Module Slot 62 Information
D8594,	D8595	= Expansion Module Slot 63 Information

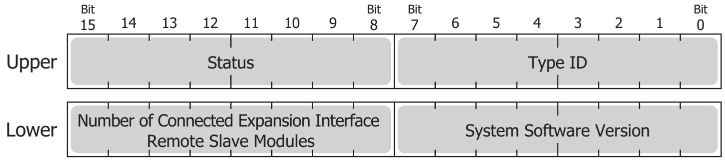
The system software version displays the version of the software written to the device.

For details on the type IDs and statuses, see "List of Type IDs and Status" on page 2-46.

■ D8596, D8597: Expansion Interface Remote Master Module Slot Information

Expansion interface remote master module type information is written to these registers.

The allocation of information (bit assignment) is as follows.



<table>
<tr><th></th><th>Bit 15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>Bit 8</th><th>Bit 7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>Bit 0</th></tr>
<tr><td>Upper</td><td colspan="8">Status</td><td colspan="8">Type ID</td></tr>
<tr><td>Lower</td><td colspan="8">Number of Connected Expansion Interface Remote Slave Modules</td><td colspan="8">System Software Version</td></tr>
</table>

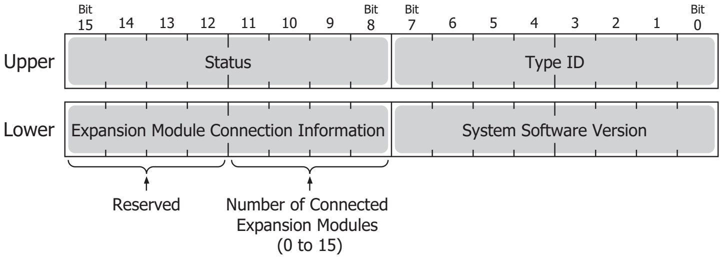
The system software version displays the version of the software written to the device.

For details on the type IDs and statuses, see "List of Type IDs and Status" on page 2-46.

■ D8598 to D8617: Expansion Interface Remote Slave Module Slot Information

Expansion interface remote slave module type information is written to these registers.

The allocation of information (bit assignment) is as follows.



Upper	Lower	
D8598,	D8599	= Expansion Interface Remote Slave Module (Unit 1) Slot Information
D8600,	D8601	= Expansion Interface Remote Slave Module (Unit 2) Slot Information
D8602,	D8603	= Expansion Interface Remote Slave Module (Unit 3) Slot Information
D8604,	D8605	= Expansion Interface Remote Slave Module (Unit 4) Slot Information
D8606,	D8607	= Expansion Interface Remote Slave Module (Unit 5) Slot Information
D8608,	D8609	= Expansion Interface Remote Slave Module (Unit 6) Slot Information
D8610,	D8611	= Expansion Interface Remote Slave Module (Unit 7) Slot Information
D8612,	D8613	= Expansion Interface Remote Slave Module (Unit 8) Slot Information
D8614,	D8615	= Expansion Interface Remote Slave Module (Unit 9) Slot Information
D8616,	D8617	= Expansion Interface Remote Slave Module (Unit 10) Slot Information

The system software version displays the version of the software written to the device.

For details on the type ID and status, see "List of Type IDs and Status" on page 2-46.

■ D8618: Refresh Time of Expansion Interface Remote Master/Slave Modules Current Value

The current value of the I/O refresh time of the expansion modules connected with the expansion interface remote master/slave modules is stored in increments of 1 ms.

■ D8619: Refresh Time of Expansion Interface Remote Master/Slave Modules Maximum Value

The maximum value of the I/O refresh time of the expansion modules connected with the expansion interface remote master/ slave modules is stored in increments of 1 ms.

■ D8630: CPU Module Ethernet Port 2 IP Settings/DNS Settings Switching

The IP settings/DNS settings for Ethernet Port 2 can be changed by writing one of the setting values in the following table to D8630 and then turning on M8333.
To use this function, enable **Enable D8630 (IP Settings / DNS Settings switching)** on **Ethernet Port 2** in **Function Area Settings** of WindLDR.
The meanings of the setting values are as follows.

Setting Value	IP Settings/DNS Settings
0	Conform to function area settings.
1	Enable DHCP.
2	Conform to special data register (D8631 to D8650) settings.

■ D8631 to D8634: CPU Module Ethernet Port 2 IP Address (Write-only)

These registers are used to write the IP address of Ethernet port 2 on the Plus CPU module.

IP address: To set as aaa.bbb.ccc.ddd, write the following.

D8631 = aaa, D8632 = bbb, D8633 = ccc, D8634 = ddd

■ D8635 to D8638: CPU Module Ethernet Port 2 Subnet Mask (Write-only)

These registers are used to write the subnet mask of Ethernet port 2 on the Plus CPU module.

Subnet mask: For aaa.bbb.ccc.ddd, write the following.

D8635 = aaa, D8636 = bbb, D8637 = ccc, D8638 = ddd

- **D8639 to D8642: CPU Module Ethernet Port 2 Default Gateway (Write-only)**

These registers are used to write the default gateway of Ethernet port 2 on the Plus CPU module.

Default gateway: For aaa.bbb.ccc.ddd, write the following.

D8639 = aaa, D8640 = bbb, D8641 = ccc, D8642 = ddd

- **D8643 to D8646: CPU Module Ethernet Port 2 Preferred DNS Server (Write-only)**

These registers are used to write the preferred DNS server of Ethernet port 2 on the Plus CPU module.

Preferred DNS server: For aaa.bbb.ccc.ddd, write the following.

D8643 = aaa, D8644 = bbb, D8645 = ccc, D8646 = ddd

- **D8647 to D8650: CPU Module Ethernet Port 2 Alternate DNS Server (Write-only)**

These registers are used to write the alternate DNS server of Ethernet port 2 on the Plus CPU module.

Alternate DNS server: For aaa.bbb.ccc.ddd, write the following.

D8647 = aaa, D8648 = bbb, D8649 = ccc, D8650 = ddd

- **D8651 to D8656: CPU Module Ethernet Port 2 MAC Address (Current Value Read-only)**

The MAC address of Ethernet port 2 on the Plus CPU module is written to the special data registers as follows.

Example: MAC address: AA-BB-CC-DD-EE-FF

D8651 = AA, D8652 = BB, D8653 = CC, D8654 = DD, D8655 = EE, D8656 = FF

- **D8657 to D8660: CPU Module Ethernet Port 2 IP Address (Current Value Read-only)**

The IP address of Ethernet port 2 on the Plus CPU module is written to the special data registers as follows.

Example: The CPU module's own IP address: aaa.bbb.ccc.ddd

D8657 = aaa, D8658 = bbb, D8659 = ccc, D8660 = ddd

- **D8661 to D8664: CPU Module Ethernet Port 2 Subnet Mask (Current Value Read-only)**

The subnet mask value of Ethernet port 2 on the Plus CPU module is written to the special data registers as follows.

Example: Subnet mask: aaa.bbb.ccc.ddd

D8661 = aaa, D8662 = bbb, D8663 = ccc, D8664 = ddd

- **D8665 to D8668: CPU Module Ethernet Port 2 Default Gateway (Current Value Read-only)**

The default gateway address of Ethernet port 2 on the Plus CPU module is written to the special data registers as follows.

Example: Default gateway: aaa.bbb.ccc.ddd

D8665 = aaa, D8666 = bbb, D8667 = ccc, D8668 = ddd

- **D8669 to D8672: CPU Module Ethernet Port 2 Preferred DNS Server (Current Value Read-only)**

The preferred DNS server address of Ethernet port 2 on the Plus CPU module is written to the special data registers as follows.

Example: Preferred DNS server: aaa.bbb.ccc.ddd

D8669 = aaa, D8670 = bbb, D8671 = ccc, D8672 = ddd

- **D8673 to D8676: CPU Module Ethernet Port 2 Alternate DNS Server (Current Value Read-only)**

The alternate DNS server address of Ethernet port 2 on the Plus CPU module is written to the special data registers as follows.

Example: Alternate DNS server: aaa.bbb.ccc.ddd

D8673 = aaa, D8674 = bbb, D8675 = ccc, D8676 = ddd

- **D8759: EMAIL Instruction Detailed Error Information (Ethernet Port 1)**

Detailed error information for the EMAIL instruction (Ethernet port 1) is written to this register.

For details, see Chapter 12 "Confirm the error detail of EMAIL instruction" in the "FC6A Series MICROSmart Communication Manual".

- **D8782: BACnet Operation Status**

The operating status of BACnet communication is written to this special data register.

For details, see Chapter 15 "Special Devices Used in BACnet/IP" in the "FC6A Series MICROSmart Communication Manual".

- **D8783: BACnet Error Information**

Information for the error that last occurred in BACnet communication is written to this special data register.

For details, see Chapter 15 "Special Devices Used in BACnet/IP in the "FC6A Series MICROSmart Communication Manual".

List of Type IDs and Status

■ Type ID

Expansion module and HMI module

Type ID		Type No.
Hexadecimal	Binary	
0x00	0000 0000	FC6A-N16B1, FC6A-N16B3
0x01	0000 0001	FC6A-R161, FC6A-T16K1, FC6A-T16P1, FC6A-T16K3, FC6A-T16P3
0x02	0000 0010	FC6A-N32B3
0x03	0000 0011	FC6A-T32K3, FC6A-T32P3
0x04	0000 0100	FC6A-N08B1, FC6A-N08A11
0x05	0000 0101	FC6A-R081, FC6A-T08K1, FC6A-T08P1
0x06	0000 0110	FC6A-M08BR1
0x07	0000 0111	FC6A-M24BR1
0x18	0001 1000	FC6A-PH1
0x19	0001 1001	FC6A-EXM2
0x1A	0001 1010	FC6A-EXM1S
0x20	0010 0000	FC6A-J2C1
0x21	0010 0001	FC6A-J4A1
0x22	0010 0010	FC6A-J8A1
0x24	0010 0100	FC6A-K4A1
0x25	0010 0101	FC6A-L06A1
0x26	0010 0110	FC6A-L03CN1
0x27	0010 0111	FC6A-J4CN1
0x28	0010 1000	FC6A-J8CU1
0x29	0010 1001	FC6A-F2M1
0x2A	0010 1010	FC6A-F2MR1
0x2B	0010 1011	FC6A-J4CH1Y
0x2C	0010 1100	FC6A-EXM1M
0x2E	0010 1110	FC6A-SIF52
0xFF	1111 1111	Not connected

Cartridge

Type ID		Type No.
Hexadecimal	Binary	
0x00	0000 0000	FC6A-PJ2A
0x01	0000 0001	FC6A-PK2AV
0x02	0000 0010	FC6A-PK2AW
0x03	0000 0011	FC6A-PJ2CP
0x06	0000 0110	FC6A-PC1
0x07	0000 0111	FC6A-PC3
0x09	0000 1001	FC6A-PTS4, FC6A-PTK4
0x0A	0000 1010	FC6A-PN4
0x0C	0000 1100	FC6A-PC4
0xFF	1111 1111	Not connected

■ Status

Status		Description
Hexadecimal	Binary	
0x00	0000 0000	Normal
0x81	1000 0001	Communication error (An error has occurred in the communication between the CPU module and an expansion module, HMI module, or cartridge.)
0x82	1000 0010	Unknown device detected (A device other than the FC6A Series MICROSmart is connected.)
0x83	1000 0011	Device setting error (No device is connected or the connected device is different from that set in the user program.)
0x84	1000 0100	Device writing error (The attempt to set the operation of the device failed.)
0x85	1000 0101	System update error (The system update failed.)
0x86	1000 0110	Expansion interface remote master module communication error (An error is occurring in the communication between the expansion interface remote master and slave modules.)

3: Instructions Reference

Introduction

This chapter provides a list of FC6A Series MICROSmart instructions and describes their functions.

The instructions are divided into basic instructions, which perform sequencing, and advanced instructions, which perform moves, comparisons, Boolean computations, binary arithmetic operations, bit shifts, and other operations.

Basic Instruction List

Symbol	Name	Function	See Page
AND	And	Series connection of NO contact	4-4
AND LOD	And Load	Series connection of circuit blocks	4-5
ANDN	And Not	Series connection of NC contact	4-4
BPP	Bit Pop	Restores the result of bit logical operation which was saved temporarily	4-6
BPS	Bit Push	Saves the result of bit logical operation temporarily	4-6
BRD	Bit Read	Reads the result of bit logical operation which was saved temporarily	4-6
CC=	Counter Comparison (=)	Equal to comparison of counter current value	4-20
CC>=	Counter Comparison (>=)	Greater than or equal to comparison of counter current value	4-20
CDP	Dual Pulse Reversible Counter	Dual pulse reversible counter (0 to 65,535)	4-12
CDPD	Double-Word Dual Pulse Reversible Counter	Double-word dual pulse reversible counter (0 to 4,294,967,295)	4-16
CNT	Adding Counter	Adding counter (0 to 65,535)	4-12
CNTD	Double-Word Adding Counter	Double-word adding counter (0 to 4,294,967,295)	4-16
CUD	Up/Down Selection Reversible Counter	Up/down selection reversible counter (0 to 65,535)	4-12
CUDD	Double-Word Up/Down Selection Reversible Counter	Double-word up/down selection reversible counter (0 to 4,294,967,295)	4-16
DC=	Data Register Comparison (=)	Equal to comparison of data register value	4-22
DC>=	Data Register Comparison (>=)	Greater than or equal to comparison of data register value	4-22
END	End	Ends a program	4-32
JEND	Jump End	Ends a jump instruction	4-31
JMP	Jump	Jumps a designated program area	4-31
LOD	Load	Stores intermediate results and reads contact status	4-1
LODN	Load Not	Stores intermediate results and reads inverted contact status	4-1
MCR	Master Control Reset	Ends a master control	4-29
MCS	Master Control Set	Starts a master control	4-29
OR	Or	Parallel connection of NO contact	4-4
OR LOD	Or Load	Parallel connection of circuit blocks	4-5
ORN	Or Not	Parallel connection of NC contact	4-4
OUT	Output	Outputs the result of bit logical operation	4-1
OUTN	Output Not	Outputs the inverted result of bit logical operation	4-1
RST	Reset	Resets output, internal relay, or shift register bit	4-3
SET	Set	Sets output, internal relay, or shift register bit	4-3
SFR	Shift Register	Forward shift register	4-24
SFRN	Shift Register Not	Reverse shift register	4-24
SOTD	Single Output Down	Falling-edge differentiation output	4-28
SOTU	Single Output Up	Rising-edge differentiation output	4-28
TIM	100-ms Timer	Subtracting 100-ms timer (0 to 6,553.5 s)	4-7
TIMO	100-ms Off-delay Timer	Subtracting 100-ms off-delay timer (0 to 6,553.5 s)	4-11

Symbol	Name	Function	See Page
TMH	10-ms Timer	Subtracting 10-ms timer (0 to 655.35 s)	4-7
TMHO	10-ms Off-delay Timer	Subtracting 10-ms off-delay timer (0 to 655.35 s)	4-11
TML	1-s Timer	Subtracting 1-s timer (0 to 65,535 s)	4-7
TMLO	1-s Off-delay Timer	Subtracting 1-s off-delay timer (0 to 65,535 s)	4-11
TMS	1-ms Timer	Subtracting 1-ms timer (0 to 65.535 s)	4-7
TMSO	1-ms Off-delay Timer	Subtracting 1-ms off-delay timer (0 to 65.535 s)	4-11

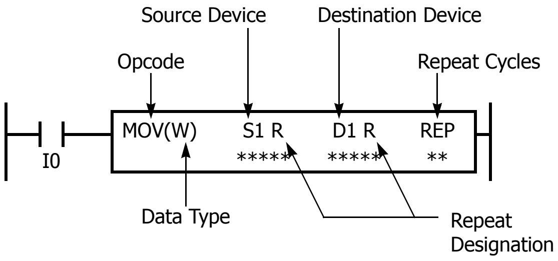
Advanced Instruction List

Group	Symbol	Name	Valid Data Type					See Page
			W	I	D	L	F	
NOP	NOP	No Operation						3-11
Move	MOV	Move	X	X	X	X	X	5-1
	MOVN	Move Not	X	X	X	X		5-5
	IMOV	Indirect Move	X		X		X	5-6
	IMOVN	Indirect Move Not	X		X			5-8
	MOVC	Move Characters						5-9
	BMOV	Block Move	X					5-11
	IBMV	Indirect Bit Move	X					5-12
	IBMVN	Indirect Bit Move Not	X					5-14
	NSET	N Data Set	X	X	X	X	X	5-15
	NRS	N Data Repeat Set	X	X	X	X	X	5-16
	XCHG	Exchange	X		X			5-17
	TCCST	Timer/Counter Current Value Store	X		X			5-18
Data Comparison	CMP=	Compare Equal To	X	X	X	X	X	6-1
	CMP<>	Compare Unequal To	X	X	X	X	X	6-1
	CMP<	Compare Less Than	X	X	X	X	X	6-1
	CMP>	Compare Greater Than	X	X	X	X	X	6-1
	CMP<=	Compare Less Than or Equal To	X	X	X	X	X	6-1
	CMP>=	Compare Greater Than or Equal To	X	X	X	X	X	6-1
	ICMP>=	Interval Compare Greater Than or Equal To	X	X	X	X	X	6-6
	LC=	Load Compare Equal To	X	X	X	X	X	6-8
	LC<>	Load Compare Unequal To	X	X	X	X	X	6-8
	LC<	Load Compare Less Than	X	X	X	X	X	6-8
	LC>	Load Compare Greater Than	X	X	X	X	X	6-8
	LC<=	Load Compare Less Than or Equal To	X	X	X	X	X	6-8
	LC>=	Load Compare Greater Than or Equal To	X	X	X	X	X	6-8
Binary Arithmetic	ADD	Addition	X	X	X	X	X	7-1
	SUB	Subtraction	X	X	X	X	X	7-1
	MUL	Multiplication	X	X	X	X	X	7-1
	DIV	Division	X	X	X	X	X	7-1
	INC	Increment	X	X	X	X		7-13
	DEC	Decrement	X	X	X	X		7-13
	ROOT	Root	X		X		X	7-14
	SUM	Sum (ADD)	X	X	X	X	X	7-15
		Sum (XOR)	X					7-15
	RNDM	Random	X					7-18
Boolean Computation	ANDW	AND Word	X		X			8-1
	ORW	OR Word	X		X			8-1
	XORW	Exclusive OR Word	X		X			8-1
Shift and Rotate	SFTL	Shift Left						9-1
	SFTR	Shift Right						9-3
	BCDLS	BCD Left Shift			X			9-5
	WSFT	Word Shift	X					9-7
	ROTL	Rotate Left	X		X			9-8
	ROTR	Rotate Right	X		X			9-10

Group	Symbol	Name	Valid Data Type					See Page
			W	I	D	L	F	
Data Conversion	HTOB	Hex to BCD	X		X			10-1
	BTOH	BCD to Hex	X		X			10-3
	HTOA	Hex to ASCII	X					10-5
	ATOH	ASCII to Hex	X					10-7
	BTOA	BCD to ASCII	X		X			10-9
	ATOB	ASCII to BCD	X		X			10-12
	ENCO	Encode						10-15
	DECO	Decode						10-16
	BCNT	Bit Count						10-17
	ALT	Alternate Output						10-18
	CVDT	Convert Data Type	X	X	X	X	X	10-19
	DTDV	Data Divide	X					10-21
	DTCB	Data Combine	X					10-22
	SWAP	Data Swap	X		X			10-23
Week Programmer	WKTIM	Week Timer						11-1
	WKTBL	Week Table						11-1
	WEEK	Weekly Timer						11-3
	YEAR	Yearly Timer						11-16
Interface	MSG	Message						12-1
	DISP	Display						12-20
	DGRD	Digital Read						12-22
Program Branching	LABEL	Label						13-1
	LJMP	Label Jump						13-1
	LCAL	Label Call						13-3
	LRET	Label Return						13-3
	DJNZ	Decrement Jump Non-zero						13-5
	DI	Disable Interrupt						15-1
	EI	Enable Interrupt						15-1
	IOREF	I/O Refresh						14-1
	HSCRF	High-speed Counter Refresh						14-3
	FRQRF	Frequency Measurement Refresh						14-4
	COMRF	Communication Refresh						14-5
Coordinate Conversion	XYFS	XY Format Set	X	X				16-1
	CVXTY	Convert X to Y	X	X				16-2
	CVYTX	Convert Y to X	X	X				16-3
	AVRG	Average	X	X	X	X	X	17-1
Pulse	PULS1	Pulse Output 1						18-1
	PULS2	Pulse Output 2						18-1
	PULS3	Pulse Output 3						18-1
	PULS4	Pulse Output 4						18-1
	PWM1	Pulse Width Modulation 1						18-8
	PWM2	Pulse Width Modulation 2						18-8
	PWM3	Pulse Width Modulation 3						18-8
	PWM4	Pulse Width Modulation 4						18-8
	RAMP1	Ramp Pulse Output 1						18-15
	RAMP2	Ramp Pulse Output 2						18-15
	RAMPL	Linear Interpolation Control						18-27
	ZRN1	Zero Return 1						18-36
	ZRN2	Zero Return 2						18-36
	ARAMP1	Advanced Ramp 1						18-46
	ARAMP2	Advanced Ramp 2						18-46
	ABS	Set Absolute Position						18-68
	JOG	JOG Operation						18-72

Group	Symbol	Name	Valid Data Type					See Page
			W	I	D	L	F	
PID Control	PID	PID Control (FC5A compatible)						19-1
	PIDA	PID Control						19-3
	PIDD	PID with Derivative Decay						19-26
Dual / Teaching Timer	DTML	1-s Dual Timer						20-1
	DTIM	100-ms Dual Timer						20-1
	DTMH	10-ms Dual Timer						20-1
	DTMS	1-ms Dual Timer						20-1
	TTIM	Teaching Timer						20-3
Trigonometric Function	RAD	Degree to Radian					X	21-1
	DEG	Radian to Degree					X	21-2
	SIN	Sine					X	21-3
	COS	Cosine					X	21-4
	TAN	Tangent					X	21-5
	ASIN	Arc Sine					X	21-6
	ACOS	Arc Cosine					X	21-7
	ATAN	Arc Tangent					X	21-8
Logarithm / Power	LOGE	Natural Logarithm					X	22-1
	LOG10	Common Logarithm					X	22-2
	EXP	Exponent					X	22-3
	POW	Power					X	22-4
File Data Processing	FIFOF	FIFO Format	X					23-1
	FIEX	First-In Execute	X					23-3
	FOEX	First-Out Execute	X					23-3
	NDSRC	N Data Search	X	X	X	X	X	23-5
Clock	TADD	Time Addition						24-1
	TSUB	Time Subtraction						24-5
	HTOS	HMS to S						24-9
	STOH	S to HMS						24-10
	HOUR	Hour Meter						24-11
Data Logging	DLOG	Data Logging						25-1
	TRACE	Data Trace						25-15
Script	SCRPT	Script	X	X	X	X	X	26-1
Flow Volume Calculation	SCALE	Convert Analog Input	X	X				27-1
	FLWA	Analog Flow Totalizer						27-9
	FLWP	Pulse Flow Totalizer	X		X			27-18
User-defined Macro	UMACRO	User-defined Macro						28-1
User Communication	TXD	Transmit						Communication Manual 5-2
	ETXD	Transmit over Ethernet						Communication Manual 5-23
	RXD	Receive						Communication Manual 5-10
	ERXD	Receive over Ethernet						Communication Manual 5-23
PING	PING	Ping						Communication Manual 10-1
E-mail	EMAIL	Send E-mail						Communication Manual 12-1

Structure of an Advanced Instruction



Repeat Designation

Specifies whether repeat is used for the device or not.

Repeat Cycles

Specifies the quantity of repeat cycles: 1 through 99.

Opcode

The opcode is a symbol used to identify the advanced instruction.

Data Type

Specifies word (W), integer (I), double word (D), long (L), or float (F) data.

Source Device

The source device specifies the 16- or 32-bit data to be processed by the advanced instruction. Some advanced instructions require two source devices.

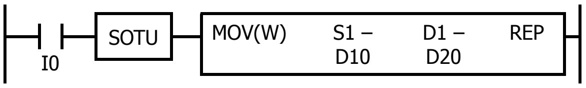
Destination Device

The destination device specifies the 16- or 32-bit data to store the result of the advanced instruction. Some advanced instructions require two destination devices.

Input Condition for Advanced Instructions

Almost all advanced instructions must be preceded by a contact, except NOP (no operation), LABEL (label), and LRET (label return) instructions. The input condition can be programmed using a bit device such as input, output, internal relay, or shift register. Timer and counter can also be used as an input condition to turn on the contact when the timer times out or the counter counts out.

While the input condition is on, the advanced instruction is executed in each scan. To execute the advanced instruction only at the rising or falling edge of the input, use the SOTU or SOTD instruction.



While the input condition is off, the advanced instruction is not executed and device statuses are held.

Source and Destination Devices

The source and destination devices specify 16- or 32-bit data, depending on the selected data type. When a bit device such as input, output, internal relay, or shift register is designated as a source or destination device, 16 or 32 points starting with the designated number are processed as source or destination data. When a word device such as timer or counter is designated as a source device, the current value is read as source data. When a timer or counter is designated as a destination device, the result of the advanced instruction is set to the preset value for the timer or counter. When a data register is designated as a source or destination device, the data is read from or written to the designated data register.

Using Timer or Counter as Source Device

Since all timer instructions—TML (1-s timer), TIM (100-ms timer), TMH (10-ms timer), and TMS (1-ms timer)—subtract from the preset value, the current value is decremented from the preset value and indicates the remaining time. As described above, when a timer is designated as a source device of an advanced instruction, the current value, or the remaining time, of the timer is read as source data. Adding counters CNT start counting at 0, and the current value is incremented up to the preset value. Reversible counters CDP and CUD start counting at the preset value and the current value is incremented or decremented from the preset value. When any counter is designated as a source device of an advanced instruction, the current value is read as source data.

Using Timer or Counter as Destination Device

As described above, when a timer or counter is designated as a destination device of an advanced instruction, the result of the advanced instruction is set to the preset value of the timer or counter. Timer and counter preset values can be 0 through 65,535.

When a timer or counter preset value is designated using a data register, the timer or counter cannot be designated as a destination of an advanced instruction. When executing such an advanced instruction, a user program execution error will result. For details of user program execution error, see "User Program Execution Error" on page A-12.

Note: When a user program execution error occurs, the result is not set to the destination.

Data Types for Advanced Instructions

When using move, data comparison, binary arithmetic, Boolean computation, bit shift/rotate, data conversion, and coordinate conversion instructions, data types can be selected from word (W), integer (I), double word (D), long (L), or float (F). For other advanced instructions, the data is processed in units of 16-bit word.

Data Type	Symbol	Bits	Quantity of Data Registers Used	Range of Decimal Values	
				Decimal	Hexdecimal
Word (Unsigned 16 bits)	**W**	16 bits	1	0 to 65,535	\$0000 to \$FFFF
Integer (Signed 16 bits)	**I**	16 bits	1	–32,768 to 32,767	\$8000 to \$7FFF
Double Word (Unsigned 32 bits)	**D**	32 bits	2	0 to 4,294,967,295	\$00000000 to \$FFFFFFFF
Long (Signed 32 bits)	**L**	32 bits	2	–2,147,483,648 to 2,147,483,647	\$80000000 to \$7FFFFFFF
Float (Floating point)	**F**	32 bits	2	-3.402823×10^{38} to 3.402823×10^{38}	—

- When a constant is set in WindLDR, it can be set as decimal or hexadecimal number. Prefix the number with "$" to set it as a hexadecimal number.

Decimal Values and Hexadecimal Storage (Word, Integer, Double, and Long Data)

The following table shows hexadecimal equivalents which are stored in the CPU, as a result of addition and subtraction of the decimal values shown:

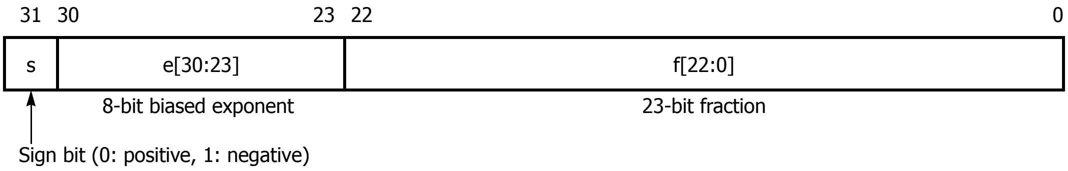
Data Type	Result of Addition	Hexadecimal Storage	Result of Subtraction	Hexadecimal Storage
Word			65,535	FFFF
	0	0000	0	0000
	65,535	FFFF	–1	(BW) FFFF
	131,071	(CY) FFFF	–65,535	(BW) 0001
			–65,536	(BW) 0000
Integer	65,534	(CY) 7FFE	65,534	(BW) 7FFE
	32,768	(CY) 0000	32,768	(BW) 0000
	32,767	7FFF	32,767	7FFF
	0	0000	0	0000
	–1	FFFF	–1	FFFF
	–32,767	8001	–32,767	8001
	–32,768	8000	–32,768	8000
	–32,769	(CY) FFFF	–32,769	(BW) FFFF
	–65,535	(CY) 8001	–65,535	(BW) 8001
Double Word			4,294,967,295	FFFFFFFF
	0	00000000	0	00000000
	4,294,967,295	FFFFFFFF	–1	(BW) FFFFFFFF
	8,589,934,591	(CY) FFFFFFFF	–4,294,967,295	(BW) 00000001
			–4,294,967,296	(BW) 00000000
Long	4,294,967,294	(CY) 7FFFFFFE	4,294,967,294	(BW) 7FFFFFFE
	2,147,483,648	(CY) 00000000	2,147,483,648	(BW) 00000000
	2,147,483,647	7FFFFFFF	2,147,483,647	7FFFFFFF
	0	00000000	0	00000000
	–1	FFFFFFFF	–1	FFFFFFFF
	–2,147,483,647	80000001	–2,147,483,647	80000001
	–2,147,483,648	80000000	–2,147,483,648	80000000
	–2,147,483,649	(CY) FFFFFFFF	–2,147,483,649	(BW) FFFFFFFF
	–4,294,967,295	(CY) 80000001	–4,294,967,295	(BW) 80000001

Floating-Point Data Format

The FC6A Series MICROSmart can specify floating-point data (F) for advanced instructions. Like double word (D) and long integer (L) data, floating-point data also uses two consecutive data registers to execute advanced instructions. The FC6A Series MICROSmart supports the floating-point data based on the single storage format of the IEEE (The Institute of Electrical and Electronics Engineers) Standard 754.

Single Storage Format

The IEEE single storage format consists of three fields: a 23-bit fraction, f; an 8-bit biased exponent, e; and 1-bit sign, s. These fields are stored contiguously in one 32-bit word, as shown in the figure below. Bits 0:22 contain the 23-bit fraction, f, with bit 0 being the least significant bit of the fraction and bit 22 being the most significant; bits 23:30 contain the 8-bit biased exponent, e, with bit 23 being the least significant bit of the biased exponent and bit 30 being the most significant; and the highest-order bit 31 contains the sign bit, s.



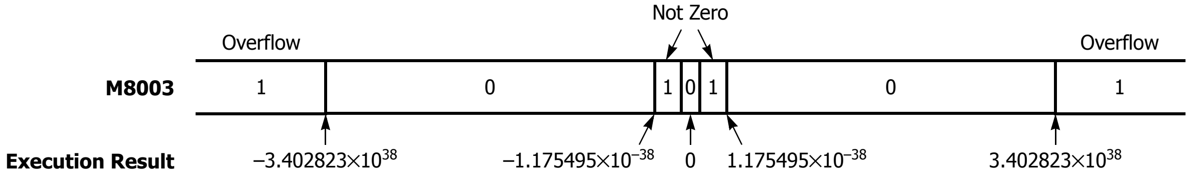
The table below shows the correspondence between the values of the three constituent fields s, e, and f and the value represented by the single storage format bit pattern. When any value out of the bit pattern is entered to the advanced instruction or when execution of advanced instructions, such as division by zero, has produced any value out of the bit pattern, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

Value	Exponent Field e	Fraction Field f	Representation in WindLDR
±0	e=0	f=0	0.0
Denormal numbers	e=0	f≠0	-1.175494E-38 to 1.175494E-38
Normal numbers	0<e<255	Arbitrary	-3.402823E+38 to -1.175494E-38 1.175494E-38 to 3.402823E+38
±∞ (± infinities)	e=255	f=0	INF
NaNs (Not a number values)		f≠0	NAN

Carry and Borrow in Floating-Point Data Processing

When advanced instructions involving floating-point data are executed, special internal relay M8003 (carry and borrow) is updated.

M8003	Execution Result	Value
1	$\neq 0$	Overflow (exceeds the range between -3.402823×10^{38} and 3.402823×10^{38})
1	0	Not zero (within the range between -1.175495×10^{-38} and 1.175495×10^{-38})
0	0	Zero



32-bit Data Storage

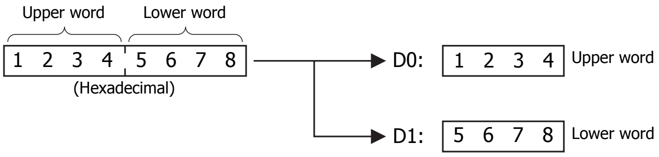
The 32-bit data for D (double word) and L (long word) is stored in devices according to the method selected in WindLDR under **Device Settings** in **Function Area Settings**.

For applicable devices and instructions, see Chapter 5 "32-bit Data Storage Setting" in the "FC6A Series MICROSmart User's Manual".

Word devices[*1]: The data storage when **From Upper Word** is selected under **Device Settings**

When D0 is specified as the source or destination device, the upper word is stored in D0 and the lower word is stored in D1.

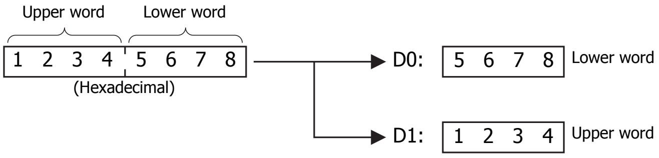
Double word data (constant)



The data storage when **From Lower Word** is selected under **Device Settings**

When D0 is specified as the source or destination device, the lower word is stored in D0 and the upper word is stored in D1.

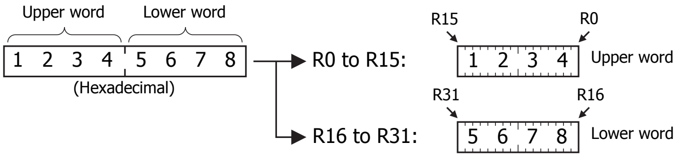
Double word data (constant)



Bit devices[*1]: The data storage when **From Upper Word** is selected under **Device Settings**

When R0 is specified as the source or destination device, the upper word is stored in R0 to R15 and the lower word is stored in R16 to R31.

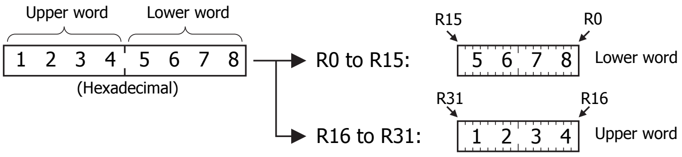
Double word data (constant)



The data storage when **From Lower Word** is selected under **Device Settings**

When R0 is specified as the source or destination device, the lower word is stored in R0-R15 and the upper word is stored in R16-R31.

Double word data (constant)



[*1] For bit devices and word devices, see "Devices" on page 2-1.

User Program Execution Errors

When an advanced instruction is executed, a user program execution error occurs when any of the following conditions are met.

- the result of the adavnced instruction is invalid
- source or destination device that is indirectly specified in the advanced instruction exceeds the valid device range
- the advanced instruction does not operate correctly

For example, the data in the source device does not comply with the normal floating-point format when the data type is Float (F).
For details about device ranges, see "Devices" on page 2-1.

When a user program execution error occurs, special internal relay M8004 turns on and the corresponding error code is stored in special data register D8006. For details about the error codes, see "User Program Execution Error" on page A-12.

When a user program execution error occurs, instructions operate as follows:

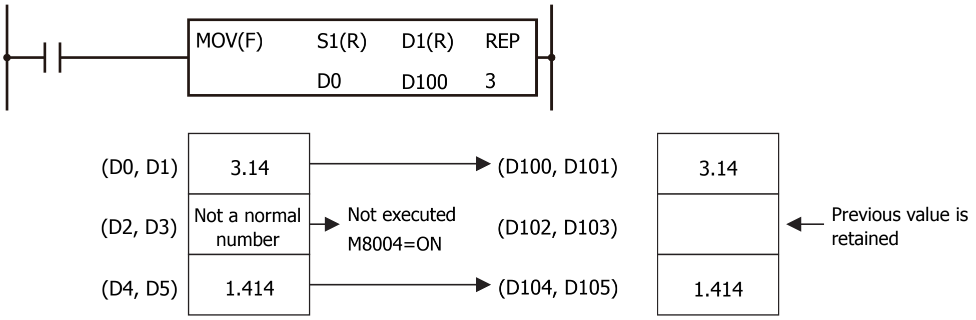
- When the source data is invalid, the execution of the advanced instruction is canceled and the data in the destination device is not changed.
- When the execution result is invalid, a value is stored in the destination device.
  For details on stored values, see the descriptions of relevant instructions.
- If a user program execution error occurred during a repeat operation, the operation is canceled and the next repeat operation is executed.
  M8004 is retained even when no further user program execution errors occur during subsequent repeat operations.

Example: User program execution error during a repeat operation

When the source data does not comply with the normal floating-point format.

When the second repeat operation is executed, special internal relay M8004 turns on because the source data is not a normal number in floating-point format.

The second repeat operation is canceled and the third repeat operation is executed.



Carry and Borrow

When the result of an operation exceeds the valid device range, a carry (CY) or borrow (BW) occurs.

Carrying and borrowing occur in the following conditions according to the data type:

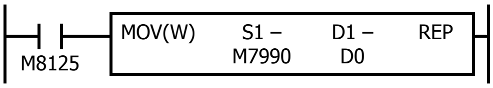
Data Type	Status
Word	Exceeds range between 0 and 65,535
Integer	Exceeds range between -32,768 and 32,767
Double Word	Exceeds range between 0 and 4,294,967,295
Long	Exceeds range between -2,147,483,648 and 2,147,483,647
Float	If an overflow or underflow occurs For details on overflow and underflow, see "Carry and Borrow in Floating-Point Data Processing" on page 3-8.

When a carry or borrow occurs, special internal relay M8003 (carry and borrow) turns on.

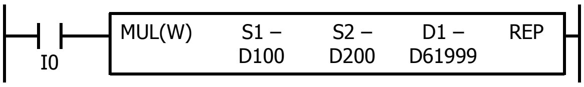
For example, when D0 has a value of FFFF (hex) and +1 is added using an INC instruction, the result is 10000 (hex), but when the data type is Word (W), 0000 (hex) is stored in D0 and 1 is stored in M8003.

Discontinuity of Device Areas

Each device area is discrete and does not continue, for example, from input to output or from output to internal relay. In addition, special internal relays M8000 through M8997 are in a separate area from internal relays M0 through M7997 and M10000 through M21247. Data registers D0 through D7999, D10000 through D61999, and D70000 through D269999, and special data registers D8000 through D8899 are in separate areas and do not continue with each other.

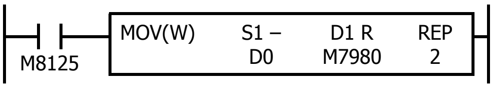


The internal relay ends at M7997. Since the MOV (move) instruction reads 16 internal relays, the last internal relay exceeds the valid range, resulting in a user program syntax error.



This program results in a user program syntax error. The destination of the MUL (multiplication) instruction requires two data registers D61999 and D62000. Since D62000 exceeds the valid range, a user program syntax error occurs.

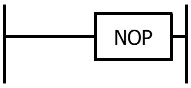
Advanced instructions execute operation only on the available devices in the valid area. If a user program syntax error is found during programming, WindLDR rejects the program instruction and shows an error message.



The MOV (move) instruction sets data of data register D0 to 16 internal relays M7980 through M7997 in the first repeat cycle. The destination of the second cycle is the next 16 internal relays M8000 through M8017, which are invalid, resulting in a user program syntax error.

For details about repeat operations of each advanced instruction, see the following chapters.

NOP (No Operation)



No operation is executed by the NOP instruction.

The NOP instruction may serve as a place holder. It can also be used to add a delay to the CPU scan time, in order to simulate communication with a machine or application, for debugging purposes.

The NOP instruction does not require an input and device.

Details of the other advanced instructions are described in the following chapters.

Device Addressing for Instruction Execution

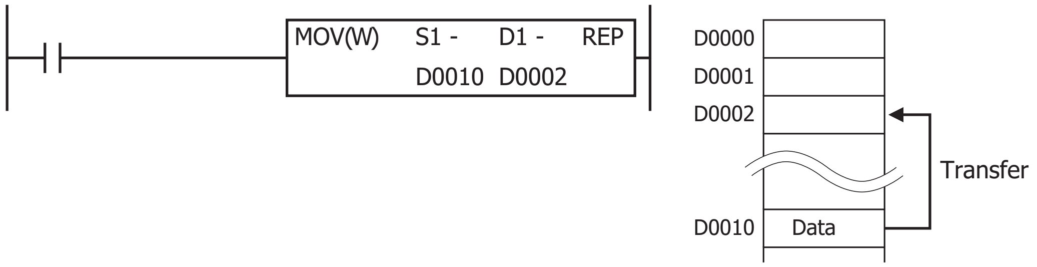
This section provides an explanation about how source and destination devices used by instructions for the execution can be specified. There are two ways to specify the device to use: Direct Addressing and Indirect Addressing.

Direct Addressing

In direct addressing, source and destination devices are specified with the devices to use.

■ Direct Addressing Operation

Example: When the source device (D0010) of the MOV(W) is directly specified.
The data in D0010 is transferred to D0002.



■ Devices that can be directly addressed

I	Q	M	R	T	C	D	P	Constant
X	X	X	X	X	X	X	X	X

■ Instructions supporting the direct addressing
All basic and advanced instructions support the direct addressing.

Indirect Addressing

Indirect Addressing Format

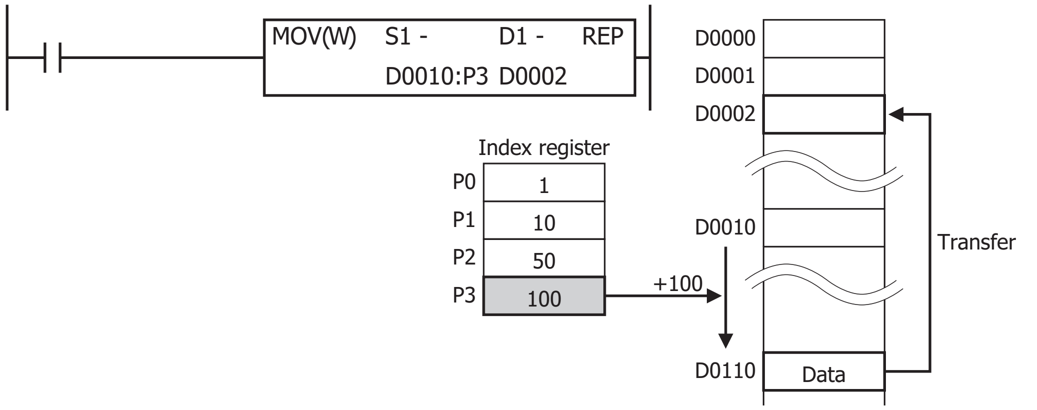
The index register Pn indicates the offset from the base device. Indirect addressing is possible by specifying source devices and destination devices with the base device and the index register. The format of the indirect addressing is as follows:

"Base Device" + ":" + "Pn (Address n of the index register)"

Example: When the base device is D0010 and the index register P3 is used.
After D0010, add ":" and "P3".
D0010:P3

Indirect Addressing Operation

Example: When the source device (D0010) of the MOV(W) instruction is indirectly specified with P3.
The data in P3 is added to the address of D0010, so D(10 + (P3)) = D (10 + (100)) = D0110, and the data in D0110 is transferred to D0002.

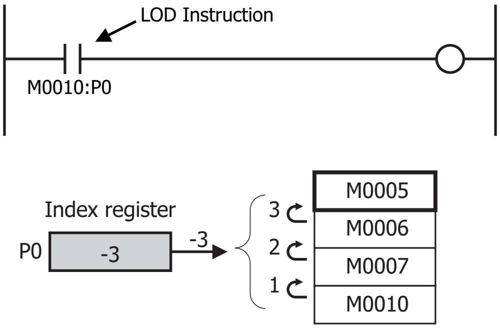


Index Registers

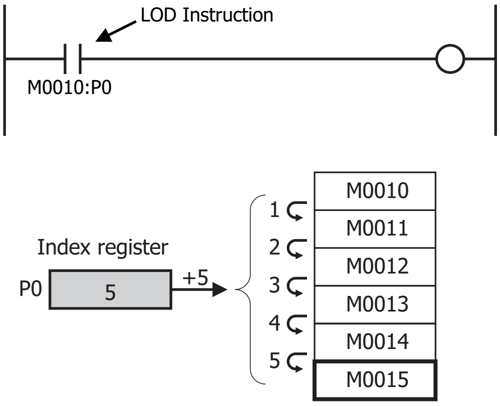
16 index registers (P0 to P15) can be used, and the data type of the index register is L (long).

Example: LOD M10:P0

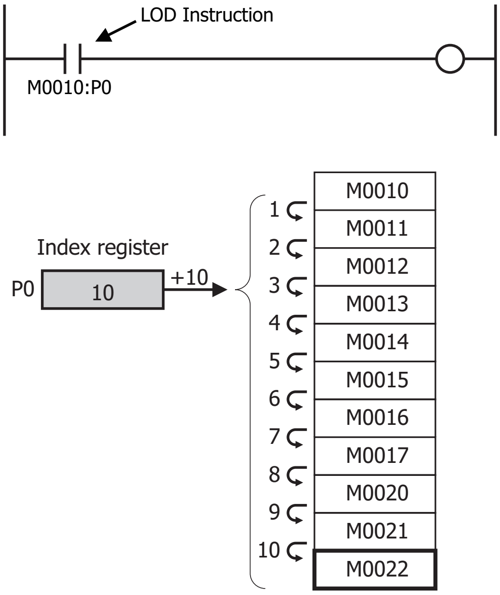
- P0=-3
  This indirect addressing indicates M0005, which is 3 bits behind of M0010.



- P0=5
  This indirect addressing indicates M0015, which is five bits ahead of M0010.

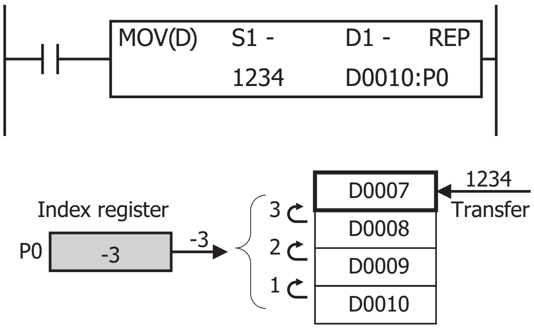


- P0=10
  This indirect addressing indicates M0022, which is 10 bits ahead of M0010.

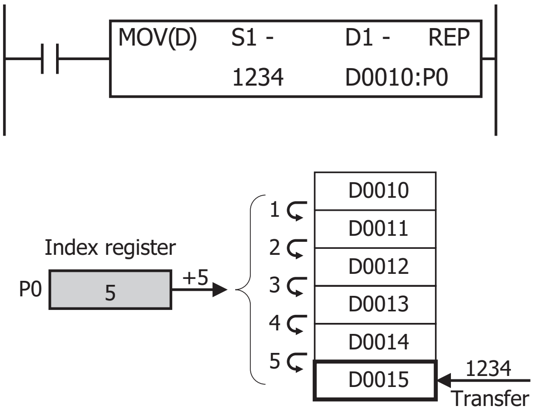


Example: MOV(D) 1234 D10:P0

- P0=-3
  This indirect addressing indicates D0007, which is 3 words behind of D0010.



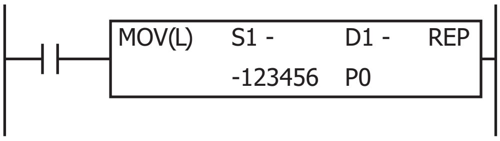
- P0=5
  This indirect addressing indicates D0015, which is 5 words ahead of D0010.



Values stored in the index registers is 32-bit and its data type is L (long). Index registers can directly specified as source and destination devices in MOV(L), ADD(L), SUB(L), and MUL(L) instructions and values in index registers can be moved or manipulated.

Example: MOV(L) -123456 P0

- -123456 is transferred to P0.



Notes:

- Index registers cannot be used in scripts.
- Indirect addressing to a different device type is not possible.
- Each of the following ranges of data registers is handled as a different device from the others:
  All-in-One CPU module: D0 to D7999, D8000 to D8499, D10000 to D55999
  Plus CPU module: D0 to D7999, D8000 to D8899, D10000 to D61999, D70000 to D269999
- If indirectly specified devices are beyond the device range, a user program execution error (1: Source/destination device exceeds range) will occur.
- When the indirect addressing is used, the base device and index register cannot be specified with the tag names.
- Indirect addressing for data register bits is not supported.
- The four instructions that can be used to modify the values of the index registers are MOV(L), ADD(L), SUB(L), and MUL(L). Specify P (index register) directly in those instructions.

Devices that can be indirectly addressed

I	Q	M	R	T	C	D	P	Constant
X	X	X	X	X	X	X	—	—

Instructions supporting the indirect addressing

■ Basic Instructions

Command	Indirect Addressing	Comments
LOD, LODN	X	Not possible when specifying a bit in a data register.
OUT, OUTN	X	Not possible when specifying a bit in a data register.
SET, RST	X	Not possible when specifying a bit in a data register.
AND, ANDN	X	Not possible when specifying a bit in a data register.
OR, ORN	X	Not possible when specifying a bit in a data register.
AND LOD	—	

Command	Indirect Addressing	Comments
OR LOD	—	
BPS, BRD, BPP	—	
TML, TIM, TMH, TIMS, TMLO, TIMO, TMHO, TIMSO	—	Indirect addressing is not supported because unique numbers are allocated.
CNT, CDP, CUD, CNTD, CDPD, CUDD	—	Indirect addressing is not supported because unique numbers are allocated.
CC=, CC>=, DC=, DC>=	—	
SFR, SFRN	—	
SOTU, SOTD	—	
MCS, MCR	—	
JMP, JEND	—	
END	—	

■ Advanced Instructions

Instruction	Indirect Addressing	Comments
NOP	—	
MOV, MOVN	X	
IMOV, IMOVN	—	
MOVC	X	
BMOV	X	
IBMV, IBMVN	—	
NSET	X	
NRS	X	
XCHG	X	
TCCST	X	
CMP=, CMP<>, CMP<, CMP>, CMP<=, CMP>=	X	
ICMP>=	X	
LC=, LC<>, LC<, LC>, LC<=, LC>=	X	
ADD	X	
SUB	X	
MUL	X	
DIV	X	
INC	X	
DEC	X	
ROOT	X	
SUM	X	
RNDM	X	
ANDW	X	
ORW	X	
XORW	X	
SFTL, SFTR	X	
BCDLS	X	
WSFT	X	
ROTL, ROTR	X	
HTOB	X	
BTOH	X	
HTOA	X	
ATOH	X	
BTOA	X	
ATOB	X	
ENCO	—	
DECO	—	
BCNT	X	

Instruction	Indirect Addressing	Comments
ALT	X	
CVDT	X	
DTDV	X	
DTCB	X	
SWAP	X	
WKTIM	—	
WKTBL	—	
WEEK	—	
YEAR	—	
MSG	—	
DISP	—	
DGRD	—	
LABEL	—	
LJMP	—	
LCAL	—	
LRET	—	
DJNZ	—	
IOREF	—	
HSCRF	—	
FRQRF	—	
COMRF	—	
DI	—	
EI	—	
XYFS	—	
CVXTY	—	
CVYTX	—	
AVRG	—	
PULS	—	
PWM	—	
RAMP	—	
RAMPL	—	
ZRN	—	
ARAMP	—	
ABS	—	
JOG	—	
PID	—	
PIDA	—	
PIDD	—	
DTML, DTIM, DTMH, DTMS	—	
TTIM	—	
RAD	X	
DEG	X	
SIN	X	
COS	X	
TAN	X	
ASIN	X	
ACOS	X	
ATAN	X	
LOGE	X	
LOG10	X	
EXP	X	
POW	X	
FIFOF	—	
FIEX	—	
FOEX	—	

Instruction	Indirect Addressing	Comments
NDSRC	—	
TADD	—	
TSUB	—	
HOUR	—	
HTOS	—	
STOH	—	
DLOG	—	
TRACE	—	
SCRPT	—	Index registers cannot be used in scripts.
SCALE	—	
FLWA	—	
FLWP	—	
UMACRO	—	Indirect addressing is possible in the instructions programmed in user-defined macros.
TXD, RXD, ETXD, ERXD	—	
PING	—	
EMAIL	—	

4: BASIC INSTRUCTIONS

Introduction

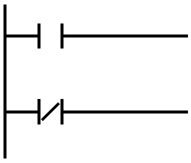
This chapter describes the basic instructions that perform sequence control.

LOD (Load) and LODN (Load Not)

The LOD instruction starts the logical operation with a NO (normally open) contact. The LODN instruction starts the logical operation with a NC (normally closed) contact.

A total of eight LOD and/or LODN instructions can be programmed consecutively.

Ladder Diagram



Valid Devices

Instruction	I	Q	M	T	C	R	D	P
LOD LODN	0-27 30-10597	0-17 30-10597	0-7997 10000-21247	0-1999	0-511	0-225	0.0-7999.15 10000.15-61999.15	—

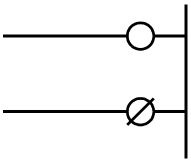
The valid device range depends on the FC6A Series MICROSmart type. For details, see "Device Addresses" on page 2-1.

Specify the bit by inserting a "." between the data register number and the bit position.

OUT (Output) and OUTN (Output Not)

The OUT instruction outputs the result of bit logical operation to the specified device. The OUTN instruction outputs the inverted result of bit logical operation to the specified device.

Ladder Diagram



Valid Devices

Instruction	I	Q	M	T	C	R	D	P
OUT OUTN	—	0-17 30-10597	0-7997 10000-21247	—	—	—	0.0-7999.15 10000.15-61999.15	—

The valid device range depends on the FC6A Series MICROSmart type. For details, see "Device Addresses" on page 2-1.

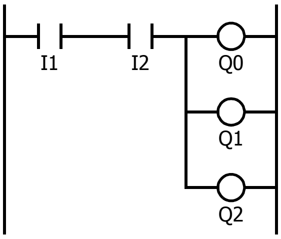
Specify the bit by inserting a "." between the data register number and the bit position.

Note: For restrictions on ladder programming of OUT and OUTN instructions, see "Restriction on Ladder Programming" on page 4-33.

Multiple OUT and OUTN

There is no limit to the number of OUT and OUTN instructions that can be programmed into one rung.

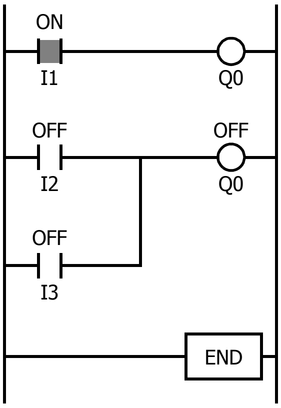
Ladder Diagram



Programming multiple outputs of the same output number is not recommended. However, if doing so, it is good practice to separate the outputs with the JMP/JEND set of instructions, or the MCS/MCR set of instructions. These instructions are detailed later in this chapter.

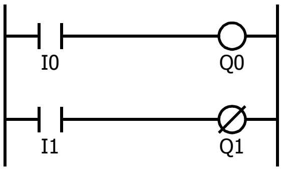
When the same output number is programmed more than once within one scan, the output nearest to the END instruction is given priority for outputting. In the example on the right, output Q0 is off.

Ladder Diagram



Examples: LOD (Load), OUT (Output), and NOT

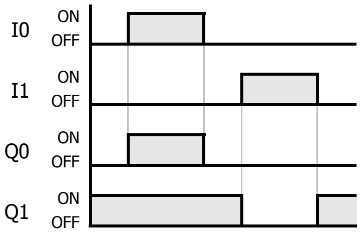
Ladder Diagram



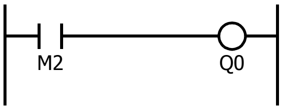
Program List

Instruction	Data
LOD	I0
OUT	Q0
LOD	I1
OUTN	Q1

Timing Chart



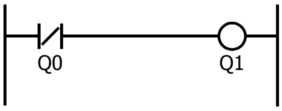
Ladder Diagram



Program List

Instruction	Data
LOD	M2
OUT	Q0

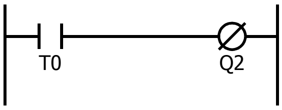
Ladder Diagram



Program List

Instruction	Data
LODN	Q0
OUT	Q1

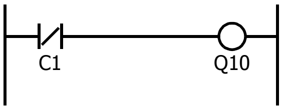
Ladder Diagram



Program List

Instruction	Data
LOD	T0
OUTN	Q2

Ladder Diagram



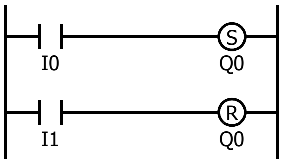
Program List

Instruction	Data
LODN	C1
OUT	Q10

SET and RST (Reset)

The SET and RST (reset) instructions are used to set (on) or reset (off) outputs, internal relays, and shift register bits. The same output can be set and reset many times within a program. SET and RST instructions operate in every scan while the input is on.

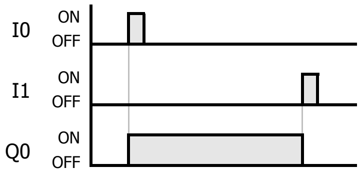
Ladder Diagram



Program List

Instruction	Data
LOD	I0
SET	Q0
LOD	I1
RST	Q0

Timing Chart



Valid Devices

Instruction	I	Q	M	T	C	R	D	P
SET RST	—	0-17 30-10597	0-7997 10000-21247	—	—	0-225	0.0-7999.15 10000.15-61999.15	—

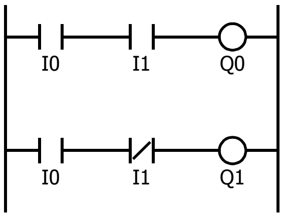
The valid device range depends on the FC6A Series MICROSmart type. For details, see "Device Addresses" on page 2-1.
Specify the bit by inserting a "." between the data register number and the bit position.

Note: For restrictions on ladder programming of SET and RST instructions, see "Restriction on Ladder Programming" on page 4-33.

AND and ANDN (And Not)

The AND instruction is used for programming a NO contact in a series. The ANDN instruction is used for programming a NC contact in a series. The AND or ANDN instruction is entered after the first set of contacts.

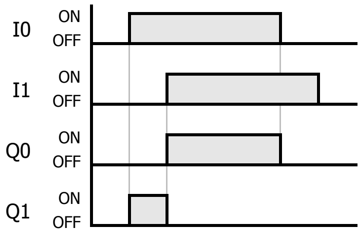
Ladder Diagram



Program List

Instruction	Data
LOD	I0
AND	I1
OUT	Q0
LOD	I0
ANDN	I1
OUT	Q1

Timing Chart



When both inputs I0 and I1 are on, output Q0 is on. When either input I0 or I1 is off, output Q0 is off.
When input I0 is on and input I1 is off, output Q1 is on. When either input I0 is off or input I1 is on, output Q1 is off.

Valid Devices

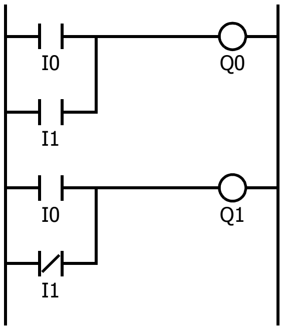
Instruction	I	Q	M	T	C	R	D	P
AND	0-27	0-17	0-7997	0-1999	0-511	0-225	0.0-7999.15	—
ANDN	30-10597	30-10597	10000-21247				10000.15-61999.15	

The valid device range depends on the FC6A Series MICROSmart type. For details, see "Device Addresses" on page 2-1.
Specify the bit by inserting a "." between the data register number and the bit position.

OR and ORN (Or Not)

The OR instruction is used for programming a NO contact in parallel. The ORN instruction is used for programming a NC contact in parallel. The OR or ORN instruction is entered after the first set of contacts.

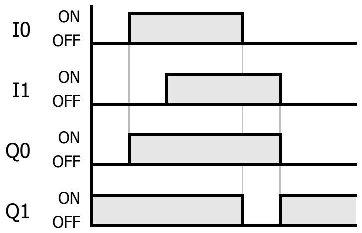
Ladder Diagram



Program List

Instruction	Data
LOD	I0
OR	I1
OUT	Q0
LOD	I0
ORN	I1
OUT	Q1

Timing Chart



When either input I0 or I1 is on, output Q0 is on. When both inputs I0 and I1 are off, output Q0 is off.
When either input I0 is on or input I1 is off, output Q1 is on. When input I0 is off and input I1 is on, output Q1 is off.

Valid Devices

Instruction	I	Q	M	T	C	R	D	P
OR	0-27	0-17	0-7997	0-1999	0-511	0-225	0.0-7999.15	—
ORN	30-10597	30-10597	10000-21247				10000.15-61999.15	

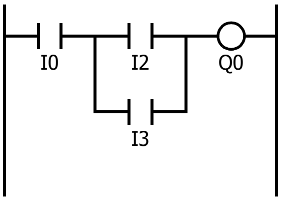
The valid device range depends on the FC6A Series MICROSmart type. For details, see "Device Addresses" on page 2-1.
Specify the bit by inserting a "." between the data register number and the bit position.

AND LOD (Load)

The AND LOD instruction is used to connect, in a series, two or more circuits starting with the LOD instruction. The AND LOD instruction is the equivalent of a "node" on a ladder diagram.

When using WindLDR, the user does not need to program the AND LOD instruction. The circuit in the ladder diagram shown below is converted into AND LOD when the ladder diagram is compiled.

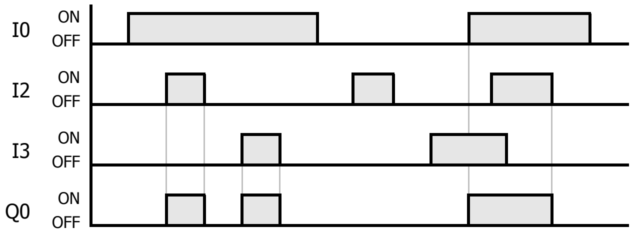
Ladder Diagram



Program List

Instruction	Data
LOD	I0
LOD	I2
OR	I3
ANDLOD	
OUT	Q0

Timing Chart



When input I0 is on and either input I2 or I3 is on, output Q0 is on.

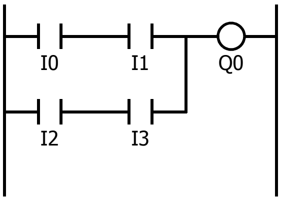
When input I0 is off or both inputs I2 and I3 are off, output Q0 is off.

OR LOD (Load)

The OR LOD instruction is used to connect, in parallel, two or more circuits starting with the LOD instruction. The OR LOD instruction is the equivalent of a "node" on a ladder diagram.

When using WindLDR, the user does not need to program the OR LOD instruction. The circuit in the ladder diagram shown below is converted into OR LOD when the ladder diagram is compiled.

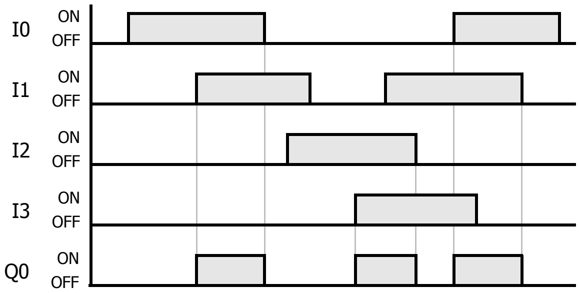
Ladder Diagram



Program List

Instruction	Data
LOD	I0
AND	I1
LOD	I2
AND	I3
ORLOD	
OUT	Q0

Timing Chart



When both inputs I0 and I1 are on or when both inputs I2 and I3 are on, output Q0 is on.

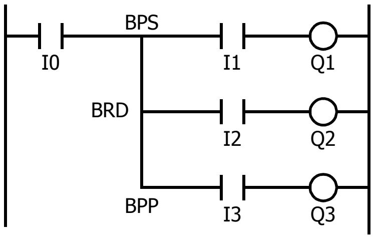
When either input I0 or I1 is off and either input I2 or I3 is off, output Q0 is off.

BPS (Bit Push), BRD (Bit Read), and BPP (Bit Pop)

The BPS (bit push) instruction is used to temporarily save the result of bit logical operation.
The BRD (bit read) instruction is used to read the result of the temporarily saved bit logical operation.
The BPP (bit pop) instruction is used to restore the result of the temporarily saved bit logical operation.

When using WindLDR, the user does not need to program the BPS, BRD, and BPP instructions. The circuit in the ladder diagram shown below is converted into BPS, BRD, and BPP when the ladder diagram is compiled.

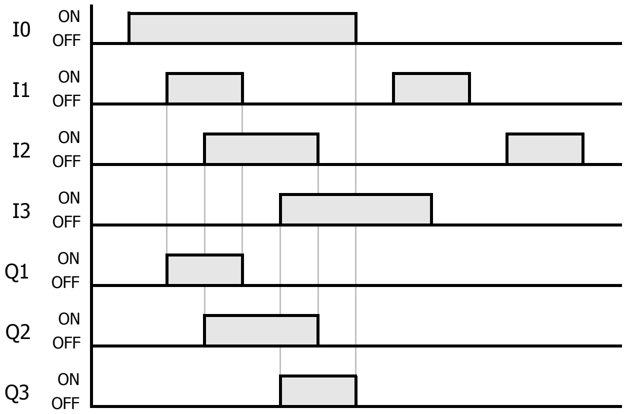
Ladder Diagram



Program List

Instruction	Data
LOD	I0
BPS	
AND	I1
OUT	Q1
BRD	
AND	I2
OUT	Q2
BPP	
AND	I3
OUT	Q3

Timing Chart



When both inputs I0 and I1 are on, output Q1 is turned on.

When both inputs I0 and I2 are on, output Q2 is turned on.

When both inputs I0 and I3 are on, output Q3 is turned on.

TML, TIM, TMH, and TMS (Timer)

Four types of on-delay timers are available; 1-s timer TML, 100-ms timer TIM, 10-ms timer TMH, and 1-ms timer TMS. A total of 2,000 on- and off-delay timers can be programmed in a user program. Each timer must be allocated to a unique number T0 through T1999.

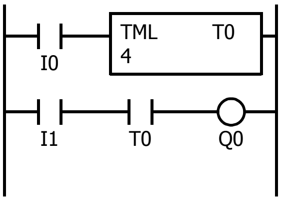
Timer	Device Address	Range	Increments	Preset Value	
TML (1-s timer)	T0 to T1999	0 to 65,535 s	1 s	Constant:	0 to 65,535
TIM (100-ms timer)	T0 to T1999	0 to 6,553.5 s	100 ms	Data registers:	D0 to D7999 D10000 to D61999
TMH (10-ms timer)	T0 to T1999	0 to 655.35 s	10 ms		
TMS (1-ms timer)	T0 to T1999	0 to 65.535 s	1 ms		

For details about device ranges, see "Device Addresses" on page 2-1.

To indirectly specify the value, specify it with a data register number, and specify the value of the data register in the range of 0 to 65,535.

TML (1-s Timer)

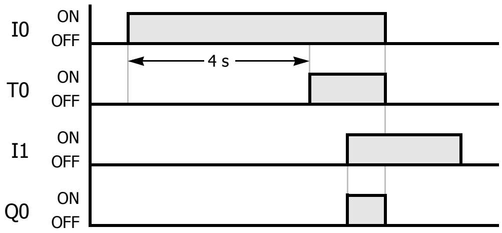
Ladder Diagram (TML)



Program List

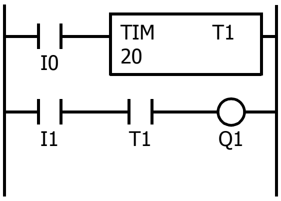
Instruction	Data
LOD	I0
TML	T0
	4
LOD	I1
AND	T0
OUT	Q0

Timing Chart



TIM (100-ms Timer)

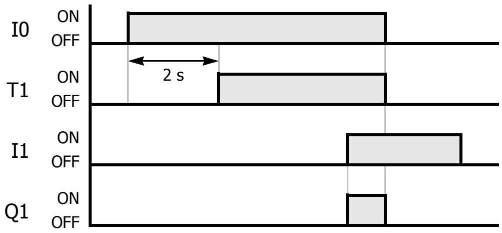
Ladder Diagram (TIM)



Program List

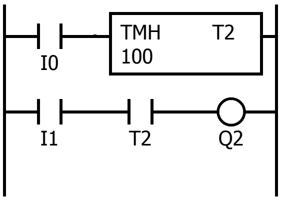
Instruction	Data
LOD	I0
TIM	T1
	20
LOD	I1
AND	T1
OUT	Q1

Timing Chart



TMH (10-ms Timer)

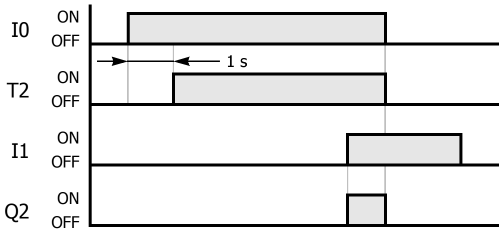
Ladder Diagram (TMH)



Program List

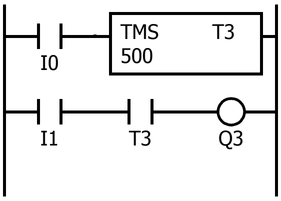
Instruction	Data
LOD	I0
TMH	T2
	100
LOD	I1
AND	T2
OUT	Q2

Timing Chart



TMS (1-ms Timer)

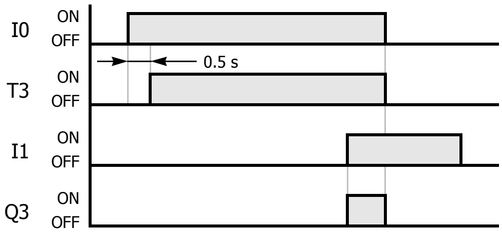
Ladder Diagram (TMS)



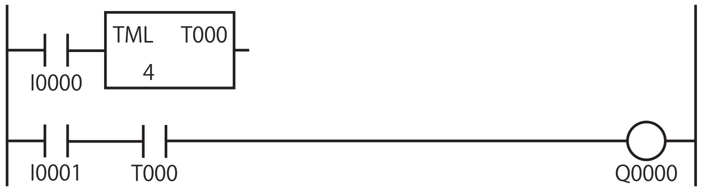
Program List

Instruction	Data
LOD	I0
TMS	T3
	500
LOD	I1
AND	T3
OUT	Q3

Timing Chart



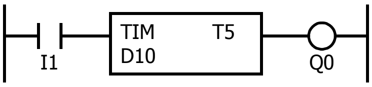
Note: Certain instructions can be programmed in a series after a timer instruction in WindLDR. These instructions are not automatically connected to the right power rail as shown in the above ladder diagram. (Refer to the below diagram.)
For details, see "Timer Circuit" on page 4-9.



Timer Circuit

The preset value 0 through 65,535 can be designated using a data register; then the data of the data register becomes the preset value. Directly after the TML, TIM, TMH, or TMS instruction, the OUT, OUTN, SET, RST, TML, TIM, TMH, TMS, TMLO, TIMO, TMHO, or TMSO instruction can be programmed.

Ladder Diagram



Program List

Instruction	Data
LOD	I1
TIM	T5 D10
OUT	Q0

Note: For restrictions on ladder programming of timer instructions, see "Restriction on Ladder Programming" on page 4-33.

- Countdown from the preset value is initiated when the operation result directly before the timer input is on.
- The timer output turns on when the current value (timed value) reaches 0.
- The current value returns to the preset value when the timer input is off.
- Timer preset and current values can be changed using WindLDR without downloading the entire program to the CPU again. From the WindLDR menu bar, select **Online** > **Monitor > Monitor**, then **Online** > **Custom** > **New Custom Monitor**.
- If a timer preset value is changed during countdown, the timer remains unchanged for that cycle. The change will be reflected in the next time cycle.
- If a timer preset value is changed to 0, then the timer stops operation, and the timer output is turned on immediately.
- If a current value is changed during countdown, the change becomes effective immediately.
- For data movement when changing, confirming, and clearing preset values, see "Changing, Confirming, and Clearing Preset Values for Timers and Counters" on page 4-19. Preset values can also be changed and confirmed using the LCD screen and pushbuttons.
- WindLDR ladder diagrams show TP (timer preset value) and TC (timer current value) in advanced instruction devices.
- The timer instructions (TML, TIM, TMH, TMS) and off-delay timer instructions (TMLO, TIMO, TMHO, TMSO) cannot be used in an interrupt program.
- If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

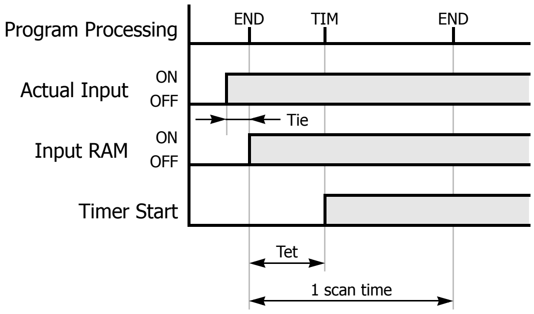
Timer Accuracy

Timer accuracy due to software configuration depends on three factors: timer input error, timer counting error, and timeout output error. These errors are not constant but vary with the user program and other causes.

Timer Input Error

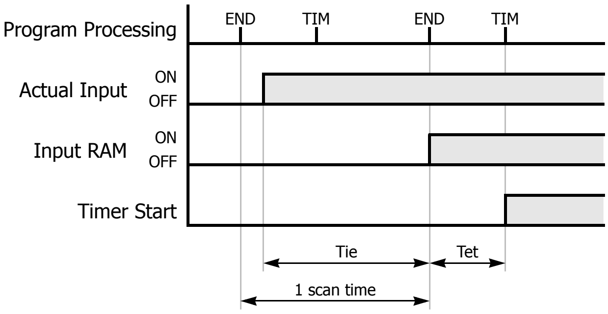
The input status is read at the END processing and stored to the input RAM. So, an error occurs depending on the timing when the timer input turns on in a scan cycle. The same error occurs on normal input and catch input. The timer input error shown below does not include input delay caused by the hardware.

Minimum Error



When the input turns on immediately before END processing, Tie is almost 0. Then the timer input error is only Tet (behind error) and is at its minimum.

Maximum Error



When the input turns on immediately after END processing, Tie is almost equal to one scan time. Then the timer input error is Tie + Tet = one scan time + Tet (behind error) and is at its maximum.

Tie: Time from input turning on to END processing
Tet: Time from END processing to the timer instruction execution

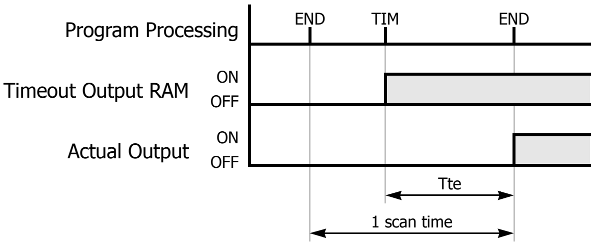
Timer Counting Error

Every timer instruction operation is individually based on asynchronous 16-bit reference timers. Therefore, an error can occur depending on the status of the asynchronous 16-bit timer when the timer instruction is executed. Use of a TMS (1-ms timer) is recommended to reduce advance errors.

Error		TML (1-s timer)	TIM (100-ms timer)	TMH (10-ms timer)	TMS (1-ms timer)
Maximum	Advance error	1,000 ms	100 ms	10 ms	1 ms
	Behind error	1 scan time	1 scan time	1 scan time	1 scan time

Timeout Output Error

The output RAM status is set to the actual output when the END instruction is processed. When timeout output turns from off to on, an error occurs depending on the position where the timer instruction is programmed in the user program. The timeout output error shown below does not include output delay caused by the hardware.



Timeout output error is equal to Tte (behind error) and can be between 0 and one scan time.

0 < Tte < 1 scan time

Tte: Time from the timer instruction execution to the END processing

Error Maximum and Minimum

Error		Timer Input Error	Timer Counting Error	Timeout Output Error	Total Error
Minimum	Advance error	0 (Note)	0	0 (Note)	0
	Behind error	Tet + Tie	0	Tte	0
Maximum	Advance error	0 (Note)	Increment	0 (Note)	Increment – (Tet + Tte)
	Behind error	1 scan time + Tet (1 scan time)	1 scan time	Tte (1 scan time)	2 scan times + (Tet + Tte)

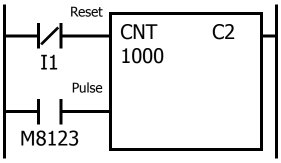
Notes: Advance error does not occur at the timer input and timeout output.
Tet + Tte = 1 scan time
Increment is 1 s (TML), 100 ms (TIM), 10 ms (TMH), or 1 ms (TMS).
The maximum advance error is: Increment – 1 scan time
The maximum behind error is: 3 scan times

The timer input error and timeout output error shown above do not include the input response time (behind error) and output response time (behind error) caused by hardware.

Power Failure Memory Protection

Timers TML, TIM, TMH, and TMS do not have power failure protection. To provide a timer with this protection use a counter instruction and special internal relay M8121 (1-s clock), M8122 (100-ms clock), or M8123 (10-ms clock).

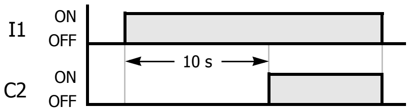
Ladder Diagram
(10-s Timer)



Program List

Instruction	Data
LODN	I1
LOD	M8123
CNT	C2
	1000

Timing Chart



Note: Designate counter C2 used in this program as a keep type counter. See Chapter 5 "Memory Backup" in the "FC6A Series MICROSmart User’s Manual".

TMLO, TIMO, TMHO, and TMSO (Off-Delay Timer)

Four types of on-delay off-delay timers are available; 1-s off-delay timer TMLO, 100-ms off-delay timer TIMO, 10-ms off-delay timer TMHO, and 1-ms off-delay timer TMSO. A total of 2,000 on- and off-delay timers can be programmed in a user program. Each timer must be allocated to a unique number T0 through T1999.

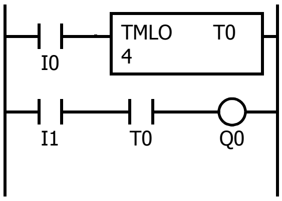
Timer	Device Address	Range	Increments	Preset Value	
TMLO (1-s off-delay timer)	T0 to T1999	0 to 65,535 s	1 s	Constant:	0 to 65,535
TIMO (100-ms off-delay timer)	T0 to T1999	0 to 6,553.5 s	100 ms	Data registers:	D0 to D7999 D10000 to D61999
TMHO (10-ms off-delay timer)	T0 to T1999	0 to 655.35 s	10 ms		
TMSO (1-ms off-delay timer)	T0 to T1999	0 to 65.535 s	1 ms		

For details about device ranges, see "Device Addresses" on page 2-1.

To indirectly specify the value, specify it with a data register number, and specify the value of the data register in the range of 0 to 65,535.

TMLO (1-s Off-delay Timer)

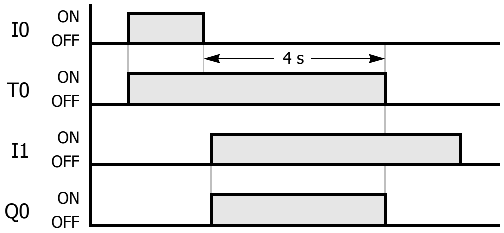
Ladder Diagram (TMLO)



Program List

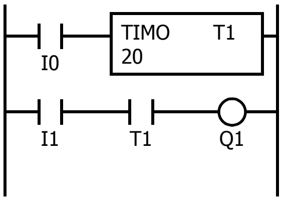
Instruction	Data
LOD	I0
TMLO	T0
	4
LOD	I1
AND	T0
OUT	Q0

Timing Chart



TIMO (100-ms Off-delay Timer)

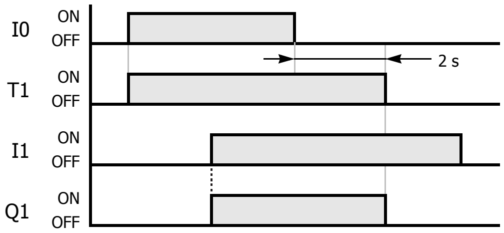
Ladder Diagram (TIMO)



Program List

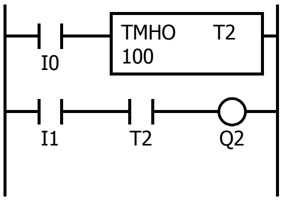
Instruction	Data
LOD	I0
TIMO	T1
	20
LOD	I1
AND	T1
OUT	Q1

Timing Chart



TMHO (10-ms Off-delay Timer)

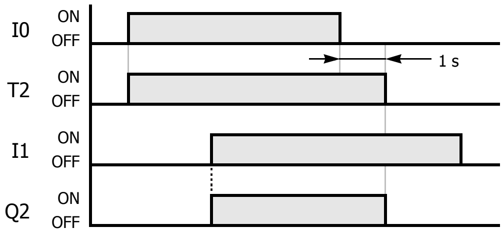
Ladder Diagram (TMHO)



Program List

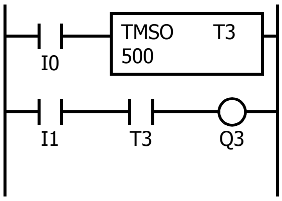
Instruction	Data
LOD	I0
TMHO	T2
	100
LOD	I1
AND	T2
OUT	Q2

Timing Chart



TMSO (1-ms Off-delay Timer)

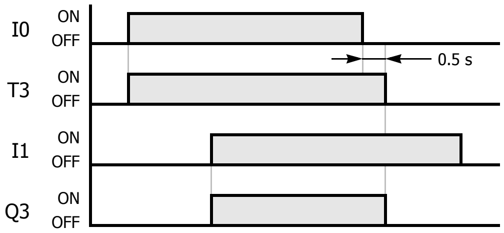
Ladder Diagram (TMSO)



Program List

Instruction	Data
LOD	I0
TMSO	T3
	500
LOD	I1
AND	T3
OUT	Q3

Timing Chart



CNT, CDP, and CUD (Counter)

Three types of counters are available; adding (up) counter CNT, dual-pulse reversible counter CDP, and up/down selection reversible counter CUD. A total of 512 counters can be programmed in a user program. Each counter must be allocated to a unique number C0 through C511.

Counter	Device Address	Preset Value	
CNT (adding counter)	C0 to C511	Constant:	0 to 65,535
CDP (dual-pulse reversible counter)	C0 to C511	Data registers:	D0 to D7999 D10000 to D61999
CUD (up/down selection reversible counter)	C0 to C511		

For details about device ranges, see "Device Addresses" on page 2-1.

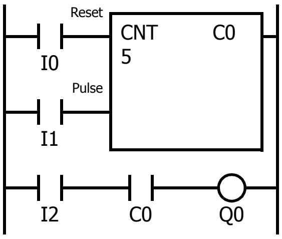
To indirectly specify the value, specify it with a data register number, and specify the value of the data register in the range of 0 to 65,535.

CNT (Adding Counter)

When counter instructions are programmed, two addresses are required. The circuit for an adding (UP) counter must be programmed in the following order: reset input, pulse input, the CNT instruction, and a counter number C0 through C511, followed by a counter preset value from 0 to 65,535.

The preset value can be designated using a decimal constant or a data register. When a data register is used, the data of the data register becomes the preset value.

Ladder Diagram

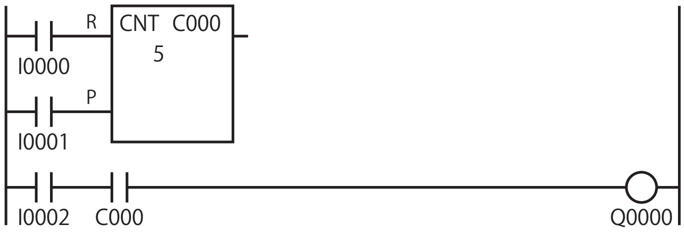


Program List

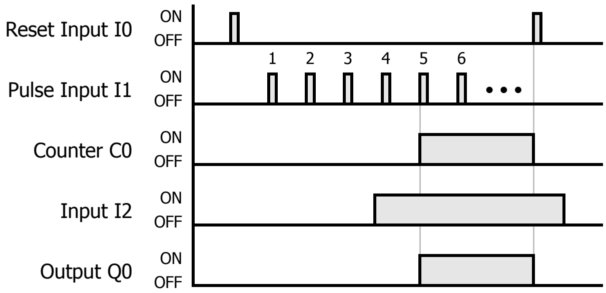
Instruction	Data
LOD	I0
LOD	I1
CNT	C0
	5
LOD	I2
AND	C0
OUT	Q0

- The CNT instruction cannot be used in an interrupt program.
- When the reset input is on, all pulse inputs are ignored.
- While the reset input is off, the counter counts the leading edges of pulse inputs and compares them with the preset value.
- When the reset input changes from off to on, the current value is reset.
- When the current value reaches the preset value, the counter turns output on. The output stays on until the reset input is turned on.

Note: Certain instructions can be programmed in a series after a counter instruction in WindLDR. These instructions are not automatically connected to the right power rail as shown in the above ladder diagram. (Refer to the below diagram.)
For details, see "Counter Circuit" on page 4-15.



Timing Chart

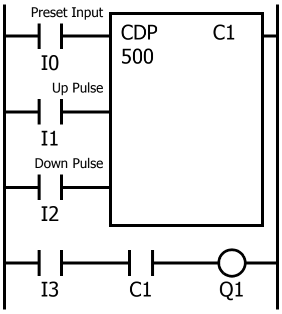


CDP (Dual-Pulse Reversible Counter)

The dual-pulse reversible counter CDP has up and down pulse inputs, so the three inputs are required. The circuit for a dual-pulse reversible counter must be programmed in the following order: preset input, up-pulse input, down-pulse input, the CDP instruction, and a counter number C0 through C511, followed by a counter preset value from 0 to 65,535.

The preset value can be designated using a decimal constant or a data register. When a data register is used, the data of the data register becomes the preset value.

Ladder Diagram

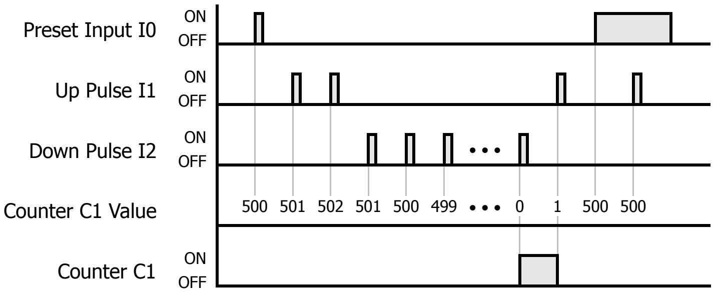


Program List

Instruction	Data
LOD	I0
LOD	I1
LOD	I2
CDP	C1
	500
LOD	I3
AND	C1
OUT	Q1

- The preset input must be turned on initially so that the current value returns to the preset value.
- The preset input must be turned off before counting may begin.
- After the current value reaches 0 (counting down), it changes to 65,535 on the next count down.
- After the current value reaches 65,535 (counting up), it changes to 0 on the next up count.
- The counter output is on only when the current value is 0.
- When the up pulse and down pulses are on simultaneously, no pulse is counted.

Timing Chart

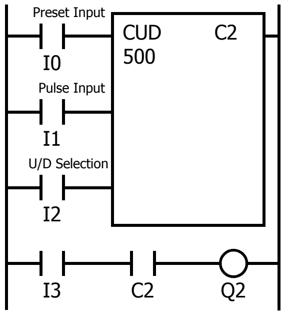


CUD (Up/Down Selection Reversible Counter)

The up/down selection reversible counter CUD has a selection input to switch the up/down gate, so the three inputs are required. The circuit for an up/down selection reversible counter must be programmed in the following order: preset input, pulse input, up/down selection input, the CUD instruction, and a counter number C0 through C511, followed by a counter preset value from 0 to 65,535.

The preset value can be designated using a decimal constant or a data register. When a data register is used, the data of the data register becomes the preset value.

Ladder Diagram

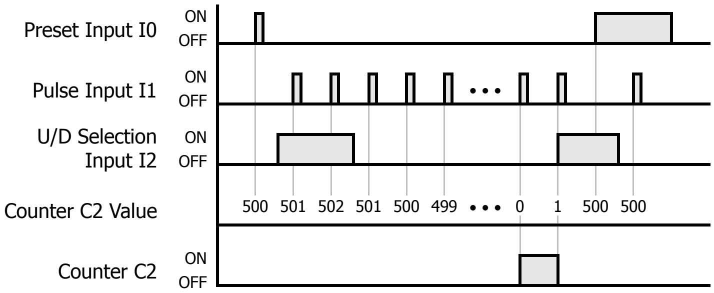


Program List

Instruction	Data
LOD	I0
LOD	I1
LOD	I2
CUD	C2
	500
LOD	I3
AND	C2
OUT	Q2

- The preset input must be turned on initially so that the current value returns to the preset value.
- The up mode is selected when the up/down selection input is on.
- The down mode is selected when the up/down selection input is off.
- After the current value reaches 0 (counting down), it changes to 65,535 on the next count down.
- After the current value reaches 65,535 (counting up), it changes to 0 on the next count up.
- The counter output is on only when the current value is 0.

Timing Chart

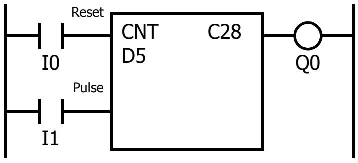


Counter Circuit

The preset value 0 through 65,535 can be designated using a data register D0 thorough D7999 or D10000 through D61999; then the data of the data register becomes the preset value.

Directly after the CNT instruction, the OUT, OUTN, SET, RST, TML, TIM, TMH, or TMS instruction can be programmed.

Ladder Diagram



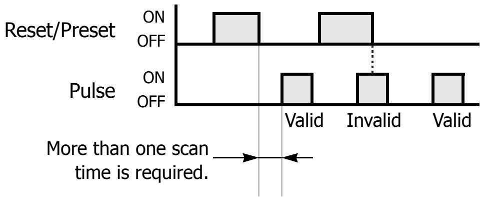
Program List

Instruction	Data
LOD	I0
LOD	I1
CNT	C28
OUT	Q0

Note: For restrictions on ladder programming of counter instructions, see "Restriction on Ladder Programming" on page 4-33.

Valid Pulse Inputs

The reset or preset input has priority over the pulse input. One scan after the reset or preset input has changed from on to off, and the counter starts counting the pulse inputs as they change from off to on.



- The reset input must be turned off before counting may begin.
- When power is off, the preset values that were changed are cleared and the original preset values are loaded.
- When power is off, the counter's current value is held, and can also be designated as "clear" type counters using Function Area Settings (see Chapter 5 "Memory Backup" in the "FC6A Series MICROSmart User's Manual".).
- Counter preset and current values can be changed using WindLDR without downloading the entire program to the CPU again. From the WindLDR menu bar, select **Online** > **Monitor** > **Monitor**, then **Online** > **Custom** > **New Custom Monitor**. Change the current value while the counter reset input is off.
- When the preset or current value is changed during counter operation, the change becomes effective immediately.
- WindLDR ladder diagrams show CP (counter preset value) and CC (counter current value) in advanced instruction devices.
- For data movement when changing, confirming, and clearing preset values, see "Changing, Confirming, and Clearing Preset Values for Timers and Counters" on page 4-19. Preset values can also be changed and confirmed using the LCD screen and pushbuttons.
- If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Counter Operation after Count out

Condition	Counter Output
When the counter has counted out, either the current value or preset value is changed.	The counter maintains the counted out status.
Before the counter has counted out, the current value is changed to a larger value than the preset.	The counter output is turned on.
The preset value is changed to 0.	The counter output is turned on without regard to the current value.
When the reset input is on, the preset value is changed to 0.	The counter output is not turned on.

CNTD, CDPD, and CUDD (Double-Word Counter)

Three types of double-word counters are available; adding (up) counter CNTD, dual-pulse reversible counter CDPD, and up/down selection reversible counter CUDD. A total of 256 double-word counters can be programmed in a user program. Each double-word counter uses 2 consecutive devices starting with the allocated device, which can be C0 through C510. Once used in a user program, counters cannot be used in any other counter instructions.

Counter	Device Address	Preset Value	
CNTD (double-word adding counter)	C0 to C510	Constant:	0 to 4,294,967,295
CDPD (double-word dual-pulse reversible counter)	C0 to C510	Data registers:	D0 to D7999 D10000 to D61999
CUDD (double-word up/down selection reversible counter)	C0 to C510		

For details about device ranges, see "Device Addresses" on page 2-1.

The preset value can be 0 through 4,294,967,295 and designated using a constant or a data register. If a data register is designated as the preset value, two consecutive data registers are used.

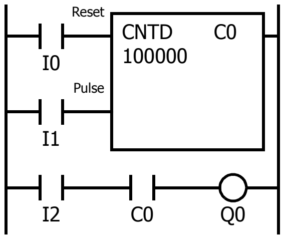
CNTD (Double-Word Adding Counter)

For a constant, specify the value in the range of 0 to 4,294,967,295.

To indirectly specify the value, specify it with a data register number, and specify the value of the data register in the range of 0 to 4,294,967,295.

The preset value can be designated using a constant or a data register. When a data register is used, the double-word data of two consecutive data registers becomes the preset value. For 32-bit data storage setting, see Chapter 5 "32-bit Data Storage Setting" in the "FC6A Series MICROSmart User's Manual".

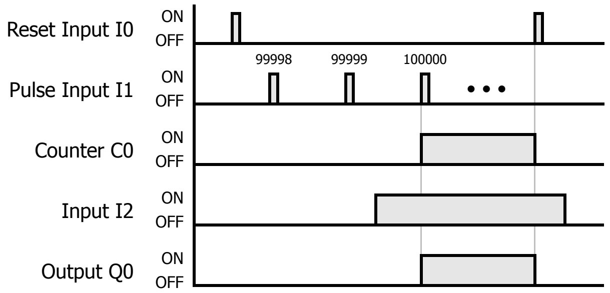
Ladder Diagram



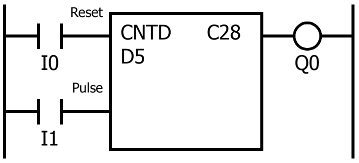
Program List

Instruction	Data
LOD	I0
LOD	I1
CNTD	C0
	100000
LOD	I2
AND	C0
OUT	Q0

Timing Chart



- The preset value 0 through 4,294,967,295 can be designated using a data register D0 through D1998; then the data of the data registers becomes the preset value.
- Directly after the CNTD instruction, the OUT, OUTN, SET, RST, TML, TIM, TMH, TMS, TMLO, TIMO, TMHO, or TMSO instruction can be programmed.



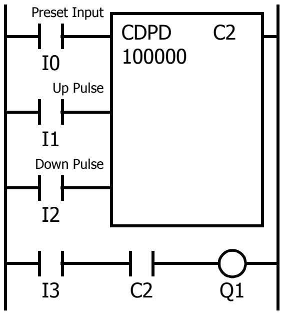
- Double-word counter instructions use two consecutive counters, and counters cannot be used more than once in a user program.
- While the reset input is off, the counter counts the leading edges of pulse inputs and compares them with the preset value.
- When the current value reaches the preset value, the counter turns output on. The output stays on until the reset input is turned on.
- When the reset input changes from off to on, the current value is reset.
- When the reset input is on, all pulse inputs are ignored.
- The reset input must be turned off before counting may begin.
- When power is off, the counter's current value is held, and can also be designated as "clear" type counters using Function Area Settings (see Chapter 5 "Memory Backup" in the "FC6A Series MICROSmart User's Manual".).
- Counter preset and current values can be changed using WindLDR without downloading the entire program to the CPU again. From the WindLDR menu bar, select **Online** > **Monitor** > **Monitor**, then **Online** > **Custom** > **New Custom Monitor**. To change a counter preset value, select DEC(D) in the pull-down list box.
- When the preset or current value is changed during counter operation, the change becomes effective immediately.
- When power is off, the changed preset values are cleared and the original preset values are loaded.
- For data movement when changing, confirming, and clearing preset values, see "Changing, Confirming, and Clearing Preset Values for Timers and Counters" on page 4-19.
- WindLDR ladder diagrams show CP (counter preset value) and CC (counter current value) in advanced instruction devices.
- The CNTD instruction cannot be used in an interrupt program.
- If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

CDPD (Double-Word Dual-Pulse Reversible Counter)

The double-word dual-pulse reversible counter CDPD has up and down pulse inputs, so the three inputs are required. The circuit for a double-word dual-pulse reversible counter must be programmed in the following order: preset input, up-pulse input, down-pulse input, the CDPD instruction, and a counter number C0 through C510, followed by a counter preset value from 0 to 4,294,967,295.

The preset value can be designated using a constant or a data register. When a data register is used, the double-word data of two consecutive data registers becomes the preset value. For 32-bit data storage setting, see Chapter 5 "32-bit Data Storage Setting" in the "FC6A Series MICROSmart User’s Manual".

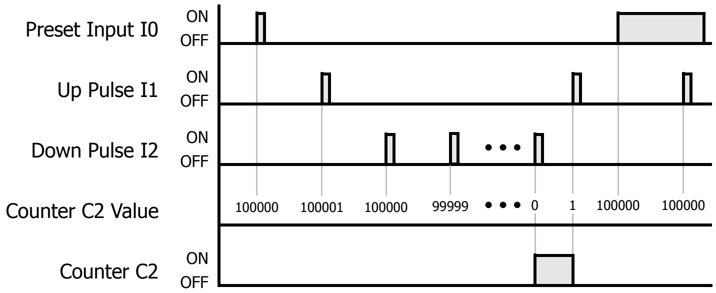
Ladder Diagram



Program List

Instruction	Data
LOD	I0
LOD	I1
LOD	I2
CDPD	C2
	100000
LOD	I3
AND	C2
OUT	Q1

Timing Chart



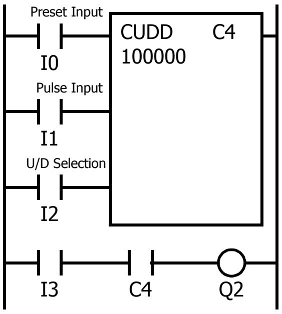
- Double-word counter instructions use two consecutive counters, and counters cannot be used more than once in a user program.
- The preset input must be turned on initially so that the current value returns to the preset value.
- The preset input must be turned off before counting may begin.
- When the up and down pulses are on simultaneously, no pulse is counted.
- The counter output is on only when the current value is 0.
- After the current value reaches 0 (counting down), it changes to 4,294,967,295 on the next count down.
- After the current value reaches 4,294,967,295 (counting up), it changes to 0 on the next count up.
- When power is off, the counter’s current value is held, and can also be designated as “clear” type counters using the Function Area Settings (see Chapter 5 "Memory Backup" in the "FC6A Series MICROSmart User’s Manual".).
- Counter preset and current values can be changed using WindLDR without downloading the entire program to the CPU again. From the WindLDR menu bar, select **Online** > **Monitor** > **Monitor**, then **Online** > **Custom** > **New Custom Monitor**. To change a counter preset value, select DEC(D) in the pull-down list box.
- When the preset or current value is changed during counter operation, the change becomes effective immediately.
- When power is off, the changed preset values are cleared and the original preset values are loaded.
- For data movement when changing, confirming, and clearing preset values, see "Changing, Confirming, and Clearing Preset Values for Timers and Counters" on page 4-19.
- WindLDR ladder diagrams show CP (counter preset value) and CC (counter current value) in advanced instruction devices. The CNPD instruction cannot be used in an interrupt program.
- If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

CUDD (Double-Word Up/Down Selection Reversible Counter)

The double-word up/down selection reversible counter CUDD has a selection input to switch the up/down gate, so the three inputs are required. The circuit for a double-word up/down selection reversible counter must be programmed in the following order: preset input, pulse input, up/down selection input, the CUDD instruction, and a counter number C0 through C510, followed by a counter preset value from 0 to 4,294,967,295.

The preset value can be designated using a constant or a data register. When a data register is used, the double-word data of two consecutive data registers becomes the preset value. For 32-bit data storage setting, see Chapter 5 "32-bit Data Storage Setting" in the "FC6A Series MICROSmart User's Manual".

Ladder Diagram

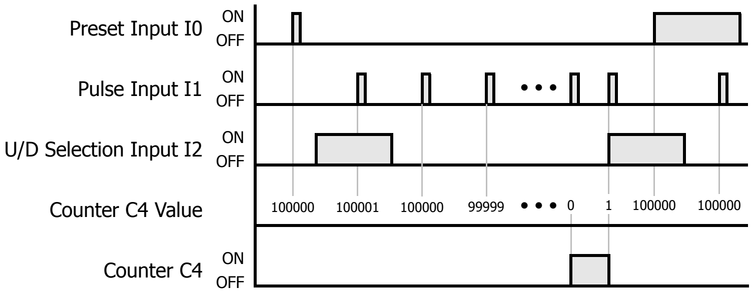


Program List

Instruction	Data
LOD	I0
LOD	I1
LOD	I2
CUDD	C4
	100000
LOD	I3
AND	C4
OUT	Q2

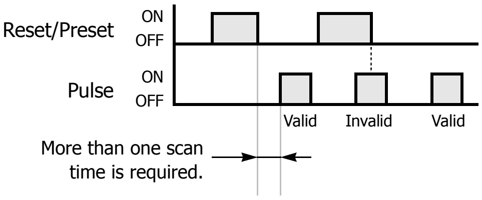
- Double-word counter instructions use two consecutive counters, and counters cannot be used more than once in a user program.
- The preset input must be turned on initially so that the current value returns to the preset value.
- The preset input must be turned off before counting may begin.
- The up mode is selected when the up/down selection input is on.
- The down mode is selected when the up/down selection input is off.
- The counter output is on only when the current value is 0.
- After the current value reaches 0 (counting down), it changes to 4,294,967,295 on the next count down.
- After the current value reaches 4,294,967,295 (counting up), it changes to 0 on the next count up.
- When power is off, the counter's current value is held, and can also be designated as "clear" type counters using the Function Area Settings (see Chapter 5 "Memory Backup" in the "FC6A Series MICROSmart User's Manual".).
- Counter preset and current values can be changed using WindLDR without downloading the entire program to the CPU again. From the WindLDR menu bar, select **Online** > **Monitor** > **Monitor**, then **Online** > **Custom** > **New Custom Monitor**. To change a counter preset value, select DEC(D) in the pull-down list box.
- When the preset or current value is changed during counter operation, the change becomes effective immediately.
- When power is off, the changed preset values are cleared and the original preset values are loaded.
- For data movement when changing, confirming, and clearing preset values, see "Changing, Confirming, and Clearing Preset Values for Timers and Counters" on page 4-19.
- WindLDR ladder diagrams show CP (counter preset value) and CC (counter current value) in advanced instruction devices. The CUDD instruction cannot be used in an interrupt program.
- If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Timing Chart



Valid Pulse Inputs

The reset or preset input has priority over the pulse input. One scan after the reset or preset input has changed from on to off, the counter starts counting the pulse inputs as they change from off to on.

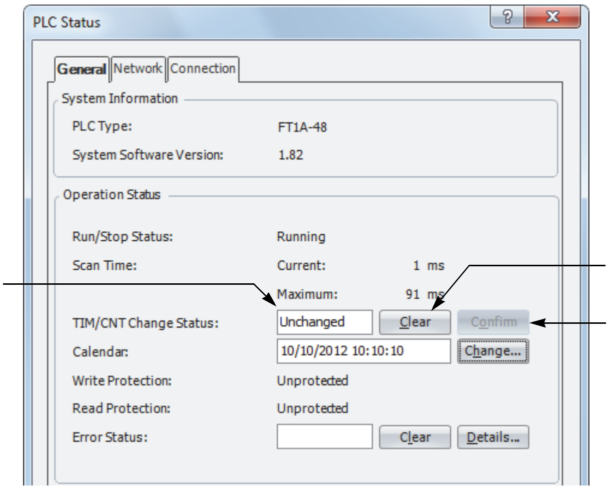


Changing, Confirming, and Clearing Preset Values for Timers and Counters

Preset values for timers and counters can be changed by selecting **Online** > **Monitor** > **Monitor**, followed by **Online** > **Custom** > **New Custom Monitor** on WindLDR for transferring a new value to the FC6A Series MICROSmart RAM as described on preceding pages. After changing the preset values temporarily, the changes can be written to the user program in the FC6A Series MICROSmart ROM or cleared from the RAM.

When the power is off, the temporarily changed preset values in the RAM are cleared and the original preset values are loaded. Confirm the changed preset values to store them in the ROM. The timer/counter preset values can be confirmed only when FC6A Series MICROSmart is stopped.

To access the PLC Status dialog box from the WindLDR menu bar, select **Online** > **Monitor** > **Monitor**, then **Online** > **Status**.



PLC Status

General | Network | Connection

System Information

PLC Type: FT1A-48

System Software Version: 1.82

Operation Status

Run/Stop Status: Running

Scan Time: Current: 1 ms

Maximum: 91 ms

TIM/CNT Change Status: Unchanged [Clear] [Confirm]

Calendar: 10/10/2012 10:10:10 [Change...]

Write Protection: Unprotected

Read Protection: Unprotected

Error Status: [Clear] [Details...]

After pressing the Clear or Confirm button, the display changes to “Unchanged.”

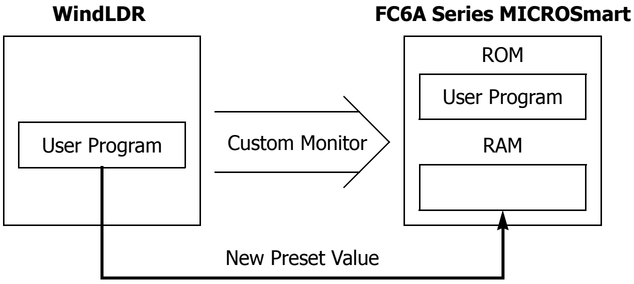
Clear Button

Confirm Button

Data movement when changing a timer/counter preset value

When changing a timer/counter preset value using Custom Monitor on WindLDR, the new preset value is written to the FC6A Series MICROSmart RAM. The user program and preset values in the ROM are not changed.

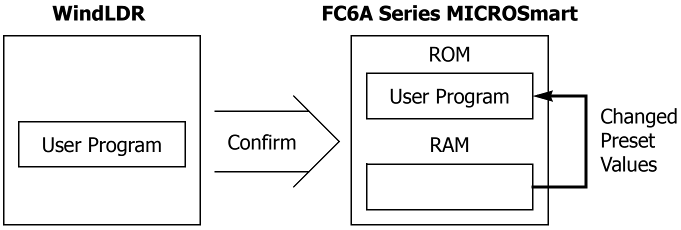
Note: The LCD screen and pushbuttons can also be used to change preset values and confirm changed preset values.



Data movement when confirming changed preset values

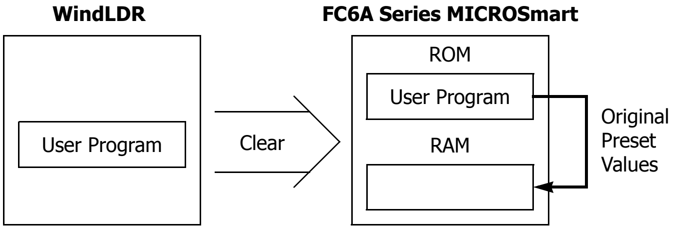
When the Confirm button is pressed before pressing the Clear button, the changed timer/counter preset values in the FC6A Series MICROSmart RAM are written to the ROM.

When uploading the user program after confirming, the user program with changed preset values is uploaded from the FC6A Series MICROSmart ROM to WindLDR.



Data movement when clearing changed preset values to restore original values

Changing preset values for timers and counters in the FC6A Series MICROSmart RAM does not automatically update preset values in the user memory, ROM. This is useful for restoring original preset values. When the Clear button is pressed before pressing the Confirm button, the changed timer/counter preset values are cleared from the RAM and the original preset values are loaded from the ROM to the RAM.



CC= and CC>= (Counter Comparison)

The CC= instruction is an equivalent comparison instruction for counter current values. This instruction will constantly compare current values to the value that has been programmed in. When the counter value equals the given value, the desired output will be initiated.

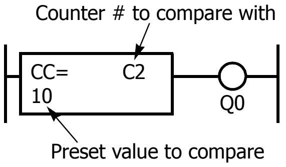
The CC>= instruction is an equal to or greater than comparison instruction for counter current values. This instruction will constantly compare current values to the value that has been programmed in. When the counter value is equal to or greater than the given value, the desired output will be initiated.

When a counter comparison instruction is programmed, two addresses are required. For a constant, specify the value in the range of 0 to 65,535.

To indirectly specify the value, specify it with a data register number, and specify the value of the data register in the range of 0 to 65,535.

The preset value can be designated using a decimal constant or a data register. When a data register is used, the data of the data register becomes the preset value.

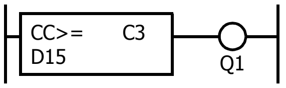
Ladder Diagram (CC=)



Program List

Instruction	Data
CC=	C2 10
OUT	Q0

Ladder Diagram (CC>=)

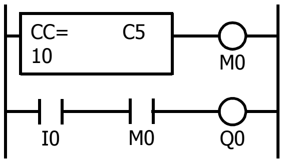


Program List

Instruction	Data
CC>=	C3 D15
OUT	Q1

- The CC= and CC>= instructions can be used repeatedly for different preset values.
- The comparison instructions only compare the current value. The status of the counter does not affect this function.
- The comparison instructions also serve as an implicit LOD instruction.
- The comparison instructions can be used with internal relays, which are ANDed or ORed at a separate program address.
- Like the LOD instruction, the comparison instructions can be followed by the AND and OR instructions.

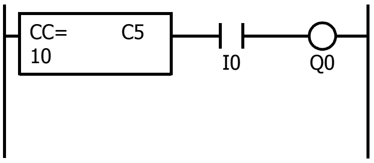
Ladder Diagram



Program List

Instruction	Data
CC=	C5 10
OUT	M0
LOD	I0
AND	M0
OUT	Q0

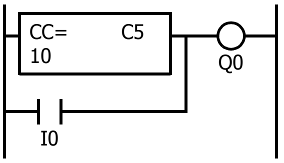
Ladder Diagram



Program List

Instruction	Data
CC=	C5 10
AND	I0
OUT	Q0

Ladder Diagram

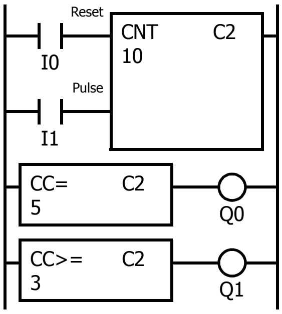


Program List

Instruction	Data
CC=	C5 10
OR	I0
OUT	Q0

Examples: CC= and CC>= (Counter Comparison)

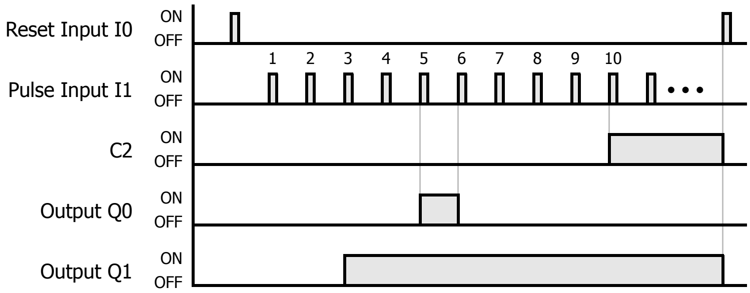
Ladder Diagram 1



Program List

Instruction	Data
LOD	I0
LOD	I1
CNT	C2
	10
CC=	C2
	5
OUT	Q0
CC³	C2
	3
OUT	Q1

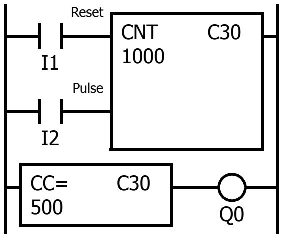
Timing Chart



Output Q0 is on when counter C2 current value is 5.

Output Q1 is turned on when counter C2 current value reaches 3 and remains on until counter C2 is reset.

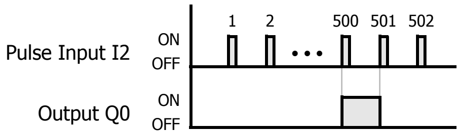
Ladder Diagram 2



Program List

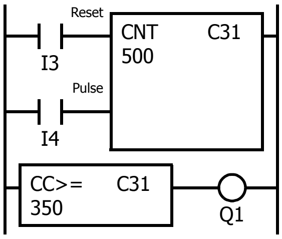
Instruction	Data
LOD	I1
LOD	I2
CNT	C30
	1000
CC=	C30
	500
OUT	Q0

Timing Chart



Output Q0 is on when counter C30 current value is 500.

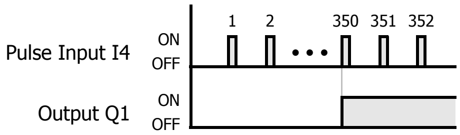
Ladder Diagram 3



Program List

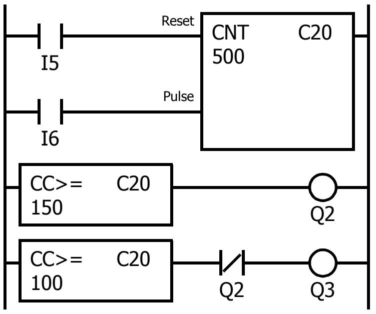
Instruction	Data
LOD	I3
LOD	I4
CNT	C31
	500
CC>=	C31
	350
OUT	Q1

Timing Chart



Output Q1 is turned on when counter C31 current value reaches 350 and remains on until counter C31 is reset.

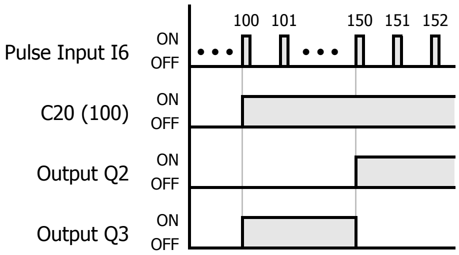
Ladder Diagram 4



Program List

Instruction	Data
LOD	I5
LOD	I6
CNT	C20
	500
CC>=	C20
	150
OUT	Q2
CC>=	C20
	100
ANDN	Q2
OUT	Q3

Timing Chart



Output Q3 is on when counter C20 current value is between 100 and 149.

When the current value of C20 is between 150 and 500, Q2 is on.

DC= and DC>= (Data Register Comparison)

The DC= instruction is an equivalent comparison instruction for data register values. This instruction will constantly compare data register values to the value that has been programmed in. When the data register value equals the given value, the desired output will be initiated.

The DC>= instruction is an equal to or greater than comparison instruction for data register values. This instruction will constantly compare data register values to the value that has been programmed in. When the data register value is equal to or greater than the given value, the desired output will be initiated.

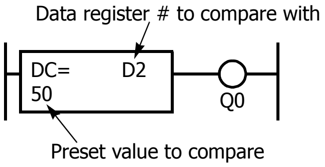
When a data register comparison instruction is programmed, two addresses are required. For a constant, specify the value in the range of 0 to 65,535.

To indirectly specify the value, specify it with a data register number, and specify the value of the data register in the range of 0 to 65,535.

The preset value can be designated using a decimal constant or a data register. When a data register is used, the data of the data register becomes the preset value.

For LC (Load Compare) instructions, see "LC= (Load Compare Equal To)" on page 6-8.

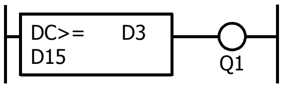
Ladder Diagram (DC=)



Program List

Instruction	Data
DC=	D2 50
OUT	Q0

Ladder Diagram (DC>=)

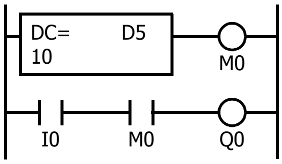


Program List

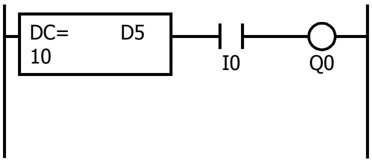
Instruction	Data
DC>=	D3 D15
OUT	Q1

- The DC= and DC>= instructions can be repeated for different preset values.
- The comparison instructions also serve as an implicit LOD instruction.
- The comparison instructions can be used with internal relays, which are ANDed or ORed at a separate program address.
- Like the LOD instruction, the comparison instructions can be followed by the AND and OR instructions.

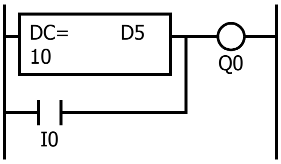
Ladder Diagram



Ladder Diagram



Ladder Diagram



Program List

Instruction	Data
DC=	D5 10
OUT	M0
LOD	I0
AND	M0
OUT	Q0

Program List

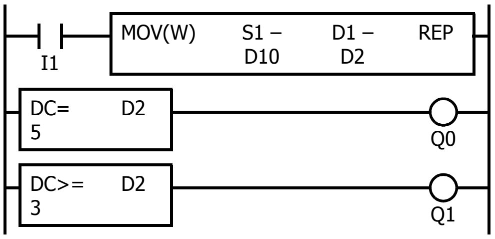
Instruction	Data
DC=	D5 10
AND	I0
OUT	Q0

Program List

Instruction	Data
DC=	D5 10
OR	I0
OUT	Q0

Examples: DC= and DC>= (Data Register Comparison)

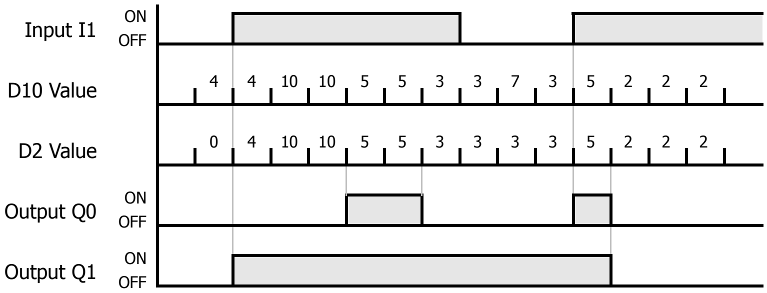
Ladder Diagram 1



Program List

Instruction	Data
LOD	I1
MOV(W)	
	D10 –
	D2 –
DC=	D2
	5
OUT	Q0
DC³	D2
	3
OUT	Q1

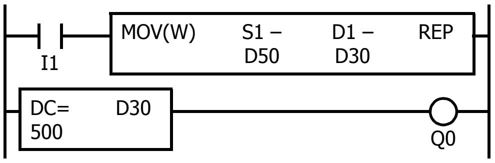
Timing Chart



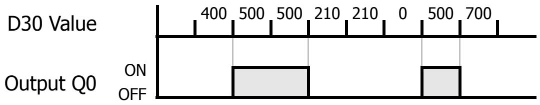
Output Q0 is on when data register D2 value is 5.

Output Q1 is on when data register D2 value is 3 or more.

Ladder Diagram 2

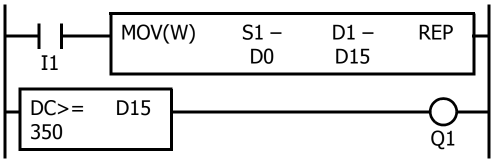


Timing Chart

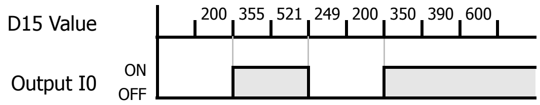


Output Q0 is on when data register D30 value is 500.

Ladder Diagram 3

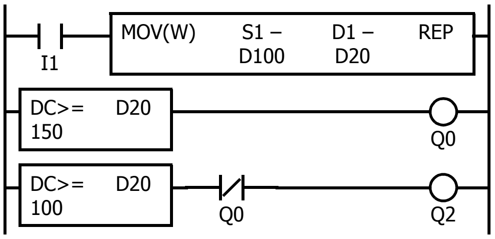


Timing Chart

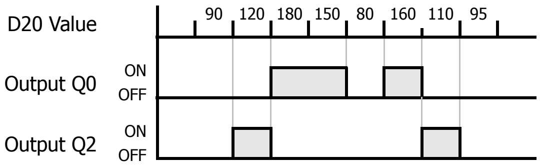


Output Q1 is on when data register D15 value is 350 or more.

Ladder Diagram 4



Timing Chart



Output Q2 is on while data register D20 value is between 149 and 100.

When the value of D20 is 150 or higher, output Q0 is on.

SFR and SFRN (Forward and Reverse Shift Register)

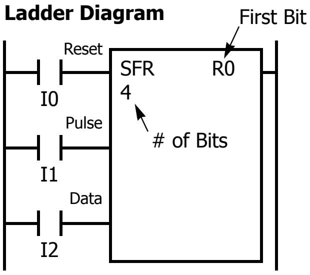
FC6A Series MICROSmart have a shift register consisting of 256 bits which are allocated to R0 through R255. Any number of available bits can be selected to form a train of bits which store on or off status. The on/off data of constituent bits is shifted in the forward direction (forward shift register) or in the reverse direction (reverse shift register) when a pulse input is turned on.

Forward Shift Register (SFR)

When SFR instructions are programmed, two addresses are always required. The SFR instruction is entered, followed by a shift register number selected from appropriate device addresses. The shift register number corresponds to the first, or head bit. The number of bits is the second required address after the SFR instruction.

The SFR instruction requires three inputs. The forward shift register circuit must be programmed in the following order: reset input, pulse input, data input, and the SFR instruction, followed by the first bit and the number of bits.

Ladder Diagram

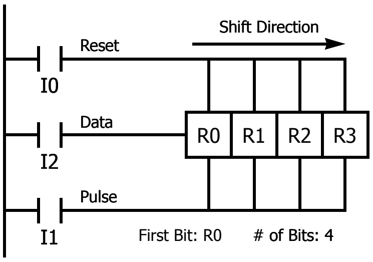


First Bit	R0 to R255
# of Bits	1 to 256

Program List

Instruction	Data
LOD LOD LOD SFR	I0 I1 I2 R0 4

Structural Diagram



Reset Input

The reset input will cause the value of each bit of the shift register to return to zero. Initialize pulse special internal relay, M8120, may be used to initialize the shift register at start-up.

Pulse Input

The pulse input triggers the data to shift. The shift is in the forward direction for a forward shift register and in reverse for a reverse shift register. A data shift will occur upon the leading edge of a pulse; that is, when the pulse *turns on*. If the pulse has been on and stays on, no data shift will occur.

Data Input

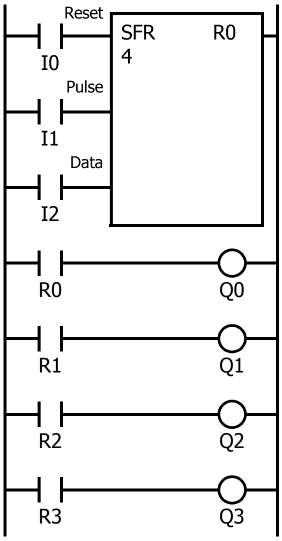
The data input is the information which is shifted into the first bit when a forward data shift occurs, or into the last bit when a reverse data shift occurs.

Note: When power is turned off, the statuses of all shift register bits are normally cleared. It is also possible to maintain the statuses of shift register bits by using the Function Area Settings as required. See Chapter 5 "Functions and Settings" - "Memory Backup" in the "FC6A Series MICROSmart User’s Manual".

Note: The SFR instruction cannot be used in an interrupt program. If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Note: For restrictions on ladder programming of shift register instructions, see "Restriction on Ladder Programming" on page 4-33.

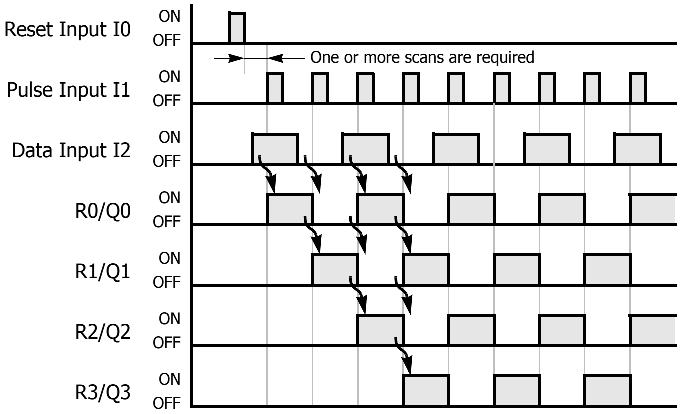
Ladder Diagram



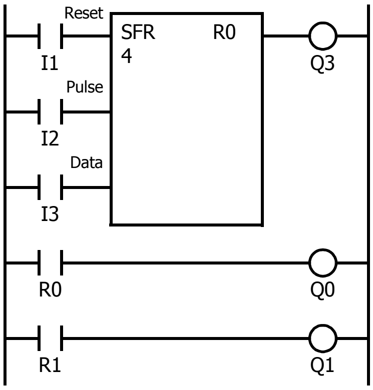
Program List

Instruction	Data
LOD	I0
LOD	I1
LOD	I2
SFR	R0 4
LOD	R0
OUT	Q0
LOD	R1
OUT	Q1
LOD	R2
OUT	Q2
LOD	R3
OUT	Q3

Timing Chart



Ladder Diagram

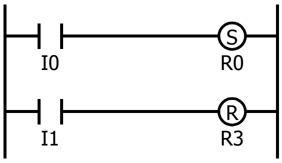


Program List

Instruction	Data
LOD	I1
LOD	I2
LOD	I3
SFR	R0 4
OUT	Q3
LOD	R0
OUT	Q0
LOD	R1
OUT	Q1

- The last bit status output can be programmed directly after the SFR instruction. In this example, the status of bit R3 is read to output Q3.
- Each bit can be loaded using the LOD R# instruction.

Setting and Resetting Shift Register Bits



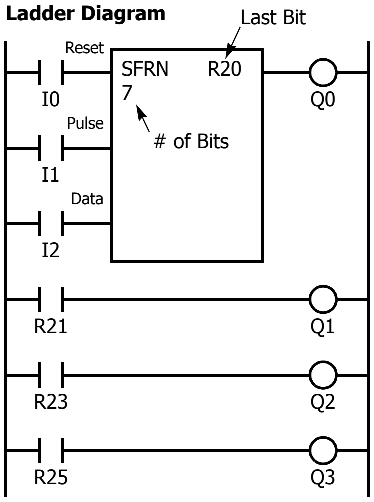
- Any shift register bit can be turned on using the SET instruction.
- Any shift register bit can be turned off using the RST instruction.
- The SET or RST instruction is actuated by any input condition.

Reverse Shift Register (SFRN)

For reverse shifting, use the SFRN instruction. When SFRN instructions are programmed, two addresses are always required. The SFRN instructions are entered, followed by a shift register number selected from appropriate device addresses. The shift register number corresponds to the lowest bit number in a string. The number of bits is the second required address after the SFRN instructions.

The SFRN instruction requires three inputs. The reverse shift register circuit must be programmed in the following order: reset input, pulse input, data input, and the SFRN instruction, followed by the last bit and the number of bits.

Ladder Diagram



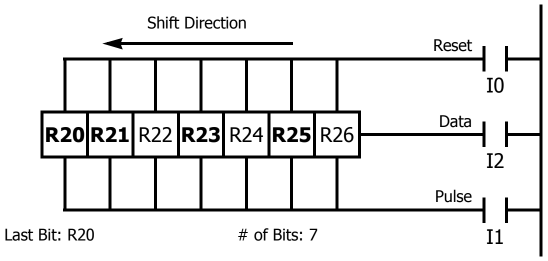
Last Bit	R0 to R255
# of Bits	1 to 256

Program List

Instruction	Data
LOD	I0
LOD	I1
LOD	I2
SFRN	R20
	7
OUT	Q0
LOD	R21
OUT	Q1
LOD	R23
OUT	Q2
LOD	R25
OUT	Q3

- The last bit status output can be programmed directly after the SFRN instruction. In this example, the status of bit R20 is read to output Q0.
- Each bit can be loaded using the LOD R# instructions.
- For details of reset, pulse, and data inputs, see "Forward Shift Register (SFR)" on page 4-24.

Structural Diagram



Note: Output is initiated only for those bits highlighted in bold print.

Note: When power is turned off, the statuses of all shift register bits are normally cleared. It is also possible to maintain the statuses of shift register bits by using the Function Area Settings as required. See Chapter 5 "Functions and Settings" - "Memory Backup" in the "FC6A Series MICROSmart User's Manual".

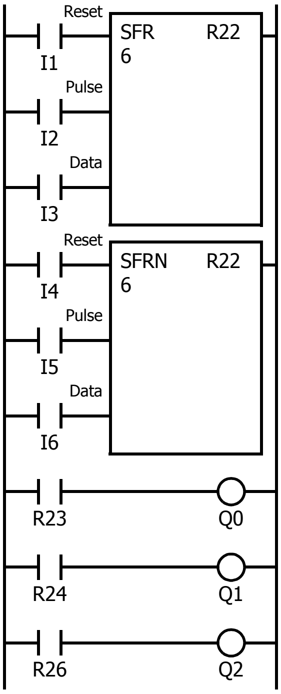
Note: The SFRN instruction cannot be used in an interrupt program. If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Note: For restrictions on ladder programming of shift register instructions, see "Restriction on Ladder Programming" on page 4-33.

Bidirectional Shift Register

A bidirectional shift register can be created by first programming the SFR instruction as detailed in the "Forward Shift Register (SFR)" on page 4-24. Next, the SFRN instruction is programed as detailed in the "Reverse Shift Register (SFRN)" on page 4-26.

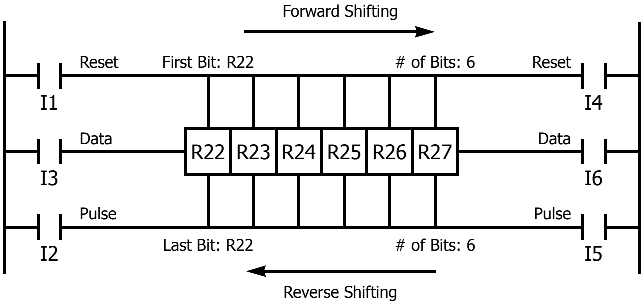
Ladder Diagram



Program List

Instruction	Data
LOD	I1
LOD	I2
LOD	I3
SFR	R22 6
LOD	I4
LOD	I5
LOD	I6
SFRN	R22 6
LOD	R23
OUT	Q0
LOD	R24
OUT	Q1
LOD	R26
OUT	Q2

Structural Diagram



Note: Output is initiated only for those bits highlighted in bold print.

SOTU and SOTD (Single Output Up and Down)

The SOTU instruction "looks for" the transition of a given input from off to on. The SOTD instruction looks for the transition of a given input from on to off. When this transition occurs, the desired output will turn on for the length of one scan. The SOTU or SOTD instruction converts an input signal to a "one-shot" pulse signal.

A total of 3,072 SOTU and SOTD instructions can be used in a user program.

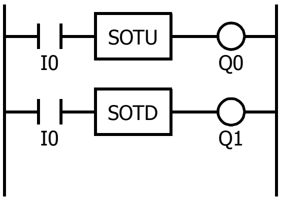
If operation is started while the given input is already on, the SOTU output will not turn on. The transition from off to on is what triggers the SOTU instruction.

The SOTU or SOTD instructions cannot be used in an interrupt program.

If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

When a CPU relay is defined as the SOTU or SOTD output, it may not operate if the scan time is not compatible with relay requirements.

Ladder Diagram

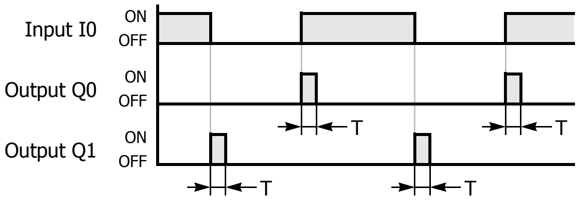


Program List

Instruction	Data
LOD	I0
SOTU	
OUT	Q0
LOD	I0
SOTD	
OUT	Q1

Note: For restrictions on ladder programming of SOTU and SOTD instructions, see "Restriction on Ladder Programming" on page 4-33.

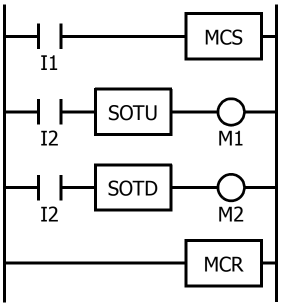
Timing Chart



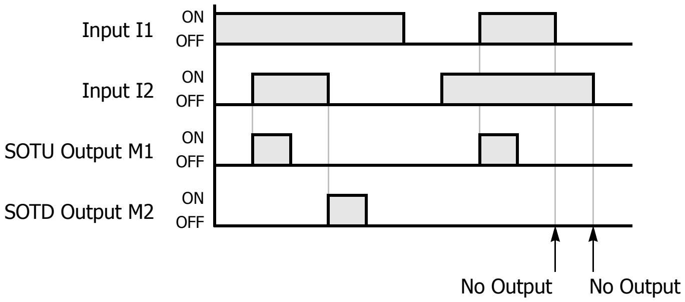
Note: "T" equals one scan time (one-shot pulse).

There is a special case when the SOTU and SOTD instructions are used between the MCS and MCR instructions (see "MCS and MCR (Master Control Set and Reset)" on page 4-29). If input I2 to the SOTU instruction turns on while input I1 to the MCS instruction is on, then the SOTU output turns on. If input I2 to the SOTD instruction turns off while input I1 is on, then the SOTD output turns on. If input I1 turns on while input I2 is on, then the SOTU output turns on. However, if input I1 turns off while input I2 is on, then the SOTD output does not turn on as shown below.

Ladder Diagram



Timing Chart



MCS and MCR (Master Control Set and Reset)

The MCS (master control set) instruction is usually used in combination with the MCR (master control reset) instruction. The MCS instruction can also be used with the END instruction, instead of the MCR instruction.

When the input preceding the MCS instruction is off, the MCS is executed so that all inputs to the portion between the MCS and the MCR are forced off. When the input preceding the MCS instruction is on, the MCS is not executed so that the program following it is executed according to the actual input statuses.

When the input condition to the MCS instruction is off and the MCS is executed, other instructions between the MCS and MCR are executed as follows:

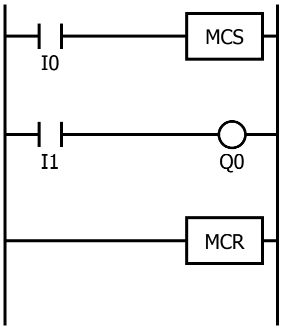
Instruction	Status
SOTU	Rising edges (ON pulses) are not detected.
SOTD	Falling edges (OFF pulses) are not detected.
OUT	All are turned off.
OUTN	All are turned on.
SET and RST	All are held in current status.
TML, TIM, TMH, and TMS	Current values are reset to zero. Timeout statuses are turned off.
CNT, CDP, and CUD	Current values are held. Pulse inputs are turned off. Countout statuses are turned off.
SFR and SFRN	Shift register bit statuses are held. Pulse inputs are turned off. The output from the last bit is turned off.

Input conditions cannot be set for the MCR instruction.

More than one MCS instruction can be used with one MCR instruction.

Corresponding MCS/MCR instructions cannot be nested within another pair of corresponding MCS/MCR instructions.

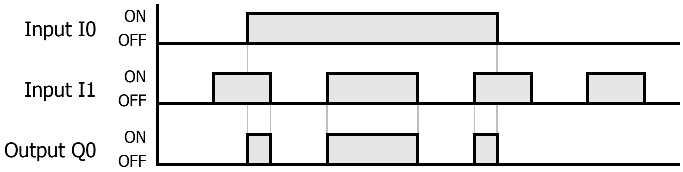
Ladder Diagram



Program List

Instruction	Data
LOD	I0
MCS	
LOD	I1
OUT	Q0
MCR	

Timing Chart

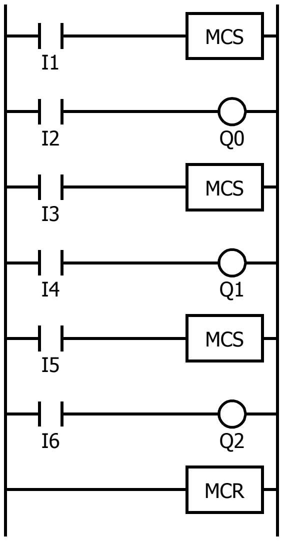


When input I0 is off, MCS is executed so that the subsequent input is forced off.

When input I0 is on, MCS is not executed so that the following program is executed according to the actual input statuses.

Multiple Usage of MCS instructions

Ladder Diagram



Program List

Instruction	Data
LOD	I1
MCS	
LOD	I2
OUT	Q0
LOD	I3
MCS	
LOD	I4
OUT	Q1
LOD	I5
MCS	
LOD	I6
OUT	Q2
MCR	

This master control circuit will give priority to I1, I3, and I5, in that order.

When input I1 is off, the first MCS is executed so that subsequent inputs I2 through I6 are forced off.

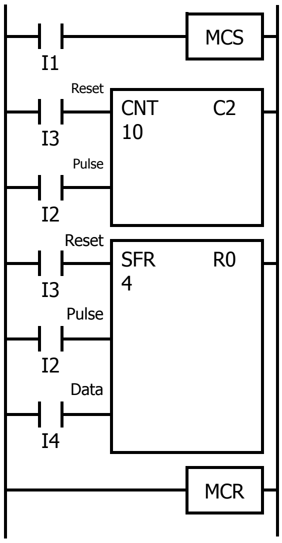
When input I1 is on, the first MCS is not executed so that the following program is executed according to the actual input statuses of I2 through I6.

When I1 is on and I3 is off, the second MCS is executed so that subsequent inputs I4 through I6 are forced off.

When both I1 and I3 are on, the first and second MCSs are not executed so that the following program is executed according to the actual input statuses of I4 through I6.

Counter and Shift Register in Master Control Circuit

Ladder Diagram

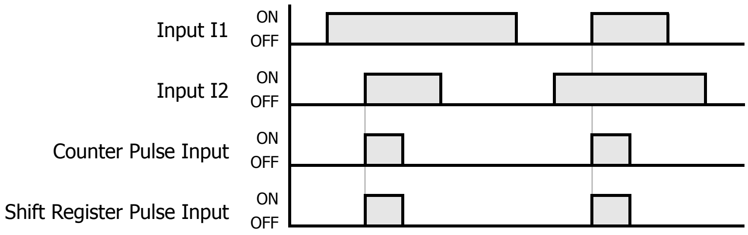


When input I1 is on, the MCS is not executed so that the counter and shift register are executed according to the actual status of subsequent inputs I2 through I4.

When input I1 is off, the MCS is executed so that subsequent inputs I2 through I4 are forced off.

When input I1 is turned on while input I2 is on, the counter and shift register pulse inputs are turned on as shown below.

Timing Chart



JMP (Jump) and JEND (Jump End)

The JMP (jump) instruction is usually used in combination with the JEND (jump end) instruction. At the end of a program, the JMP instruction can also be used with the END instruction, instead of the JEND instruction.

These instructions are used to proceed through the portion of the program between the JMP and the JEND *without* processing. This is similar to the MCS/MCR instructions, except that the portion of the program between the MCS and MCR instruction *is* executed.

When the operation result immediately before the JMP instruction is on, the JMP is valid and the program is *not* executed. When the operation result immediately before the JMP instruction is off, the JMP is invalid and the program is executed.

When the input condition to the JMP instruction is on and the JMP is executed, other instructions between the JMP and JEND are executed as follows:

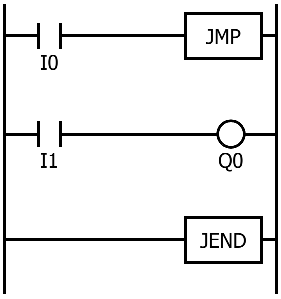
Instruction	Status
SOTU	Rising edges (ON pulses) are not detected.
SOTD	Falling edges (OFF pulses) are not detected.
OUT and OUTN	All are held in current status.
SET and RST	All are held in current status.
TML, TIM, TMH, and TMS	Current values are held. Timeout statuses are held.
CNT, CDP, and CUD	Current values are held. Pulse inputs are turned off. Countout statuses are held.
SFR and SFRN	Shift register bit statuses are held. Pulse inputs are turned off. The output from the last bit is held.

Input conditions cannot be set for the JEND instruction.

More than one JMP instruction can be used with one JEND instruction.

Corresponding JMP/JEND instructions cannot be nested within another pair of corresponding JMP/JEND instructions.

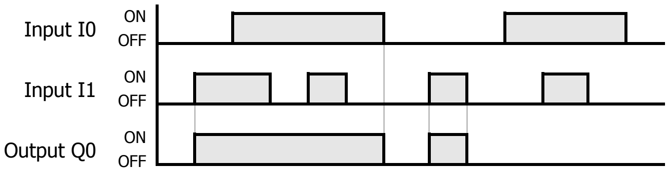
Ladder Diagram



Program List

Instruction	Data
LOD	I0
JMP	
LOD	I1
OUT	Q0
JEND	

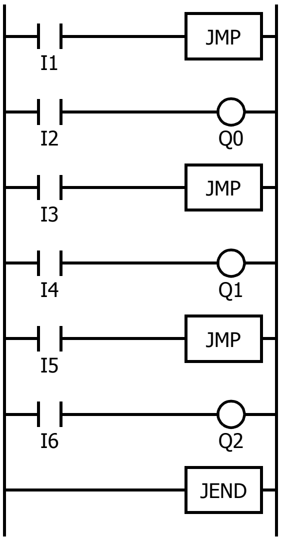
Timing Chart



When input I0 is on, JMP is executed so that the subsequent output status is held.

When input I0 is off, JMP is not executed so that the following program is executed according to the actual input statuses.

Ladder Diagram



Program List

Instruction	Data
LOD	I1
JMP	
LOD	I2
OUT	Q0
LOD	I3
JMP	
LOD	I4
OUT	Q1
LOD	I5
JMP	
LOD	I6
OUT	Q2
JEND	

This jump circuit will give priority to I1, I3, and I5, in that order.

When input I1 is on, the first JMP is executed so that subsequent output statuses of Q0 through Q2 are held.

When input I1 is off, the first JMP is not executed so that the following program is executed according to the actual input statuses of I2 through I6.

When I1 is off and I3 is on, the second JMP is executed so that subsequent output statuses of Q1 and Q2 are held.

When both I1 and I3 are off, the first and second JMPs are not executed so that the following program is executed according to the actual input statuses of I4 through I6.

END

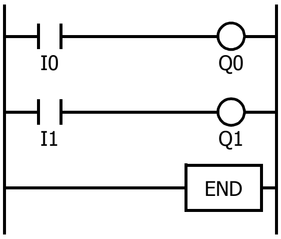
The END instruction is always required at the end of a program; however, it is not necessary to program the END instruction after the last programmed instruction. WindLDR automatically appends the END instruction at the end of a program.

A *scan* is the execution of all instructions from address zero to the END instruction. The time required for this execution is referred to as one *scan time*. The scan time varies with respect to program length, which corresponds to the address where the END instruction is found.

During the scan time, program instructions are processed sequentially. This is why the output instruction closest to the END instruction has priority over a previous instruction for the same output. No output is initiated until all logic within a scan is processed.

Output occurs simultaneously, and this is the first part of the END instruction execution. The second part of the END instruction execution is to monitor all inputs, also done simultaneously. Then program instructions are ready to be processed sequentially once again.

Ladder Diagram

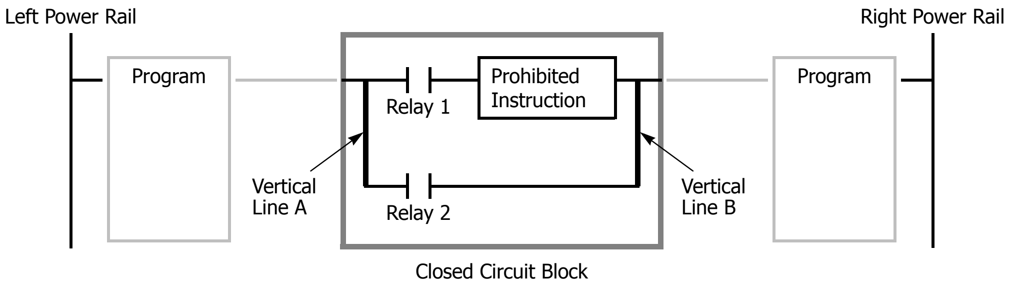


Program List

Instruction	Data
LOD	I0
OUT	Q0
LOD	I1
OUT	Q1
END	

Restriction on Ladder Programming

Due to the structure of WindLDR, the following ladder diagram cannot be programmed — a closed circuit block is formed by vertical lines, except for right and left power rails, and the closed circuit block contains one or more prohibited instructions shown in the table below.

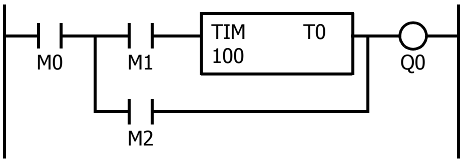


Prohibited Instructions	OUT, OUTN, SET, RST, TML, TIM, TMH, TMS, TMLO, TIMO, TMHO, TMSO, CNT, CDP, CUD, CNTD, CDPD, CUDD, SFR, SFRN, SOTU, SOTD
Error Detection	When converting the ladder program, an error message is shown, such as “TIM follows an invalid device.” Conversion fails to create mnemonics and the program is not downloaded to the FC6A Series MICROSmart.

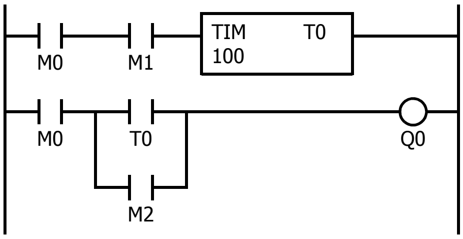
Modifying Prohibited Ladder Programs

Intended operation can be performed by modifying the prohibited ladder program as shown in the examples below:

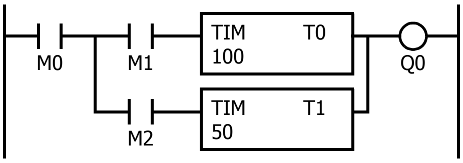
Prohibited Ladder Program 1



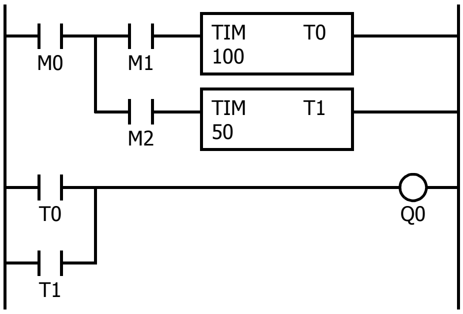
Modified Ladder Program 1



Prohibited Ladder Program 2



Modified Ladder Program 2



5: Move Instructions

Introduction

Data can be moved using the MOV (move), MOVN (move not), IMOV (indirect move), or IMOVN (indirect move not) instruction. The moved data is 16- or 32-bit data, and the repeat operation can also be used to increase the quantity of data moved. In the MOV or MOVN instruction, the source and destination device are designated by S1 and D1 directly. In the IMOV or IMOVN instruction, the source and destination device are determined by the offset values designated by S2 and D2 added to source device S1 and destination device D1.

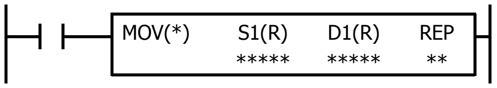
The BMOV (block move) instruction is useful to move consecutive blocks of timer, counter, and data register values.

The IBMV (indirect bit move) and IBMVN (indirect bit move not) instructions move one bit of data from a source device to a destination device. Both devices are determined by adding an offset to the device.

NSET (N data set) and NRS (N data repeat set) instructions can be used to set values to a group of devices. The XCHG (exchange) instruction is used to swap word or double-word data between two devices. The current timer or counter values can be changed using the TCCST (timer/counter current value store) instruction.

Since the move instructions are executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

MOV (Move)



S1 → D1
When input is on, 16- or 32-bit data from device assigned by S1 is moved to device assigned by D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	First device address to move	X	X	X	X	X	X	X	X	X	1-99
D1 (Destination 1)	First device address to move to	—	X	▲	X	X	X	X	X	—	1-99

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S1, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP) which can be 0 through 65,535.

When F (float) data is selected, only data register and constant can be designated as S1, and only data register can be designated as D1.

When F (float) data is selected and S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For user program execution errors, see "User Program Execution Errors" on page 3-10.

P (index register) can be used only when the data type is L (long).

Valid Data Types

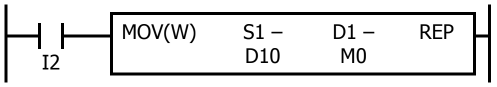
W (word)	X
I (integer)	X
D (double word)	X
L (long)	X
F (float)	X

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source or destination, 16 points (word or integer data) or 32 points (double-word or long data) are used. When repeat is assigned for a bit device, the quantity of device bits increases in 16- or 32-point increments.

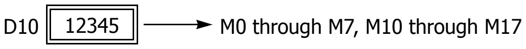
When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word or integer data) or 2 points (double-word, long, or float data) are used. When repeat is assigned for a word device, the quantity of device words increases in 1- or 2-point increments.

Examples: MOV

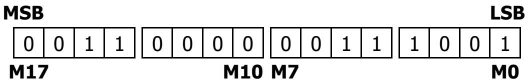
Data Type: Word



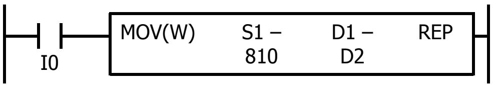
D10 → M0
When input I2 is on, data in data register D10 assigned by source device S1 is moved to 16 internal relays starting with M0 assigned by destination device D1.



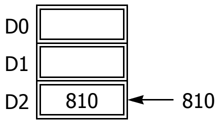
Data in the source data register is converted into 16-bit binary data, and the ON/OFF statuses of the 16 bits are moved to internal relays M0 through M7 and M10 through M17. M0 is the LSB (least significant bit). M17 is the MSB (most significant bit).



Data Type: Word

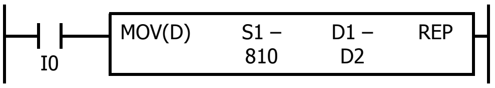


810 → D2
When input I0 is on, constant 810 assigned by source device S1 is moved to data register D2 assigned by destination device D1.

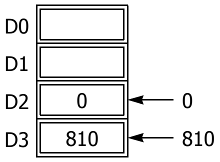


Data move operation for integer data is the same as for word data.

Data Type: Double Word

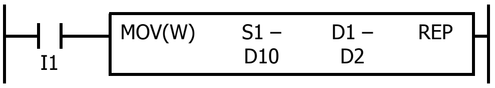


810 → D2·D3
When input I0 is on, constant 810 assigned by source device S1 is moved to data registers D2 and D3 assigned by destination device D1.

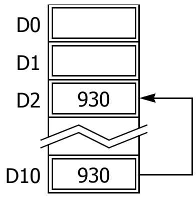


Data move operation for the long data type is the same as for the double-word data type.

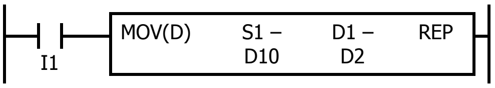
Data Type: Word



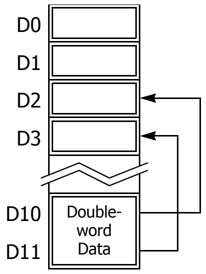
D10 → D2
When input I1 is on, data in data register D10 assigned by source device S1 is moved to data register D2 assigned by destination device D1.



Data Type: Double Word



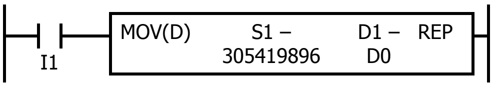
D10·D11 → D2·D3
When input I1 is on, data in data registers D10 and D11 assigned by source device S1 is moved to data registers D2 and D3 assigned by destination device D1.

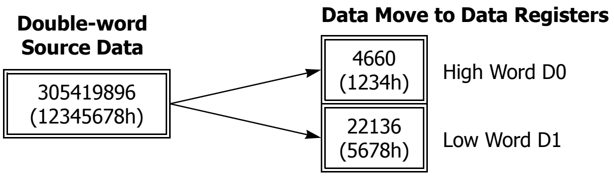


Double-word Data Move in Data Registers

When a data register, timer, or counter is selected as a double-word device, the upper-word data is loaded from or stored to the first device selected. The lower-word data is loaded from or stored to the subsequent device.

Double-word Destination Device: Data Register



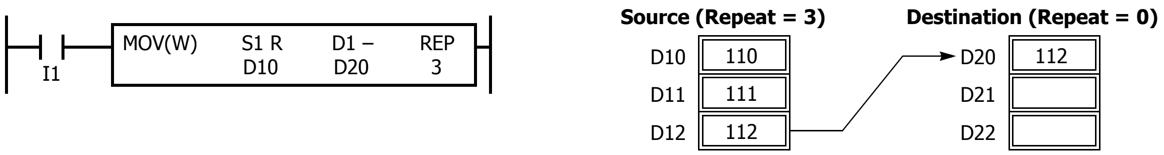


Repeat Operation in the Move Instructions

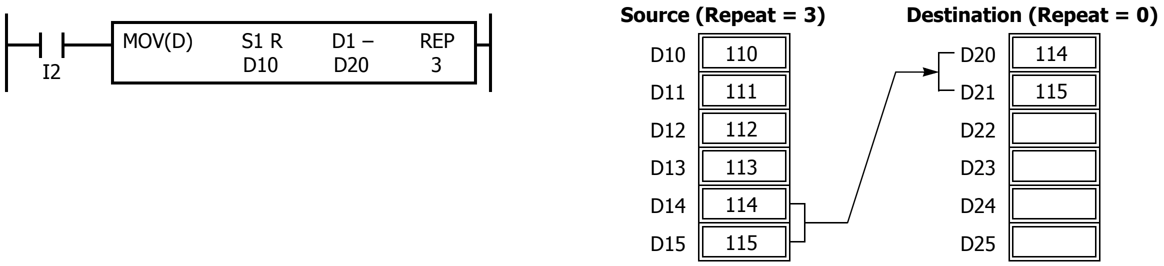
Repeat Source Device

When the S1 (source) is set to repeat, as many devices as the repeat cycles, starting with the device designated by S1, are moved to the destination. As a result, only the last of the source devices is moved to the destination.

- **Data Type: Word**



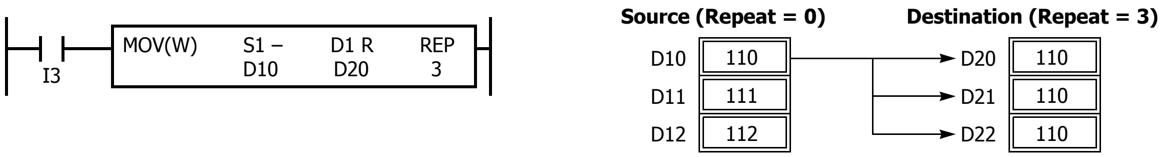
- **Data Type: Double Word**



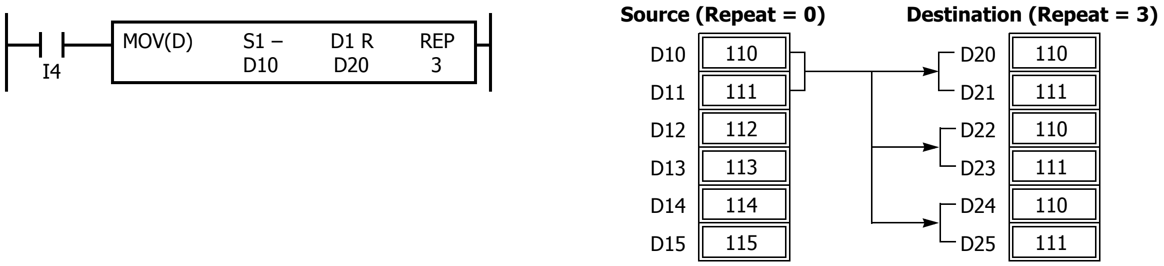
Repeat Destination Device

When the D1 (destination) is designated to repeat, the source device designated by S1 is moved to all destination devices as many times as the repeat cycles, starting with the destination designated by D1.

- **Data Type: Word**



- **Data Type: Double Word**

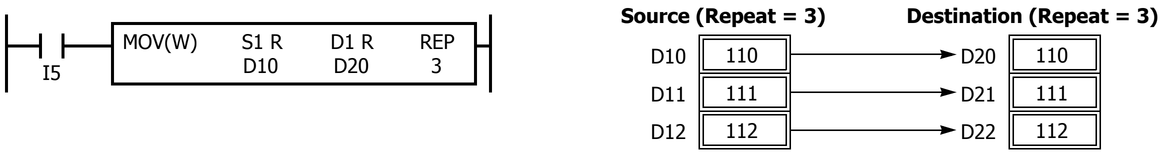


Repeat Source and Destination Devices

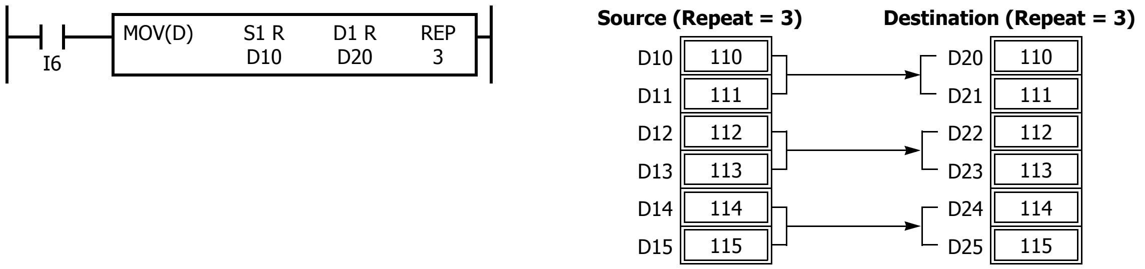
When both S1 (source) and D1 (destination) are set to repeat, as many devices as the repeat cycles, starting with the device designated by S1, are moved to the same quantity of devices starting with the device designated by D1.

Note: The BMOV (block move) instruction has the same effect as the MOV instruction with both the source and destination designated to repeat.

- **Data Type: Word**

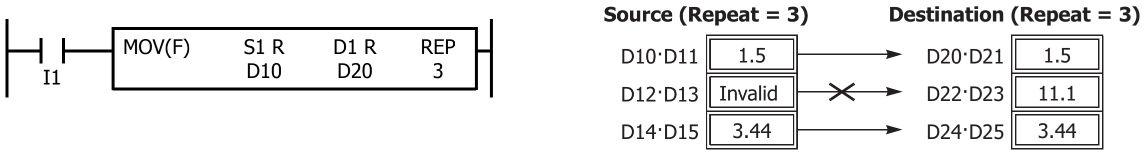


- **Data Type: Double Word**



- **Data Type: Float**

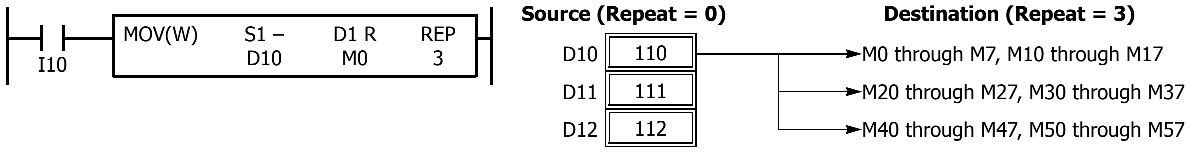
When the source data does not comply with the normal floating-point format in any repeat operation, a user program execution error occurs, and the source data is not moved to the destination.



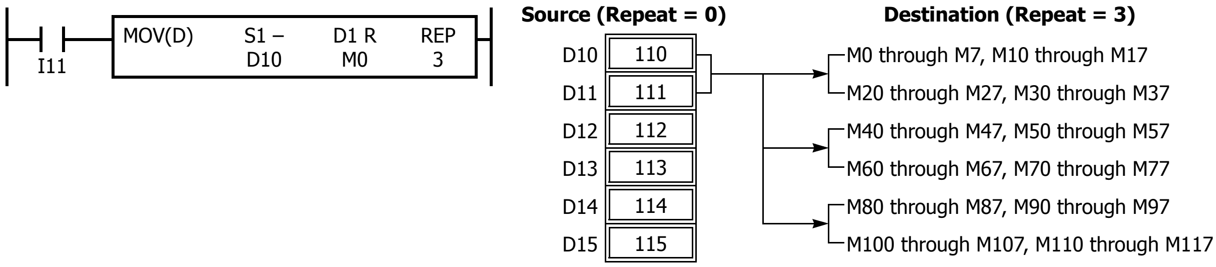
Repeat Bit Devices

The MOV (move) instruction moves 16-bit data (word or integer data) or 32-bit data (double-word or integer data). When a bit device such as input, output, internal relay, or shift register is designated as the source or destination device, 16 or 32 bits starting with the one designated by S1 or D1 are the target data. If a repeat operation is designated for a bit device, the target data increases in 16- or 32-bit increments, depending on the selected data type.

- **Data Type: Word**

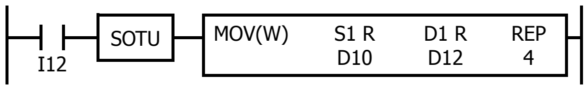

I10
MOV(W)
S1 –
D10
D1 R
M0
REP
3
Source (Repeat = 0)
Destination (Repeat = 3)
D10
110
D11
111
D12
112
M0 through M7, M10 through M17
M20 through M27, M30 through M37
M40 through M47, M50 through M57

- **Data Type: Double Word**

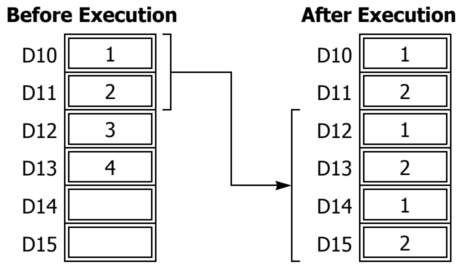

I11
MOV(D)
S1 –
D10
D1 R
M0
REP
3
Source (Repeat = 0)
Destination (Repeat = 3)
D10
110
D11
111
D12
112
D13
113
D14
114
D15
115
M0 through M7, M10 through M17
M20 through M27, M30 through M37
M40 through M47, M50 through M57
M60 through M67, M70 through M77
M80 through M87, M90 through M97
M100 through M107, M110 through M117

Overlapped Devices by Repeat

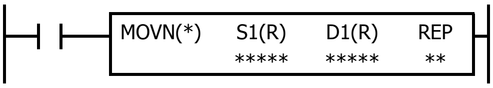
If the repeat operation is set for both the source and destination and if a portion of the source and destination areas overlap each other, then the source data in the overlapped area also changes.



Source: D10 through D13 (Repeat = 4)
Destination: D12 through D15 (Repeat = 4)



MOVN (Move Not)



S1 NOT → D1

When input is on, 16- or 32-bit data from device assigned by S1 is inverted bit by bit and moved to device assigned by D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	First device address to move	X	X	X	X	X	X	X	—	X	1-99
D1 (Destination 1)	First device address to move to	—	X	▲	X	X	X	X	—	—	1-99

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S1, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP) which can be 0 through 65,535.

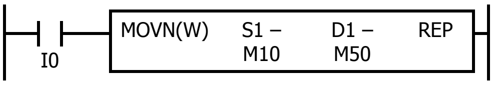
Valid Data Types

W (word)	X
I (integer)	X
D (double word)	X
L (long)	X
F (float)	—

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source or destination, 16 points (word or integer data) or 32 points (double-word or long data) are used. When repeat is set for a bit device, the quantity of device bits increases in 16- or 32-point increments.

When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word or integer data) or 2 points (double-word or long data) are used. When repeat is set for a word device, the quantity of device words increases in 1- or 2-point increments.

Examples: MOVN



M10 NOT → M50

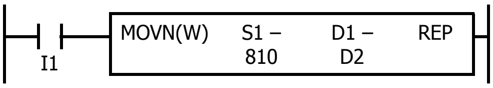
When input I0 is on, the 16 internal relays, starting with M10 assigned by source device S1, are inverted bit by bit and moved to 16 internal relays starting with M50 assigned by destination device D1.

M10 through M17, M20 through M27 NOT → M50 through M57, M60 through M67

	MSB							S1								LSB
Before inversion (M0027-M0010):	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	1

	MSB							D1								LSB
After inversion (M0067-M0050):	1	1	0	0	1	1	1	1	1	1	0	0	0	1	1	0

The ON/OFF statuses of the 16 internal relays M10 through M17 and M20 through M27 are inverted and moved to 16 internal relays M50 through M57 and M60 through M67. M50 is the LSB (least significant bit), and M67 is the MSB (most significant bit).

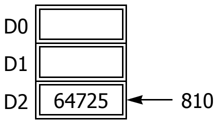


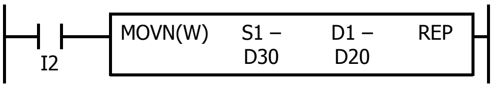
810 NOT → D2

When input I1 is on, decimal constant 810 assigned by source device S1 is converted into 16-bit binary data, and the ON/OFF statuses of the 16 bits are inverted and moved to data register D2 assigned by destination device D1.

	MSB							S1								LSB
Before inversion (810):	0	0	0	0	0	0	1	1	0	0	1	0	1	0	1	0

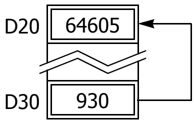
	MSB							D1								LSB
After inversion (64725):	1	1	1	1	1	1	0	0	1	1	0	1	0	1	0	1



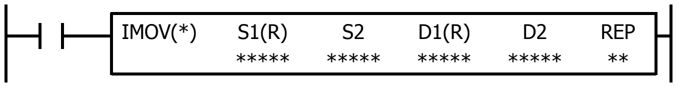


D30 NOT → D20

When input I2 is on, the data in data register D30, assigned by S1, is inverted bit by bit and moved to data register D20 assigned by D1.



IMOV (Indirect Move)



S1 + S2 → D1 + D2

When input is on, the values contained in devices assigned by S1 and S2 are added together to determine the data source. The 16- or 32-bit data is then moved to the destination, which is determined by the sum of values contained in devices assigned by D1 and D2.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Base address to move from	X	X	X	X	X	X	X	—	—	1-99
S2 (Source 2)	Offset for S1	X	X	X	X	X	X	X	—	—	—
D1 (Destination 1)	Base address to move to	—	X	▲	X	X	X	X	—	—	1-99
D2 (Destination 2)	Offset for D1	X	X	X	X	X	X	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S1, S2, or D2, the device data is the timer/counter current value (TC or CC). When T (timer) or C (counter) is used as D1, the device data is the timer/counter preset value (TP or CP) which can be 0 through 65,535.

When F (float) data is selected, only data register can be designated as S1 or D1.

When F (float) data is selected and S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart.

Source device S2 or destination device D2 does not have to be assigned. If S2 or D2 is not assigned, the source or destination device is determined by S1 or D1 without offset.

Make sure that the source data determined by S1 + S2 and the destination data determined by D1 + D2 are within the valid device range. If the derived source or destination device exceeds the valid device range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Data Types

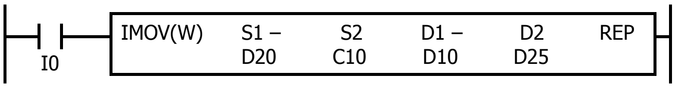
W (word)	X
I (integer)	—
D (double word)	X
L (long)	—
F (float)	X

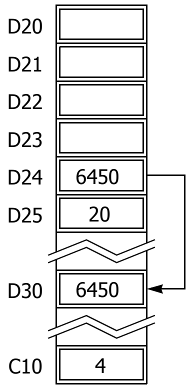
When S1 (source 1) is a word device and D1 (destination 1) is a Q, M, or R bit device, the data is transferred as 16 points or 32 bits, depending on the data type. When repeat is assigned for a bit device, the quantity of device bits increases in 16- or 32-point increments.

When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word data) or 2 points (double-word or float data) are used. When repeat is assigned for a word device, the quantity of device words increases in 1- or 2-point increments.

Example: IMOV

- **Data Type: Word**





D20 + C10 → D10 + D25

Source device S1 and destination device D1 determine the type of device. Source device S2 and destination device D2 are the offset values to determine the source and destination devices.

If the current value of counter C10 assigned by source device S2 is 4, the source data is determined by adding the offset value to data register D20 assigned by source device S1:

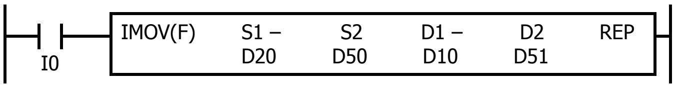
D(20 + 4) = D24

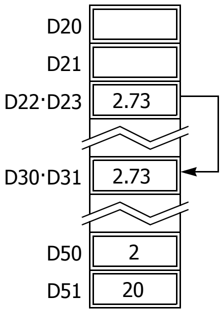
If data register D25 contains a value of 20, the destination is determined by adding the offset value to data register D10 assigned by destination device D1:

D(10 + 20) = D30

As a result, when input I0 is on, the data in data register D24 is moved to data register D30.

- **Data Type: Float**





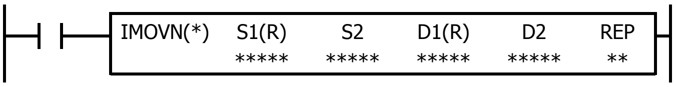
D20 + D50 → D10 + D51

If data register D50 contains a value of 2, the source data is determined by adding the offset value to data register D20 assigned by source device S1: D(20 + 2) = D22

If data register D51 contains a value of 20, the destination is determined by adding the offset value to data register D10 assigned by destination device D1: D(10 + 20) = D30

As a result, when input I0 is on, the data in data registers D22·D23 is moved to data registers D30·D31.

IMOVN (Indirect Move Not)



S1 + S2 NOT → D1 + D2

When input is on, the values contained in devices assigned by S1 and S2 are added together to determine the data source. The 16- or 32-bit data is then inverted and moved to the destination, which is determined by the sum of values contained in devices assigned by D1 and D2.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Base address to move from	X	X	X	X	X	X	X	—	—	1-99
S2 (Source 2)	Offset for S1	X	X	X	X	X	X	X	—	—	—
D1 (Destination 1)	Base address to move to	—	X	▲	X	X	X	X	—	—	1-99
D2 (Destination 2)	Offset for D1	X	X	X	X	X	X	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S1, S2, or D2, the device data is the timer/counter current value (TC or CC). When T (timer) or C (counter) is used as D1, the device data is the timer/counter preset value (TP or CP), 0 through 65,535.

Source device S2 or destination device D2 does not have to be assigned. If S2 or D2 is not assigned, the source or destination device is determined by S1 or D1 without offset.

Make sure that the source data determined by S1 + S2 and the destination data determined by D1 + D2 are within the valid device range. If the derived source or destination device exceeds the valid device range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For user program execution errors, see "User Program Execution Errors" on page 3-10.

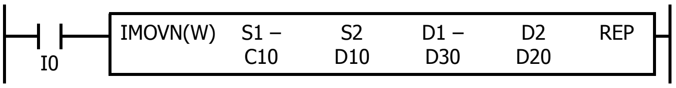
Valid Data Types

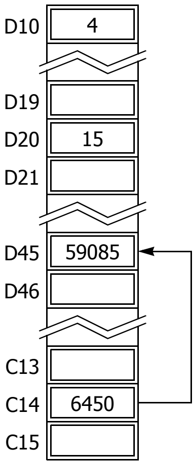
W (word)	X
I (integer)	—
D (double word)	X
L (long)	—
F (float)	—

When S1 (source 1) is a word device and D1 (destination 1) is a Q, M, or R bit device, the data is transferred as 16 points or 32 bits, depending on the data type. When repeat is set for a bit device, the quantity of device bits increases in 16- or 32-point increments.

When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word data) or 2 points (double-word data) are used. When repeat is assigned for a word device, the quantity of device words increases in 1- or 2-point increments.

Example: IMOVN





C10 + D10 NOT → D30 + D20

Source device S1 and destination device D1 determine the type of device. Source device S2 and destination device D2 are the offset values used to determine the source and destination devices.

If the data of data register D10 assigned by source device S2 is 4, then the source data is determined by adding the offset value to counter C10 assigned by source device S1:

C(10 + 4) = C14

If data register D20 assigned by destination device D2 contains a value of 15, then the destination is determined by adding the offset value to data register D30 assigned by destination device D1:

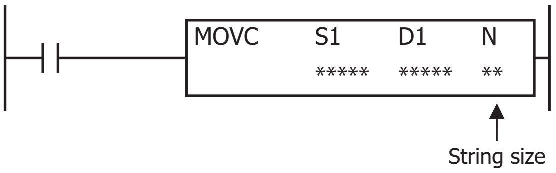
D(30 + 15) = D45

As a result, when input I0 is on, the current value of counter C14 is inverted and moved to data register D45.

MOVC (Move Characters)

The MOVC instruction moves a character string with the specified character set.

Ladder Diagram



Operation

When the input is on, a NULL terminator (0x00) (1 byte) is added to the character string specified by S1, and that data is sequentially transferred to the devices specified by D1 starting from the upper byte of the device.

S1='1234' ⇒

Device	Stored Value	
	Upper Byte	Lower Byte
D0000	'1'=0x31	'2'=0x32
D0001	'3'=0x33	'4'=0x34
D0002	0x00	
D0003		

Notes:

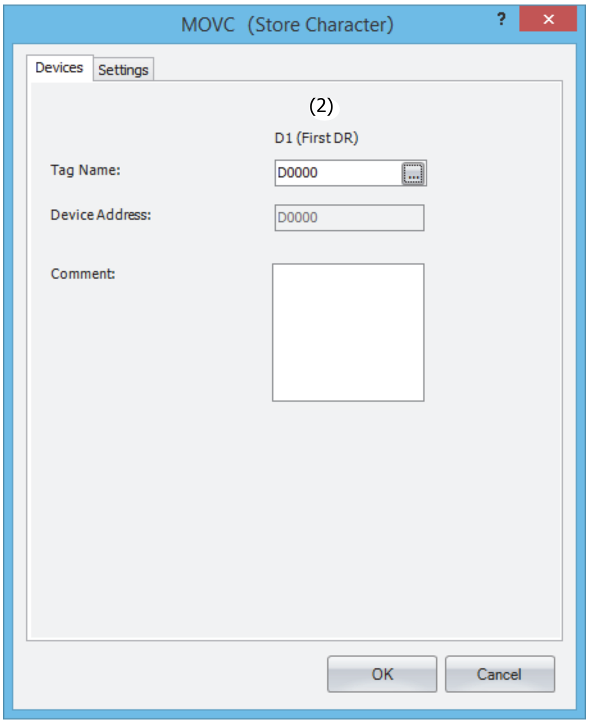
- The maximum size of the character string that can be specified as S1 is 1,023 bytes.
- This instruction can be used to store a character sting in the data registers that are specified in EMAIL instruction for inserting data registers.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Character string	—	—	—	—	—	—	—	—	X	—
D1 (Destination 1)	Transfer destination	—	—	—	—	—	—	X*1	—	—	—

*1 Special data registers cannot be used.

Settings

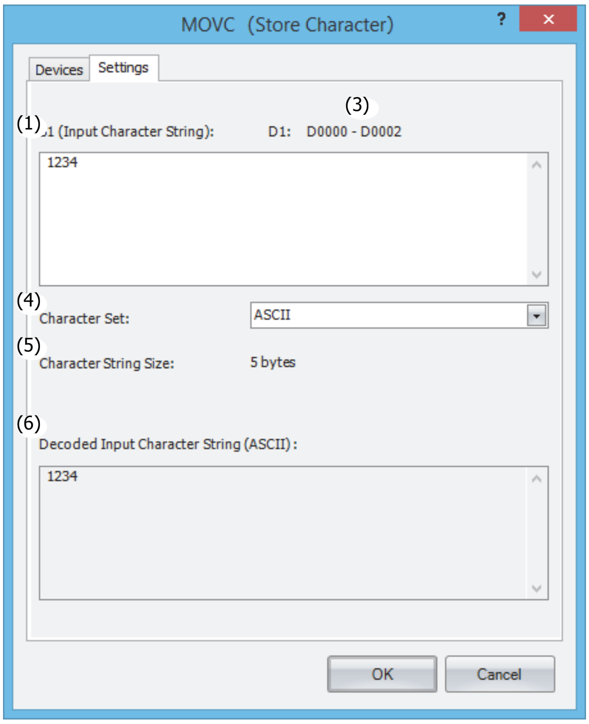


D1 (First DR)

Tag Name: D0000

Device Address: D0000

Comment:



1 (Input Character String): D1: D0000 - D0002

1234

Character Set: ASCII

Character String Size: 5 bytes

Decoded Input Character String (ASCII):

1234

■S1 (source 1) setting

(1) Input character string

Specify a character string within a maximum length of 1,023 bytes, regardless of whether single-byte or double-byte characters are used. When multiple lines are entered, the line feed character (0D0Ah) is inserted between those lines.

■D1 (destination 1) setting

(2) First DR

Specify the first data register of the data registers where the character string is stored.

■Other settings

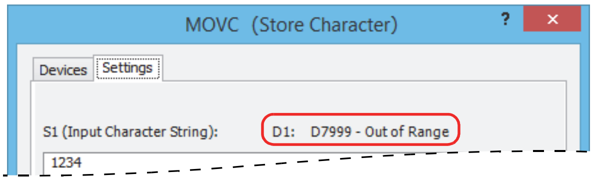
(3) Device range

The device range displays the range of data registers, starting from the specified data register, that are required to store the character string specified by the input character string (1).

The range of data registers is not displayed in the following cases:

- The byte length of the character string specified by the input character string (1) exceeds 1,023 bytes.
- A device other than a data register (excluding special data registers) is specified as the first DR (2), or the first DR (2) is not configured.

Note: The device range is displayed as follows when the specified data registers are out of range.



S1 (Input Character String): D1: D7999 - Out of Range

1234

(4) Character set

Select the character set of the character string from the following six types of character sets:

- ASCII
- Unicode (UTF-8)
- Japanese (ISO-2022-JP)
- Japanese (Shift-JIS)
- Chinese (GB2312)
- Western European (ISO 8859-1 (Latin 1))

(5) Character string size

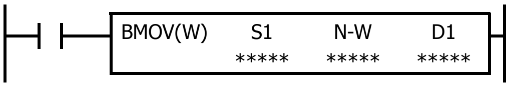
The character string size displays the byte length of the input character string.

The displayed value is the byte length of the input character string (1) plus the NULL terminator (1 byte) added to that character string.

(6) Decoded input character string

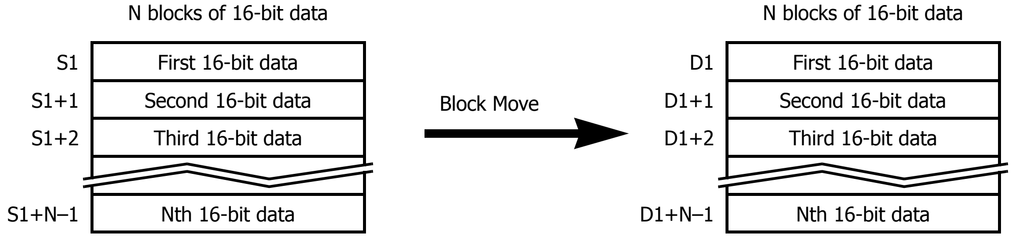
The character string input in Input character string (1) is decoded with the character set specified by Character set (4) and displayed in this text box.

BMOV (Block Move)



S1, S1+1, S1+2, ... , S1+N–1 → D1, D1+1, D1+2, ... , D1+N–1

When input is on, N blocks of 16-bit word data starting with device assigned by S1 are moved to N blocks of destinations, starting with device assigned by D1. N-W specifies the quantity of blocks to move.



Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	First device address to move	X	X	X	X	X*1	X*1	X	—	—	—
N-W (N words)	Quantity of blocks to move	X	X	X	X	X*1	X*1	X	—	X*2	—
D1 (Destination 1)	First device address to move to	—	X	X*3	X	X*4	X*4	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

*1 When T/C is used as S1 or NW, this is the current value area.

*2 You can specify the number between 1 and 6,000.

*3 Special internal relays cannot be used.

*4 When T/C is used as D1, this is the preset value area.

Make sure that the last source data determined by S1+N–1 and the last destination data determined by D1+N–1 are within the valid device range. If the derived source or destination device exceeds the valid device range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Data Types

W (word)	X
I (integer)	—
D (double word)	—
L (long)	—
F (float)	—

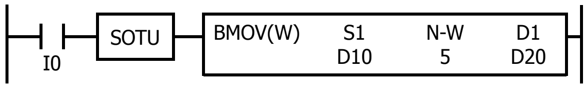
When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source, N-W, or destination, 16 points (word data) are used.

When a word device such as T (timer), C (counter), or D (data register) is assigned as the source, N-W, or destination, 1 point (word data) is used.

Special Internal Relay M8024: BMOV/WSFT Executing Flag

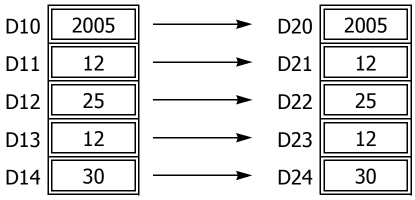
While the BMOV or WSFT is executed, M8024 turns on. When completed, M8024 turns off. If the CPU is powered down while executing BMOV or WSFT, M8024 remains on when the CPU is powered up again.

Example: BMOV

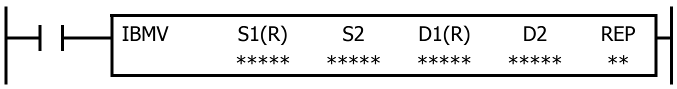


D10 through D14 → D20 through D24

When input I0 is turned on, data of 5 data registers, starting with D10 assigned by source device S1, is moved to 5 data registers starting with D20 assigned by destination device D1.



IBMV (Indirect Bit Move)



S1 + S2 → D1 + D2

When input is on, the values contained in devices assigned by S1 and S2 are added together to determine the source of data. The 1-bit data is then moved to the destination, which is determined by the sum of values contained in devices assigned by D1 and D2.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Base address to move from	X	X	X	X	—	—	X	—	0 or 1	1-99
S2 (Source 2)	Offset for S1	X	X	X	X	X	X	X	—	0-65,535	—
D1 (Destination 1)	Base address to move to	—	X	▲	X	—	—	X	—	—	1-99
D2 (Destination 2)	Offset for D1	X	X	X	X	X	X	X	—	0-65,535	—

For valid device address ranges, see "Device Addresses" on page 2-1.

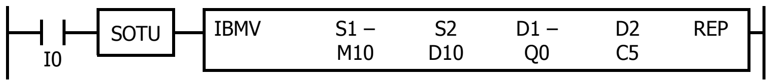
▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S2 or D2, the timer/counter current value (TC or CC) is displayed.

Make sure that the last source data determined by S1+S2 and the last destination data determined by D1+D2 are within the valid device range. If the derived source or destination device exceeds the valid device range, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Source device S2 or destination device D2 does not have to be assigned. If S2 or D2 is not assigned, the source or destination device is determined by S1 or D1 without offset.

Examples: IBMV

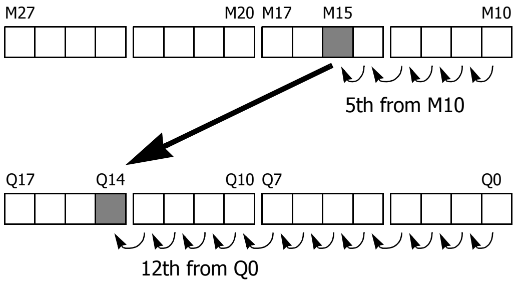


M10 + D10 → Q0 + C5

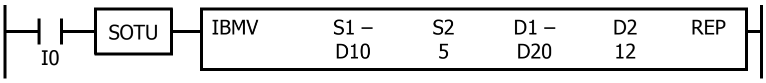
Source device S1 and destination device D1 determine the type of device. Source device S2 and destination device D2 are the offset values to determine the source and destination devices.

If the value of data register D10 assigned by source device S2 is 5, the source data is determined by adding the offset value to internal relay M10 assigned by source device S1.

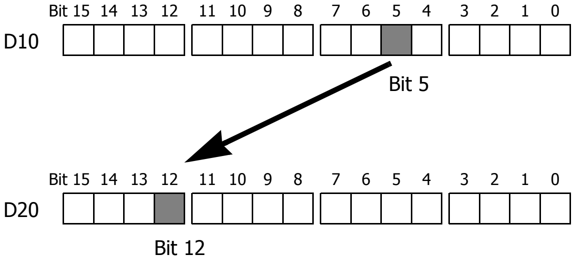
If the current value of counter C5 assigned by destination device D2 is 12, the destination is determined by adding the offset value to output Q0 assigned by destination device D1.



As a result, when input I0 is on, the ON/OFF status of internal relay M15 is moved to output Q14.



D10 + 5 → D20 + 12


Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
D10
Bit 5
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
D20
Bit 12

Since source device S1 is a data register and the value of source device S2 is 5, the source data is bit 5 of data register D10 assigned by source device S1.

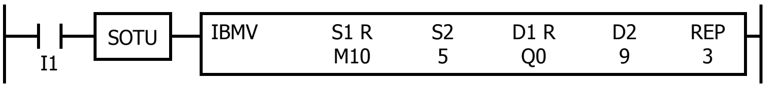
Since destination device D1 is a data register and the value of source device D2 is 12, the destination is bit 12 of data register D20 assigned by destination device D1.

As a result, when input I0 is on, the ON/OFF status of data register D10 bit 5 is moved to data register D20 bit 12.

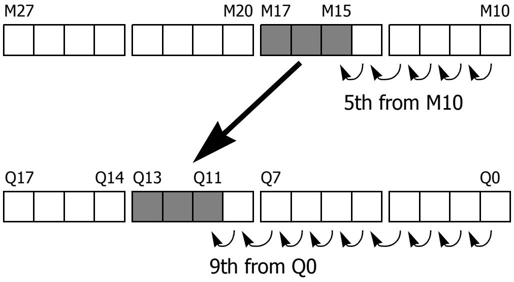
Repeat Operation in the Indirect Bit Move Instructions

Repeat Bit Devices (Source and Destination)

If a repeat operation is set for bit devices such as input, output, internal relay, or shift register, as many bit devices as the repeat cycles are moved.



M10 + 5 → Q0 + 9
Repeat = 3



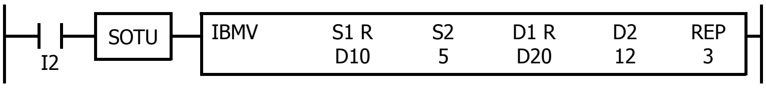
Since source device S1 is internal relay M10 and the value of source device S2 is 5, the source data is 3 internal relays starting with M15.

Since destination device D1 is output Q0 and the value of destination device D2 is 9, the destination is 3 outputs starting with Q11.

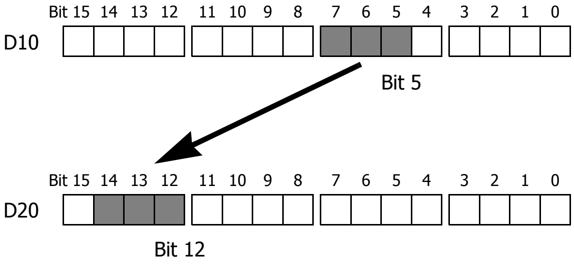
As a result, when input I1 is on, the ON/OFF statuses of internal relays M15 through M17 are moved to outputs Q11 through Q13.

Repeat Word Devices (Source and Destination)

If a repeat operation is for word devices such as data register, as many bit devices as the repeat cycles in the designated data register are moved.



D10 + 5 → D20 + 12
Repeat = 3

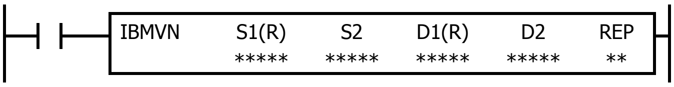

Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
D10
Bit 5
Bit 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
D20
Bit 12

Since source device S1 is data register D10 and the value of source device S2 is 5, the source data is 3 bits starting with bit 5 of data register D10.

Since destination device D1 is data register D20 and the value of destination device D2 is 12, the destination is 3 bits starting with bit 12 of data register D20.

As a result, when input I2 is on, the ON/OFF statuses of data register D10 bits 5 through 7 are moved to data register D20 bits 12 through 14.

IBMVN (Indirect Bit Move Not)



S1 + S2 NOT → D1 + D2

When input is on, the values contained in devices assigned by S1 and S2 are together added to determine the data source. The 1-bit data is then inverted and moved to the destination, which is determined by the sum of values contained in devices assigned by D1 and D2.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Base address to move from	X	X	X	X	—	—	X	—	0 or 1	1-99
S2 (Source 2)	Offset for S1	X	X	X	X	X	X	X	—	0-65,535	—
D1 (Destination 1)	Base address to move to	—	X	▲	X	—	—	X	—	—	1-99
D2 (Destination 2)	Offset for D1	X	X	X	X	X	X	X	—	0-65,535	—

For valid device address ranges, see "Device Addresses" on page 2-1.

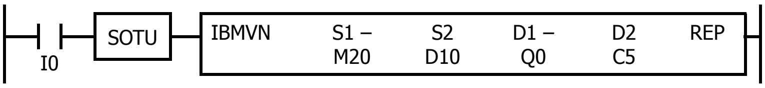
▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S2 or D2, the timer/counter current (TC or CC) value is displayed.

Make sure that the last source data determined by S1+S2 and the last destination data determined by D1+D2 are within the valid device range. If the derived source or destination device exceeds the valid device range, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Source device S2 or destination device D2 does not have to be used. If S2 or D2 are not used, the source or destination device is determined by S1 or D1 without offset.

Examples: IBMVN

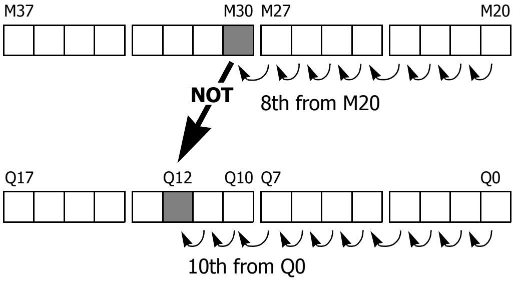


M20 + D10 NOT → Q0 + C5

Source device S1 and destination device D1 determine the type of device. Source device S2 and destination device D2 are the offset values used to determine the source and destination devices.

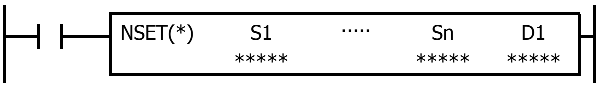
If the value of data register D10 assigned by source device S2 is 8, the source data is determined by adding the offset value to internal relay M20 assigned by source device S1.

If the current value of counter C5 assigned by destination device D2 is 10, the destination is determined by adding the offset value to output Q0 assigned by destination device D1.



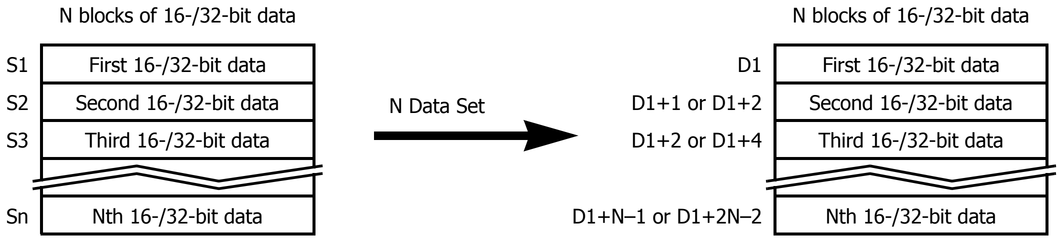
As a result, when input I0 is on, the ON/OFF status of internal relay M30 is inverted and moved to output Q12.

NSET (N Data Set)



S1, S2, S3, ... , Sn → D1, D2, D3, ... , Dn

When input is on, N blocks of 16- or 32-bit data in devices assigned by S1, S2, S3, ... , Sn are moved to N blocks of destinations, starting with device assigned by D1.



Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	First device address to move	X	X	X	X	X	X	X	—	X	—
D1 (Destination 1)	First device address to move to	—	X	▲	X	X	X	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S1, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP).

When F (float) data is selected, only data register and constant can be designated as S1, and only data register can be designated as D1.

When F (float) data is selected and S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart.

Make sure that the last destination data determined by D1+N-1 (word or integer data) or D1+2N-2 (double-word, long, or float data) is within the valid device range. If the derived destination device exceeds the valid device range, a user program execution error will result, turning on special internal relay M8004 and ERROR LED on the FC6A Series MICROSmart. For user program execution errors, see "User Program Execution Errors" on page 3-10.

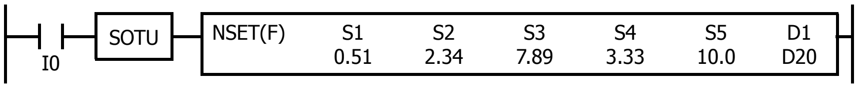
Valid Data Types

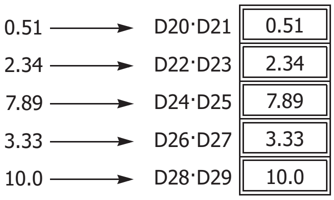
W (word)	X
I (integer)	X
D (double word)	X
L (long)	X
F (float)	X

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source or destination, 16 points (word or integer data) or 32 points (double-word or long data) are used.

When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word or integer data) or 2 points (double-word, long, or float data) are used.

Example: NSET(F)

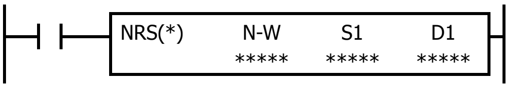




Five constants 0.51, 2.34, 7.89, 3.33, and 10.0 → D20 through D29

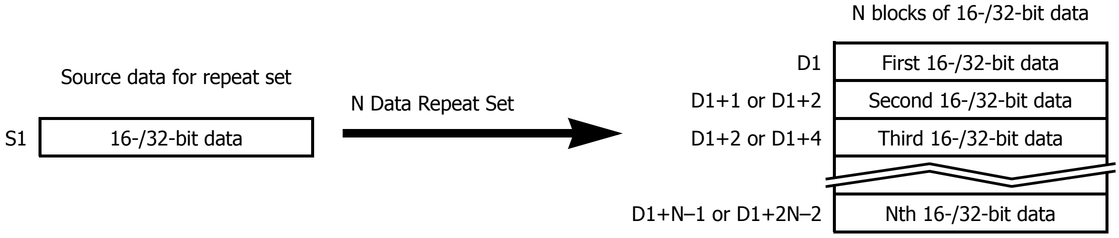
When input I0 is turned on, 5 constants assigned by source devices S1 through S5 are moved to 10 data registers starting with D20 assigned by destination device D1.

NRS (N Data Repeat Set)



S1 → D1, D2, D3, ... , Dn–1

When input is on, 16- or 32-bit data assigned by S1 is set to N blocks of destinations, starting with device assigned by D1.



Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
N-W (N blocks)	Quantity of blocks to move	X	X	X	X	X	X	X	—	X	—
S1 (Source 1)	First device address to move	X	X	X	X	X	X	X	—	X	—
D1 (Destination 1)	First device address to move to	—	X	▲	X	X	X	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

For the N-W, 1 word (16 bits) is always used without regard to the data type.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as N-W or S1, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP).

When F (float) data is selected, only data register and constant can be designated as S1, and only data register can be designated as D1.

When F (float) data is selected and S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart.

Make sure that the last destination data determined by D1+N-1 (word or integer data) or D1+2N-2 (double-word, long, or float data) is within the valid device range. If the derived destination device exceeds the valid device range, a user program execution error will result, turning on special internal relay M8004 and ERROR LED on the FC6A Series MICROSmart. For user program execution errors, see "User Program Execution Errors" on page 3-10.

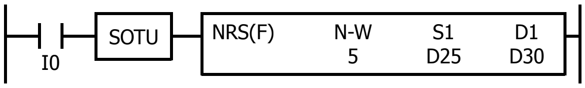
Valid Data Types

W (word)	X
I (integer)	X
D (double word)	X
L (long)	X
F (float)	X

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source or destination, 16 points (word or integer data) or 32 points (double-word or long data) are used.

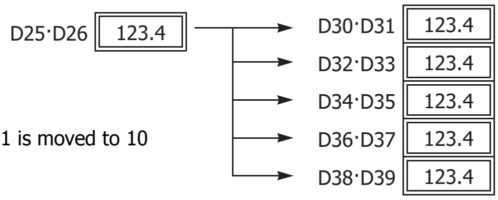
When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word or integer data) or 2 points (double-word, long, or float data) are used.

Example: NRS(F)

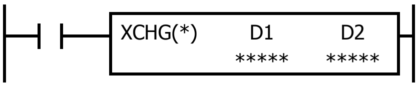


D25·D26 → D30 through D39

When input I0 is turned on, data of data registers D25·D26 assigned by source device S1 is moved to 10 data registers starting with D30 assigned by destination device D1.



XCHG (Exchange)



Word data:	D1 ↔ D2
Double-word data:	D1·D1+1 → D2, D2+1

When input is on, the 16- or 32-bit data in devices assigned by D1 and D2 are exchanged with each other.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
D1 (Destination 1)	First device address to exchange	—	X	▲	X	—	—	X	—	—	—
D2 (Destination 2)	First device address to exchange	—	X	▲	X	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1 or D2.

Valid Data Types

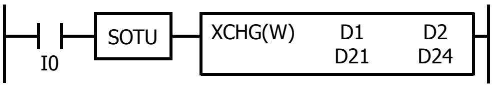
W (word)	X
I (integer)	—
D (double word)	X
L (long)	—
F (float)	—

When a bit device such as Q (output), M (internal relay), or R (shift register) is assigned as the destination, 16 points (word data) or 32 points (double-word data) are used.

When a word device such as D (data register) is assigned as the destination, 1 point (word data) or 2 points (double-word data) are used.

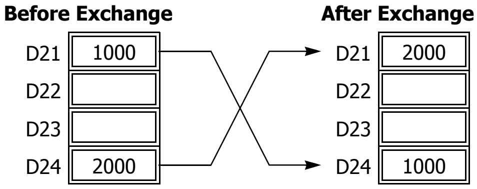
Examples: XCHG

- **Data Type: Word**

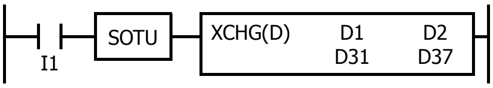


D21 ↔ D24

When input I0 is turned on, data of data registers D21 and D24 assigned by devices D1 and D2 are exchanged with each other.

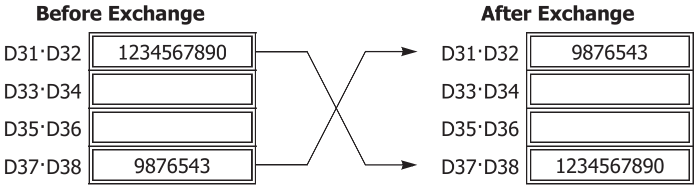


- **Data Type: Double Word**

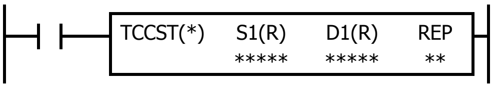


D31·D32 ↔ D37·D38

When input I1 is turned on, data of data registers D31·D32 and D37·D38 assigned by devices D1 and D2 are exchanged with each other.



TCCST (Timer/Counter Current Value Store)



S1 → D1
When input is on, 16- or 32-bit data assigned by S1 is displayed and stored to the current value of device assigned by D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	First device address to move	X	X	X	X	X	X	X	—	X	1-99
D1 (Destination 1)	First device address to move to	—	—	—	—	X	X	—	—	—	1-99

For valid device address ranges, see "Device Addresses" on page 2-1.

When T (timer) or C (counter) is used as S1, the timer/counter current value (TC or CC) is displayed. T (timer) or C (counter) is used as D1, and the data is written in as a current value (TP or CP).

Since the TCCST instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	X
I (integer)	—
D (double word)	X
L (long)	—
F (float)	—

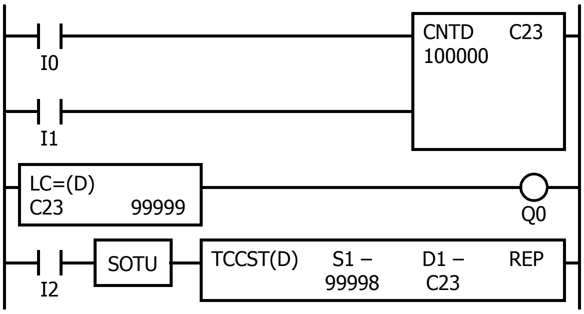
When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source, 16 points (word data) or 32 points (double-word data) are used. When repeat is assigned for a bit device, the quantity of device bits increases in 16- or 32-point increments.

When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word data) or 2 points (double-word data) are used. When repeat is assigned for a word device, the quantity of device words increases in 1- or 2-point increments.

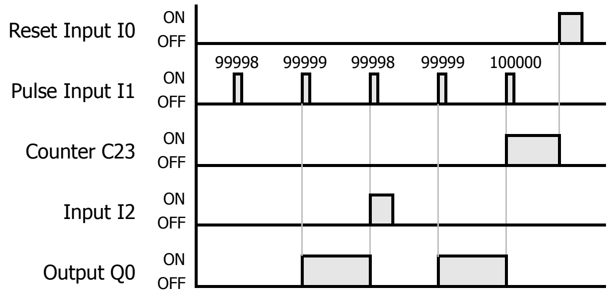
Example: TCCST

When input I2 is turned on, 99,998 is written to the current value of counter C23.

Ladder Diagram



Timing Chart



6: Data Comparison Instructions

Introduction

Data can be compared using data comparison instructions, such as equal to, unequal to, less than, greater than, less than or equal to, and greater than or equal to. When the comparison result is true, an output or internal relay is turned on. The repeat operation can also be used to compare more than one set of data.

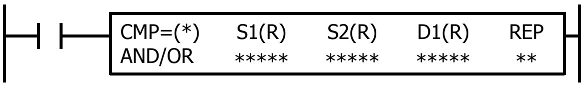
When the repeat operation is enabled, repeated comparison results of CMP instructions can be selected from AND or OR operation, and the result is outputted to an output or internal relay.

Three values can also be compared using the ICMP>= instruction.

Load comparison instructions load comparison results so that the following instructions can be initiated.

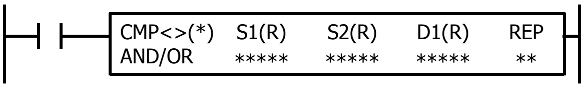
Since the data comparison instructions are executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

CMP= (Compare Equal To)



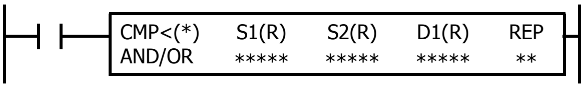
Data type W or I: S1 = S2 → D1 on
Data type D, L, or F: S1·S1+1 = S2·S2+1 → D1 on
When input is on, 16- or 32-bit data assigned by source devices S1 and S2 are compared. When S1 data is equal to S2 data, destination device D1 is turned on. When the condition is not met, D1 is turned off.

CMP<> (Compare Unequal To)



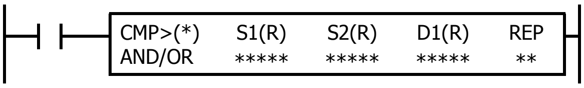
Data type W or I: S1≠ S2 → D1 on
Data type D, L, or F: S1·S1+1 ≠ S2·S2+1 → D1 on
When input is on, 16- or 32-bit data assigned by source devices S1 and S2 are compared. When S1 data is not equal to S2 data, destination device D1 is turned on. When the condition is not met, D1 is turned off.

CMP< (Compare Less Than)



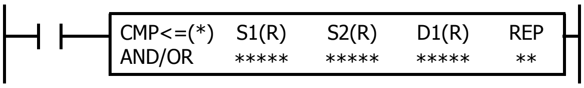
Data type W or I: S1 < S2 → D1 on
Data type D, L, or F: S1·S1+1 < S2·S2+1 → D1 on
When input is on, 16- or 32-bit data assigned by source devices S1 and S2 are compared. When S1 data is less than S2 data, destination device D1 is turned on. When the condition is not met, D1 is turned off.

CMP> (Compare Greater Than)



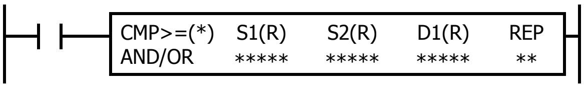
Data type W or I: S1 > S2 → D1 on
Data type D, L, or F: S1·S1+1 > S2·S2+1 → D1 on
When input is on, 16- or 32-bit data assigned by source devices S1 and S2 are compared. When S1 data is greater than S2 data, destination device D1 is turned on. When the condition is not met, D1 is turned off.

CMP<= (Compare Less Than or Equal To)



Data type W or I: S1 ≤ S2 → D1 on
Data type D, L, or F: S1·S1+1 ≤ S2·S2+1 → D1 on
When input is on, 16- or 32-bit data assigned by source devices S1 and S2 are compared. When S1 data is less than or equal to S2 data, destination device D1 is turned on. When the condition is not met, D1 is turned off.

CMP>= (Compare Greater Than or Equal To)



Data type W or I: S1 ≥ S2 → D1 on
Data type D, L, or F: S1·S1+1 ≥ S2·S2+1 → D1 on
When input is on, 16- or 32-bit data assigned by source devices S1 and S2 are compared. When S1 data is greater than or equal to S2 data, destination device D1 is turned on. When the condition is not met, D1 is turned off.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
Repeat Result	Logical AND or OR operation	—	—	—	—	—	—	—	—	—	—
S1 (Source 1)	Data to compare	X	X	X	X	X	X	X	—	X	1-99
S2 (Source 2)	Data to compare	X	X	X	X	X	X	X	—	X	1-99
D1 (Destination 1)	Comparison output	—	X	▲	—	—	—	—	—	—	1-99

For valid device address ranges, see "Device Addresses" on page 2-1.

When only S1 and/or S2 is repeated, the logical operation type can be selected from AND or OR.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S1 or S2, the timer/counter current value (TC or CC) is displayed.

When F (float) data is selected, only data register and constant can be designated as S1 and S2.

When F (float) data is selected and S1 or S2 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Data Types

W (word)	X
I (integer)	X
D (double word)	X
L (long)	X
F (float)	X

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source, 16 points (word or integer data) or 32 points (double-word or long data) are used. When repeat is assigned for a bit device, the quantity of device bits increases in 16- or 32-point increments.

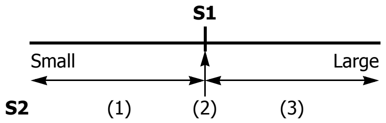
When a word device such as T (timer), C (counter), or D (data register) is assigned as the source, 1 point (word or integer data) or 2 points (double-word, long, or float data) are used. When repeat is assigned for a word device, the quantity of device words increases in 1- or 2-point increments.

When an output or internal relay is assigned as the destination, only 1 point is used regardless of the selected data type. When repeat is assigned for the destination, outputs or internal relays as many as the repeat cycles are used.

Special Internal Relays M8150, M8151, and M8152 in CMP=

Three special internal relays are available to indicate the comparison result of the CMP= instruction. Depending on the result, one of the three special internal relays turns on.

When S1 > S2, M8150 turns on.
When S1 = S2, M8151 turns on.
When S1 < S2, M8152 turns on.



S2 Value	M8150	M8151	M8152	D1 Status
(1) S1 > S2	**ON**	OFF	OFF	OFF
(2) S1 = S2	OFF	**ON**	OFF	**ON**
(3) S1 < S2	OFF	OFF	**ON**	OFF

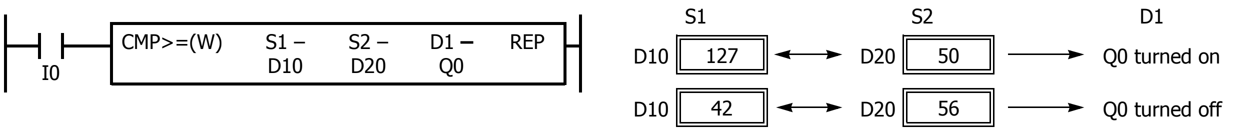
When repeat is designated, the comparison result of the last repeat cycle turns on one of the three special internal relays.

When more than one CMP= or ICMP>= instruction is used, M8150, M8151, or M8152 indicates the result of the instruction that was executed last.

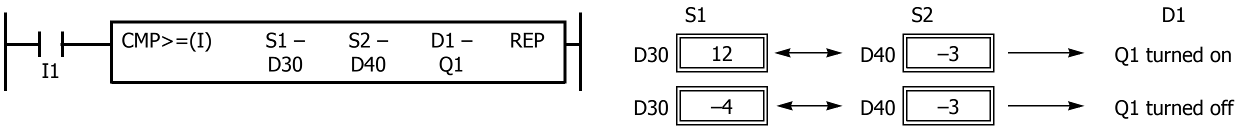
Examples: CMP>=

The following examples are described using the CMP≥ instruction. Data comparison operation for all other data comparison instructions is the same for the CMP≥ instruction.

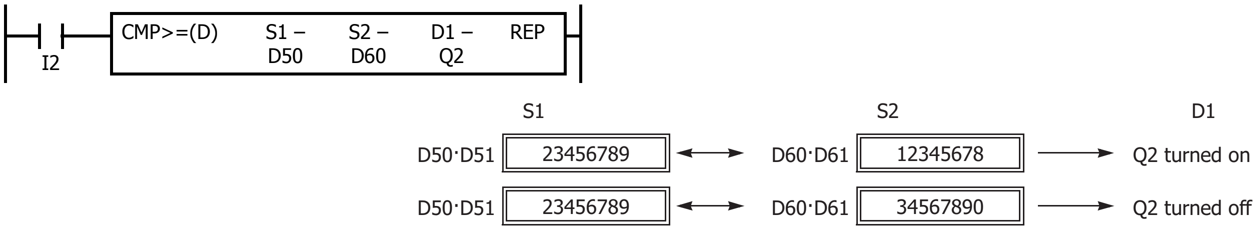
- **Data Type: Word**


I0
CMP>=(W)
S1 – D10
S2 – D20
D1 – Q0
REP
S1
S2
D1
D10 127 ⟷ D20 50 → Q0 turned on
D10 42 ⟷ D20 56 → Q0 turned off

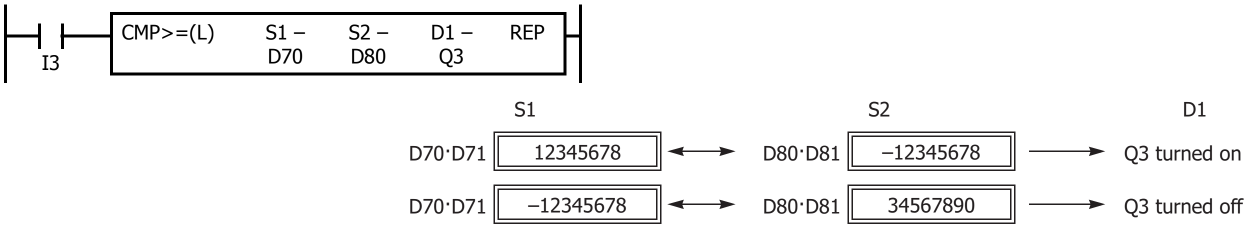
- **Data Type: Integer**


I1
CMP>=(I)
S1 – D30
S2 – D40
D1 – Q1
REP
S1
S2
D1
D30 12 ⟷ D40 –3 → Q1 turned on
D30 –4 ⟷ D40 –3 → Q1 turned off

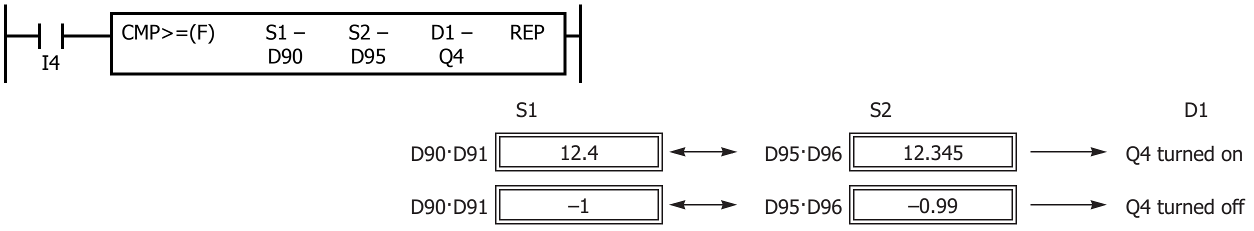
- **Data Type: Double Word**


I2
CMP>=(D)
S1 – D50
S2 – D60
D1 – Q2
REP
S1
S2
D1
D50·D51 23456789 ⟷ D60·D61 12345678 → Q2 turned on
D50·D51 23456789 ⟷ D60·D61 34567890 → Q2 turned off

- **Data Type: Long**


I3
CMP>=(L)
S1 – D70
S2 – D80
D1 – Q3
REP
S1
S2
D1
D70·D71 12345678 ⟷ D80·D81 –12345678 → Q3 turned on
D70·D71 –12345678 ⟷ D80·D81 34567890 → Q3 turned off

- **Data Type: Float**


I4
CMP>=(F)
S1 – D90
S2 – D95
D1 – Q4
REP
S1
S2
D1
D90·D91 12.4 ⟷ D95·D96 12.345 → Q4 turned on
D90·D91 –1 ⟷ D95·D96 –0.99 → Q4 turned off

Repeat Operation in the Data Comparison Instructions

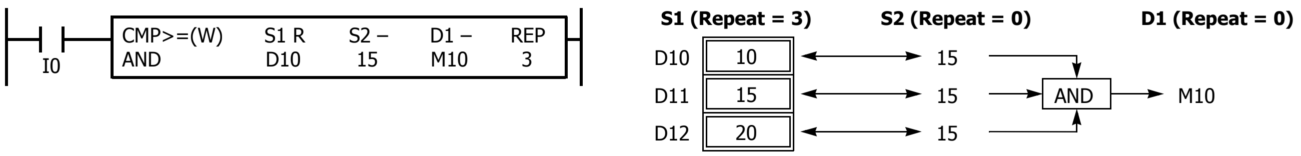
The following examples use the CMP>= instruction of word and double word data. Repeat operation for all other data comparison instructions and other data types is the same as the following examples.

When the repeat operation is enabled, repeated comparison results of CMP instructions can be selected from AND or OR operation, and the result is output to an output or internal relay.

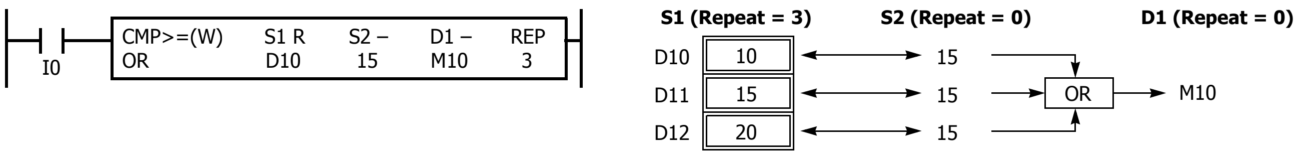
Repeat One Source Device

When only S1 (source) is designated to repeat, source devices (as many as the repeat cycles, starting with the device designated by S1) are compared with the device designated by S2. The comparison results are ANDed or ORed and set to the destination device designated by D1.

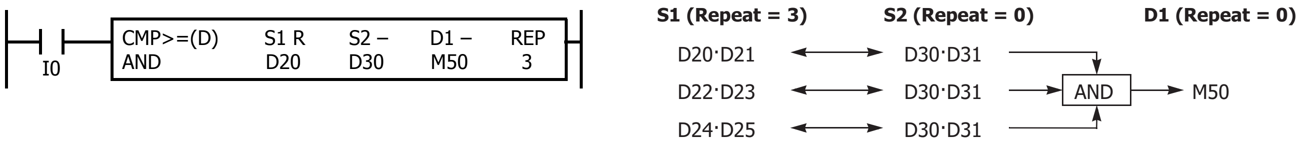
- **Data Type: Word (Repeat Logical Operation AND)**



- **Data Type: Word (Repeat Logical Operation OR)**



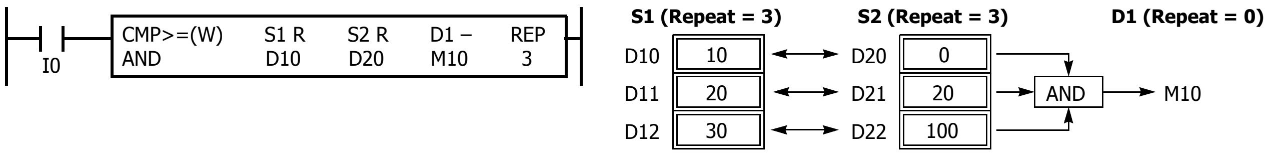
- **Data Type: Double Word (Repeat Logical Operation AND)**



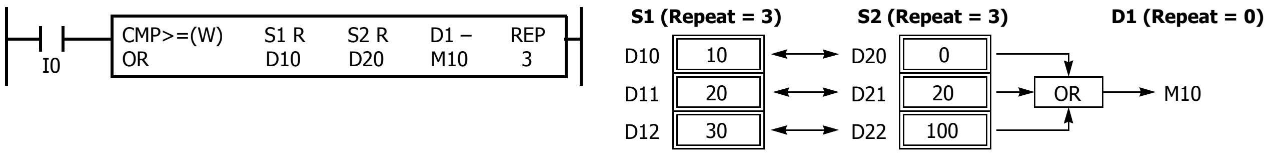
Repeat Two Source Devices

When S1 (source) and S2 (source) are designated to repeat, source devices (as many as the repeat cycles, starting with the devices designated by S1 and S2) are compared with each other. The comparison results are ANDed or ORed and set to the destination device designated by D1.

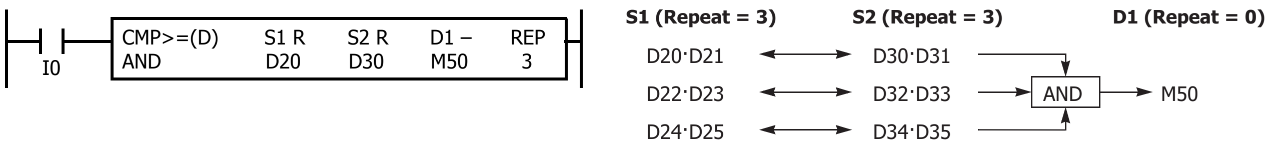
- **Data Type: Word (Repeat Logical Operation AND)**



- **Data Type: Word (Repeat Logical Operation OR)**



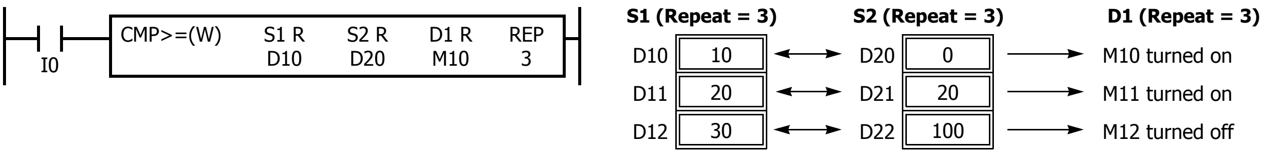
- **Data Type: Double Word (Repeat Logical Operation AND)**



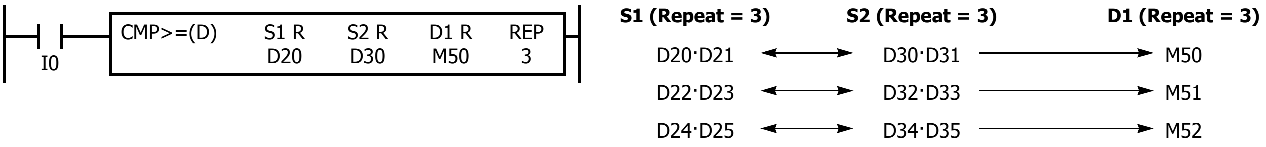
Repeat Source and Destination Devices

When S1, S2 (source), and D1 (destination) are designated to repeat, source devices (as many as the repeat cycles, starting with the devices designated by S1 and S2) are compared with each other. The comparison results are set to destination devices (as many as the repeat cycles, starting with the device designated by D1).

- **Data Type: Word**

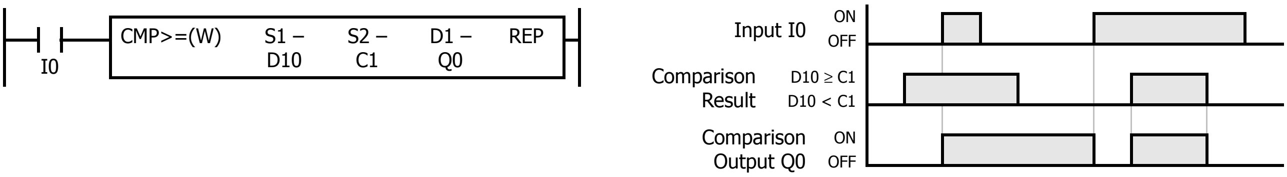


- **Data Type: Double Word**

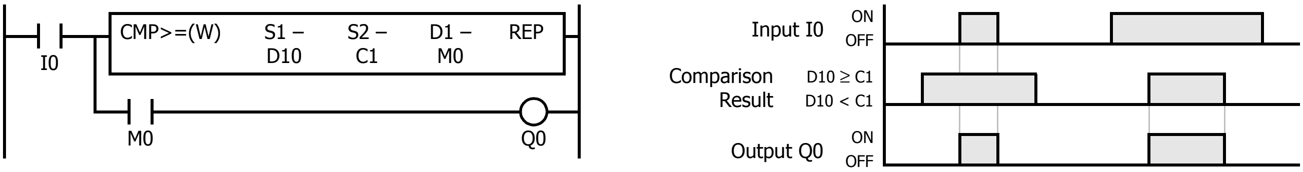


Comparison Output Status

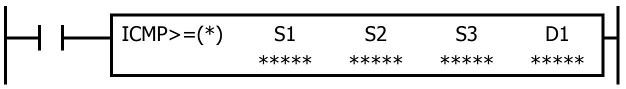
The comparison output is usually maintained while the input to the data comparison instruction is off. If the comparison output is on, the on status is maintained when the input is turned off as demonstrated by this program.



This program turns the output off when the input is off.



ICMP>= (Interval Compare Greater Than or Equal To)



Data type W or I: S1 ≥ S2 ≥ S3 → D1 on
Data type D, L, F: S1·S1+1 ≥ S2·S2+1 ≥ S3·S3+1 → D1 on

When input is on, the 16- or 32-bit data assigned by S1, S2, and S3 are compared. When the condition is met, destination device D1 is turned on. When the condition is not met, D1 is turned off.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Data to compare	X	X	X	X	X	X	X	—	X	—
S2 (Source 2)	Data to compare	X	X	X	X	X	X	X	—	X	—
S3 (Source 3)	Data to compare	X	X	X	X	X	X	X	—	X	—
D1 (Destination 1)	Comparison output	—	X	▲	—	—	—	—	—	—	—

For the valid device address range, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S1, S2, or S3, the timer/counter current value (TC or CC) is displayed.

When F (float) data is selected, only data register and constant can be designated as S1, S2, and S3.

When F (float) data is selected and S1, S2, or S3 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart.

When the data of S1 is smaller than that of S3 (S1 < S3), a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For user program execution errors, see "User Program Execution Errors" on page 3-10.

W (word)	X
I (integer)	X
D (double word)	X
L (long)	X
F (float)	X

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source, 16 points (word or integer data) or 32 points (double-word or long data) are used.

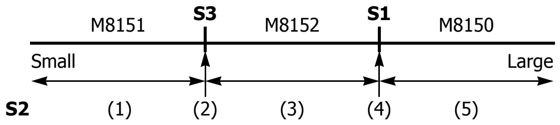
When a word device such as T (timer), C (counter), or D (data register) is assigned as the source, 1 point (word or integer data) or 2 points (double-word, long, or float data) are used.

The destination uses only one output or internal relay regardless of the selected data type.

Special Internal Relays M8150, M8151, and M8152 in ICMP>=

Three special internal relays are available to indicate the comparison result of the ICMP>= instruction. Depending on the result, one of the three special internal relays turns on. S1 must always be greater than or equal to S3 (S1 ≥ S3).

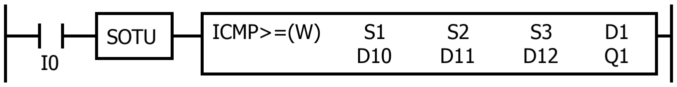
When S2 > S1, M8150 turns on.
When S2 < S3, M8151 turns on.
When S1 > S2 > S3, M8152 turns on.



S2 Value	M8150	M8151	M8152	D1 Status
(1) S2 < S3	OFF	**ON**	OFF	OFF
(2) S2 = S3	OFF	OFF	OFF	**ON**
(3) S3 < S2 < S1	OFF	OFF	**ON**	**ON**
(4) S2 = S1	OFF	OFF	OFF	**ON**
(5) S2 > S1	**ON**	OFF	OFF	OFF

When more than one ICMP>= or CMP= instruction is used, M8150, M8151, or M8152 indicates the result of the instruction that was executed last.

Example: ICMP>=

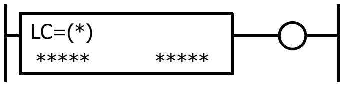


D10 ≥ D11 ≥ D12 → Q1 goes on

When input I0 is turned on, data of data registers D10, D11, and D12 assigned by source devices S1, S2, and S3 are compared. When the condition is met, internal relay Q1 assigned by destination device D1 is turned on. When the condition is not met, Q1 is turned off.

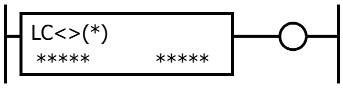
	S1			S2			S3		D1	M8150	M8151	M8152	M8004
D10	17	>	D11	15	=	D12	15	→	Q1 goes on	OFF	OFF	OFF	OFF
D10	15	<	D11	18	<	D12	19	→	Q1 goes off	ON	ON	OFF	ON

LC= (Load Compare Equal To)



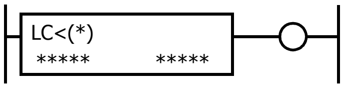
Data type W or I: S1 = S2
Data type D, L, or F: S1·S1+1 = S2·S2+1
This instruction constantly compares 16- or 32- bit data assigned by S1 and S2. When S1 data is equal to S2 data, the output to the following instructions is turned on. When the condition is not met, the output is turned off.

LC<> (Load Compare Unequal To)



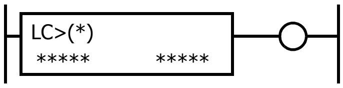
Data type W or I: S1 ≠ S2
Data type D, L, or F: S1·S1+1 ≠ S2·S2+1
This instruction constantly compares 16- or 32- bit data assigned by S1 and S2. When S1 data is not equal to S2 data, the output to the following instructions is turned on. When the condition is not met, the output is turned off.

LC< (Load Compare Less Than)



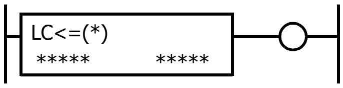
Data type W or I: S1 < S2
Data type D, L, or F: S1·S1+1 < S2·S2+1
This instruction constantly compares 16- or 32- bit data assigned by S1 and S2. When S1 data is less than S2 data, the output to the following instructions is turned on. When the condition is not met, the output is turned off.

LC> (Load Compare Greater Than)



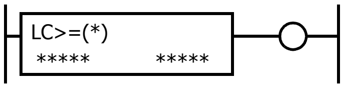
Data type W or I: S1 > S2
Data type D, L, or F: S1·S1+1 > S2·S2+1
This instruction constantly compares 16- or 32- bit data assigned by S1 and S2. When S1 data is greater than S2 data, the output to the following instructions is turned on. When the condition is not met, the output is turned off.

LC<= (Load Compare Less Than or Equal To)



Data type W or I: S1 ≦ S2
Data type D, L, or F: S1·S1+1 ≦ S2·S2+1
This instruction constantly compares 16- or 32- bit data assigned by S1 and S2. When S1 data is less than or equal to S2 data, the output to the following instructions is turned on. When the condition is not met, the output is turned off.

LC>= (Load Compare Greater Than or Equal To)



Data type W or I: S1 ≥ S2
Data type D, L, or F: S1·S1+1 ≥ S2·S2+1
This instruction constantly compares 16- or 32- bit data assigned by S1 and S2. When S1 data is greater than or equal to S2 data, the output to the following instructions is turned on. When the condition is not met, the output is turned off.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Data to compare	X	X	X	X	X	X	X	—	X	—
S2 (Source 2)	Data to compare	X	X	X	X	X	X	X	—	X	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When T (timer) or C (counter) is used, the timer/counter current value (TC or CC) is displayed.

When F (float) data is selected, only data register and constant can be designated.

When F (float) data is selected and S1 or S2 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. The output to the following instructions is turned off. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Data Types

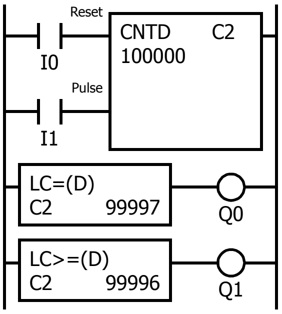
W (word)	X
I (integer)	X
D (double word)	X
L (long)	X
F (float)	X

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned, 16 points (word or integer data) or 32 points (double-word or long data) are used.

When a word device such as T (timer), C (counter), or D (data register) is assigned, 1 point (word or integer data) or 2 points (double-word, long, or float data) are used.

Examples: LC

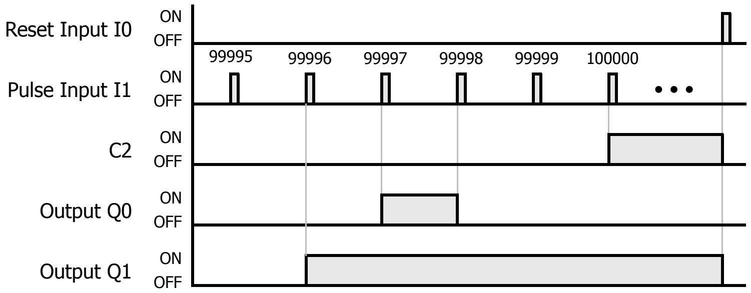
Ladder Diagram 1



Program List

Instruction	Data
LOD	I0
LOD	I1
CNTD	C2
	100000
LC=(D)	C2
	99997
OUT	Q0
LC>=(D)	C2
	99996
OUT	Q1

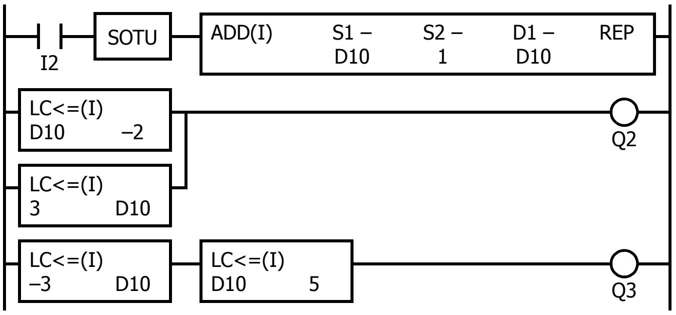
Timing Chart



Output Q0 is on when counter C2 current value is 99,997.

Output Q1 is turned on when counter C2 current value reaches 99,996 and remains on until counter C2 is reset.

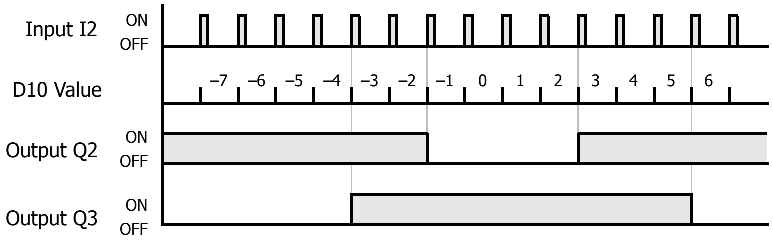
Ladder Diagram 2



Program List

Instruction	Data
LOD	I2
SOTU	
ADD(I)	D10
	1
	D10
LC<=(I)	D10
	–2
LC<=(I)	3
	D10
ORLOD	
OUT	Q2
LC<=(I)	–3
	D10
LC<=(I)	D10
	5
ANDLOD	
OUT	Q3

Timing Chart



Output Q2 is on when data register D10 is less than or equal to –2 and greater than or equal to 3.

Output Q3 is on while data register D10 is between –3 and 5.

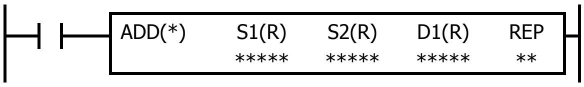
7: Binary Arithmetic Instructions

Introduction

This chapter describes the arithmetic operation instructions that perform arithmetic based on specified data and store those results in devices. For addition and subtraction devices, internal relay M8003 is used to carry or to borrow.

The ROOT instruction can be used to calculate the square root of the value stored in one or two data registers.

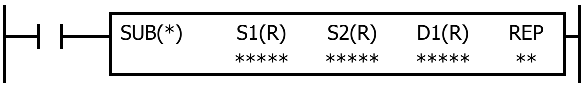
ADD (Addition)



Data type W or I:	S1 + S2 → D1, CY
Data type D, L, or F:	S1·S1+1 + S2·S2+1 → D1·D1+1, CY

When input is on, 16- or 32-bit data assigned by source devices S1 and S2 are added together. The result is set to destination device D1 and internal relay M8003 (carry or borrow).

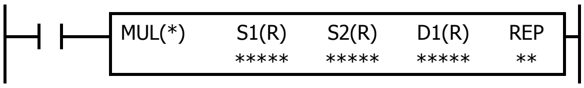
SUB (Subtraction)



Data type W or I:	S1 – S2 → D1, BW
Data type D, L, or F:	S1·S1+1 – S2·S2+1 → D1·D1+1, BW

When input is on, 16- or 32-bit data assigned by source device S2 is subtracted from 16- or 32-bit data assigned by source device S1. The result is set to destination device D1 and internal relay M8003 (carry or borrow).

MUL (Multiplication)

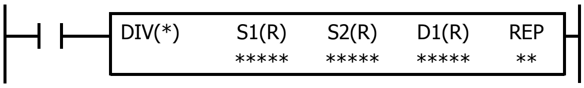


Data type W or I:	S1 × S2 → D1·D1+1
Data type D, L, or F:	S1·S1+1 × S2·S2+1 → D1·D1+1

When input is on, 16- or 32-bit data assigned by source device S1 is multiplied by 16- or 32-bit data assigned by source device S2. The result is set to destination device D1.

When the result exceeds the valid range for data types D or L, the ERR LED and special internal relay M8004 (user program execution error) are turned on.

DIV (Division)



Data type W or I:	S1 ÷ S2 → D1 (quotient), D1+1 (remainder)
Data type D or L:	S1·S1+1 ÷ S2·S2+1 → D1·D1+1 (quotient), D1+2·D1+3 (remainder)
Data type F:	S1·S1+1 ÷ S2·S2+1 →D1·D1+1 (quotient)

When input is on, 16- or 32-bit data assigned by source device S1 is divided by 16- or 32-bit data assigned by source device S2. The quotient is set to 16- or 32-bit destination device D1, and the remainder is set to the next 16- or 32-bit data. Data type F does not generate a remainder.

When S2 is 0 (dividing by 0), the ERR LED and special internal relay M8004 (user program execution error) are turned on.

A user program execution error also occurs in the following division operations.

Data type I:	–32768 ÷ (–1)
Data type L:	–2147483648 ÷ (–1)

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Data for calculation	X	X	X	X	X	X	X	X	X	1-99
S2 (Source 2)	Data for calculation	X	X	X	X	X	X	X	X	X	1-99
D1 (Destination 1)	Destination to store results	—	X	▲	X	X	X	X	X	—	1-99

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S1 or S2, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP) which can be 0 through 65,535.

When F (float) data is selected, only data register and constant can be designated as S1 and S2.

When F (float) data is selected and S1 or S2 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For user program execution errors, see "User Program Execution Errors" on page 3-10.

P (index register) can be used when ADD, SUB, or MUL is programmed with the data type L (long).

Since the binary arithmetic instructions are executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	X
I (integer)	X
D (double word)	X
L (long)	X
F (float)	X

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source, 16 points (word or integer data) or 32 points (double-word, long, or float data) are used. When repeat is assigned for a bit device, the quantity of device bits increases in 16- or 32-point increments.

When a word device such as T (timer), C (counter), or D (data register) is assigned as the source, 1 point (word or integer data) or 2 points (double-word, long, or float data) are used. When repeat is assigned for a word device, the quantity of device words increases in 1- or 2-point increments.

Using Carry or Borrow Signals

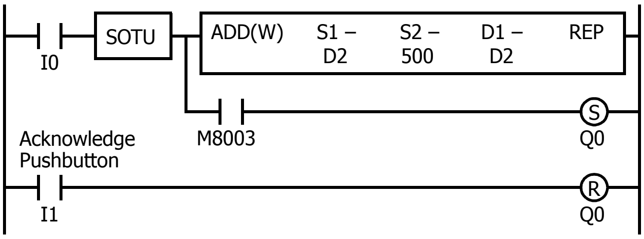
When the D1 (destination) data exceeds the valid data range as a result of any binary arithmetic operation, a carry or borrow occurs, and special internal relay M8003 is turned on.

Data Type	Carry/borrow Occurs when D1 Exceeds the Range between
W (word)	0 and 65,535
I (integer)	–32,768 and 32,767
D (double word)	0 and 4,294,967,295
L (long)	–2,147,483,648 and 2,147,483,647
F (float)	-3.402823×10^{38} and -1.175495×10^{-38} 1.175495×10^{-38} and 3.402823×10^{38}

Examples: ADD

- **Data Type: Word**

This example demonstrates the use of a carry signal from special internal relay M8003 to set an alarm signal.

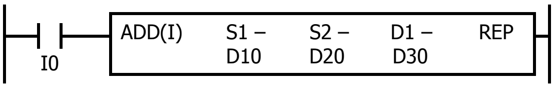


D2 + 500 → D2

When a carry occurs, output Q0 is assigned as a warning indicator.

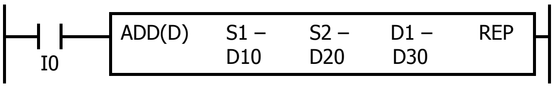
When the acknowledge pushbutton (input I1) is pressed, the warning indicator is reset.

- **Data Type: Integer**



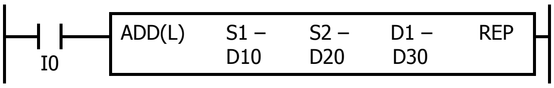
D10 −4 + D20 −11 → D30 −15

- **Data Type: Double Word**



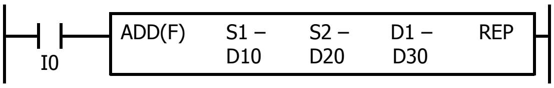
D10·D11 1957400 + D20·D21 4112600 → D30·D31 6070000

- **Data Type: Long**



D10·D11 216283 + D20·D21 −964355 → D30·D31 −748072

- **Data Type: Float**

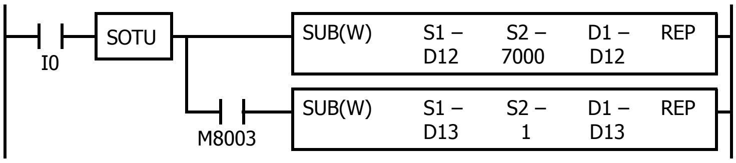


D10·D11 1.414 + D20·D21 3.14 → D30·D31 4.554

Example: SUB

- **Data Type: Word**

The following example demonstrates the use of special internal relay M8003 to process a borrow.


I0
SOTU
SUB(W) S1 – D12 S2 – 7000 D1 – D12 REP
M8003
SUB(W) S1 – D13 S2 – 1 D1 – D13 REP

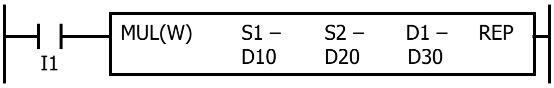
D12 − 7000 → D12

Borrow is processed so that the number of times a borrow occurs is subtracted from D13.

When a borrow occurs, D13 is decremented by one.

Examples: MUL

- **Data Type: Word**

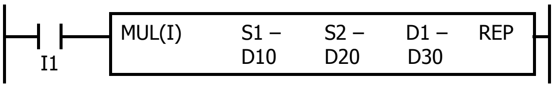


D10 500 (01F4h) × D20 600 (0258h) ⟶ D30·D31 300000 (000493E0h)

When input I1 is on, data of D10 is multiplied by data of D20, and the result is set to D30 and D31.

D30	4 (0004h)
D31	37856 (93E0h)

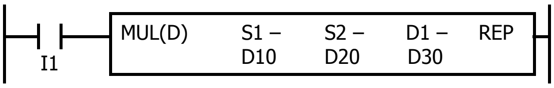
- **Data Type: Integer**



D10 –500 (FE0Ch) × D20 600 (0258h) ⟶ D30·D31 –300000 (FFFB6C20h)

D30	65531 (FFFBh)
D31	27680 (6C20h)

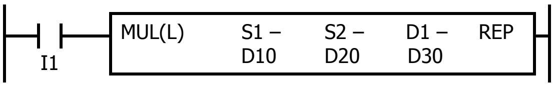
- **Data Type: Double Word**



D10·D11 100000 × D20·D21 5000 ⟶ D30·D31 500000000

Note: In multiplication of double word data, the lower 32-bit data of the result is set to destination device D1·D1+1.

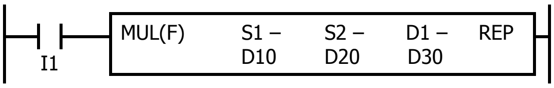
- **Data Type: Long**



D10·D11 –100000 × D20·D21 –5000 ⟶ D30·D31 500000000

Note: In multiplication of long data, the lower 32-bit data of the result is set to destination device D1·D1+1.

- **Data Type: Float**

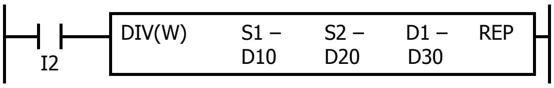


D10·D11 4.554 × D20·D21 1.414 ⟶ D30·D31 6.439356

Note: Since the destination uses two word devices in the multiplication operation, data register such as D7999 cannot be used as destination device D1. When using a bit device such as internal relay for destination, 32 internal relays are required; so internal relay such as M7961 cannot be used as destination device D1.

Examples: DIV

- **Data Type: Word**



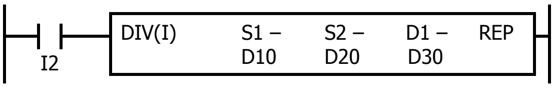
D10 $\boxed{50}$ ÷ D20 $\boxed{7}$ ⟶ D30 $\boxed{7}$ D31 $\boxed{1}$

Quotient Remainder

When input I2 is on, D10 data is divided by D20 data. The quotient is set to D30, and the remainder is set to D31.

Note: Since the destination uses two word devices in the division operation of word data, data register such as D7999 cannot be used as destination device D1. When using a bit device, such as internal relay for destination, 32 internal relays are required; so internal relay such as M7961 cannot be used as destination device D1.

- **Data Type: Integer**

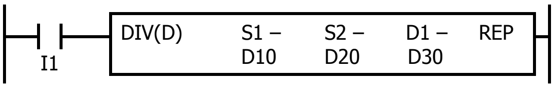


D10 $\boxed{50}$ ÷ D20 $\boxed{-7}$ ⟶ D30 $\boxed{-7}$ D31 $\boxed{1}$

Quotient Remainder

Note: Since the destination uses two word devices in the division operation of integer data, data register such as D7999 cannot be used as destination device D1. When using a bit device, such as internal relay for destination, 32 internal relays are required; so internal relay such as M7961 cannot be used as destination device D1.

- **Data Type: Double Word**

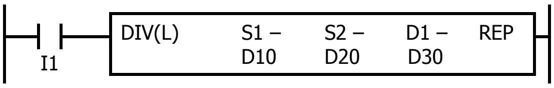


D10·D11 $\boxed{100000}$ ÷ D20·D21 $\boxed{70000}$ ⟶ D30·D31 $\boxed{1}$ D32·D33 $\boxed{30000}$

Quotient Remainder

Note: Since the destination uses four word devices in the division operation of double-word data, data registers such as D7997 cannot be used as destination device D1. When using a bit device such as internal relay for destination, 64 internal relays are required; so internal relay such as M7921 cannot be used as destination device D1.

- **Data Type: Long**

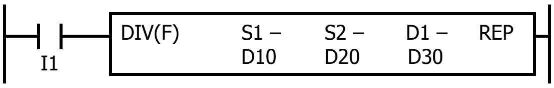


D10·D11 $\boxed{100000}$ ÷ D20·D21 $\boxed{-70000}$ ⟶ D30·D31 $\boxed{-1}$ D32·D33 $\boxed{30000}$

Quotient Remainder

Note: Since the destination uses four word devices in the division operation of long data, data registers such as D7997 cannot be used as destination device D1. When using a bit device, such as internal relay for destination, 64 internal relays are required; so internal relay such as M7921 cannot be used as destination device D1.

- **Data Type: Float**



D10·D11 $\boxed{4.43996}$ ÷ D20·D21 $\boxed{3.14}$ ⟶ D30·D31 $\boxed{1.414}$

Quotient

Note: Since the destination uses two word devices in the division operation of float data, data register such as D7999 cannot be used as destination device D1.

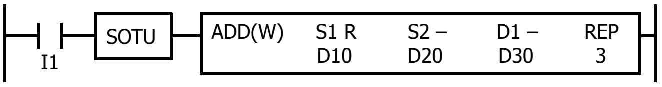
Repeat Operation in the ADD and SUB Instructions

Source devices S1 and S2 and destination device D1 can be assigned to repeat individually or in combination. When destination device D1 is not set to repeat, the final result is set to destination device D1. When repeat is assigned, as many consecutive devices as the repeat cycles, starting with the designated device, are used. Since the repeat operation works similarly on the ADD (addition) and SUB (subtraction) instructions, the following examples are described using the ADD instruction.

Repeat One Source Device

- **Data Type: Word and Integer**

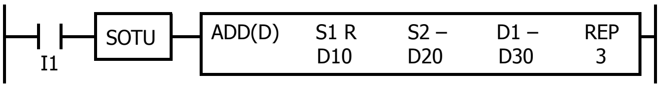
When only S1 (source) is assigned to repeat, the final result is set to destination device D1.



S1 (Repeat = 3)			S2 (Repeat = 0)			D1 (Repeat = 0)	
D10	10	+	D20	25	→	D30	(35)
D11	15	+	D20	25	→	D30	(40)
D12	20	+	D20	25	→	D30	45

- **Data Type: Double Word, Long, and Float**

When only S1 (source) is assigned to repeat, the final result is set to destination device D1·D1+1.

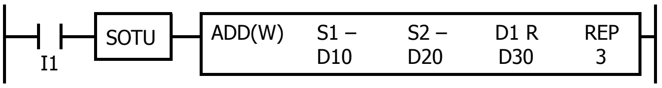


S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 0)
D10·D11	+	D20·D21	→	(D30·D31)
D12·D13	+	D20·D21	→	(D30·D31)
D14·D15	+	D20·D21	→	D30·D31

Repeat Destination Device Only

- **Data Type: Word and Integer**

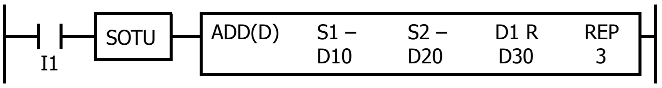
When only D1 (destination) is assigned to repeat, the same result is set to 3 devices starting with D1.



S1 (Repeat = 0)			S2 (Repeat = 0)			D1 (Repeat = 3)	
D10	10	+	D20	25	→	D30	35
D10	10	+	D20	25	→	D31	35
D10	10	+	D20	25	→	D32	35

- **Data Type: Double Word, Long, and Float**

When only D1 (destination) is assigned to repeat, the same result is set to 3 devices starting with D1·D1+1.

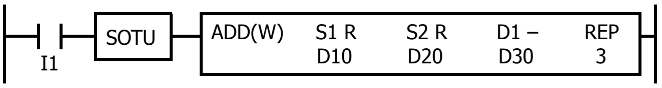


S1 (Repeat = 0)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10·D11	+	D20·D21	→	D30·D31
D10·D11	+	D20·D21	→	D32·D33
D10·D11	+	D20·D21	→	D34·D35

Repeat Two Source Devices

- **Data Type: Word and Integer**

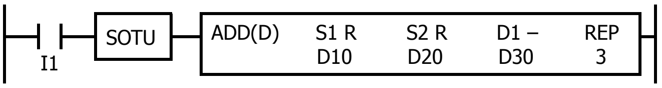
When S1 and S2 (source) are assigned to repeat, the final result is set to destination device D1.



S1 (Repeat = 3)			S2 (Repeat = 3)			D1 (Repeat = 0)	
D10	10	+	D20	25	→	D30	(35)
D11	15	+	D21	35	→	D30	(50)
D12	20	+	D22	45	→	D30	65

- **Data Type: Double Word, Long, and Float**

When S1 and S2 (source) are assigned to repeat, the final result is set to destination device D1·D1+1.

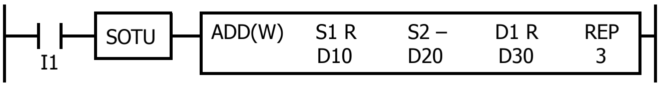


S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 0)
D10·D11	+	D20·D21	→	(D30·D31)
D12·D13	+	D22·D23	→	(D30·D31)
D14·D15	+	D24·D25	→	D30·D31

Repeat Source and Destination Devices

- **Data Type: Word and Integer**

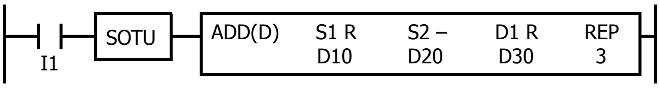
When S1 (source) and D1 (destination) are assigned to repeat, different results are set to 3 devices starting with D1.



S1 (Repeat = 3)			S2 (Repeat = 0)			D1 (Repeat = 3)	
D10	10	+	D20	25	→	D30	35
D11	15	+	D20	25	→	D31	40
D12	20	+	D20	25	→	D32	45

- **Data Type: Double Word, Long, and Float**

When S1 (source) and D1 (destination) are assigned to repeat, different results are set to 3 devices starting with D1·D1+1.

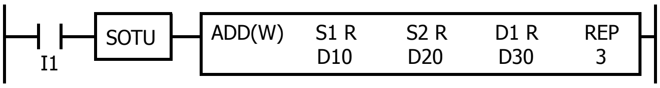


S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10·D11	+	D20·D21	→	D30·D31
D12·D13	+	D20·D21	→	D32·D33
D14·D15	+	D20·D21	→	D34·D35

Repeat All Source and Destination Devices

- **Data Type: Word and Integer**

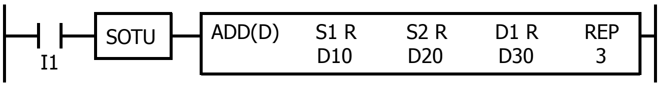
When all devices are assigned to repeat, different results are set to 3 devices starting with D1.



S1 (Repeat = 3)			S2 (Repeat = 3)			D1 (Repeat = 3)	
D10	10	+	D20	25	→	D30	35
D11	15	+	D21	35	→	D31	50
D12	20	+	D22	45	→	D32	65

- **Data Type: Double Word, Long, and Float**

When all devices are assigned to repeat, different results are set to 3 devices starting with D1·D1+1.



S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 3)
D10·D11	+	D20·D21	→	D30·D31
D12·D13	+	D22·D23	→	D32·D33
D14·D15	+	D24·D25	→	D34·D35

Note: Special internal relay M8003 (carry/borrow) is turned on when a carry or borrow occurs in the last repeat operation. When a user program execution error occurs in any repeat operation, special internal relay M8004 (user program execution error) and the ERR LED are turned on and maintained while operations for other instructions continues.

Repeat Operation in the MUL Instruction

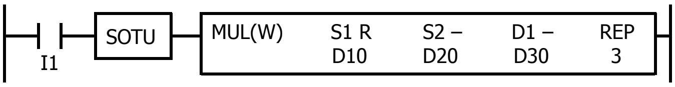
Since the MUL (multiplication) instruction uses two destination devices, the result is stored to destination devices as described below. Source devices S1 and S2 and destination device D1 can be designated to repeat individually or in combination. When destination device D1 is not assigned to repeat, the final result is set to destination device D1 and D1+1. When repeat is designated, consecutive devices as many as the repeat cycles starting with the designated device are used.

Since the repeat operation works similarly on word and integer data, the following examples are described using word data.

Repeat One Source Device

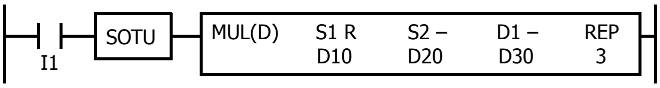
When only S1 (source) is assigned to repeat, the final result is set to destination device D1·D1+1.

- **Data Type: Word and Integer**



S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 0)
D10	×	D20	→	(D30·D31)
D11	×	D20	→	(D30·D31)
D12	×	D20	→	D30·D31

- **Data Type: Double Word, Long, and Float**

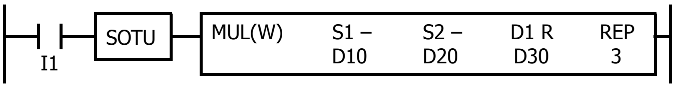


S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 0)
D10·D11	×	D20·D21	→	(D30·D31)
D12·D13	×	D20·D21	→	(D30·D31)
D14·D15	×	D20·D21	→	D30·D31

Repeat Destination Device Only

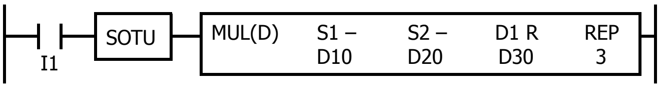
When only D1 (destination) is assigned to repeat, the same result is set to 3 devices starting with D1·D1+1.

- **Data Type: Word and Integer**



S1 (Repeat = 0)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10	×	D20	→	D30·D31
D10	×	D20	→	D32·D33
D10	×	D20	→	D34·D35

- **Data Type: Double Word, Long, and Float**

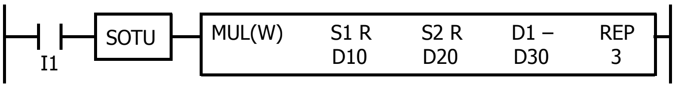


S1 (Repeat = 0)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10·D11	×	D20·D21	→	D30·D31
D10·D11	×	D20·D21	→	D32·D33
D10·D11	×	D20·D21	→	D34·D35

Repeat Two Source Devices

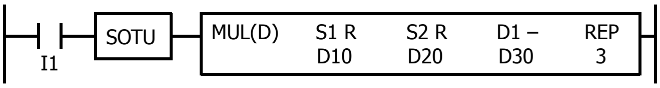
When S1 and S2 (source) are assigned to repeat, the final result is set to destination device D1·D1+1.

- **Data Type: Word and Integer**



S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 0)
D10	×	D20	→	(D30·D31)
D11	×	D21	→	(D30·D31)
D12	×	D22	→	D30·D31

- **Data Type: Double Word, Long, and Float**

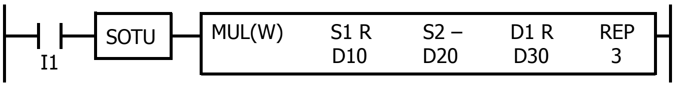


S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 0)
D10·D11	×	D20·D21	→	(D30·D31)
D12·D13	×	D22·D23	→	(D30·D31)
D14·D15	×	D24·D25	→	D30·D31

Repeat Source and Destination Devices

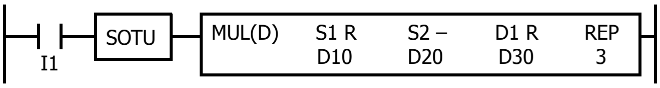
When S1 (source) and D1 (destination) are assigned to repeat, different results are set to 3 devices starting with D1·D1+1.

- **Data Type: Word and Integer**



S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10	×	D20	⟶	D30·D31
D11	×	D20	⟶	D32·D33
D12	×	D20	⟶	D34·D35

- **Data Type: Double Word, Long, and Float**

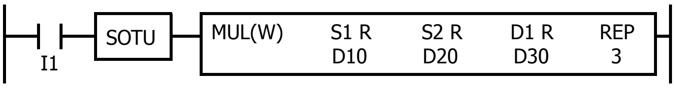


S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10·D11	×	D20·D21	⟶	D30·D31
D12·D13	×	D20·D21	⟶	D32·D33
D14·D15	×	D20·D21	⟶	D34·D35

Repeat All Source and Destination Devices

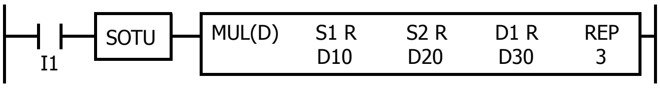
When all devices are assigned to repeat, different results are set to 3 devices starting with D1·D1+1.

- **Data Type: Word and Integer**



S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 3)
D10	×	D20	⟶	D30·D31
D11	×	D21	⟶	D32·D33
D12	×	D22	⟶	D34·D35

- **Data Type: Double Word, Long, and Float**



S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 3)
D10·D11	×	D20·D21	⟶	D30·D31
D12·D13	×	D22·D23	⟶	D32·D33
D14·D15	×	D24·D25	⟶	D34·D35

Repeat Operation in the DIV Instruction

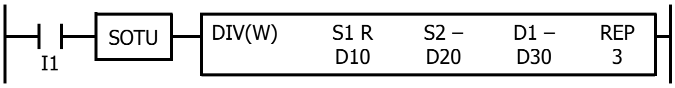
Since the DIV (division) instruction (except float data) uses two destination devices, the quotient and remainder are stored as described below. Source devices S1 and S2 and destination device D1 can be assigned to repeat individually or in combination. When destination device D1 is not assigned to repeat, the final result is set to destination device D1 (quotient) and D1+1 (remainder). When repeat is assigned, assigned as many consecutive devices as the repeat cycles starting with the designated device are used.

Division instructions in float data do not generate remainders and use two consecutive data registers to store quotients. When repeat is assigned for destination of float data, as many consecutive data registers as the repeat cycles are used.

Repeat One Source Device

- **Data Type: Word and Integer**

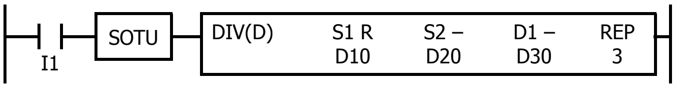
When only S1 (source) is assigned to repeat, the final result is set to destination devices D1 and D1+1.



S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 0)	
D10	÷	D20	→	(D30)	(D31)
D11	÷	D20	→	(D30)	(D31)
D12	÷	D20	→	D30	D31
				Quotient	Remainder

- **Data Type: Double Word and Long**

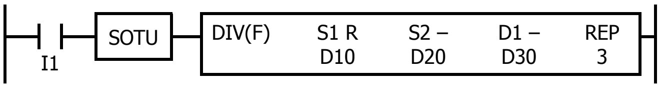
When only S1 (source) is assigned to repeat, the final result is set to destination devices D1·D1+1 and D1+2·D1+3.



S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 0)	
D10·D11	÷	D20·D21	→	(D30·D31)	(D32·D33)
D12·D13	÷	D20·D21	→	(D30·D31)	(D32·D33)
D14·D15	÷	D20·D21	→	D30·D31	D32·D33
				Quotient	Remainder

- **Data Type: Float**

When only S1 (source) is assigned to repeat, the final result is set to destination devices D1·D1+1.

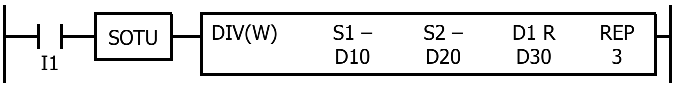


S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 0)
D10·D11	÷	D20·D21	→	(D30·D31)
D12·D13	÷	D20·D21	→	(D30·D31)
D14·D15	÷	D20·D21	→	D30·D31
				Quotient

Repeat Destination Device Only

- **Data Type: Word and Integer**

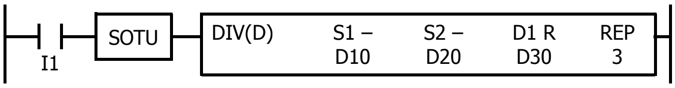
When only D1 (destination) is assigned to repeat, the same result is set to 6 devices starting with D1.



S1 (Repeat = 0)		S2 (Repeat = 0)		D1 (Repeat = 3)	
D10	÷	D20	→	D30	D33
D10	÷	D20	→	D31	D34
D10	÷	D20	→	D32	D35
				Quotient	Remainder

- **Data Type: Double Word and Long**

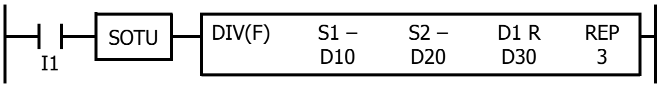
When only D1 (destination) is assigned to repeat, the same result is set to 6 devices starting with D1·D1+1.



S1 (Repeat = 0)		S2 (Repeat = 0)		D1 (Repeat = 3)	
D10·D11	÷	D20·D21	→	D30·D31	D36·D37
D10·D11	÷	D20·D21	→	D32·D33	D38·D39
D10·D11	÷	D20·D21	→	D34·D35	D40·D41
				Quotient	Remainder

- **Data Type: Float**

When only D1 (destination) is assigned to repeat, the same result is set to 3 devices starting with D1·D1+1.

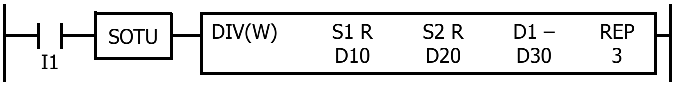


S1 (Repeat = 0)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10·D11	÷	D20·D21	→	D30·D31
D10·D11	÷	D20·D21	→	D32·D33
D10·D11	÷	D20·D21	→	D34·D35
				Quotient

Repeat Two Source Devices

- **Data Type: Word and Integer**

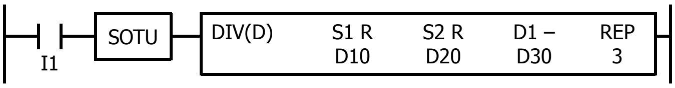
When S1 and S2 (source) are assigned to repeat, the final result is set to destination devices D1 and D1+1.



S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 0)	
D10	÷	D20	→	(D30)	(D31)
D11	÷	D21	→	(D30)	(D31)
D12	÷	D22	→	D30	D31
				Quotient	Remainder

- **Data Type: Double Word and Long**

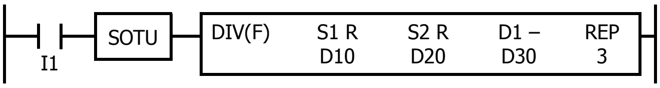
When S1 and S2 (source) are assigned to repeat, the final result is set to destination devices D1·D1+1 and D1+2·D1+3.



S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 0)	
D10·D11	÷	D20·D21	→	(D30·D31)	(D32·D33)
D12·D13	÷	D22·D23	→	(D30·D31)	(D32·D33)
D14·D15	÷	D24·D25	→	D30·D31	D32·D33
				Quotient	Remainder

- **Data Type: Float**

When S1 and S2 (source) are assigned to repeat, the final result is set to destination devices D1·D1+1.

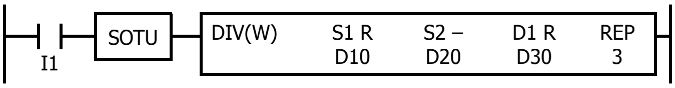


S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 0)
D10·D11	÷	D20·D21	→	(D30·D31)
D12·D13	÷	D22·D23	→	(D30·D31)
D14·D15	÷	D24·D25	→	D30·D31
				Quotient

Repeat Source and Destination Devices

- **Data Type: Word and Integer**

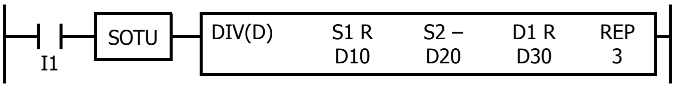
When S1 (source) and D1 (destination) are assigned to repeat, different results are set to 6 devices starting with D1.



S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 3)	
D10	÷	D20	→	D30	D33
D11	÷	D20	→	D31	D34
D12	÷	D20	→	D32	D35
				Quotient	Remainder

- **Data Type: Double Word and Long**

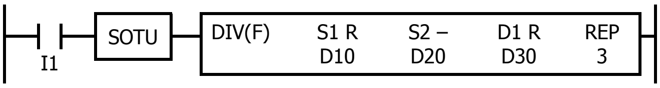
When S1 (source) and D1 (destination) are assigned to repeat, different results are set to 6 devices starting with D1·D1+1.



S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 3)	
D10·D11	÷	D20·D21	→	D30·D31	D36·D37
D12·D13	÷	D20·D21	→	D32·D33	D38·D39
D14·D15	÷	D20·D21	→	D34·D35	D40·D41
				Quotient	Remainder

- **Data Type: Float**

When S1 (source) and D1 (destination) are assigned to repeat, different results are set to 3 devices starting with D1·D1+1.

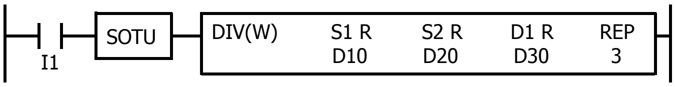


S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10·D11	÷	D20·D21	→	D30·D31
D12·D13	÷	D20·D21	→	D32·D33
D14·D15	÷	D20·D21	→	D34·D35
				Quotient

Repeat All Source and Destination Devices

- **Data Type: Word and Integer**

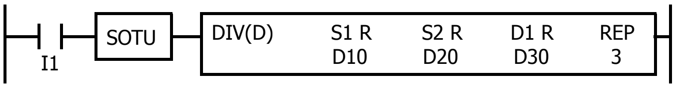
When all devices are assigned to repeat, different results are set to 6 devices starting with D1.



S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 3)	
D10	÷	D20	→	D30	D33
D11	÷	D21	→	D31	D34
D12	÷	D22	→	D32	D35
				Quotient	Remainder

- **Data Type: Double Word and Long**

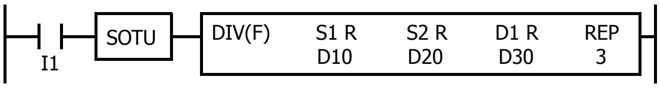
When all devices are assigned to repeat, different results are set to 6 devices starting with D1·D1+1.



S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 3)	
D10·D11	÷	D20·D21	→	D30·D31	D36·D37
D12·D13	÷	D22·D23	→	D32·D33	D38·D39
D14·D15	÷	D24·D25	→	D34·D35	D40·D41
				Quotient	Remainder

- **Data Type: Float**

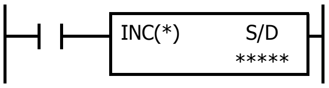
When all devices are assigned to repeat, different results are set to 3 devices starting with D1·D1+1.



S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 3)
D10·D11	÷	D20·D21	→	D30·D31
D12·D13	÷	D22·D23	→	D32·D33
D14·D15	÷	D24·D25	→	D34·D35
				Quotient

Note: When a user program execution error occurs in any repeat operation, special internal relay M8004 (user program execution error) and the ERR LED are turned on and maintained while operations for other instructions continues.

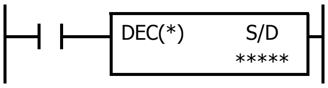
INC (Increment)



Data type W or I:	S/D + 1 → S/D
Data type D or L:	S/D·S/D+1 + 1 → S/D·S/D+1

When input is on, one is added to the 16- or 32-bit data assigned by device S/D and the result is stored to the same device.

DEC (Decrement)



Data type W or I:	S/D – 1 → S/D
Data type D or L:	S/D·S/D+1 – 1 → S/D·S/D+1

When input is on, one is subtracted from the 16- or 32-bit data assigned by device S/D and the result is stored to the same device.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S/D (Source/Destination)	Device to increment data	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

Since the INC and DEC instructions are executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	X
I (integer)	X
D (double word)	X
L (long)	X
F (float)	—

When a word device such as D (data register) is assigned as the source/destination, 1 point (word or integer data) or 2 points (double-word or long data) are used.

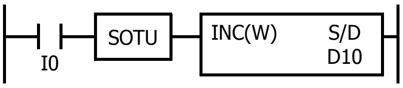
Increment beyond Limits

When the S/D value is at its maximum and is incremented by one, the value returns to 0, turning on the carry (M8003).

Decrement beyond Limits

When the S/D value is at its minimum and is decremented by one, the value returns to its maximum value (word or double-word data) or to –1 (integer or long data), turning on the internal relay M8003 (carry or borrow).

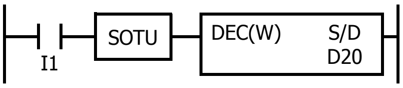
Example: INC



D10 $\boxed{100}$ + 1 → D10 $\boxed{101}$

When input I0 is turned on, the data of D10 is incremented by one.
If SOTU is not programmed, the data of D10 is incremented in each scan.

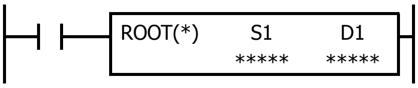
Example: DEC



D20 $\boxed{100}$ – 1 → D20 $\boxed{99}$

When input I1 is turned on, the data of D20 is decremented by one.
If SOTU is not programmed, the data of D20 is decremented in each scan.

ROOT (Root)



Data type W: $\sqrt{S1} \rightarrow D1$

When input is on, the square root of the device assigned by S1 is extracted and stored to the destination assigned by D1.

The square root is calculated to two decimals, omitting the figures below the second place of decimals, and multiplied by 100.

Data type D: $\sqrt{S1 \cdot S1+1} \rightarrow D1 \cdot D1+1$

When input is on, the square root of the device assigned by S1·S1+1 is extracted and is stored to the destination assigned by D1·D1+1.

The square root is calculated to two decimals, omitting the figures below the second place of decimals, and multiplied by 100.

Data type F: $\sqrt{S1 \cdot S1+1} \rightarrow D1 \cdot D1+1$

When input is on, the square root of the device assigned by S1·S1+1 is extracted and is stored to the destination assigned by D1·D1+1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Binary data	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When F (float) data is selected and source device S1 contains a negative value, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When F (float) data is selected and S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed. For user program execution errors, see "User Program Execution Errors" on page 3-10.

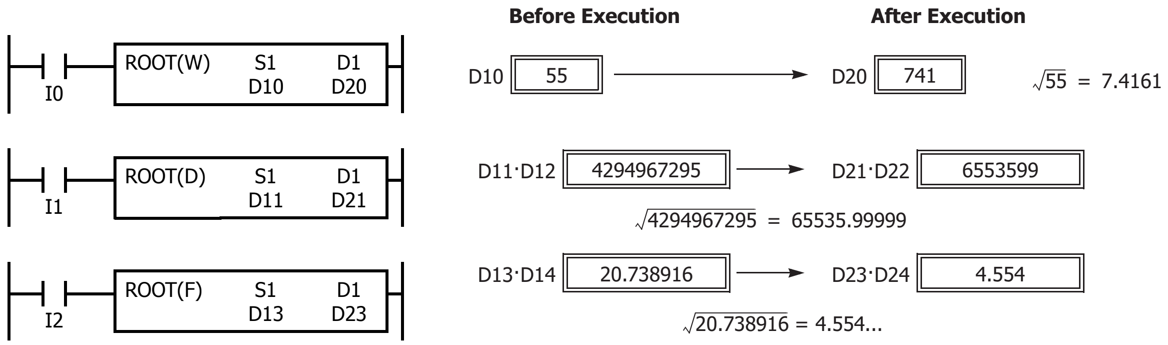
Since the ROOT instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

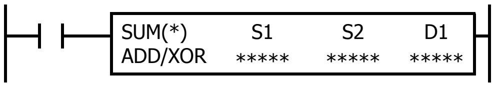
W (word)	X
I (integer)	—
D (double word)	X
L (long)	—
F (float)	X

When a word device such as D (data register) is assigned as the source or destination, 1 point (word data) or 2 points (double-word or float data) are used.

Examples: ROOT



SUM (Sum)



Calculates the total of assigned data, depending on the calculation option.

ADD:
When input is on, N blocks of 16- or 32-bit data starting at device assigned by S1 are added together and the result is stored to the device assigned by D1. S2 specifies the quantity of data blocks.

XOR:
When input is on, N blocks of 16-bit data starting at the device assigned by S1 are XORed and the result is stored to the device assigned by D1. S2 specifies the quantity of data blocks.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	First device address to calculate	—	—	—	—	X	X	X	—	—	—
S2 (Source 2)	Quantity of data blocks	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When T (timer) or C (counter) is used as S1, the timer/counter current value (TC or CC) is displayed.

When F (float) data is selected, only a data register can be designated as S1.

For source S2, 1 word is always used without regard to the data type.

When F (float) data is selected and S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart.

When S2 is 0 or exceeds the correct value range for the selected device, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Data Types

Calculation	ADD	XOR
W (word)	X	X
I (integer)	X	—
D (double word)	X	—
L (long)	X	—
F (float)	X	—

When ADD is selected, all data types can be used.

When XOR is selected, only W (word) data can be used.

When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word or integer data) or 2 points (double-word, long, or float data) are used.

Quantity of Source and Destination Devices

Depending on the ADD or XOR operation for W (word) and I (integer) data, the destination uses a different quantity of devices.

Operation	W (word), I (integer)		D (double word), L (long), F (float)	
ADD	S1, S2: D1:	1 word device 2 word devices	S1, D1: S2:	2 word devices 1 word device
XOR	S1, S2, D1:	1 word device	—	

Carry and Borrow

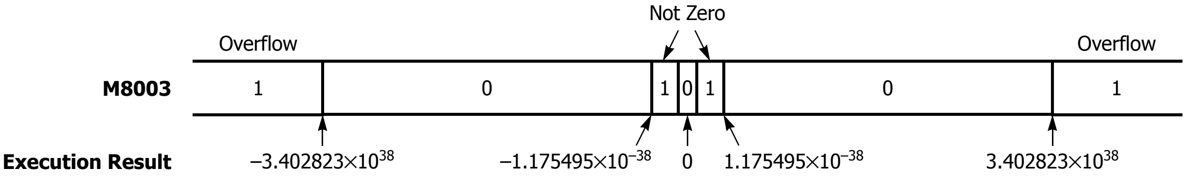
In advanced instructions involving D (double word), L (long), or F (floating point) data, special internal relay M8003 (carry and borrow) is turned on when the execution of the instruction results in the following value.

Data Type	M8003	Execution Result
D (double word)	1	Exceeds the range between 0 to 4,294,967,295
L (long)	1	Exceeds the range between –2,147,483,648 to 2,147,483,647
F (float)	1	See the figure below.

Carry and Borrow in Floating-Point Data Processing

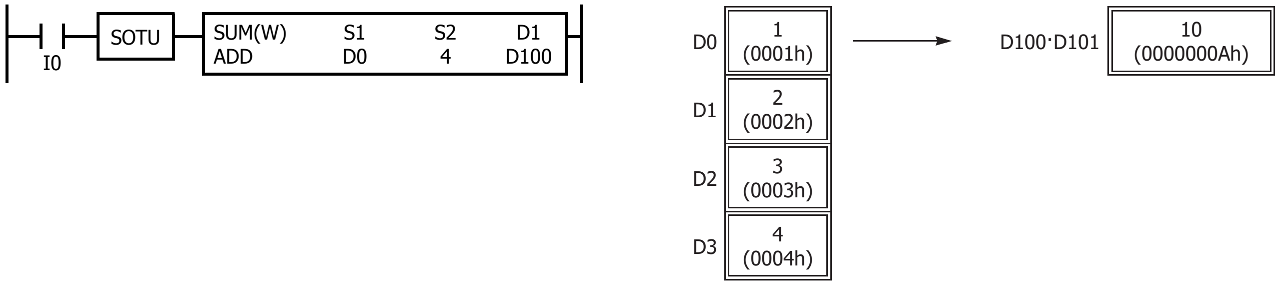
When advanced instructions involving floating-point data are executed, special internal relay M8003 (carry and borrow) is updated.

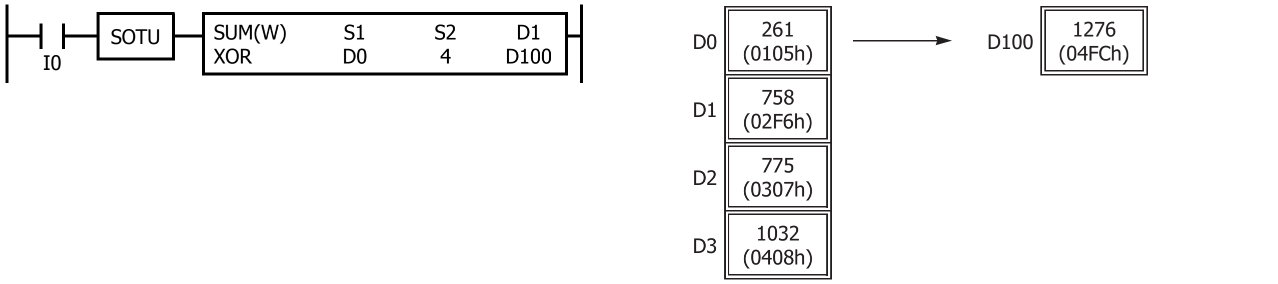
M8003	Execution Result	Value
1	**≠ 0**	Overflow (exceeds the range between -3.402823×10^{38} and 3.402823×10^{38})
1	**0**	Not zero (within the range between -1.175495×10^{-38} and 1.175495×10^{-38})
0	**0**	Zero



Examples: SUM

- **Data Type: Word**

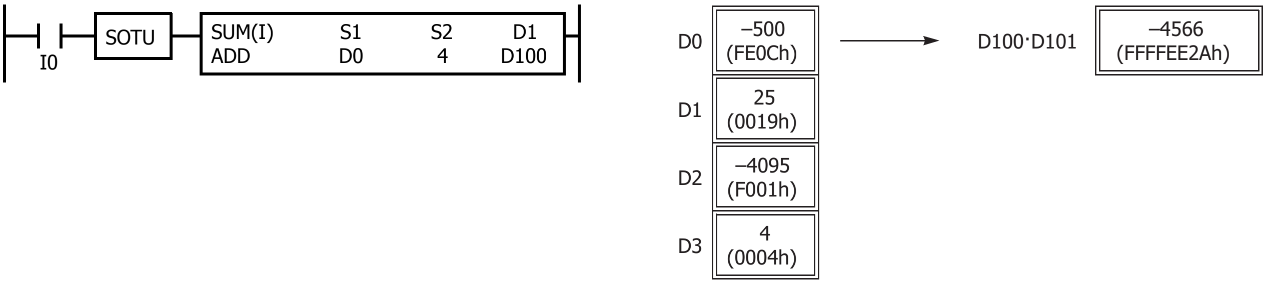




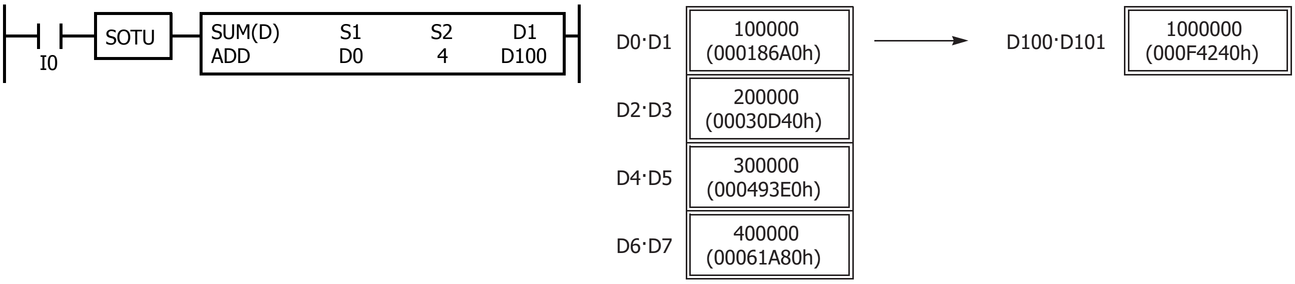
XOR Operation

	D0	(0105h)	0000 0001 0000 0101
	D1	(02F6h)	0000 0010 1111 0110
	D2	(0307h)	0000 0011 0000 0111
XOR	D3	(0408h)	0000 0100 0000 1000
	D100	(04FCh)	0000 0100 1111 1100

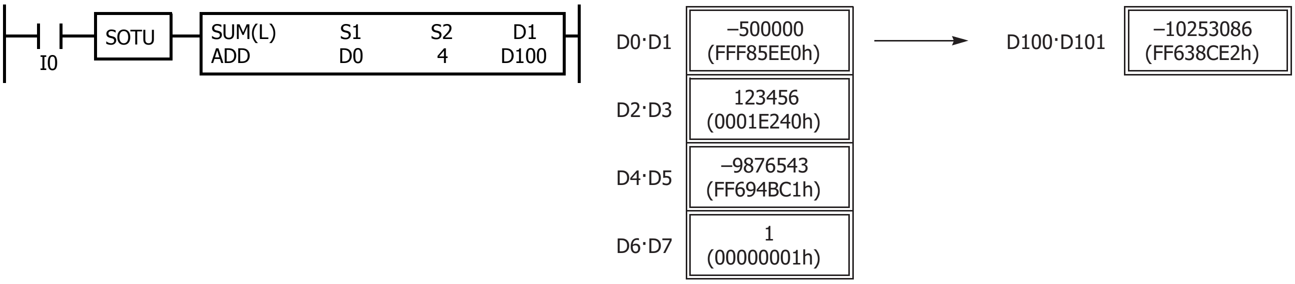
- **Data Type: Integer**



- **Data Type: Double Word**

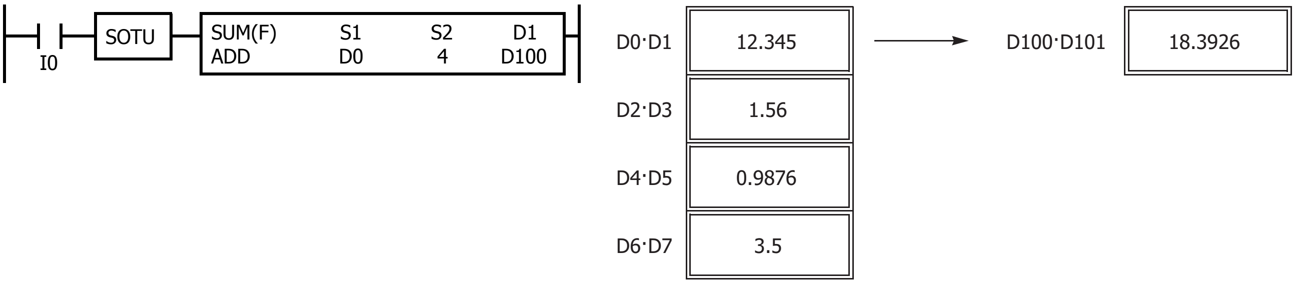


- **Data Type: Long**

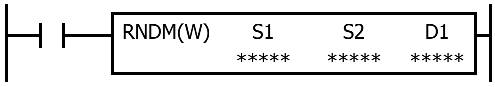


I0
SOTU
SUM(L) ADD
S1 D0
S2 4
D1 D100
D0·D1 –500000 (FFF85EE0h)
D2·D3 123456 (0001E240h)
D4·D5 –9876543 (FF694BC1h)
D6·D7 1 (00000001h)
D100·D101 –10253086 (FF638CE2h)

- **Data Type: Float**



RNDM (Random)



Generates pseudorandom numbers.

When the input is on, a pseudorandom number is generated with the data specified by S1 as the lower limit and the data specified by S2 as the upper limit, and that value is stored in D1.

The result is stored to the destination designated by device D1.

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Minimum value of pseudorandom numbers	—	—	—	—	—	—	X	—	X*1	—
S2 (Source 2)	Maximum value of pseudorandom numbers	—	—	—	—	—	—	X	—	X*1	—
D1 (Destination 1)	Destination to store results	—	—	—	—	—	—	X	—	—	—

*1 0 to 32,767 can be input.

For the valid device address range, see "Device Addresses" on page 2-1.

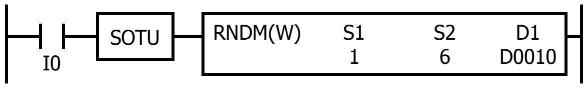
- When the data in S1 or S2 is outside the range of 0 to 32,767, the result is a user program execution error.
- When the data specified by S1 is the same value as the data specified by S2 or a larger value, the result is a user program execution error. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Data Types

W (word)	X
I (integer)	—
D (double word)	—
L (long)	—
F (float)	—

When a word device such as D (data register) is designated as the source or destination, 1 point (word) is used.

Example: RNDM



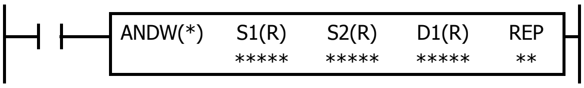
When the input is turned on, a pseudorandom number is generated in the range of 1 to 6, and the result is stored in D0010.

8: Boolean Computation Instructions

Introduction

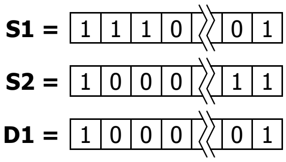
Boolean computations use the AND, OR, and exclusive OR statements as carried out by the ANDW, ORW, and XORW instructions in the word or double-word data, respectively.

ANDW (AND Word)



S1 · S2 → D1

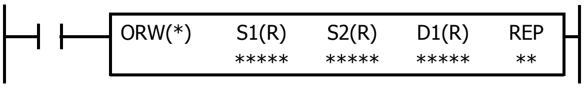
When input is on, 16- or 32-bit data assigned by source devices S1 and S2 are ANDed, bit by bit. The result is set to destination device D1.



AND Truth Table

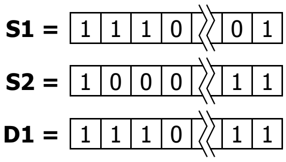
S1	S2	D1
0	0	0
0	1	0
1	0	0
1	1	1

ORW (OR Word)



S1 + S2 → D1

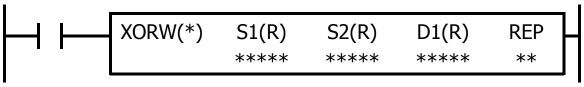
When input is on, 16- or 32-bit data assigned by source devices S1 and S2 are ORed, bit by bit. The result is set to destination device D1.



AND Truth Table

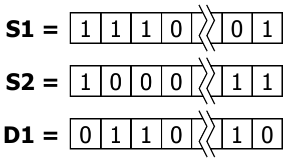
S1	S2	D1
0	0	0
0	1	1
1	0	1
1	1	1

XORW (Exclusive OR Word)



S1 ⊕ S2 → D1

When input is on, 16- or 32-bit data assigned by source devices S1 and S2 are exclusive ORed, bit by bit. The result is set to destination device D1.



AND Truth Table

S1	S2	D1
0	0	0
0	1	1
1	0	1
1	1	0

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Data for computation	X	X	X	X	X	X	X	—	X	1-99
S2 (Source 2)	Data for computation	X	X	X	X	X	X	X	—	X	1-99
D1 (Destination 1)	Destination to store results	—	X	▲	X	X	X	X	—	—	1-99

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be assigned as D1.

When T (timer) or C (counter) is used as S1 or S2, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP) which can be 0 through 65,535.

Since the Boolean computation instructions are executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

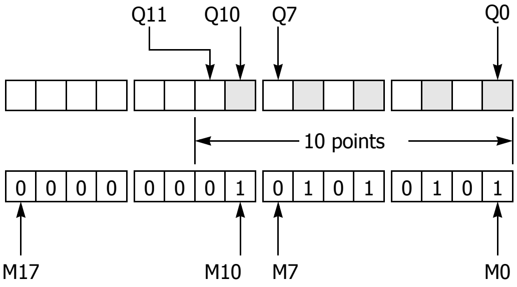
W (word)	X
I (integer)	—
D (double word)	X
L (long)	—
F (float)	—

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source or destination, 16 points (word data) or 32 points (double-word data) are used. When repeat is assigned for a bit device, the quantity of device bits increases in 16- or 32-point increments.

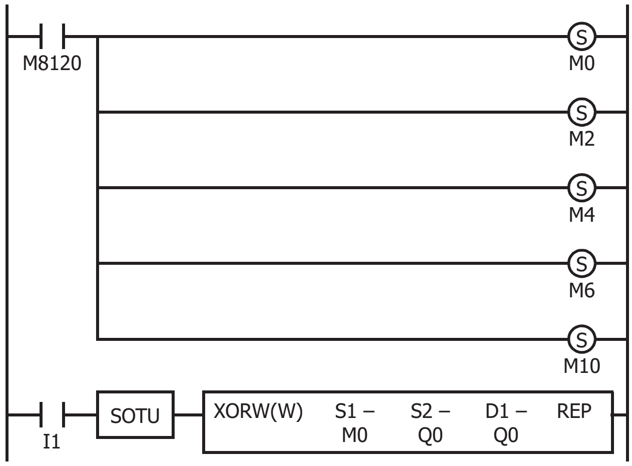
When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word data) or 2 points (double-word data) are used. When repeat is assigned for a word device, the quantity of device words increases in 1- or 2-point increments.

Example: XORW

To convert optional output status among a series of 10 output points, use the XORW instruction in combination with 10 internal relay points.



This program will invert the status of the shaded outputs on the left from on to off, and those not shaded from off to on.



Ten outputs Q0 through Q11 are assigned to 10 internal relays M0 through M11.

Five internal relays M0, M2, M4, M6, and M10 are set by initialize pulse special internal relay M8120.

When input I1 is turned on, the XORW instruction is executed to invert the status of outputs Q0, Q2, Q4, Q6, and Q10.

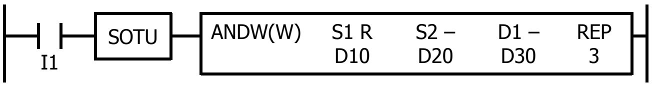
Repeat Operation in the ANDW, ORW, and XORW Instructions

Source devices S1 and S2 and destination device D1 can be assigned to repeat individually or in combination. When destination device D1 is not designated set to repeat, the final result is set to destination device D1. When repeat is assigned, consecutive devices as many as the repeat cycles starting with the designated device are used. Since the repeat operation works similarly on the ANDW (AND word), ORW (OR word), and XORW (exclusive OR word) instructions, the following examples use the ANDW instruction.

Repeat One Source Device

- **Data Type: Word**

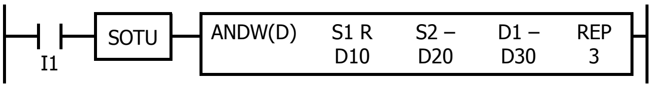
When only S1 (source) is assigned to repeat, the final result is set to destination device D1.



S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 0)
D10	·	D20	→	(D30)
D11	·	D20	→	(D30)
D12	·	D20	→	D30

- **Data Type: Double Word**

When only S1 (source) is assigned to repeat, the final result is set to destination device D1·D1+1.

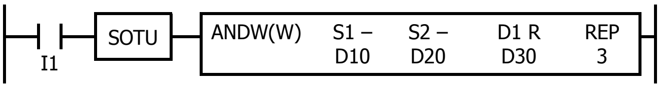


S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 0)
D10·D11	·	D20·D21	→	(D30·D31)
D12·D13	·	D20·D21	→	(D30·D31)
D14·D15	·	D20·D21	→	D30·D31

Repeat Destination Device Only

- **Data Type: Word**

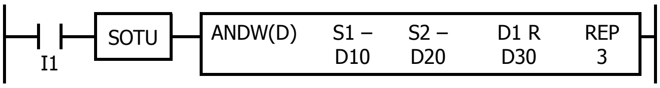
When only D1 (destination) is assigned to repeat, the same result is set to 3 devices starting with D1.



S1 (Repeat = 0)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10	·	D20	→	D30
D10	·	D20	→	D31
D10	·	D20	→	D32

- **Data Type: Double Word**

When only D1 (destination) is assigned to repeat, the same result is set to 3 devices starting with D1·D1+1.

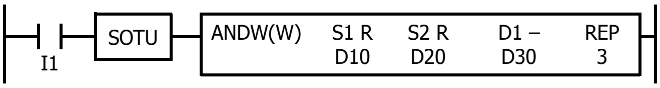


S1 (Repeat = 0)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10·D11	·	D20·D21	→	D30·D31
D10·D11	·	D20·D21	→	D32·D33
D10·D11	·	D20·D21	→	D34·D35

Repeat Two Source Devices

- **Data Type: Word**

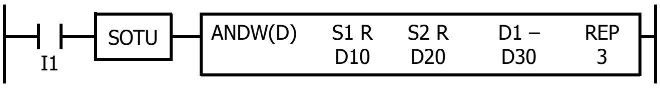
When S1 and S2 (source) are assigned to repeat, the final result is set to destination device D1.



S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 0)
D10	·	D20	→	(D30)
D11	·	D21	→	(D30)
D12	·	D22	→	D30

- **Data Type: Double Word**

When S1 and S2 (source) are assigned to repeat, the final result is set to destination device D1·D1+1.

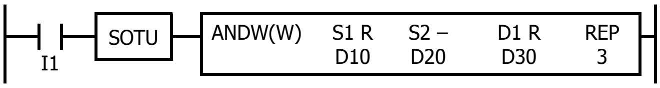


S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 0)
D10·D11	·	D20·D21	→	(D30·D31)
D12·D13	·	D22·D23	→	(D30·D31)
D14·D15	·	D24·D25	→	D30·D31

Repeat Source and Destination Devices

- **Data Type: Word**

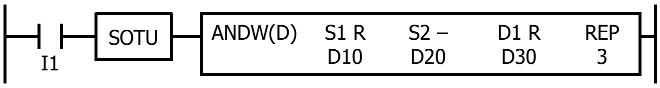
When S1 (source) and D1 (destination) are assigned to repeat, different results are set to 3 devices starting with D1.



S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10	·	D20	→	D30
D11	·	D20	→	D31
D12	·	D20	→	D32

- **Data Type: Double Word**

When S1 (source) and D1 (destination) are assigned to repeat, different results are set to 3 devices starting with D1·D1+1.

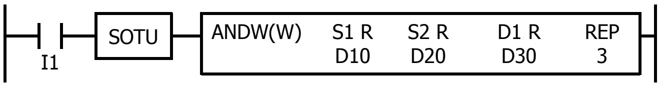


S1 (Repeat = 3)		S2 (Repeat = 0)		D1 (Repeat = 3)
D10·D11	·	D20·D21	→	D30·D31
D12·D13	·	D20·D21	→	D32·D33
D14·D15	·	D20·D21	→	D34·D35

Repeat All Source and Destination Devices

- **Data Type: Word**

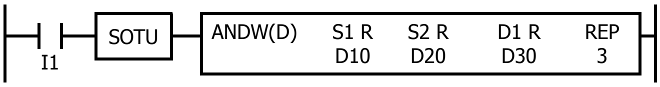
When all devices are assigned to repeat, different results are set to 3 devices starting with D1.



S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 3)
D10	·	D20	→	D30
D11	·	D21	→	D31
D12	·	D22	→	D32

- **Data Type: Double Word**

When all devices are assigned to repeat, different results are set to 3 devices starting with D1·D1+1.



S1 (Repeat = 3)		S2 (Repeat = 3)		D1 (Repeat = 3)
D10·D11	·	D20·D21	→	D30·D31
D12·D13	·	D22·D23	→	D32·D33
D14·D15	·	D24·D25	→	D34·D35

Note: When a user program error occurs in any repeat operation, special internal relay M8004 (user program execution error) and the ERROR LED are turned on and maintained while operations for other instructions continues. For an advanced instruction that has caused a user program execution error due to an error in the source devices, results are not set to any destination.

9: SHIFT / ROTATE INSTRUCTIONS

Introduction

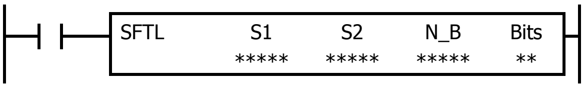
Bit shift instructions are used to shift the data string starting with source device S1 to the left or right by 1 to 15 bits as designated. The data string can be 1 to 65,535 bits. The result is set to the source device S1 and special internal relay M8003 (carry or borrow). The LSB or MSB is filled with 0 or 1 as designated.

Bit shift and rotate instructions are used to shift the 16- or 32-bit data string in the designated source device S1 to the left or right by the quantity of bits designated. The result is set to the source device S1 and special internal relay M8003 (carry or borrow).

The BCD left shift instruction shifts the BCD digits in two consecutive data registers to the left.

The word shift instruction is used to move 16-bit data to a destination data register and shifts down the data of subsequent data registers as many as designated.

SFTL (Shift Left)

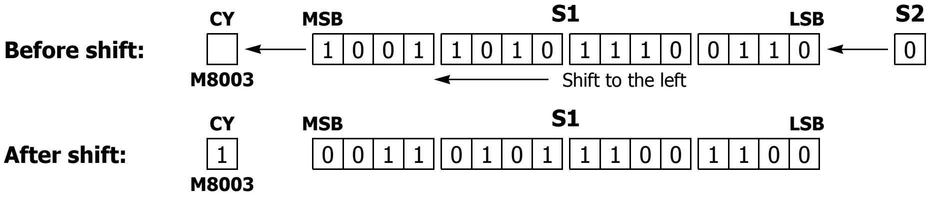


CY ← S1

When input is on, N_B-bit data string starting with source device S1 is shifted to the left by the quantity of bits assigned by device Bits.

The result is set to source device S1, and the last bit status shifted out is set to special internal relay M8003 (carry or borrow). Zero or 1 assigned by source device S2 is set to the LSB.

- **S2 = 0, N_B = 16, Bits = 1**



Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	First data for bit shift	—	X	▲	X	—	—	X	—	—	—
S2 (Source 2)	Data to shift into the LSB	X	X	X	X	—	—	—	—	0 or 1	—
N_B	Number of bits in the data string	—	—	—	—	—	—	X	—	1-65,535	—
Bits	Quantity of bits to shift	—	—	—	—	—	—	X	—	1-15	—

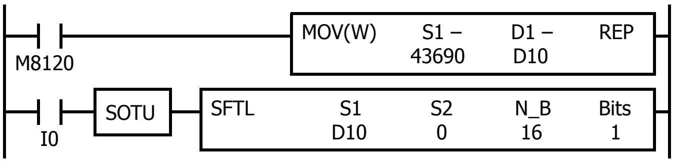
For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as S1.

Since the SFTL instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Examples: SFTL

- **N_B = 16 bits**



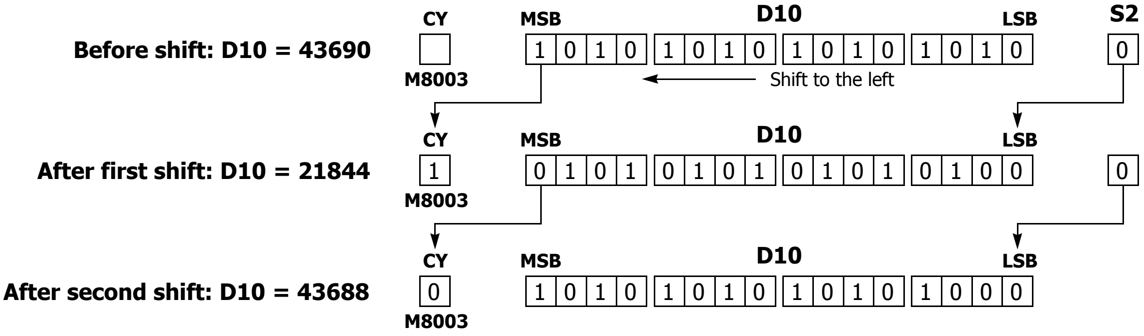
M8120
MOV(W) S1 – 43690 D1 – D10 REP
I0
SOTU
SFTL S1 D10 S2 0 N_B 16 Bits 1

M8120 is the initialize pulse special internal relay.

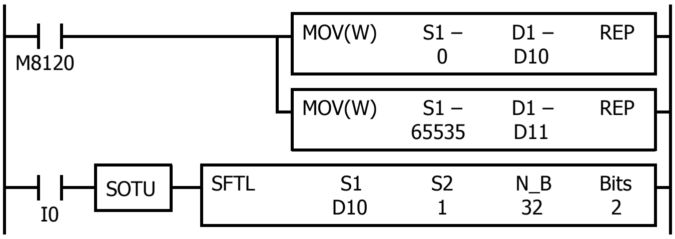
When the CPU starts operation, the MOV (move) instruction sets 43,690 to data register D10.

Each time input I0 is turned on, 16-bit data of data register D10 is shifted to the left by 1 bit as assigned by device Bits. The last bit status shifted out is set to special internal relay M8003 (carry or borrow). Zeros are set to the LSB.

Bits to shift = 1



- **N_B = 32 bits**



M8120
MOV(W) S1 – 0 D1 – D10 REP
MOV(W) S1 – 65535 D1 – D11 REP
I0
SOTU
SFTL S1 D10 S2 1 N_B 32 Bits 2

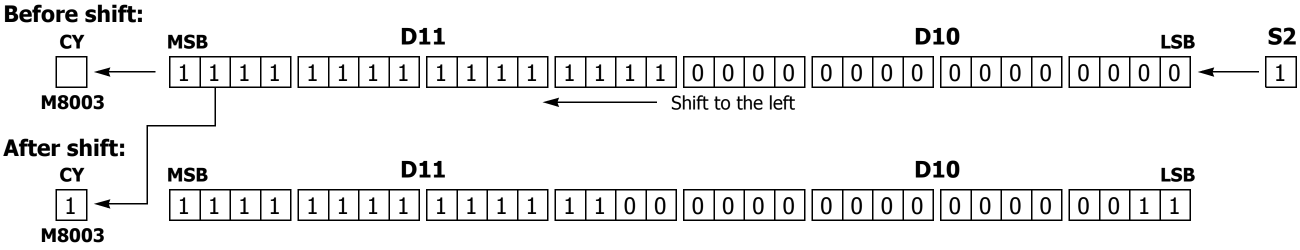
M8120 is the initialize pulse special internal relay.

When the CPU starts operation, the MOV (move) instructions set 0 and 65,535 to data registers D10 and D11, respectively.

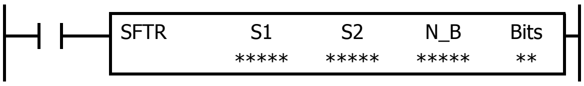
Each time input I0 is turned on, 32-bit data of data registers D10 and D11 is shifted to the left by 2 bits as assigned by device Bits. D10 is the low word, and D11 is the high word.

The last bit status shifted out is set to a carry (special internal relay M8003). Ones are set to the LSBs.

Bits to shift = 2



SFTR (Shift Right)

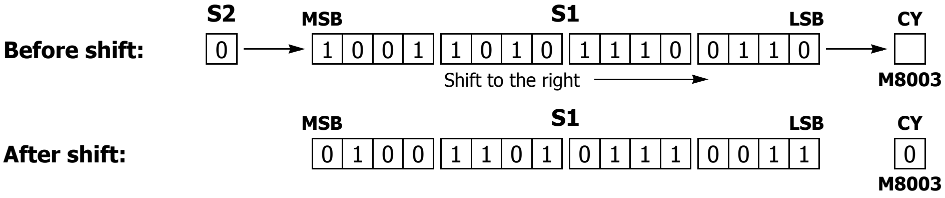


S1 → CY

When input is on, N_B-bit data string starting with source device S1 is shifted to the right by the quantity of bits assigned by device Bits.

The result is set to source device S1, and the last bit status shifted out is set to a carry (special internal relay M8003). Zero or 1 assigned by source device S2 is set to the MSB.

- **S2 = 0, N_B = 16, Bits = 1**



Valid Devices

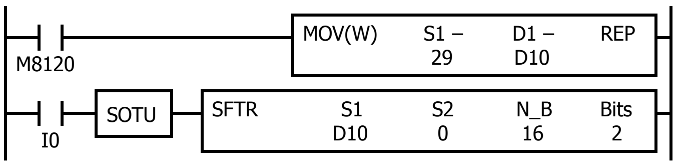
Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	First data for bit shift	—	X	▲	X	—	—	X	—	—	—
S2 (Source 2)	Data to shift into the MSB	X	X	X	X	—	—	—	—	0 or 1	—
N_B	Number of bits in the data string	—	—	—	—	—	—	X	—	1-65,535	—
Bits	Quantity of bits to shift	—	—	—	—	—	—	X	—	1-15	—

For valid device address ranges, see "Device Addresses" on page 2-1.

Since the SFTR instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Example: SFTR

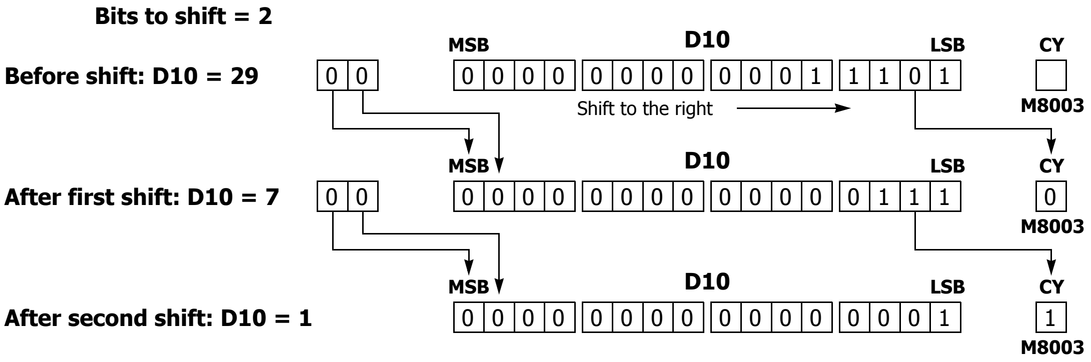
- **Data Type: Word**



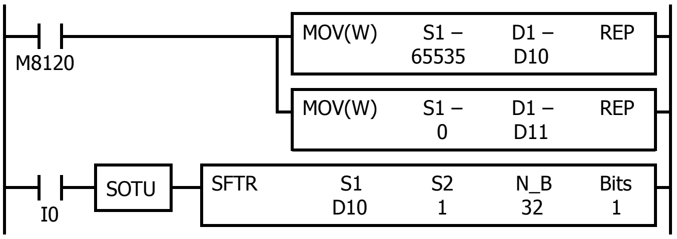
M8120 is the initialize pulse special internal relay.

When the CPU starts operation, the MOV (move) instruction sets 29 to data register D10.

Each time input I0 is turned on, 16-bit data of data register D10 is shifted to the right by 2 bits as assigned by device Bits. The last bit status shifted out is set to special internal relay M8003 (carry or borrow). Zeros are set to the MSB.



- **Data Type: Double Word**

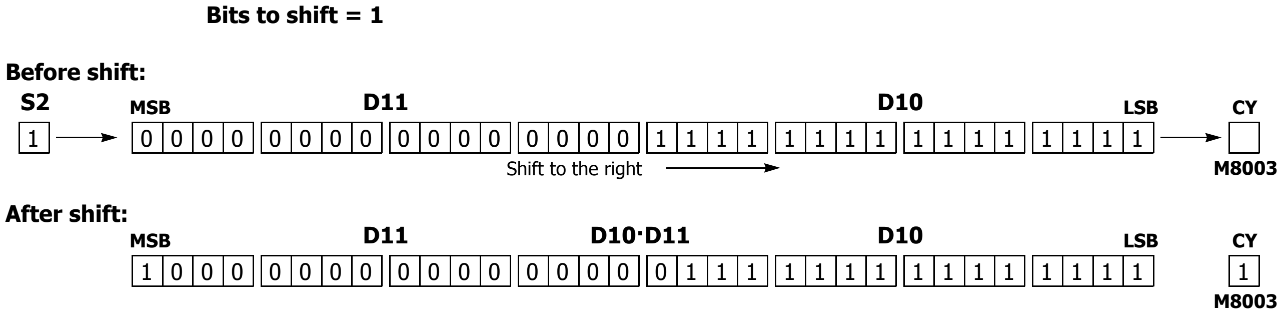


M8120 is the initialize pulse special internal relay.

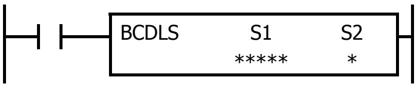
When the CPU starts operation, the MOV (move) instructions set 65,535 and 0 to data registers D10 and D11, respectively.

Each time input I0 is turned on, 32-bit data of data registers D10 and D11 is shifted to the right by 1 bit as assigned by device Bit. D10 is the low word, and D11 is the high word.

The last bit status shifted out is set to special internal relay M8003 (carry or borrow). Ones are set to the MSB.



BCDLS (BCD Left Shift)



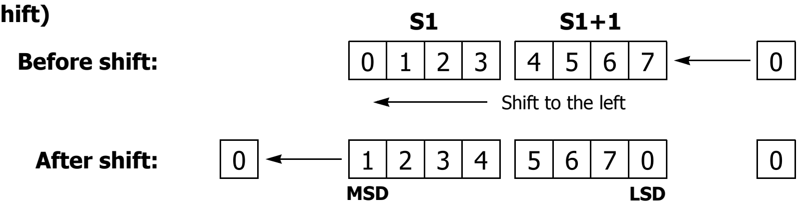
When input is on, the 32-bit binary data assigned by S1 is converted into 8 BCD digits, shifted to the left by the quantity of digits assigned by S2, and converted back to 32-bit binary data.

Valid values for each of S1 and S1+1 are 0 through 9,999.

The number of digits that can be shifted is 1 through 7.

Zero is set to the lowest digit after each shift.

When S2 = 1 (digits to shift)



Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Data for BCD shift	—	—	—	—	—	—	X	—	—	—
S2 (Source 2)	Quantity of digits to shift	X	X	X	X	X	X	X	—	1-7	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When T (timer) or C (counter) is used as S2, the timer/counter current value (TC or CC) is displayed.

The number of digits that can be shifted as S2 are 1 through 7.

Make sure that the source data determined by S1 and S1+1 is between 0 and 9,999 for each data register. If either source data is over 9,999, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. When S2 is higher than 7, a user program execution error will also result.

When a user program execution error occurs, the execution of the instruction is canceled. The data in S1 and S1+1 is left unchanged and the next instruction is executed. For user program execution errors, see "User Program Execution Errors" on page 3-10.

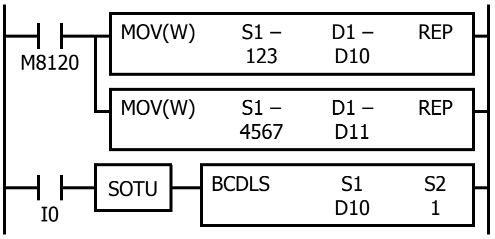
Valid Data Types

When a word device such as D (data register) is assigned as source S1, 2 points (double-word data) are used.

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as source S2, 16 points are used.

When a word device such as T (timer), C (counter), or D (data register) is assigned as source S2, 1 point is used.

Example: BCDLS



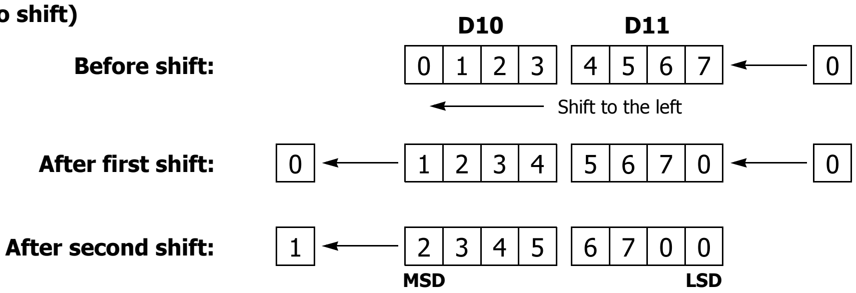
M8120 is the initialize pulse special internal relay.

When the CPU starts operation, the MOV (move) instructions set 123 and 4,567 to data registers D10 and D11, respectively.

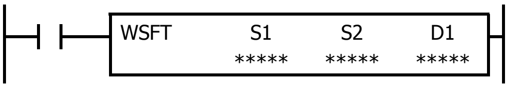
Each time input I0 is turned on, the 32-bit binary data of data registers D10 and D11 assigned by S1 is converted into 8 BCD digits, shifted to the left by 1 digit as assigned by device S2, and converted back to 32-bit binary data.

Zero is set to the lowest digit after each shift.

When S2 = 1 (digits to shift)

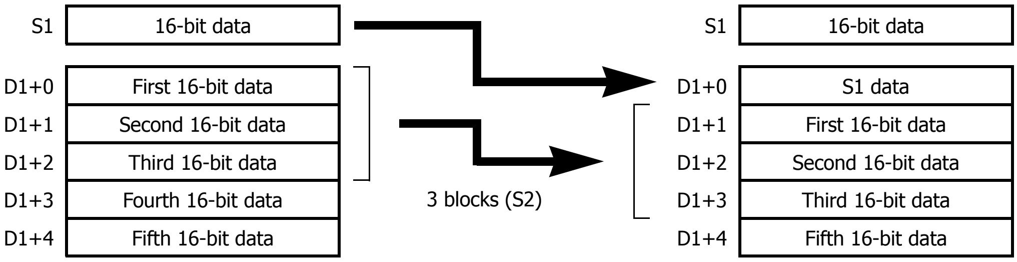


WSFT (Word Shift)



When input is on, N blocks of 16-bit word data, starting with device assigned by D1, are shifted up to the next 16-bit positions. At the same time, the data assigned by device S1 is moved to the device assigned by D1. S2 specifies the quantity of blocks to move.

When S2 = 3 (quantity of blocks to shift)



Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Source data for word shift	X	X	X	X	X	X	X	—	X	—
S2 (Source 2)	Quantity of blocks to shift	X	X	X	X	X	X	X	—	X	—
D1 (Destination 1)	First device address to shift	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When T (timer) or C (counter) is used as S1 or S2, the timer/counter current value (TC or CC) is displayed.

Valid Data Types

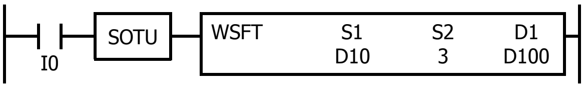
When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as source S1 or S2, 16 points are used.

When a word device such as T (timer), C (counter), or D (data register) is assigned as source S1 or S2, 1 point is used.

Special Internal Relay M8024: BMOV/WSFT Executing Flag

While the BMOV or WSFT is executed, M8024 turns on. When completed, M8024 turns off. If the CPU is powered down while executing BMOV or WSFT, M8024 remains on when the CPU is powered up again.

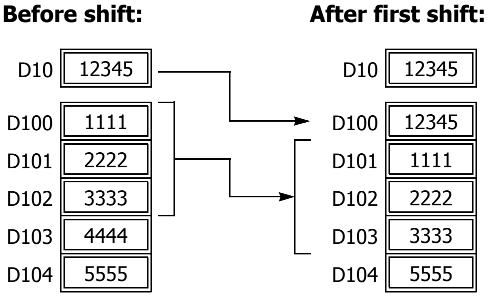
Example: WSFT



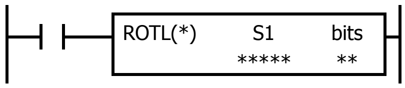
D100 through D102 → D101 through D103

D10 → D100

When input I0 is turned on, data of 3 data registers, starting with D100 assigned by destination device D1, is shifted to the next data registers. Data of data register D10, assigned by source device S1, is moved to D100 assigned by destination device D1.



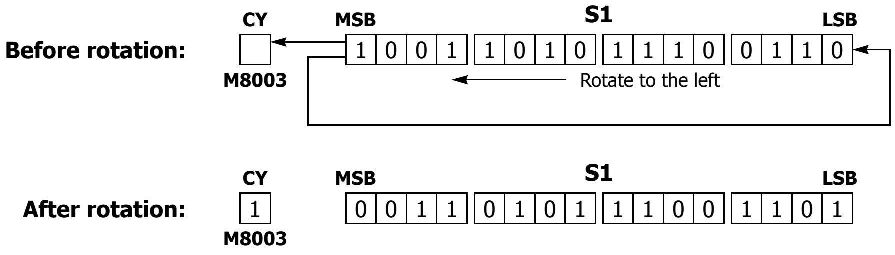
ROTL (Rotate Left)



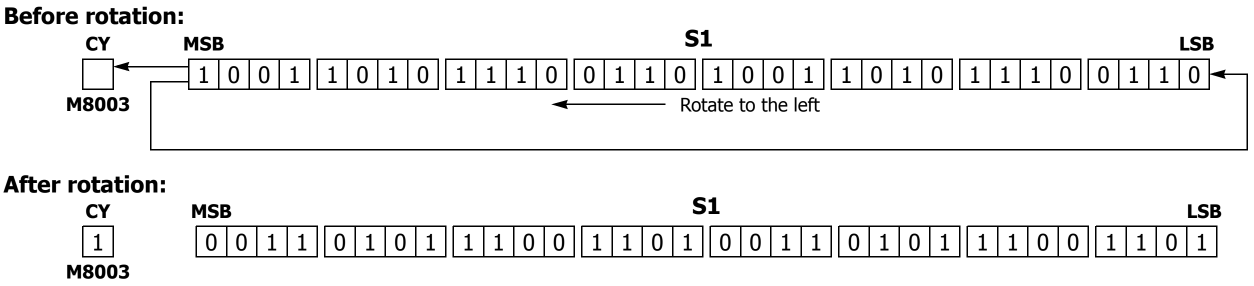
When input is on, 16- or 32-bit data of the assigned source device S1 is rotated to the left by the quantity of bits assigned by device bits.

The result is set to the source device S1, and the last bit status rotated out is set to a carry (special internal relay M8003).

- **Data Type: Word (bits to rotate = 1)**



- **Data Type: Double Word (bits to rotate = 1)**



Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Data for bit rotation	—	X	▲	X	—	—	X	—	—	—
bits	Quantity of bits to rotate	—	—	—	—	—	—	—	—	1-15, 1-31	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as S1.

The quantity of bits to rotate can be 1 through 15 for word data, or 1 through 31 for double-word data.

Since the ROTL instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

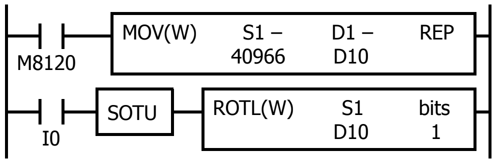
W (word)	X
I (integer)	—
D (double word)	X
L (long)	—
F (float)	—

When a bit device such as Q (output), M (internal relay), or R (shift register) is assigned as the source, 16 points (word data) or 32 points (double-word data) are used.

When a word device such as D (data register) is assigned as the source, 1 point (word data) or 2 points (double-word data) are used.

Example: ROTL

- **Data Type: Word**



M8120 is the initialize pulse special internal relay.

When the CPU starts operation, the MOV (move) instruction sets 40,966 to data register D10.

Each time input I0 is turned on, 16-bit data of data register D10 is rotated to the left by 1 bit as assigned by device bits.

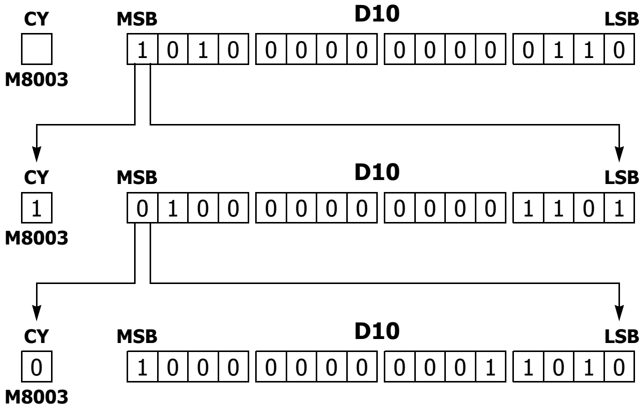
The status of the MSB is set to special internal relay M8003 (carry or borrow).

Bits to rotate = 1

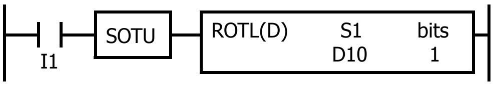
Before rotation: D10 = 40966

After first rotation: D10 = 16397

After second rotation: D10 = 32794



- **Data Type: Double Word**

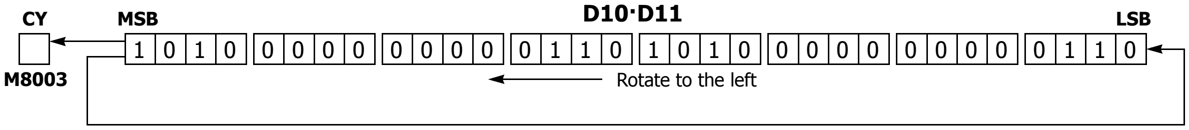


Each time input I1 is turned on, 32-bit data of data registers D10 and D11 is rotated to the left by 1 bit as assigned by device bits.

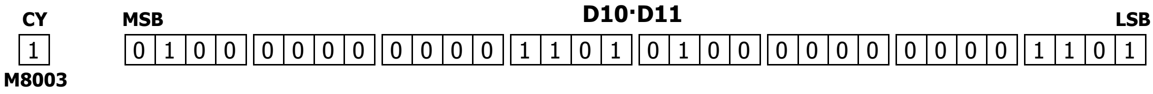
The status of the MSB is set to special internal relay M8003 (carry or borrow).

Bits to rotate = 1

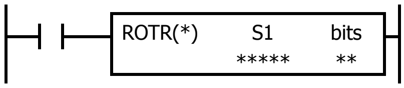
Before rotation: D10·D11 = 2,684,788,742



After rotation: D10·D11 = 1,074,610,189



ROTR (Rotate Right)

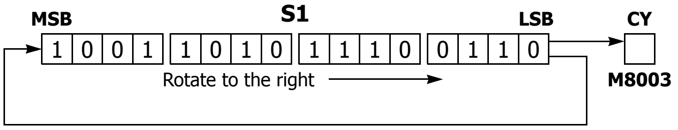


When input is on, 16- or 32-bit data of the assigned source device S1 is rotated to the right by the quantity of bits assigned by device bits.

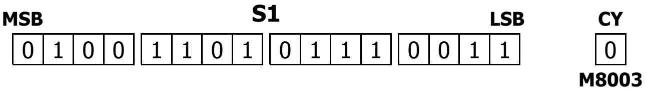
The result is set to the source device S1, and the last bit status rotated out is set to special internal relay M8003 (carry or borrow).

- **Data Type: Word (bits to rotate = 1)**

Before rotation:

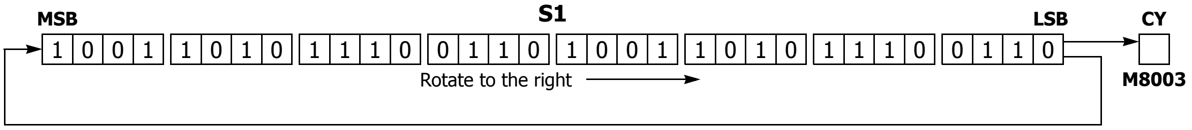


After rotation:

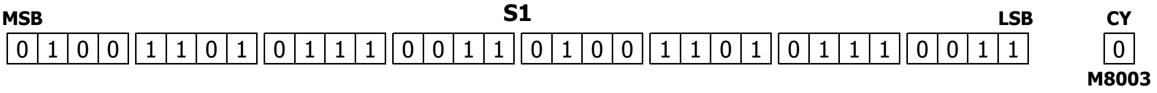


- **Data Type: Double Word (bits to rotate = 1)**

Before rotation:



After rotation:



Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Data for bit rotation	—	X	▲	X	—	—	X	—	—	—
bits	Quantity of bits to rotate	—	—	—	—	—	—	—	—	1-15, 1-31	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as S1.

The number of bits that can be rotated is 1 through 15 for word data, or 1 through 31 for double-word data.

Since the ROTR instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

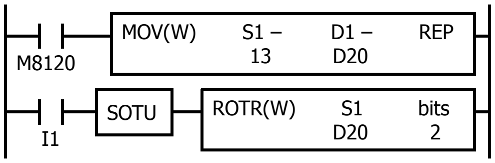
W (word)	X
I (integer)	—
D (double word)	X
L (long)	—
F (float)	—

When a bit device such as Q (output), M (internal relay), or R (shift register) is assigned as the source, 16 points (word data) or 32 points (double-word data) are used.

When a word device such as D (data register) is assigned as the source, 1 point (word data) or 2 points (double-word data) are used.

Example: ROTR

- **Data Type: Word**



M8120 is the initialize pulse special internal relay.

When the CPU starts operation, the MOV (move) instruction sets 13 to data register D20.

Each time input I1 is turned on, 16-bit data of data register D20 is rotated to the right by 2 bits as assigned by device bits.

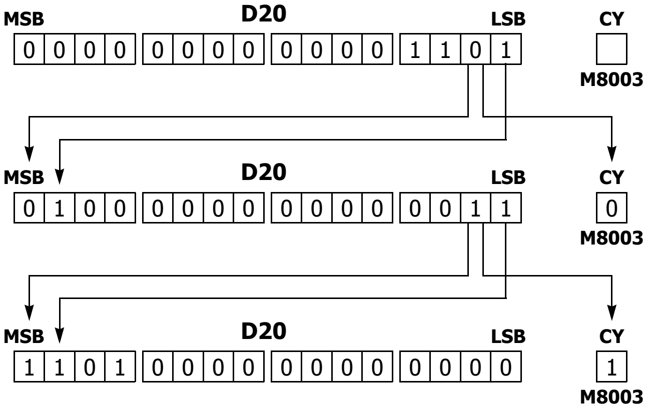
The last bit status rotated out is set to special internal relay M8003 (carry or borrow).

Bits to rotate = 2

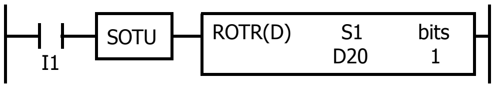
Before rotation: D20 = 13

After first rotation: D20 = 16387

After second rotation: D20 = 53248



- **Data Type: Double Word**

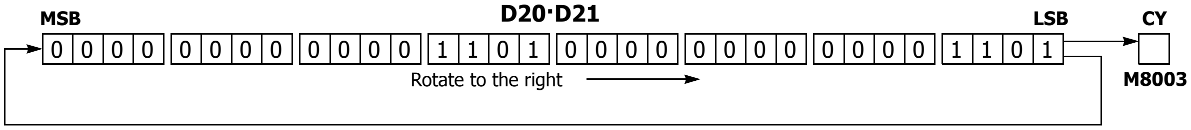


Each time input I1 is turned on, 32-bit data of data registers D20 and D21 is rotated to the right by 1 bit as assigned by device bits.

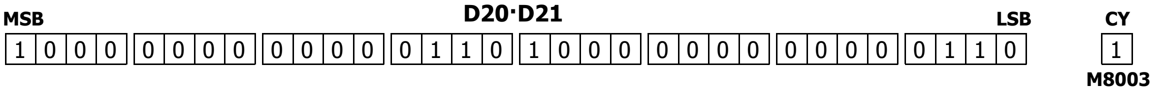
The last bit status rotated out is set to special internal relay M8003 (carry or borrow).

Bits to rotate = 1

Before rotation: D20·D21 = 851,981



After rotation: D20·D21 = 2,147,909,638



10: Data Conversion Instructions

Introduction

This chapter describes the data conversion instructions that convert data to the specified format.

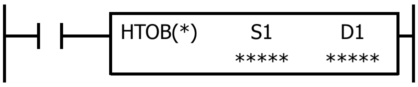
The ENCO (encode), DECO (decode), and BCNT (bit count) instructions processes bit device data.

The ALT (alternate output) instruction turns an output on and off each time an input button is pressed.

The CVDT (convert data) instruction converts data types among W (word), I (integer), D (double word), L (long), and F (float).

The DTDV and DTCB instructions convert data between two one-byte data and one word data. The SWAP exchanges upper and lower byte- or word-data for word- or double-word-data respectively.

HTOB (Hex to BCD)



S1 → D1

When input is on, the 16- or 32-bit data assigned by S1 is converted into BCD and stored to the destination assigned by device D1.

Valid values for the source device are 0 through 9,999 for word data, and 0 through 99,999,999 for double-word data.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Binary data to convert	X	X	X	X	X	X	X	—	X	—
D1 (Destination 1)	Destination to store conversion results	—	X	▲	X	X	X	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S1, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP) which can be 0 through 65,535.

When the data type is W (word), the valid range of S1 (binary data) is 0 to 9,999.

When the data type is D (double word), the valid range of S1 (binary data) is 0 to 99,999,999.

Make sure that the source designated by S1 is within the valid value range. If the source data exceeds the valid range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Since the HTOB instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

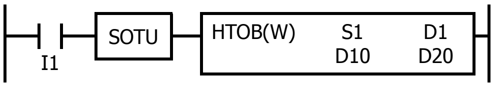
W (word)	X
I (integer)	—
D (double word)	X
L (long)	—
F (float)	—

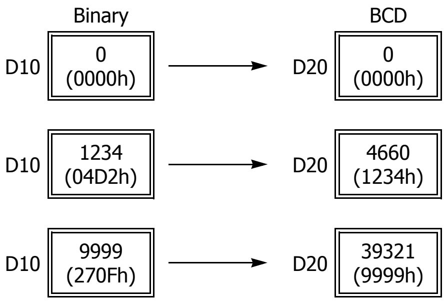
When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source, 16 points (word data) or 32 points (double-word data) are used.

When a word device such as T (timer), C (counter), or D (data register) is assigned as the source, 1 point (word data) or 2 points (double-word data) are used.

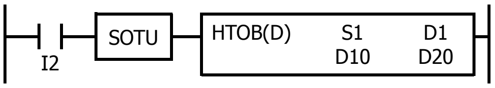
Examples: HTOB

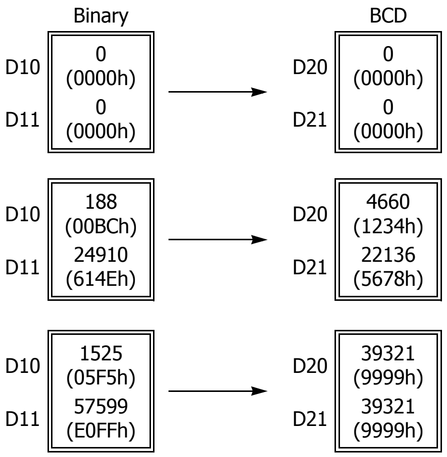
- **Data Type: Word**



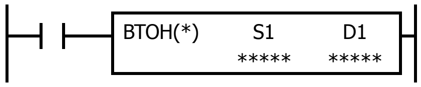


- **Data Type: Double Word**





BTOH (BCD to Hex)



S1 → D1

When input is on, the BCD data assigned by S1 is converted into 16- or 32-bit binary data and stored to the destination assigned by device D1.

Valid values for the source device are 0 through 9,999 (BCD) for word data, and 0 through 99,999,999 (BCD) for double-word data.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	BCD data to convert	X	X	X	X	X	X	X	—	X	—
D1 (Destination 1)	Destination to store conversion results	—	X	▲	X	X	X	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S1, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP) which can be 0 through 65,535.

When the data type is W (word), the valid range of S1 (BCD data) is 0 to 9,999.

When the data type is D (double word), the valid range of S1 (BCD data) is 0 to 99,999,999.

Make sure that each digit of the source designated by S1 is 0 through 9. If the source data exceeds the valid range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Since the BTOH instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

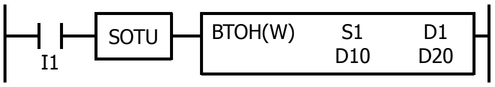
W (word)	X
I (integer)	—
D (double word)	X
L (long)	—
F (float)	—

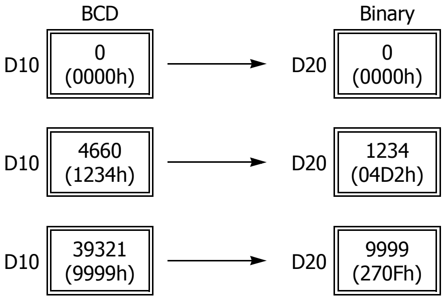
When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source, 16 points (word data) or 32 points (double-word data) are used.

When a word device such as T (timer), C (counter), or D (data register) is assigned as the source, 1 point (word data) or 2 points (double-word data) are used.

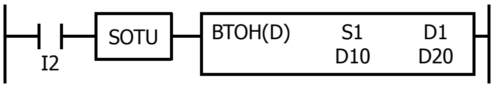
Examples: BTOH

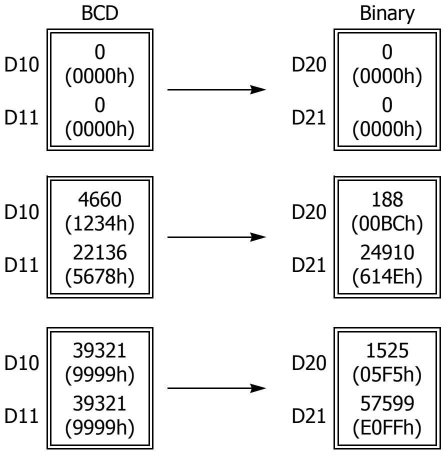
- **Data Type: Word**



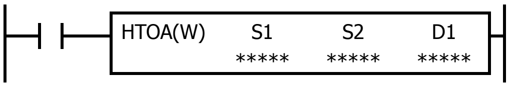


- **Data Type: Double Word**





HTOA (Hex to ASCII)



S1 → D1, D1+1, D1+2, D1+3

When input is on, the 16-bit binary data assigned by S1 is read from the lowest digit as many times as the quantity of digits assigned by S2, converted into ASCII data, and stored to the destination starting with the device assigned by D1.

The quantity of digits to convert can be 1 through 4.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Binary data to convert	X	X	X	X	X	X	X	—	X	—
S2 (Source 2)	Quantity of digits to convert	X	X	X	X	X	X	X	—	1-4	—
D1 (Destination 1)	Destination to store conversion results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When T (timer) or C (counter) is used as S1 or S2, the timer/counter current value (TC or CC) is displayed.

Make sure that the quantity of digits designated by S2 is within the valid range. If the S2 data exceeds the valid range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Since the HTOA instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

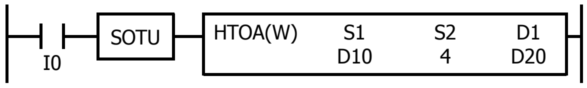
W (word)	X
I (integer)	—
D (double word)	—
L (long)	—
F (float)	—

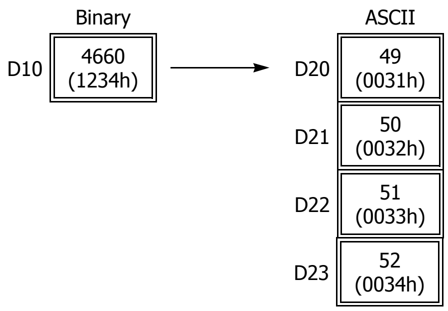
When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source, 16 points (word data) are used.

When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word data) is used.

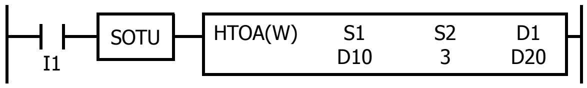
Examples: HTOA

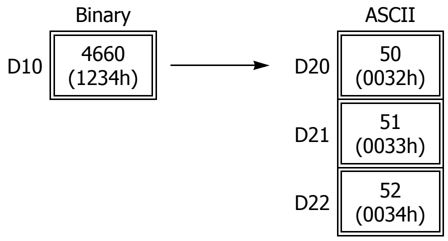
- **Quantity of Digits: 4**



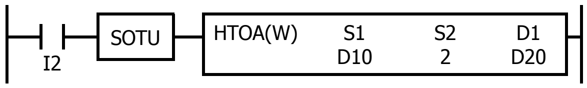


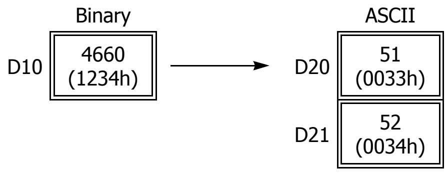
- **Quantity of Digits: 3**



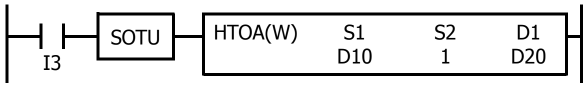


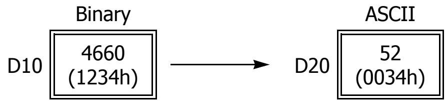
- **Quantity of Digits: 2**



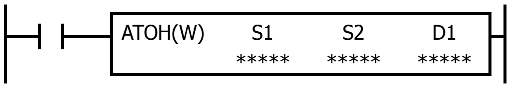


- **Quantity of Digits: 1**





ATOH (ASCII to Hex)



S1, S1+1, S1+2, S1+3 → D1

When input is on, the ASCII data, assigned by S1, is converted into 16-bit binary data. The number of times is determined by the quantity of digits assigned by S2. The data is then stored to the destination assigned by device D1.

Valid values for source data to convert are 30h to 39h and 41h to 46h.

The quantity of digits to convert can be 1 through 4.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	ASCII data to convert	—	—	—	—	—	—	X	—	—	—
S2 (Source 2)	Quantity of digits to convert	X	X	X	X	X	X	X	—	1-4	—
D1 (Destination 1)	Destination to store conversion results	—	X	▲	X	X	X	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S2, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP) which can be 0 through 65,535.

Valid values for source S1 data to convert are 30h to 39h and 41h to 46h. Make sure that the values for each source designated by S1 and the quantity of digits designated by S2 are within the valid range. If the S1 or S2 data exceeds the valid range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Since the ATOH instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

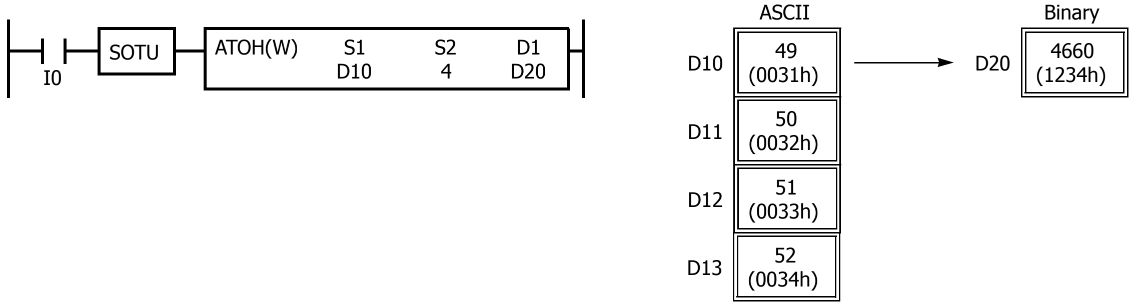
W (word)	X
I (integer)	—
D (double word)	—
L (long)	—
F (float)	—

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source or destination, 16 points (word data) are used.

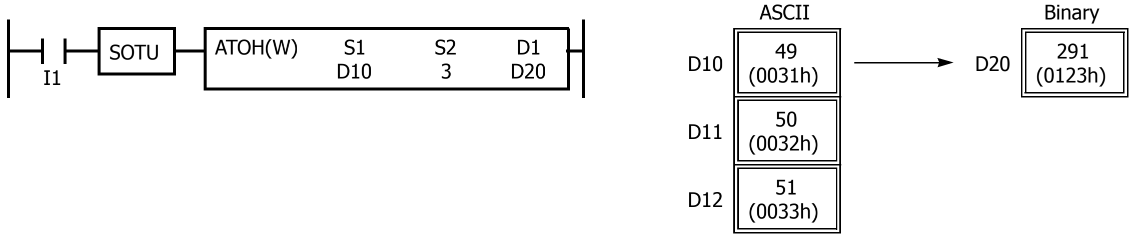
When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word data) is used.

Examples: ATOH

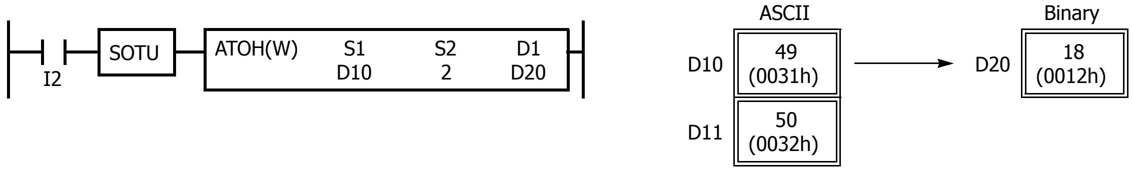
- **Quantity of Digits: 4**



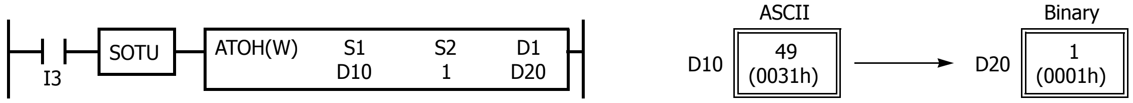
- **Quantity of Digits: 3**



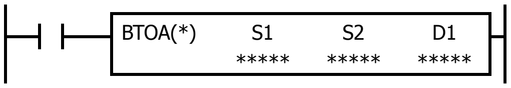
- **Quantity of Digits: 2**



- **Quantity of Digits: 1**



BTOA (BCD to ASCII)



Word data: S1 → D1, D1+1, D1+2, D1+3, D1+4
Double-word data: S1·S1+1 → D1, D1+1, D1+2, ... , D1+9

When input is on, the 16- or 32-bit binary data assigned by S1 is converted into BCD, and converted into ASCII data. The data is read from the lowest digit as many times as the quantity of digits assigned by S2. The result is stored to the destination starting with the device assigned by D1.

The number of digits that can be converted is 1 through 5 for word data, and 1 through 10 for double-word data.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Binary data to convert	X	X	X	X	X	X	X	—	X	—
S2 (Source 2)	Quantity of digits to convert	X	X	X	X	X	X	X	—	1-5, 1-10	—
D1 (Destination 1)	Destination to store conversion results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When T (timer) or C (counter) is used as S1 or S2, the timer/counter current value (TC or CC) is displayed.

When the data type is W (word), the valid range of S2 (quantity of digits to convert) is 1 to 5.

When the data type is D (double word), the valid range of S2 (quantity of digits to convert) is 1 to 10.

Make sure that the quantity of digits designated by S2 is within the valid range. If the S2 data exceeds the valid range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

Make sure that the last destination data determined by D1+S2-1 is within the valid device range. If the derived destination device exceeds the valid device range, a user program execution error will result, turning on special internal relay M8004 and ERROR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Since the BTOA instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

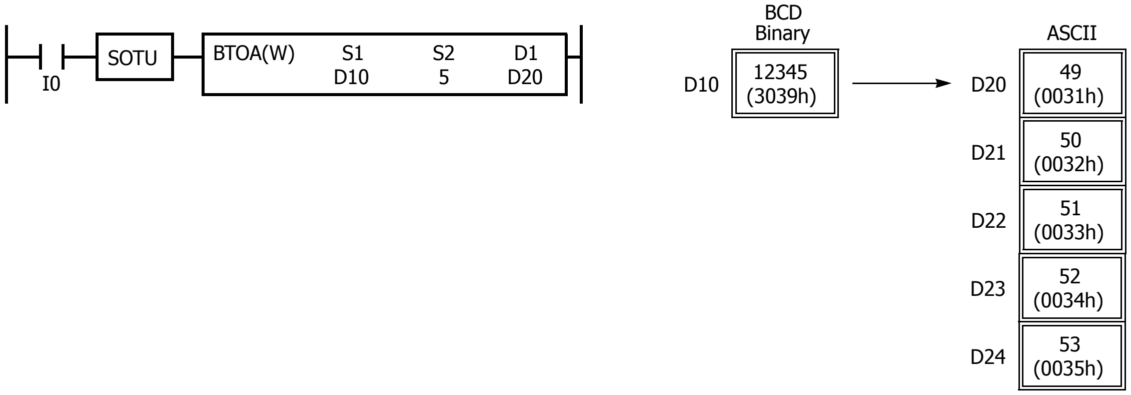
W (word)	X
I (integer)	—
D (double word)	X
L (long)	—
F (float)	—

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source, 16 points (word data) or 32 points (double-word data) are used.

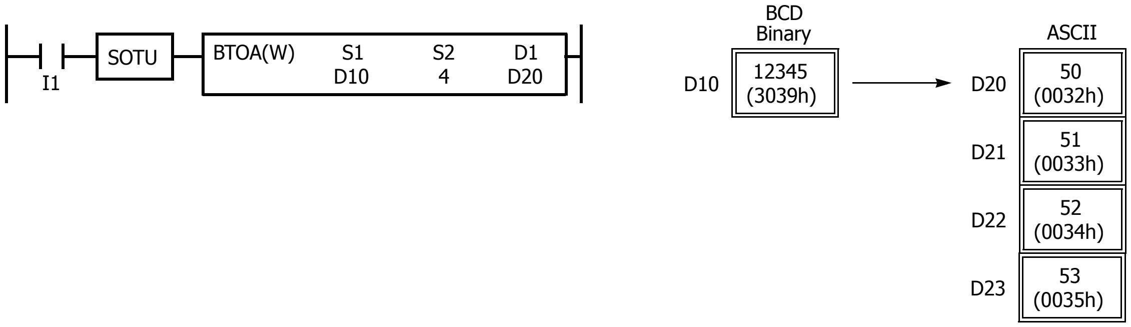
When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word data) or 2 points (double-word data) are used.

Examples: BTOA(W)

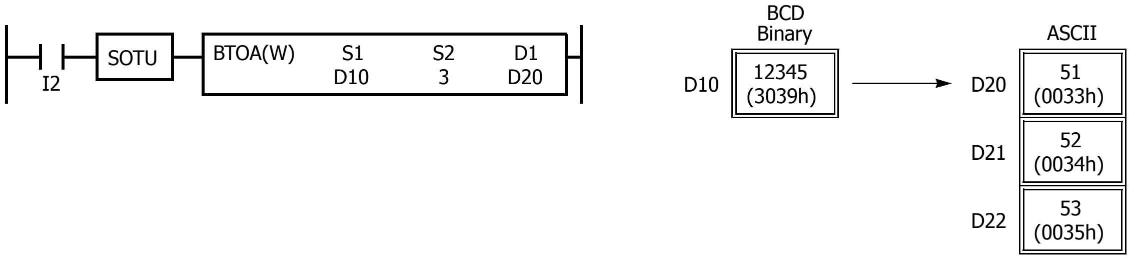
- **Quantity of Digits: 5**



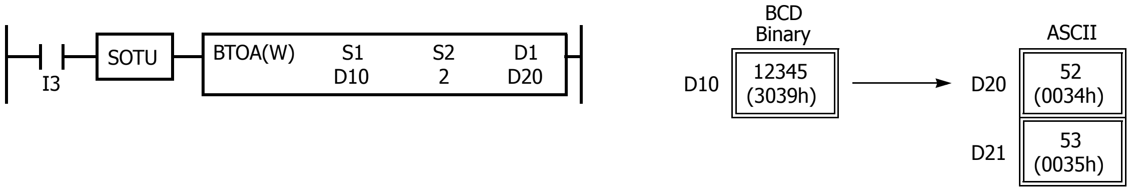
- **Quantity of Digits: 4**



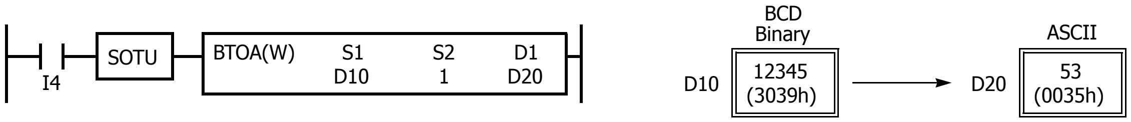
- **Quantity of Digits: 3**



- **Quantity of Digits: 2**

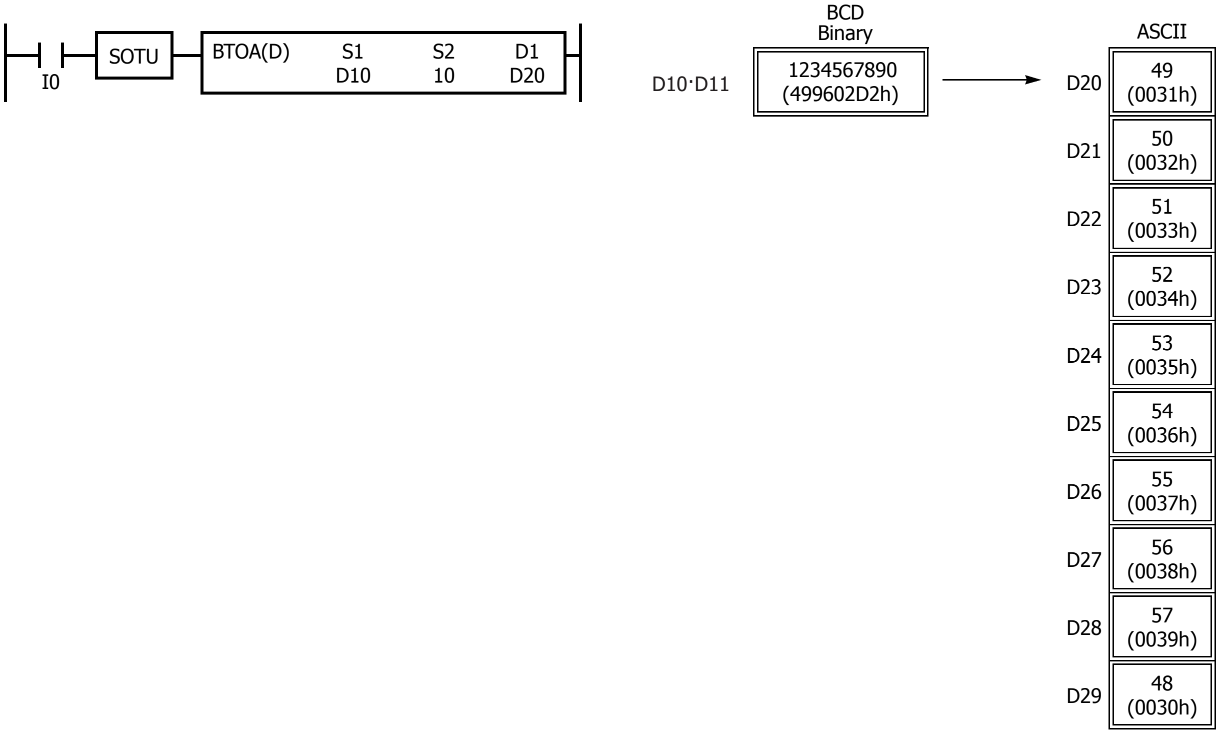


- **Quantity of Digits: 1**

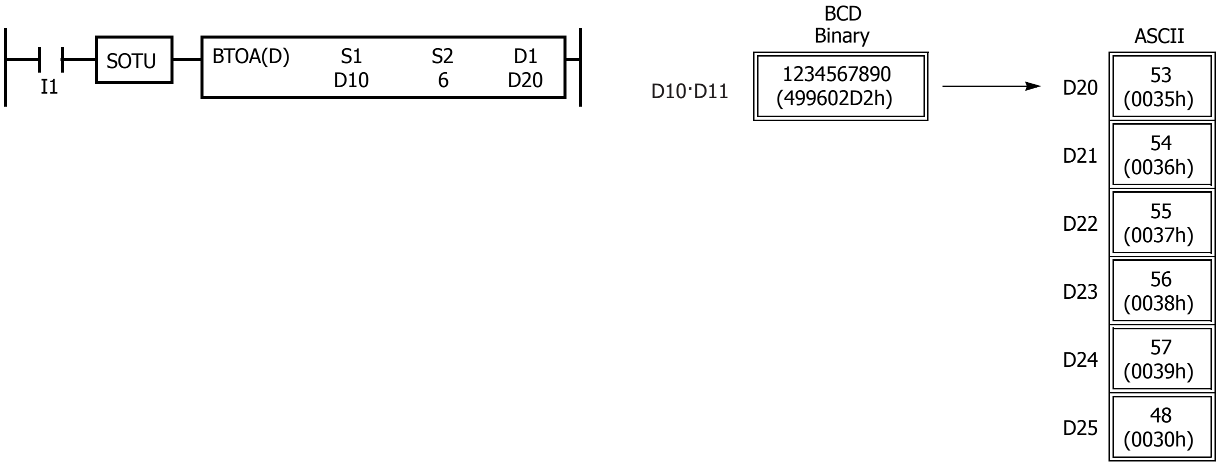


Examples: BTOA(D)

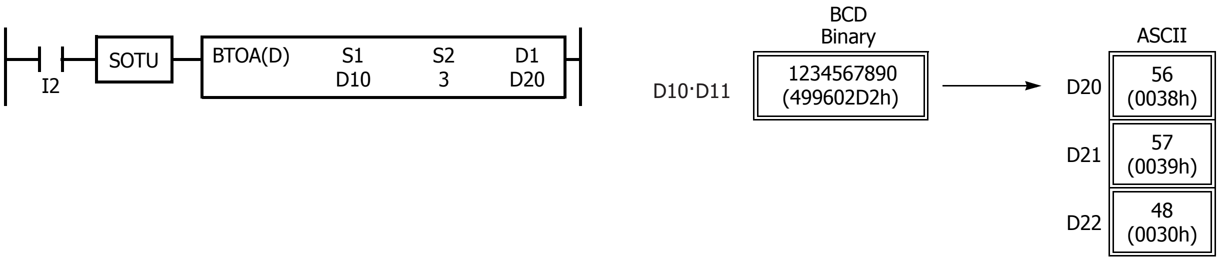
- **Quantity of Digits: 10**



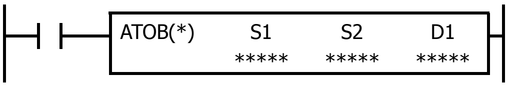
- **Quantity of Digits: 6**



- **Quantity of Digits: 3**



ATOB (ASCII to BCD)



Word data:	S1, S1+1, S1+2, S1+3, S1+4 → D1
Double-word data:	S1, S1+1, S1+2, ... , S1+9 → D1·D1+1

When input is on, the ASCII data assigned by S1 as many times as the quantity of digits assigned by S2 is converted into BCD, and converted into 16- or 32-bit binary data. The result is stored to the destination assigned by device D1.

Valid values for source data to convert are 30h through 39h.

The number of digits that can be converted is through 5 for word data, and 1 through 10 for double-word data.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	ASCII data to convert	—	—	—	—	—	—	X	—	—	—
S2 (Source 2)	Quantity of digits to convert	X	X	X	X	X	X	X	—	1-5, 1-10	—
D1 (Destination 1)	Destination to store conversion results	—	X	▲	X	X	X	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S2, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP).

Valid values for source S1 data to convert are 30h through 39h. When the data type is W (word), the valid range of S2 (quantity of digits to convert) is 1 to 5.

When the data type is D (double word), the valid range of S2 (quantity of digits to convert) is 1 to 10.

Make sure that the values for each source designated by S1 and the quantity of digits designated by S2 are within the valid range. If the S1 or S2 data exceeds the valid range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

Make sure that the last source data determined by S1+S2-1 is within the valid device range. If the derived source device exceeds the valid device range, a user program execution error will result, turning on special internal relay M8004 and ERROR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Since the ATOB instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

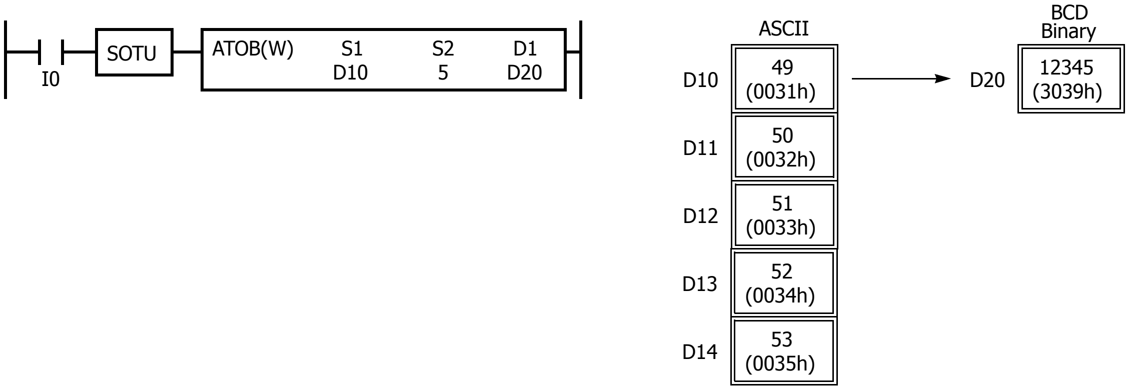
W (word)	X
I (integer)	—
D (double word)	X
L (long)	—
F (float)	—

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source or destination, 16 points (word data) or 32 points (double-word data) are used.

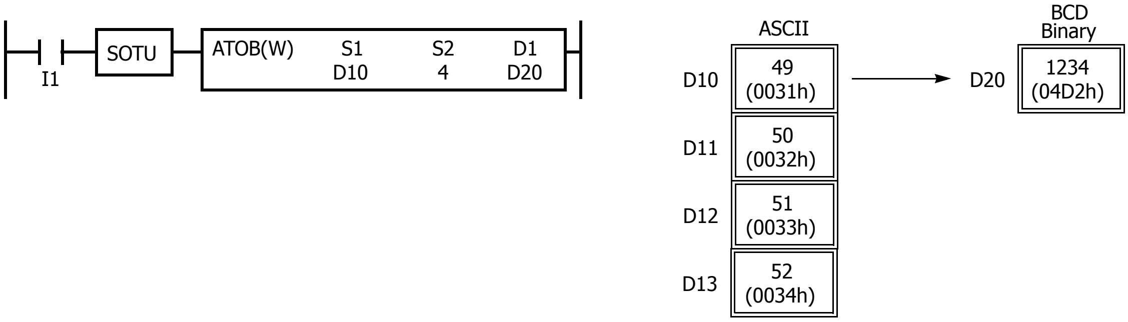
When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word data) or 2 points (double-word data) are used.

Examples: ATOB(W)

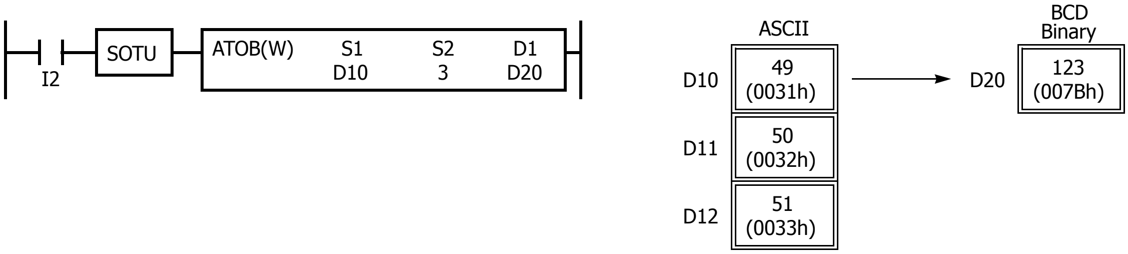
- **Quantity of Digits: 5**



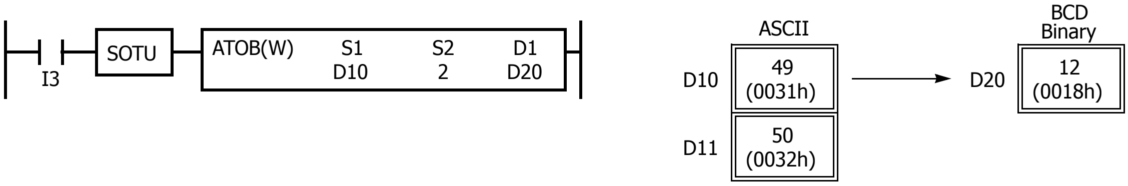
- **Quantity of Digits: 4**



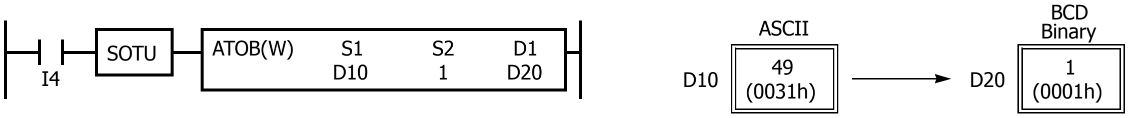
- **Quantity of Digits: 3**



- **Quantity of Digits: 2**

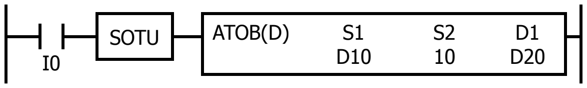


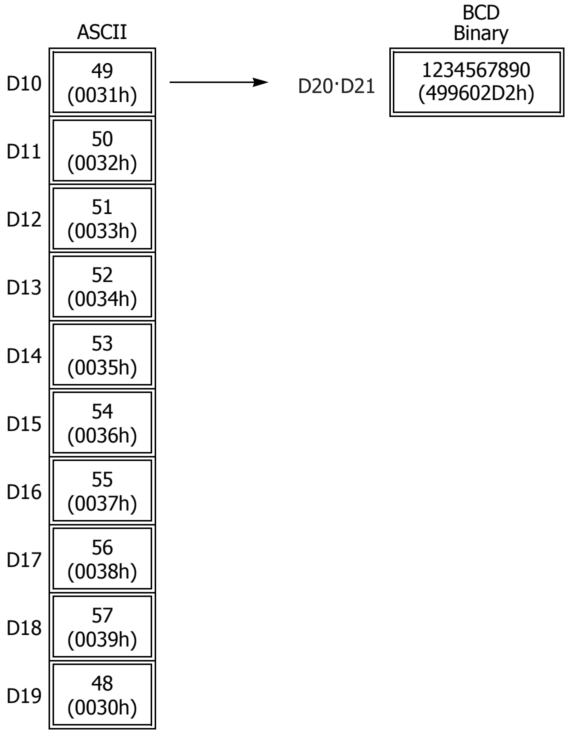
- **Quantity of Digits: 1**



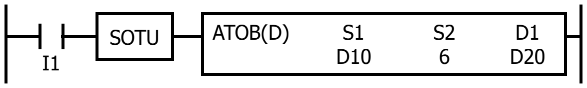
Examples: ATOB(D)

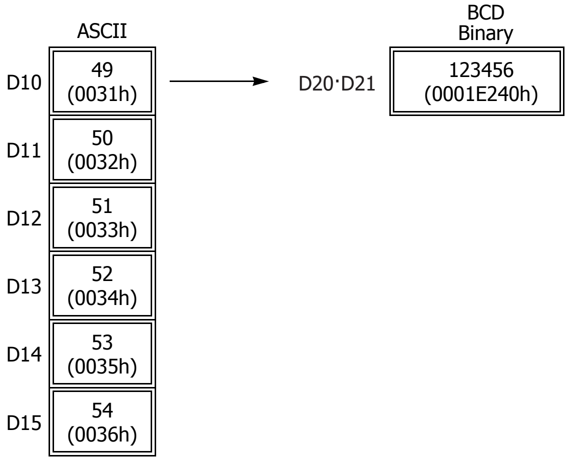
- **Quantity of Digits: 10**



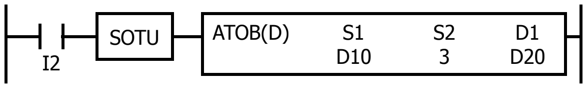


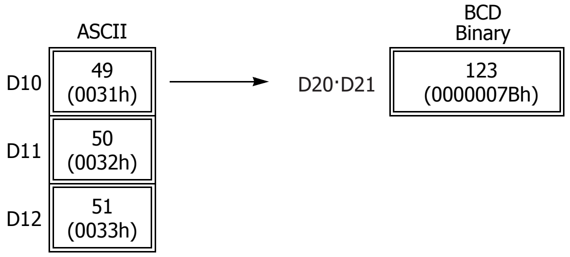
- **Quantity of Digits: 6**



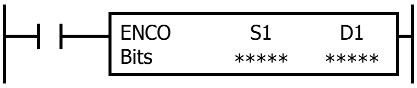


- **Quantity of Digits: 3**





ENCO (Encode)



When input is on, a search begins for the first bit that is set to on. The search begins at S1 until the first set (on) point is located. The number of points from S1 to the first set point (offset) is stored to the destination assigned by device D1.

If no point is on in the searched area, 65,535 is stored to D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	First bit to start search	X	X	X	X	—	—	X	—	—	—
D1 (Destination 1)	Destination to store search results	—	X	▲	X	—	—	X	—	—	—
Bits	Quantity of bits searched	—	—	—	—	—	—	—	—	1-256	—

For valid device address ranges, see "Device Addresses" on page 2-1.

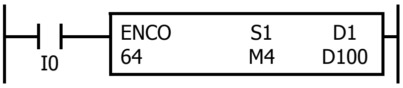
▲ Special internal relays cannot be designated as D1.

Make sure that the search area designated by S1 plus Bits is within the valid value range. If the source data exceeds the valid range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed. For user program execution errors, see "User Program Execution Errors" on page 3-10.

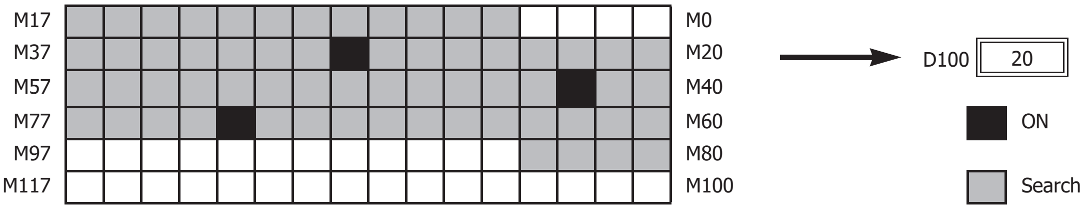
Since the ENCO instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

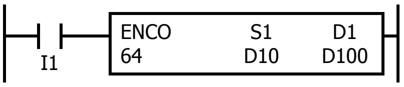
Examples: ENCO



When input I0 is on, a search begins for a bit that is set on in an array of 64 bits starting at internal relay M4 assigned by device S1.

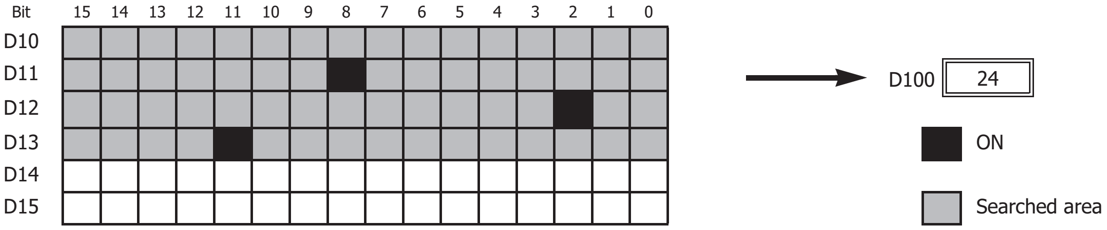
Since internal relay M30 is the first point that is on, the offset from the first search point is 20, and 20 is stored to data register D100 assigned by device D1.



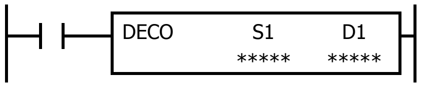


When input I1 is on, a search begins for a bit that is set on in an array of 64 bits starting at bit 0 of data register D10 assigned by device S1.

Since bit 8 of data register D11 is the first point that is on, the offset from the first search point is 24, and 24 is stored to data register D100 assigned by device D1.



DECO (Decode)



When input is on, the values contained in devices assigned by S1 and D1 are added together to determine the destination, and the bit is then turned on.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Offset	X	X	X	X	—	—	X	—	X	—
D1 (Destination 1)	First bit to count offset	—	X	▲	X	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

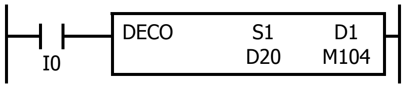
▲ Special internal relays cannot be designated as D1.

Valid values for the offset designated by source device S1 are 0 through 255. Make sure that the offset designated by S1 and the last bit of destination data determined by the sum of S1 and D1 are within the valid value range. If the offset or destination data exceeds the valid range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed. For user program execution errors, see "User Program Execution Errors" on page 3-10.

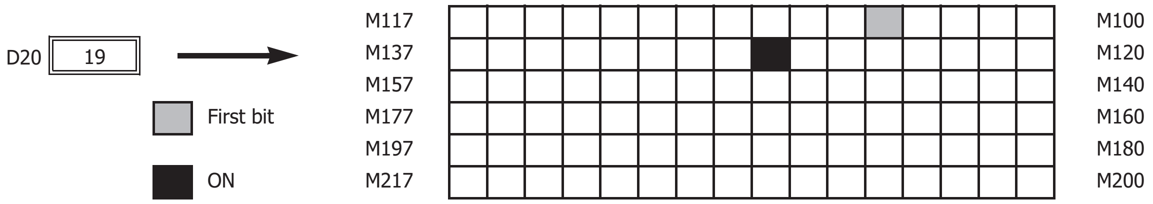
Since the DECO instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

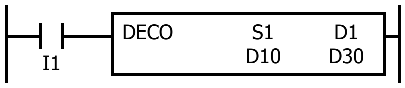
Examples: DECO



When input I0 is on, the destination bit is determined by adding the value contained in data register D20 assigned by device S1 to internal relay M104 assigned by destination device D1.

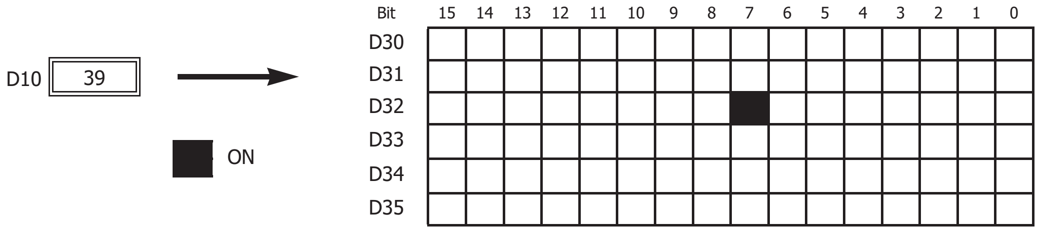
Since the 19th bit from internal relay M104 is internal relay M127, the bit is then turned on.



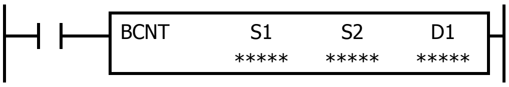


When input I1 is on, the destination bit is determined by adding the value contained in data register D10 assigned by device S1 to data register D30 assigned by destination device D1.

Since the 39th bit from data register D30 bit 0 is data register D32 bit 7, the bit is then turned on.



BCNT (Bit Count)



When input is on, a search begins for the total number of bits that are on an array of consecutive bits starting at the point assigned by source device S1. Source device S2 assigns the quantity of bits searched. The quantity of bits which are on is stored to the destination assigned by device D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	First bit to start search	X	X	X	X	—	—	X	—	—	—
S2 (Source 2)	Quantity of bits searched	X	X	X	X	X	X	X	—	X	—
D1 (Destination 1)	Destination to store quantity of ON bits	—	X	▲	X	X	X	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

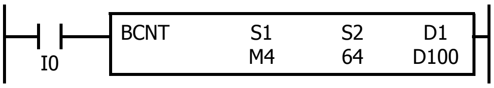
When T (timer) or C (counter) is used as S2, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP) which can be 0 through 65,535.

The valid range of S2 (quantity of bits searched) is 1 to 256. Make sure that the search area designated by S1 plus S2 is within the valid value range. If the source data exceeds the valid range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed. For user program execution errors, see "User Program Execution Errors" on page 3-10.

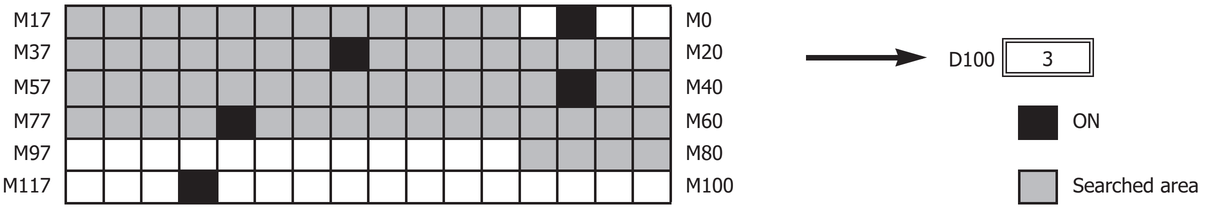
Since the BCNT instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

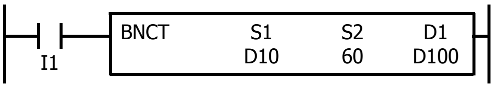
Examples: BCNT



When input is on, a search begins for the total number of bits that are on in an array of 64 bits starting at internal relay M4 assigned by source device S1.

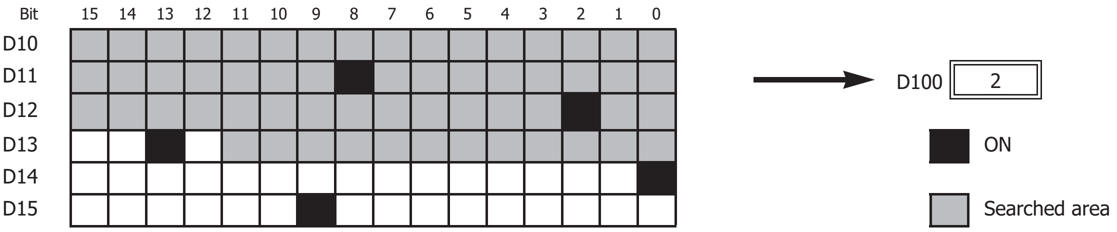
Since 3 bits are on in the searched area, the quantity is stored to data register D100 assigned by destination device D1.



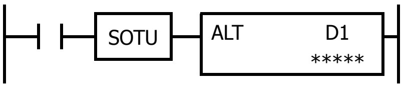


When input I0 is on, a search begins for the total number of bits that are on in an array of 60 bits starting at bit 0 of data register D10 assigned by device S1.

Since 2 bits are on among the 60 bits, 2 is stored to data register D100 assigned by device D1.



ALT (Alternate Output)



When input is turned on, output, internal relay, or shift register bit assigned by D1 is turned on and remains on after the input is turned off.

When input is turned on again, the assigned output, internal relay, or shift register bit is turned off.

The ALT instruction must be used with a SOTU or SOTD instruction, otherwise the assigned output, internal relay, or shift register bit repeats to turn on and off in each scan.

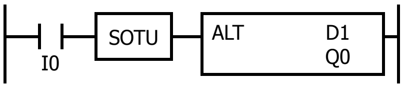
Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
D1 (Destination 1)	Bit to turn on and off	—	X	X	X	—	—	—	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

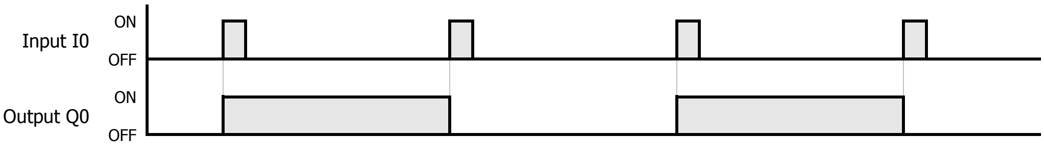
Since the ALT instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction must be used. For the SOTU instruction and the SOTD instruction, see "SOTU and SOTD (Single Output Up and Down)" on page 4-28.

Example: ALT

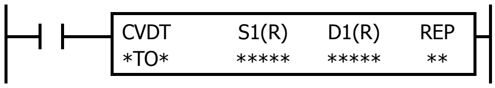


When input I0 is turned on, output Q0 assigned by device D1 is turned on and remains after input I0 is turned off.

When input I0 is turned on again, output Q0 is turned off.



CVDT (Convert Data Type)



S1 → D1

When input is on, the data type of the 16- or 32-bit data assigned by S1 is converted and stored to the destination assigned by device D1.

Data types can be assigned for the source and destination, separately

Data Type	W, I	D, L, F
Source	S1	S1·S1+1
Destination	D1	D1·D1+1

When the same data type is assigned for both source and destination, the CVDT instruction has the same function as the MOV instruction.

Unless F (float) data is selected for both source and destination, only the integral number is moved, omitting the fraction.

When the source data exceeds the range of destination data, the destination stores a value closest to the source data within the destination data.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	First device address to convert data type	X*1	X*1	X*1	X*1	X*1*2	X*2	X	—	X	1-99
D1 (Destination 1)	First device address to store converted data	—	X*1	X*1*3	X*1	X*1*4	X*1*4	X	—	—	1-99

For valid device address ranges, see "Device Addresses" on page 2-1.

*1 Cannot be used with data type F (float).

*2 When T/C is used as S1, this is the current value area.

*3 Special internal relays cannot be designated as D1.

*4 When T (timer) or C (counter) is used as S1, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP) which can be 0 through 65,535.

When a user program execution error occurs, the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Data Types

W (word)	X
I (integer)	X
D (double word)	X
L (long)	X
F (float)	X

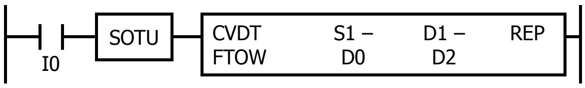
When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source or destination, 16 points (word or integer data) or 32 points (double-word, long, or float data) are used. When repeat is assigned for a bit device, the quantity of device bits increases in 16- or 32-point increments.

When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word or integer data) or 2 points (double-word, long, or float data) are used. When repeat is assigned for a word device, the quantity of device words increases in 1- or 2-point increments.

Examples: CVDT

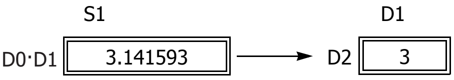
- **Data Type: Either S1 or D1 is not F (float)**

Unless F (float) data is selected for both source and destination, only the integral number is moved, omitting the fraction.



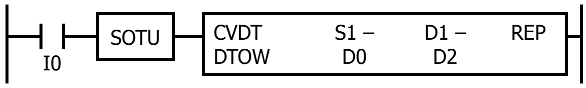
Device	Data Type	Value
Source	F	3.141593
Destination	W	3

When input I0 is turned on, 3 is stored to data register D2.



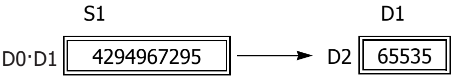
- **Data Type: S1 has a larger data range than D1**

When the source data exceeds the range of destination data type, the destination stores a value closest to the source data within the destination data type.

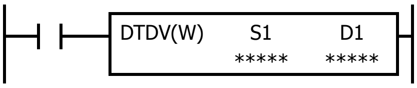


Device	Data Type	Value
Source	D	4,294,967,295
Destination	W	65,535

When input I0 is turned on, 65,535 is stored to data register D2.



DTDV (Data Divide)



S1 → D1, D1+1

When input is on, the 16-bit binary data assigned by S1 is divided into upper and lower bytes. The upper byte data is stored to the destination assigned by device D1. The lower byte data is stored to the device next to D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Binary data to divide	X	X	X	X	X	X	X	—	X	—
D1 (Destination 1)	Destination to store results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When T (timer) or C (counter) is used as S1, the timer/counter current value (TC or CC) is displayed.

Destination device D1 uses 2 data registers starting with the device assigned by D1.

Since the DTDV instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

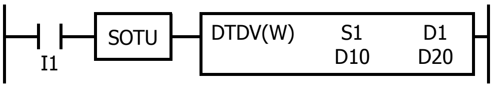
Valid Data Types

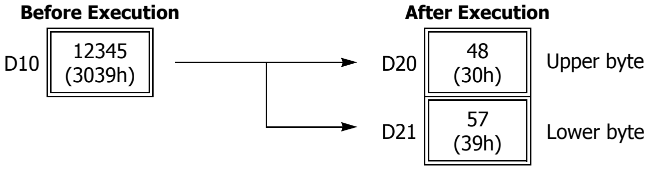
W (word)	X
I (integer)	—
D (double word)	—
L (long)	—
F (float)	—

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source, 16 points (word data) are used.

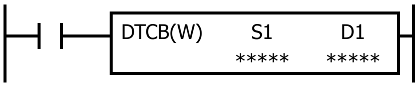
When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word data) is used.

Example: DTDV





DTCB (Data Combine)



S1, S1+1 → D1

When input is on, the lower-byte data is displayed from 2 consecutive sources starting with device assigned by S1 and combined to make 16-bit data. The lower byte data from the first source device is moved to the upper byte of the destination assigned by device D1, and the lower byte data from the next source device is moved to the lower byte of the destination.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Binary data to combine	—	—	—	—	—	—	X	—	—	—
D1 (Destination 1)	Destination to store results	—	X	▲	X	X	X	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP) which can be 0 through 65,535.

Source device S1 uses 2 data registers starting with the device assigned by S1.

Since the DTCB instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

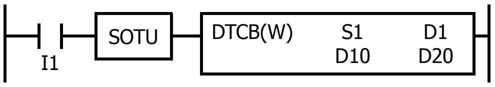
Valid Data Types

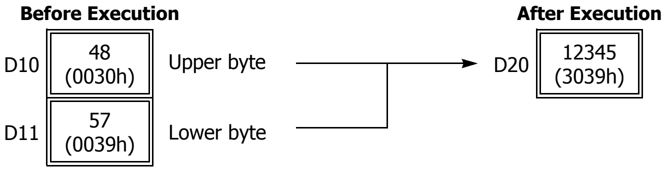
W (word)	X
I (integer)	—
D (double word)	—
L (long)	—
F (float)	—

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the destination, 16 points (word data) are used.

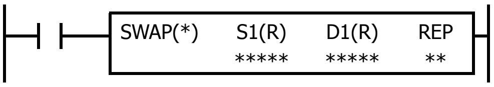
When a word device such as T (timer), C (counter), or D (data register) is assigned as the source or destination, 1 point (word data) is used.

Example: DTCB





SWAP (Data Swap)



S1 → D1
When input is on, upper and lower byte- or word-data of a word- or double-word-data assigned by S1 are exchanged, and the result is stored to destination assigned by D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Binary data to swap	—	—	—	—	—	—	X	—	—	1-99
D1 (Destination 1)	Destination to store conversion result	—	—	—	—	—	—	X	—	—	1-99

For valid device address ranges, see "Device Addresses" on page 2-1.

Since the SWAP instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

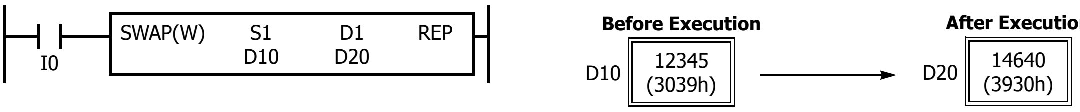
W (word)	X
I (integer)	—
D (double word)	X
L (long)	—
F (float)	—

When a D (data register) is assigned as the source or destination, 1 point (word data) or 2 points (double-word data) are used. When repeat is assigned, the quantity of device words increases in 1- or 2-point increments.

Examples: SWAP

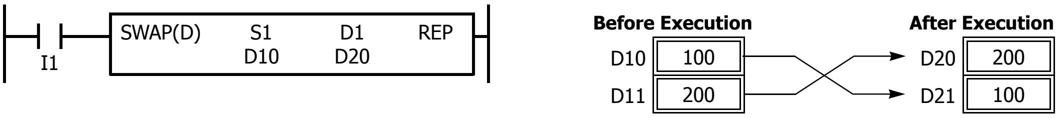
- **Data Type: W (word)**

When input I0 is turned on, upper- and lower-byte data of the 16-bit data in data register D10 assigned by source device S1 are exchanged, and the result is stored to data register D20 assigned by destination device D1.



- **Data Type: D (double-word)**

When input I1 is turned on, upper- and lower-word data of the 32-bit data in data registers D10 and D11 assigned by source device S1 are exchanged, and the result is stored to data registers D20 and D21 assigned by destination device D1.

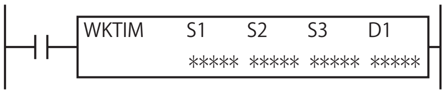


11: Week Programmer Instructions

This chapter describes the week programmer instructions that are used to turn outputs on and off on the specified days of the week or dates and times.

WKTIM (Week Timer)

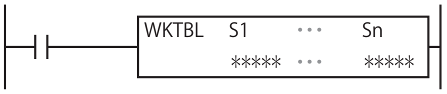
Compares the specified day of the week, start time, and end time with the current time and outputs that result.



Note: The WKTIM instruction is compatible with the FC5A Series MICROSmart WKTIM instruction. For details on the WKTIM instruction, see Chapter 9 "WKTIM (Week Timer)" in the "FC5A Series MICROSmart Pentra User's Manual Advanced Volume".

WKTBL (Week Table)

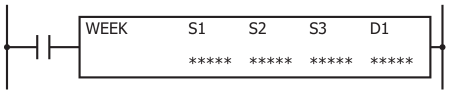
Sets the specified month and day as a special date.



Note: The WKTBL instruction is compatible with the FC5A Series MICROSmart WKTBL instruction. For details on the WKTBL instruction, see Chapter 9 "WKTBL (Week Table)" in the "FC5A Series MICROSmart Pentra User's Manual Advanced Volume".

WEEK (Weekly Timer)

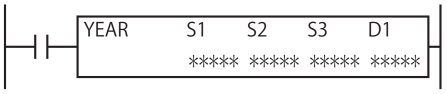
Compares the specified day of the week, ON time, and OFF time with the current time and outputs that result.



Note: For details on the WEEK instruction, see "WEEK (Weekly Timer)" on page 11-3.

YEAR (Yearly Timer)

Compares the specified date with the current date and outputs that result. This allows the special dates within a one-year period to be specified.



Note: For details on the YEAR instruction, see "YEAR (Yearly Timer)" on page 11-16.

Differences in the week programmer instructions

There are four types of week programmer instructions: the WKTIM instruction, the WKTBL instruction, the WEEK instruction, and the YEAR instruction. The initial values of the WEEK and YEAR instructions, such as the day of the week, ON time, OFF time, and special dates, can be easily configured in the dialog box for the instruction. The initial values of the WKTIM and WKTBL instructions must be configured using data registers.

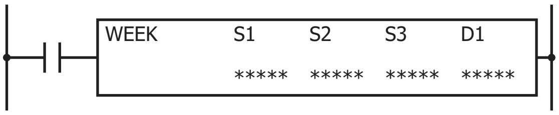
Comparing the WKTIM and WEEK instructions

Item	WKTIM Instruction	WEEK Instruction
Number of instructions that can be used in one ladder program	No limit	10
Number of patterns that can be configured in one instruction	1 pattern	20 patterns
Pulse output function	No	Yes
Initial value configuration method	Configure initial values using data registers	Configure initial values using the dialog box in WindLDR
Configuration pattern preview function	No	Yes

Comparing the WKTBL and YEAR instructions

Item	WKTBL Instruction	YEAR Instruction
Number of instructions that can be used in one ladder program	No limit	10
Number of patterns that can be configured in one instruction	20 patterns	20 patterns
Pulse output function	No	Yes
Calendar specification methods	Specify date	Specify date Specify day of the week Specify end of the month
Initial value configuration method	Configure initial values using data registers in the ladder program	Configure initial values using the dialog box in WindLDR
Configuration pattern preview function	No	Yes

WEEK (Weekly Timer)



The WEEK instruction compares the specified day of the week, ON time, and OFF time with the current time and outputs that result.

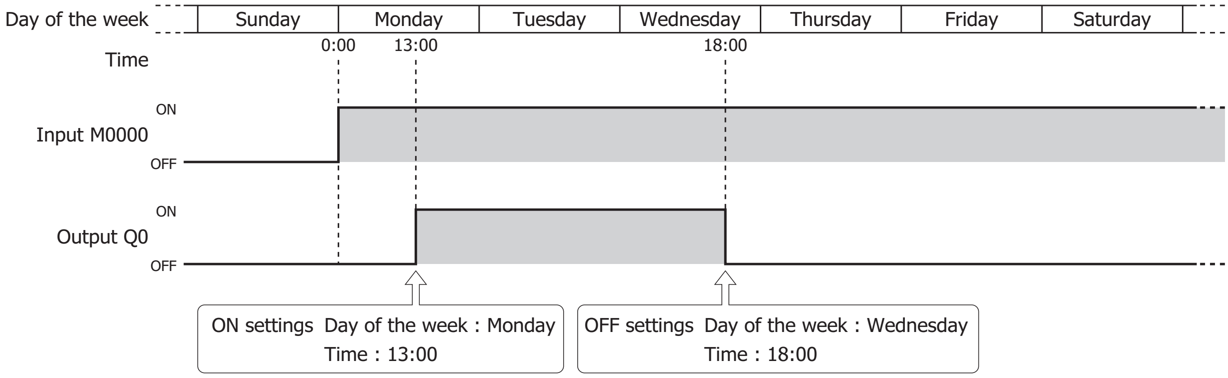
When the input is on, the output specified by D1 is turned on when the day of the week and the time specified by the ON settings match the current day of the week and time.

The output specified by D1 is also turned off when the day of the week and the time specified by the OFF settings match the current day of the week and time.

If the ON settings are Monday 13:00 and the OFF settings are Wednesday 18:00, output D1 turns on and off as follows.

When the WEEK instruction input is ON on Monday 13:00, output D1 turns ON on Monday 13:00 and it turns OFF on Wednesday 18:00.

(In this diagram the WEEK instruction input is turned on at 0:00 on Monday)



Notes:

- A maximum of 10 WEEK instructions can be used in a user program.
- Normally the output is only updated when the current time and the time for the ON/OFF settings matches, but when the WEEK instruction input changes from off to on, the output state at the current time is determined according to the ON/OFF settings and the output is turned on or off. For details, see "Timing Chart when the Input Turns On during the Configured Interval" on page 11-10.
- When pulse output is enabled, output is turned on for one scan at the ON settings time. For pulse output, see "6. Pulse Output" on page 11-5 or "6. Pulse Output" on page 11-8.
- When the ON time is set to a value larger than 2,359, the OFF time is set to a value larger than 2,400, or the lower 2 digits of ON/OFF time is set to a value larger than 59, a user program execution error will occur.
- When the day of the week is not specified, a user program execution error will occur.
- The WEEK instruction cannot be used in an interrupt program. If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	First data register	—	—	—	—	—	—	▲	—	—	—
S2 (Source 2)	Initialization input	—	—	X	—	—	—	—	—	—	—
S3 (Source 3)	Number of parameter tabs	—	—	—	—	—	—	—	—	X	—
D1 (Destination 1)	Output port	—	X	▲	—	—	—	—	—	—	—

▲ Special data registers cannot be designated as S1. Special internal relays cannot be designated as D1.

Settings

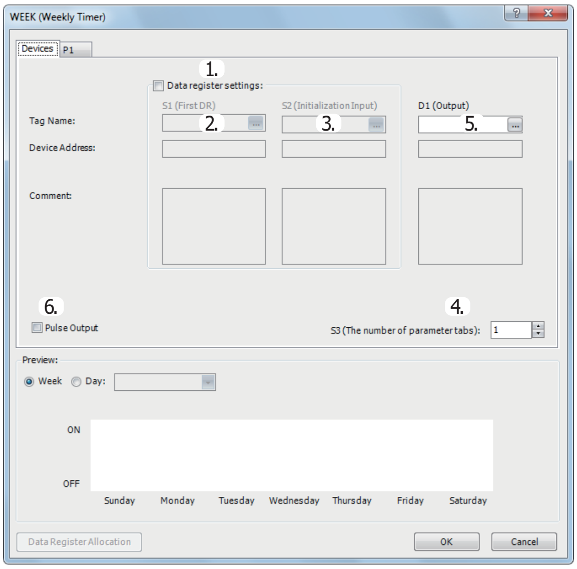
There are two methods to configure the day of the week and the time for the WEEK instruction. Specify the method on the **Devices** tab with **Data register settings**.

- Configuring the day of the week and the time as fixed settings
  The day of the week and the time for the ON/OFF settings are uniquely determined. The day of the week and the time for the ON/OFF settings cannot be changed while the FC6A Series MICROSmart is running. For details, see "To configure the day of the week and the time as fixed settings" on page 11-4.
- Configuring the day of the week and the time by specifying data registers
  The day of the week and the time for the ON/OFF settings are configured according to the values stored in the specified data registers. The day of the week and the time for the ON/OFF settings can be changed while the FC6A Series MICROSmart is running. For details, see "To configure the days of the week and the time by specifying data registers" on page 11-7.

To configure the day of the week and the time as fixed settings

The day of the week and the time for the ON/OFF settings are uniquely determined. The day of the week and the time for the ON/OFF settings cannot be changed while the FC6A Series MICROSmart is running.

• Devices tab



1. Data register settings

This setting determines whether to configure the day of the week and the time as fixed settings or to indirectly specify them with data registers.

To configure the day of the week and the time as fixed settings, clear this check box.

☐ **Unselected check box**

The day of the week and the time are fixed settings. The day of the week and the time are configured on the parameter tabs. The day of the week and the time cannot be changed while the FC6A Series MICROSmart is running. For the settings, see "Parameter tab" on page 11-5.

Note: When this check box is selected, the settings for the day of the week and the time are indirectly specified with data registers. Configure the day of the week and the time by using data registers. The day of the week and the time can be changed while the FC6A Series MICROSmart is running. For directions specific to data registers, see "To configure the days of the week and the time by specifying data registers" on page 11-7.

2. S1 (source 1): First Data Register

This setting is not configured when configuring the day of the week and the time as fixed settings.

3. S2 (source 2): Initialization Input

This setting is not configured when configuring the day of the week and the time as fixed settings.

4. S3 (source 3): Number of parameter tabs

This setting configures the number of parameter tabs. When this value is increased or decreased, the number of parameter tabs displayed on the dialog box increases or decreases. 6 bytes of the user program region are used for each parameter tab.
For the settings, see "Parameter tab" on page 11-5.

5. D1 (destination 1): Output

This setting configures the output for the results when the configured day of the week and time and the current day of the week and time are compared.

Setting	Description
Tag Name	Specifies the tag name or the device address for each device.
Device Address	Shows the device address that corresponds to the tag name.
Comment	Shows the comment for the device address. This item can be edited.

6. Pulse Output

This setting configures the operation for D1 (output). This setting is applied to all parameter tabs.

☑ **Selected check box**

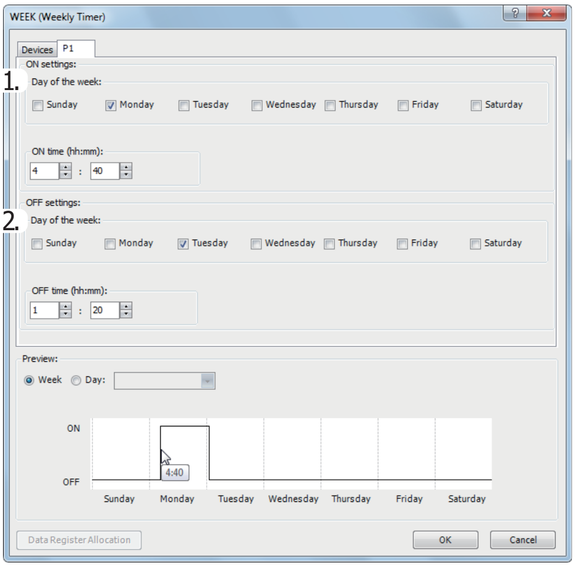
When the day of the week and the time configured in the ON settings matches the current day of the week and the time, the output is turned on for only one scan.

☐ **Unselected check box**

The output is turned on and off according to the ON settings and the OFF settings.

- **Parameter tab**

This tab configures the ON/OFF settings for the output. A maximum of 20 parameter tabs can be configured for one WEEK instruction.



WEEK (Weekly Timer)

Devices | P1

ON settings:

1. Day of the week:
☐ Sunday ☑ Monday ☐ Tuesday ☐ Wednesday ☐ Thursday ☐ Friday ☐ Saturday

ON time (hh:mm): 4 : 40

OFF settings:

2. Day of the week:
☐ Sunday ☐ Monday ☑ Tuesday ☐ Wednesday ☐ Thursday ☐ Friday ☐ Saturday

OFF time (hh:mm): 1 : 20

Preview:
● Week ○ Day:

1. ON settings

This section configures the day of the week and the time to turn on the output. The output is turned on at the specified time for the specified day of the week.

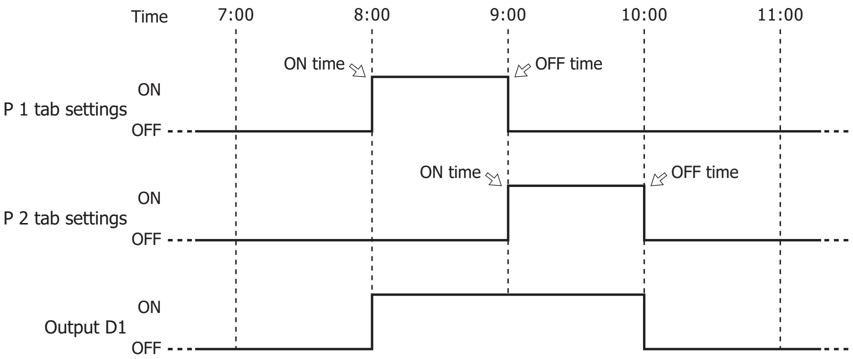
Setting	Description	Range
Day of the week	Specifies the days of the week.	—
ON time	Used to enter the time. Set the time in a range from 00:00 to 23:59.	Hour: 0 to 23
		Minute: 0 to 59

2. OFF settings

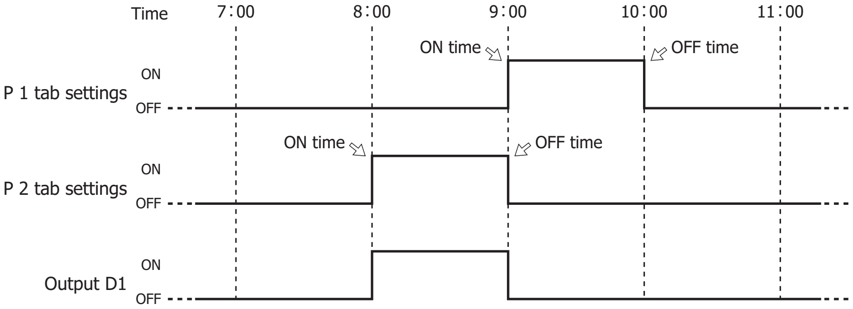
This section configures the day of the week and the time to turn off the output. The output is turned off at the specified time for the specified day of the week.

Setting	Description	Range
Day of the week	Specifies the days of the week.	—
OFF time	Used to enter the time. Set the time in a range from 00:00 to 24:00.	Hour: 0 to 24
		Minute: 0 to 59

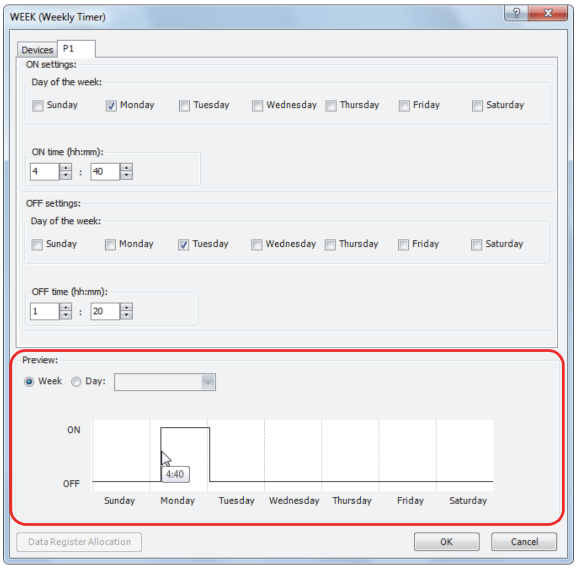
Note: When the time is duplicated in the settings on other tabs, the settings on the tab with the larger tab number are valid.
For example, if **ON time** is 8:00 and **OFF time** is 9:00 on the **P 1** tab and **ON time** is 9:00 and **OFF time** is 10:00 on the **P 2** tab, the 9:00 setting is duplicated on the two tabs and **OFF time** for the **P 1** tab is disabled. In this situation, the output is on from 8:00 to 10:00.



If **ON time** is 9:00 and **OFF time** is 10:00 on the **P 1** tab and **ON time** is 8:00 and **OFF time** is 9:00 on the **P 2** tab, the 9:00 setting is duplicated on the two tabs and **ON time** for the **P 1** tab is disabled. In this situation, the output is on from 8:00 to 9:00.



- **Preview**


WEEK (Weekly Timer)

Devices | P1

ON settings:

Day of the week:

□ Sunday ☑ Monday □ Tuesday □ Wednesday □ Thursday □ Friday □ Saturday

ON time (hh:mm): 4 : 40

OFF settings:

Day of the week:

□ Sunday □ Monday ☑ Tuesday □ Wednesday □ Thursday □ Friday □ Saturday

OFF time (hh:mm): 1 : 20

Preview:

● Week ○ Day:

The preview shows the ON/OFF state for the output based on the settings configured on the parameter tabs as a time chart. The preview can be shown as a week or a day.

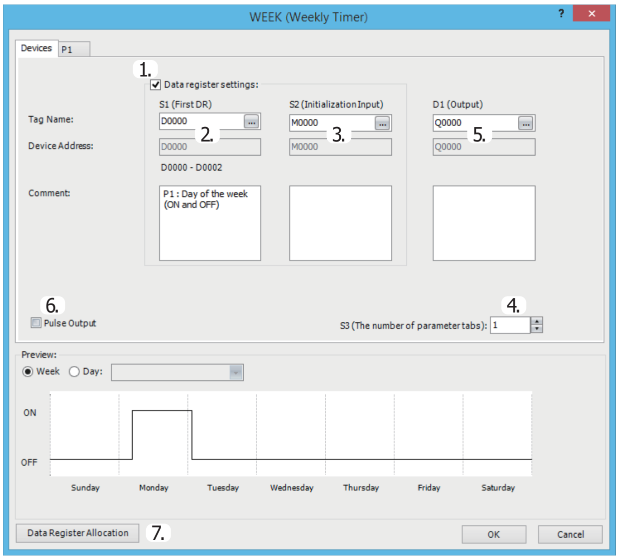
Setting	Description
Week	Select this radio button when showing the preview as a week.
Day	Select this radio button when showing the preview as a day.

To configure the days of the week and the time by specifying data registers

The day of the week and the time for the ON/OFF settings are configured according to the values stored in the specified data registers. The day of the week and the time for the ON/OFF settings can be changed while the FC6A Series MICROSmart is running.

Note: When the day of the week or time in the ON/OFF settings is modified with the WEEK instruction input turned on, it is not reflected in the output operation until the current day of the week and time match the day of the week and time in the changed ON/OFF settings.
To immediately reflect this in the output operation, turn the WEEK instruction off and then on again after changing the day of the week or time.

- **Devices tab**



WEEK (Weekly Timer)

Devices | P1

☑ Data register settings:

	S1 (First DR)	S2 (Initialization Input)	D1 (Output)
Tag Name:	D0000	M0000	Q0000
Device Address:	D0000	M0000	Q0000
	D0000 - D0002		
Comment:	P1 : Day of the week (ON and OFF)		

□ Pulse Output

S3 (The number of parameter tabs): 1

Preview:

● Week ○ Day:

1. Data register settings

This setting selects whether to configure the day of the week and the time as fixed settings or to indirectly specify them with data registers. To configure the day of the week and the time by specifying data registers, select this check box.

☑ Selected check box

The day of the week and the time settings are indirectly specified by data registers. Configure the day of the week and the time by using data registers. The day of the week and the time can be changed while the FC6A Series MICROSmart is running. For allocating the data register region, see "Data Register Allocation" on page 11-9.

The data registers can be initialized with the days of the week and the times configured on the parameter tabs by turning on the initialization input. For initialization, see "3. S2 (source 2): Initialization Input" on page 11-8.

Note: When this check box is not selected, the day of the week and the time are fixed settings. The day of the week and the time are configured on the parameter tabs. The day of the week and the time cannot be changed while the FC6A Series MICROSmart is running. For fixed settings, see "To configure the day of the week and the time as fixed settings" on page 11-4.

2. S1 (source 1): First Data Register

This setting specifies the start of the data register region to store the days of the week and time settings for the WEEK instruction.

This setting is only used when indirectly specifying the settings for the WEEK instruction with data registers.

Setting	Description
Tag Name	Specifies the tag name or the device address for the device.
Device Address	Shows the device address that corresponds to the tag name.
Used Data Registers	Shows the range of data registers used to store the settings. This item changes when the device address or the number of parameter tabs changes.
Comment	Shows the comment for the device address. This item can be edited.

For allocating the data register region, see "Parameter tab" on page 11-8.

3. S2 (source 2): Initialization Input

This setting specifies the device to initialize the days of the week and the times stored in the data register region that starts from S1 (source 1).

The values configured on the parameter tabs are stored in the data registers by turning on the initialization input.

This setting is only used when indirectly specifying the settings for the WEEK instruction with data registers.

4. S3 (source 3): The number of parameter tabs

This setting configures the number of parameter tabs.
This setting is shared in common with "To configure the day of the week and the time as fixed settings". See "4. S3 (source 3): Number of parameter tabs" on page 11-4.

5. D1 (destination 1): Output

This setting configures the output for the results when the configured day of the week and time and the current day of the week and time are compared.
This setting is shared in common with "To configure the day of the week and the time as fixed settings". See "5. D1 (destination 1): Output" on page 11-5.

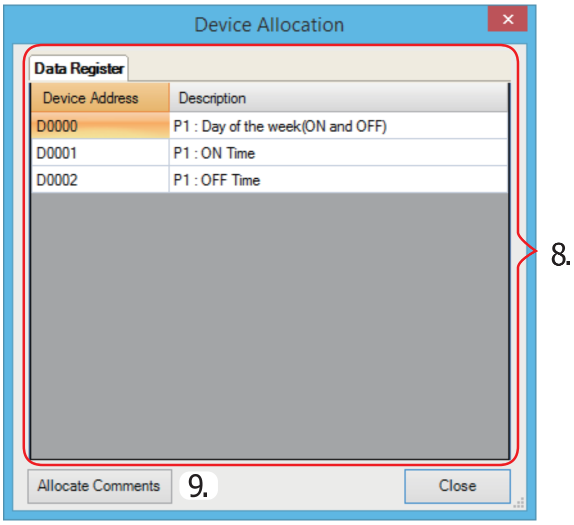
6. Pulse Output

This setting configures the operation for D1 (output). This setting is applied to all parameter tabs and is shared in common with "To configure the day of the week and the time as fixed settings". See "6. Pulse Output" on page 11-5.

7. Data Register Allocation

Click this button to display the **Device Allocation** dialog box. As shown below, a table showing the data registers and their corresponding WEEK instruction settings is displayed on the dialog box (8). Click **Allocate Comments** (9) and you can configure the comments for the data registers that correspond to the names of the settings.
This button is only used when indirectly specifying the settings for the WEEK instruction with data registers.

• Device Allocation dialog box



Device Address	Description
D0000	P1 : Day of the week(ON and OFF)
D0001	P1 : ON Time
D0002	P1 : OFF Time

• Parameter tab

This tab configures the ON/OFF settings for the output. A maximum of 20 parameter tabs can be configured for 1 WEEK instruction.
If indirectly specifying the settings for the WEEK instruction with data registers, the settings configured on the parameter tabs are stored in the data registers when the initialization input is turned on.

This setting is shared in common with "To configure the day of the week and the time as fixed settings". See "Parameter tab" on page 11-5.

• Preview

The preview shows the ON/OFF state for the output based on the settings configured on the parameter tabs as a time chart.
This function is shared in common with "To configure the day of the week and the time as fixed settings". See "Preview" on page 11-6.

Data Register Allocation

If indirectly specifying the settings for the WEEK instruction with data registers, the settings are allocated to the data registers as follows.

Storage Destination	Data Size (word)	R (Read)/W (Write)	Setting	
Start address+0	1	R/W	P 1 tab	Day of the week
Start address+1	1	R/W		ON time
Start address+2	1	R/W		OFF time
Start address+3	1	R/W	P 2 tab	Day of the week
Start address+4	1	R/W		ON time
Start address+5	1	R/W		OFF time
• • •	• • •	• • •		• • •
Start address+57	1	R/W	P 20 tab	Day of the week
Start address+58	1	R/W		ON time
Start address+59	1	R/W		OFF time

- **Day of the week data register allocation**

The day of the week for the ON settings and the day of the week for the OFF settings are allocated as bits in 1 data register as follows.

ON settings day of the week								OFF settings day of the week							
Bit 15	14	13	12	11	10	9	Bit 8	Bit 7	6	5	4	3	2	1	Bit 0
Reserved	Saturday	Friday	Thursday	Wednesday	Tuesday	Monday	Sunday	Reserved	Saturday	Friday	Thursday	Wednesday	Tuesday	Monday	Sunday

0: Disabled (check box not selected)
1: Enabled (check box selected)

Data register (1 word)

Example day of the week settings

[To configure the output to turn on Monday and to turn off Friday]

Monday is enabled in the ON settings, Friday is enabled in the OFF settings

ON settings								OFF settings							
Bit 15	14	13	12	11	10	9	Bit 8	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	0	0	0	1	0		0	1	0	0	0	0	0
Reserved	Saturday	Friday	Thursday	Wednesday	Tuesday	Monday	Sunday	Reserved	Saturday	Friday	Thursday	Wednesday	Tuesday	Monday	Sunday

Day of the week setting (ON): 0000010
Day of the week setting (OFF): 0100000
The value of the data register is 1000100000 (binary) = 544 (decimal).

[To configure the output to turn on Monday and Thursday and to turn off Tuesday and Saturday]

Monday and Thursday are enabled in the ON settings, Tuesday and Saturday are enabled in the OFF settings

ON settings								OFF settings							
Bit 15	14	13	12	11	10	9	Bit 8	Bit 7	6	5	4	3	2	1	Bit 0
	0	0	1	0	0	1	0		1	0	0	0	1	0	0
Reserved	Saturday	Friday	Thursday	Wednesday	Tuesday	Monday	Sunday	Reserved	Saturday	Friday	Thursday	Wednesday	Tuesday	Monday	Sunday

Day of the week setting (ON): 0010010
Day of the week setting (OFF): 1000100
The value of the data register is 1001001000100 (binary) = 4676 (decimal).

The ON time and the OFF time are stored in the data register as follows.

12:34 ⬇
For 12:34 | 1234 (Decimal) |
Data register (1 word)

23:45 ⬇
For 23:45 | 2345 (Decimal) |
Data register (1 word)

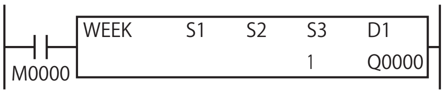
Timing Chart when the Input Turns On during the Configured Interval

When the input turns on or off during the interval between the ON settings and the OFF settings, and when the input is turns on or off after 0:00 on the date configured by the ON settings while pulse output is enabled, the timing chart is as follows.

■ When pulse output is disabled

Setting details

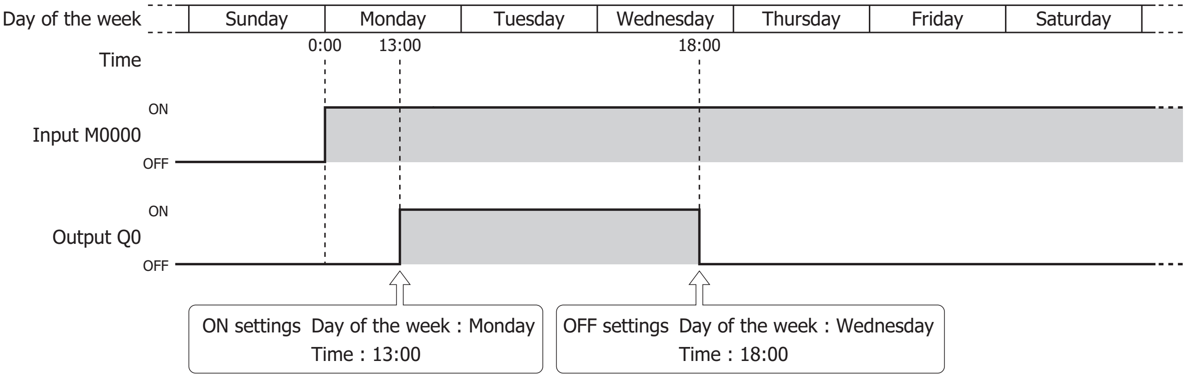
ON Settings	Monday 13:00
OFF Settings	Wednesday 18:00
Output	Q0



[When the input is on before the day of the week and time specified by the ON settings]

On Monday at 0:00 when input M0000 is turned on, the result of the current date compared with the ON and OFF settings is not within the ON and OFF settings (Monday 13:00 to Wednesday 18:00). Therefore, the output remains off.

Afterward, on Monday at 13:00, output Q0 turns on, and it turns off on Wednesday at 18:00.

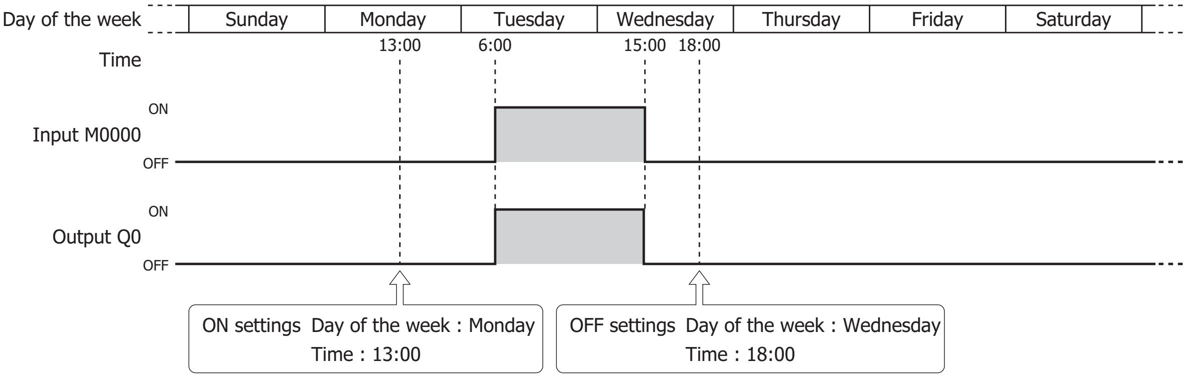


ON settings Day of the week : Monday
Time : 13:00

OFF settings Day of the week : Wednesday
Time : 18:00

[When the input is turned on or off during the interval between the ON and OFF settings]

On Tuesday at 6:00 when input M0000 is turned on, the result of the current date compared with the ON and OFF settings is within the ON and OFF settings (Monday 13:00 to Wednesday 18:00). Therefore, the output turns on. Afterward, if input M0000 is turned off before the OFF settings, the output also turns off at the same time.



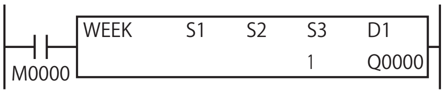
ON settings Day of the week : Monday
Time : 13:00

OFF settings Day of the week : Wednesday
Time : 18:00

■ When pulse output is enabled

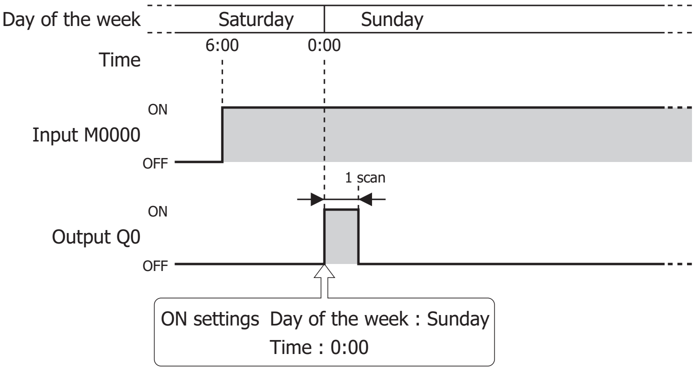
Setting details

ON settings	Sunday 0:00
Output	Q0



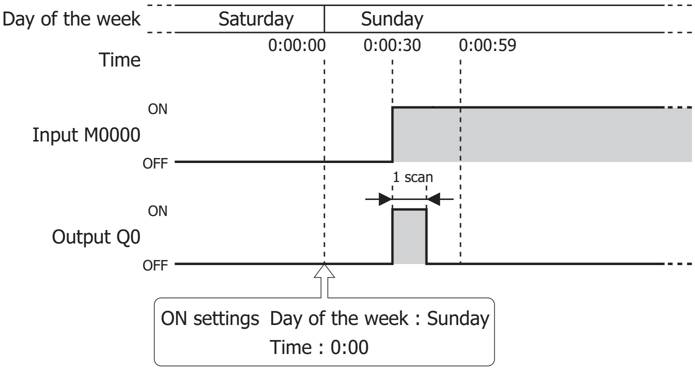
[When the input turns on before the day of the week and time specified by the ON settings]

On Sunday at 0:00, input M0000 is on, so output Q0 turns on for one scan only on Sunday at 0:00.



[When the input turns on during the day of the week and time specified by the ON settings]

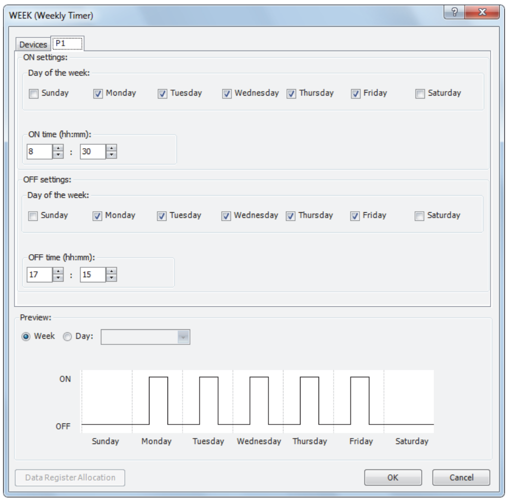
On Sunday at 0:00:30 (between Sunday 0:00:00 and 0:00:59), when input M0000 changes from off to on, output Q1 turns on for one scan only.



Examples: WEEK

[To turn on output Q0 Monday to Friday each week from 8:30 to 17:15]

Parameter tab



ON settings:

Day of the week:

□ Sunday ☑ Monday ☑ Tuesday ☑ Wednesday ☑ Thursday ☑ Friday □ Saturday

ON time (hh:mm): 8 : 30

OFF settings:

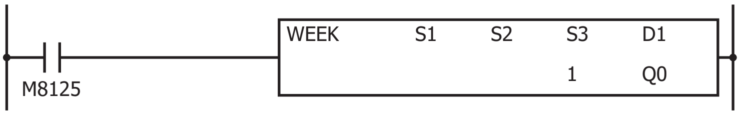
Day of the week:

□ Sunday ☑ Monday ☑ Tuesday ☑ Wednesday ☑ Thursday ☑ Friday □ Saturday

OFF time (hh:mm): 17 : 15

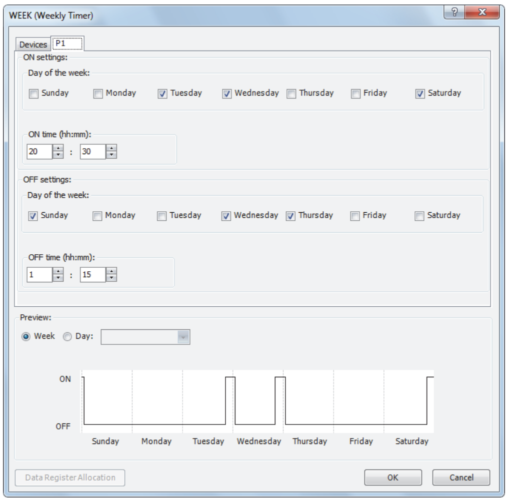
● Week ○ Day:

Configure the tab as shown above and set **D1** to Q0.



[To turn on output Q0 Tuesday, Wednesday, and Saturday each week from 20:30 to 1:15 the next day]

Parameter tab



ON settings:

Day of the week:

□ Sunday □ Monday ☑ Tuesday ☑ Wednesday □ Thursday □ Friday ☑ Saturday

ON time (hh:mm): 20 : 30

OFF settings:

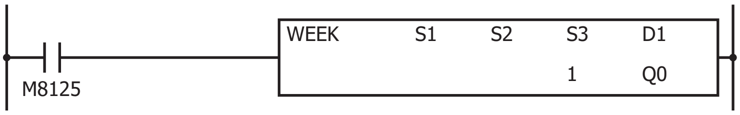
Day of the week:

☑ Sunday □ Monday □ Tuesday ☑ Wednesday ☑ Thursday □ Friday □ Saturday

OFF time (hh:mm): 1 : 15

● Week ○ Day:

Configure the tab as shown above and set **D1** to Q0.

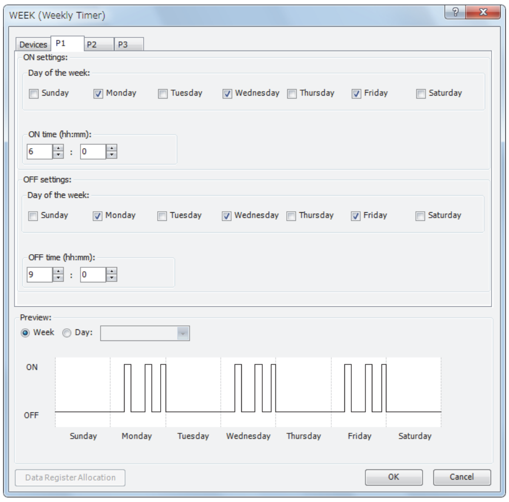


[To turn on output Q0 Monday, Wednesday, and Friday each week from 6:00 to 9:00, 15:00 to 18:00, and 22:00 to 0:00 the next day]

Parameter tab

Configure the settings using three tabs.

On P 1 tab, configure the output to turn on Monday, Wednesday, and Friday from 6:00 to 9:00.



WEEK (Weekly Timer)

Devices | P1 | P2 | P3

ON settings:

Day of the week:

□ Sunday ☑ Monday □ Tuesday ☑ Wednesday □ Thursday ☑ Friday □ Saturday

ON time (hh:mm): 6 : 0

OFF settings:

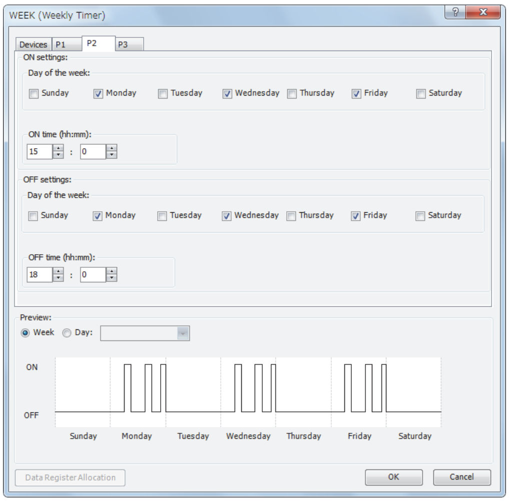
Day of the week:

□ Sunday ☑ Monday □ Tuesday ☑ Wednesday □ Thursday ☑ Friday □ Saturday

OFF time (hh:mm): 9 : 0

Preview: ● Week ○ Day:

On P 2 tab, configure the output to turn on Monday, Wednesday, and Friday from 15:00 to 18:00.



WEEK (Weekly Timer)

Devices | P1 | P2 | P3

ON settings:

Day of the week:

□ Sunday ☑ Monday □ Tuesday ☑ Wednesday □ Thursday ☑ Friday □ Saturday

ON time (hh:mm): 15 : 0

OFF settings:

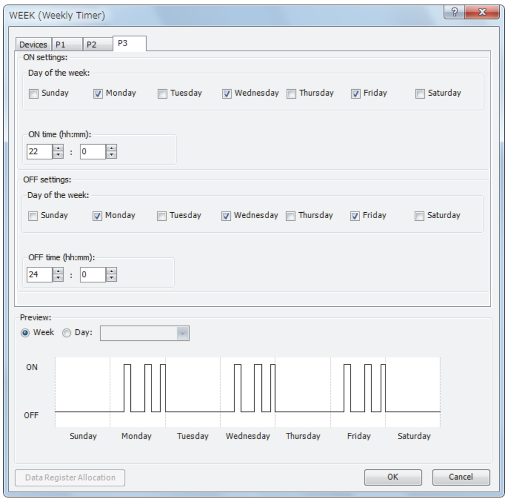
Day of the week:

□ Sunday ☑ Monday □ Tuesday ☑ Wednesday □ Thursday ☑ Friday □ Saturday

OFF time (hh:mm): 18 : 0

Preview: ● Week ○ Day:

On P 3 tab, configure the output to turn on Monday, Wednesday, and Friday from 22:00 to 0:00 the next day.



WEEK (Weekly Timer)

Devices | P1 | P2 | P3

ON settings:

Day of the week:

□ Sunday ☑ Monday □ Tuesday ☑ Wednesday □ Thursday ☑ Friday □ Saturday

ON time (hh:mm): 22 : 0

OFF settings:

Day of the week:

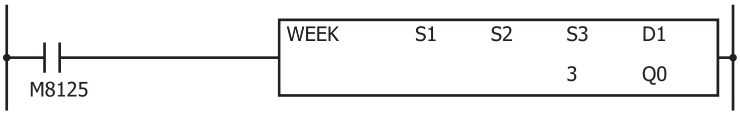
□ Sunday ☑ Monday □ Tuesday ☑ Wednesday □ Thursday ☑ Friday □ Saturday

OFF time (hh:mm): 24 : 0

Preview:

● Week ○ Day:

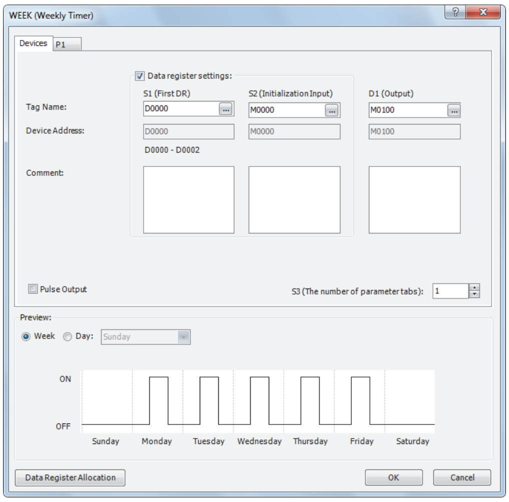
Configure the tabs as shown above and set **D1** to Q0.



[To indirectly specify the settings with data registers]

This example describes turning on output M0100 Monday to Friday each week from 8:30 to 17:15 as an example.
Select the **Data register settings** check box and set S1 to D0000 and S2 to M0000.

Parameter tab



WEEK (Weekly Timer)

Devices | P1

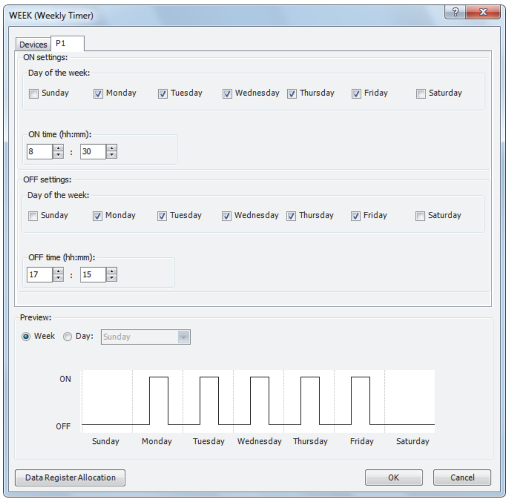
☑ Data register settings:

	S1 (First DR)	S2 (Initialization Input)	D1 (Output)
Tag Name:	D0000	M0000	M0100
Device Address:	D0000 D0000 - D0002	M0000	M0100
Comment:			

□ Pulse Output

S3 (The number of parameter tabs): 1

Preview: ● Week ○ Day: Sunday



WEEK (Weekly Timer)

Devices | P1

ON settings:

Day of the week: □ Sunday ☑ Monday ☑ Tuesday ☑ Wednesday ☑ Thursday ☑ Friday □ Saturday

ON time (hh:mm): 8 : 30

OFF settings:

Day of the week: □ Sunday ☑ Monday ☑ Tuesday ☑ Wednesday ☑ Thursday ☑ Friday □ Saturday

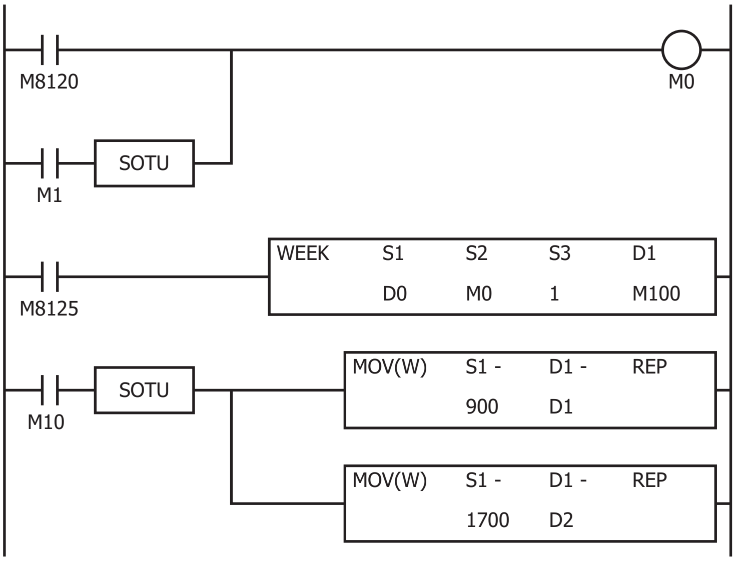
OFF time (hh:mm): 17 : 15

Preview: ● Week ○ Day: Sunday

Data register allocation

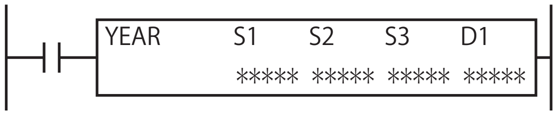
The settings on the **P 1** tab are allocated to data registers D0 to D2 as shown in the table below. The settings configured on the parameter tab are stored in D0 to D2 by turning on initialization input S2.

Data Register	Setting		Initial Setting
D0	P 1 tab	Day of the week setting	15934 (Monday to Friday, both ON settings and OFF settings)
D1		ON time	830
D2		OFF time	1715



- The initialization input (M1) turns on with the first scan and the initial settings configured on the **P 1** tab are stored in D0 to D2.
- The WEEK instruction starts operating according to the values of data registers D0 to D2.
- When M10 turns on, the ON time (D1) changes to 9:00 and the OFF time (D2) changes to 17:00.
- When M1 turns on, all of the WEEK instruction settings (D0 to D2) return to the initial settings.

YEAR (Yearly Timer)

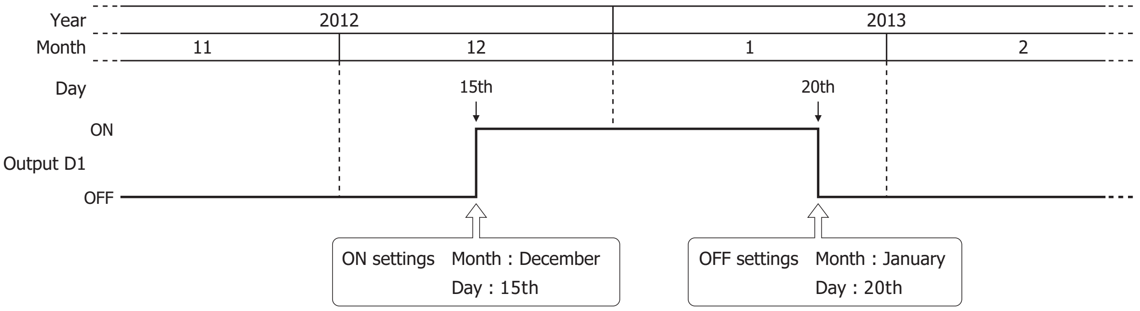


The YEAR instruction compares the specified date with the current date and outputs that result. With this instruction you can specify special dates within a one year period (A "special date" is a date configured with ON/OFF settings specified by the YEAR instruction).

When the input is on, the output specified by D1 is turned on when the date specified by the ON settings matches the current date.

The output is turned off when the date specified by the OFF settings matches the current date.

When the ON settings are December 15, 2012 and the OFF settings are January 20, 2013, output D1 turns on and off as follows.



Notes:

- A maximum of 10 YEAR instructions can be used in a user program.
- Normally the output is only updated when the current date and the date for the ON/OFF settings matches, but when the YEAR instruction input changes from off to on, the output state at the current date is determined according to the ON/OFF settings and the output is turned on or off.
  For details, see "Timing Chart when the Input Turns On during the Configured Interval" on page 11-25.
- When pulse output is enabled, output is turned on for one scan at the instant (0:00) the date changes to the ON date.
  When the YEAR instruction input changes from off to on, the output state at the current date is judged according to the ON/OFF settings, and the output is turned on for 1 scan only.
  For pulse output, see "6. Pulse Output" on page 11-18 or "6. Pulse Output" on page 11-22.
- When the year data is outside the range of 2000 to 2099, the month data is outside the range of 1 to 12, the day data is outside the range of 1 to 31, the week data is outside the range of 1 to 5, and the day of the week data is outside the range of 0 to 6, a user program execution error will occur.
- The YEAR instruction cannot be used in an interrupt program.
  If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.
  For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.
- When the yearly setting or monthly setting is enabled and a date that does not exist according to the month or year is set for the ON setting or the OFF setting, the output turns on or off on the first day of the next month.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	First data register	—	—	—	—	—	—	▲	—	—	—
S2 (Source 2)	Initialization input	—	—	X	—	—	—	—	—	—	—
S3 (Source 3)	Number of parameter tabs	—	—	—	—	—	—	—	—	X	—
D1 (Destination 1)	Output port	—	X	▲	—	—	—	—	—	—	—

▲ Special data registers cannot be designated as S1. Special internal relays cannot be designated as D1.

Settings

There are two methods to configure dates for the YEAR instruction. Specify the method on the **Devices** tab with **Data register settings**.

- Configure the dates as a fixed setting

  The dates for the ON/OFF settings are uniquely determined. The dates for the ON/OFF settings cannot be changed while the FC6A Series MICROSmart is running. For details, see "To configure the dates as a fixed setting" on page 11-17.

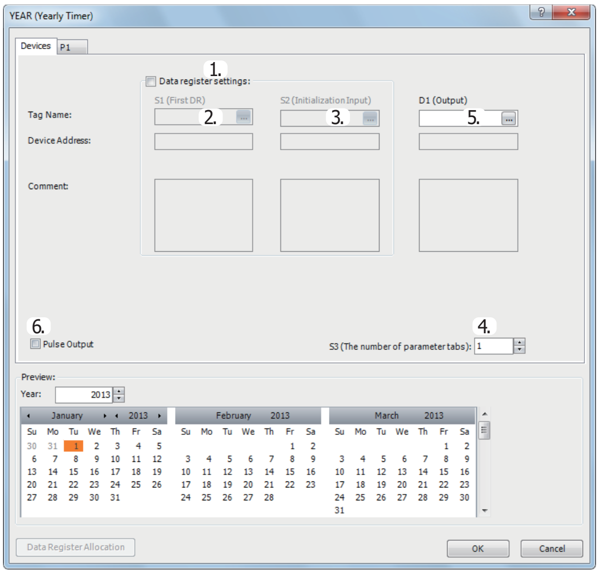
- Configuring the dates by specifying data registers

  The dates for the ON/OFF settings are configured according to the values stored in the specified data registers. The dates for the ON/OFF settings can be changed while the FC6A Series MICROSmart is running. For details, see "To configure the dates by specifying data registers" on page 11-21.

To configure the dates as a fixed setting

The dates for the ON/OFF settings are uniquely determined. The dates for the ON/OFF settings cannot be changed while the FC6A Series MICROSmart is running.

● Devices tab



YEAR (Yearly Timer)

Devices | P1

□ Data register settings:

	S1 (First DR)	S2 (Initialization Input)	D1 (Output)
Tag Name:			
Device Address:			
Comment:			

□ Pulse Output

S3 (The number of parameter tabs): 1

Preview:

Year: 2013

January 2013

Su	Mo	Tu	We	Th	Fr	Sa
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

February 2013

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28		

March 2013

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

1. Data register settings

This setting selects whether to configure YEAR instruction dates as fixed settings or to indirectly specify them with data registers.

To configure the dates as fixed settings, clear this check box.

□ **Unselected check box**

The dates are fixed settings.
The dates are configured on the parameter tabs. The dates cannot be changed while the FC6A Series MICROSmart is running. For the settings, see "Parameter tab" on page 11-5.

Note: When this check box is selected, the settings for dates are indirectly specified with data registers. Configure the dates by using data registers. The dates can be changed while the FC6A Series MICROSmart is running. For indirection specification with data registers, see "To configure the dates by specifying data registers" on page 11-21.

2. S1 (source 1): First Data Register

This setting is not configured when configuring the dates as fixed settings.

3. S2 (source 2): Initialization Input

This setting is not configured when configuring the dates as fixed settings.

4. S3 (source 3): Number of parameter tabs

This setting configures the number of parameter tabs. When this value is increased or decreased, the number of parameter tabs displayed on the dialog box increases or decreases. 10 bytes of the user program region are used for each **Parameter** tab.
For the settings, see "Parameter tab" on page 11-18.

5. D1 (destination 1): Output

This setting configures the output for the results when the configured dates and the current date are compared.

Setting	Description
Tag Name	Specifies the tag name or the device address for each device.
Device Address	Shows the device address that corresponds to the tag name.
Comment	Shows the comment for the device address. This item can be edited.

6. Pulse Output

This setting configures the operation for D1 (output). This setting is applied to all parameter tabs.

☑ **Selected check box**

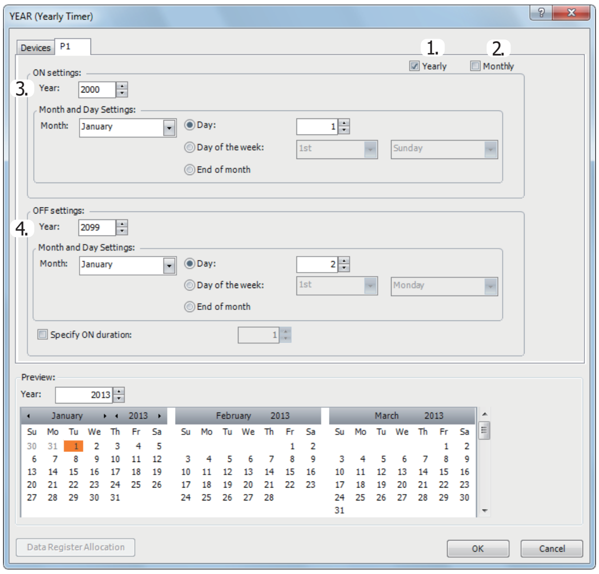
When the current date changes to the date in the ON settings, the output is turned on for only one scan.

☐ **Unselected check box**

The output is turned on and off according to the ON settings and the OFF settings.

• Parameter tab

This tab configures the settings for the output. A maximum of 20 parameter tabs can be configured for 1 YEAR instruction.



☑ Yearly ☐ Monthly

ON settings:
Year: 2000
Month and Day Settings:
Month: January
● Day: 1
○ Day of the week: 1st Sunday
○ End of month

OFF settings:
Year: 2099
Month and Day Settings:
Month: January
● Day: 2
○ Day of the week: 1st Monday
○ End of month

☐ Specify ON duration: 1

Preview:
Year: 2013

1. Yearly

When **Yearly** is selected, the **Month and Day settings** are valid every year.

At this time, you can configure whether or not to make the **Month and Day settings** valid yearly from a year until another year with **Year**.

2. Monthly

The settings for the selected tab are valid each month. At this time, **Month** is disabled.

3. ON settings

This section configures the date to turn on the output. The output is turned on at 0:00 on the configured date.

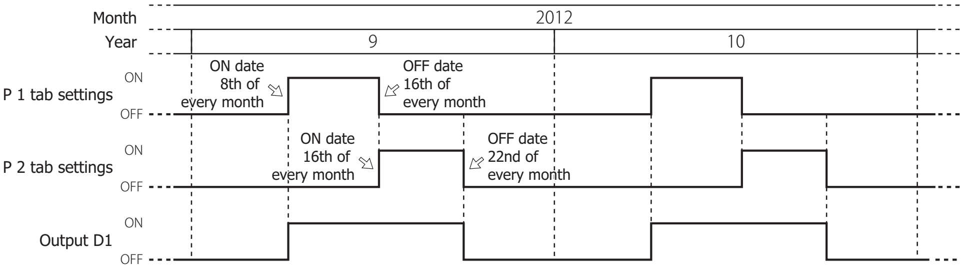
Setting		Description	Range
Year		Specifies the year to turn on the output.	2000 to 2099
Month and Day Setting	Month	Specifies the month to turn on the output.	1 to 12
	Day	Specifies the day to turn on the output.	1 to 31
	Day of the week	Specifies the day as the day of the week to turn on the output. Specify the week in the range of the 1st to the 5th (or the last) week and set the day of the week.	1 to 6
	End of month	Specify the end of month to turn on the output on the last day of the month.	—

4. OFF settings

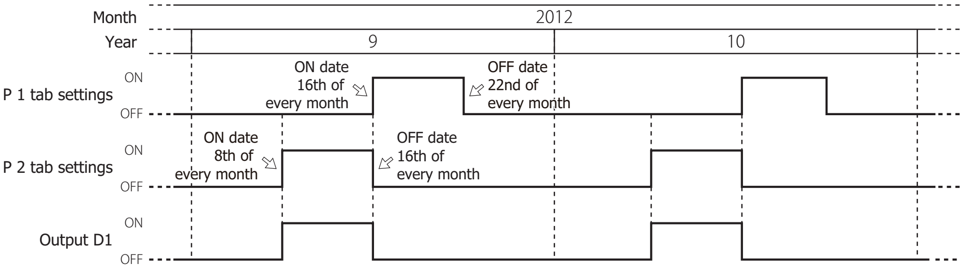
This section configures the date to turn off the output. The output is turned off at 0:00 on the configured date.

Setting		Description	Range
Year		Specifies the year to turn off the output.	2000 to 2099
Month and Day Setting	Month	Specifies the month to turn off the output.	1 to 12
	Day	Specifies the day to turn off the output.	1 to 31
	Day of the week	Specifies the day as the day of the week to turn on the output. Specify the week in the range of the 1st to the 5th (or the last) week and set the day of the week.	1 to 6
	End of month	Specify the end of month to turn off the output on the last day of the month.	—
Specify ON duration		This item specifies whether or not to turn off the output after the number of specified days from when it was turned on. If this setting is enabled, the OFF settings are disabled. This setting can be set in a range from 1 day to 30 days.	1 to 30

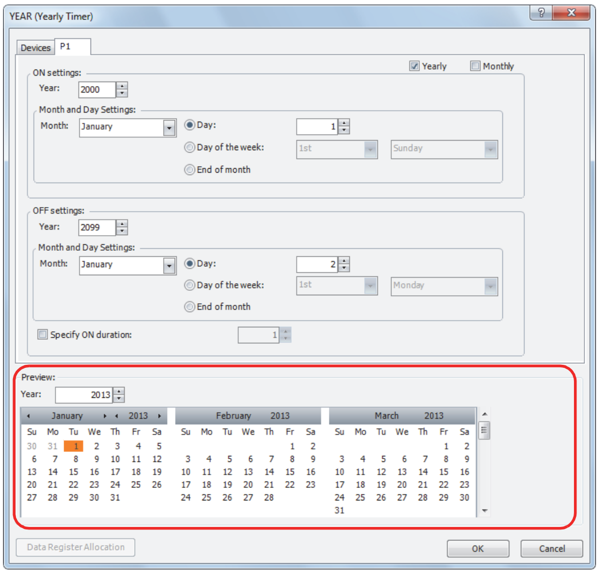
Note: When the date is duplicated in the settings on other tabs, the settings on the tab with the larger tab number are valid.
For example, if the ON date is the 8th of every month and the OFF date is the 16th of every month on the **P 1** tab and the ON date is the 16th of every month and the OFF date is the 22nd of every month on the **P 2** tab, the 16th of every month setting is duplicated on the two tabs and the ON settings for the **P 2** tab are valid. In this situation, the output is on from the 8th to the 22nd of every month.


Month
Year
2012
9
10
P 1 tab settings
ON
OFF
ON date 8th of every month
OFF date 16th of every month
P 2 tab settings
ON
OFF
ON date 16th of every month
OFF date 22nd of every month
Output D1
ON
OFF

If the ON date is the 16th of every month and the OFF date is the 22nd of every month on the **P 1** tab and the ON date is the 8th of every month and the OFF date is the 16th of every month on the **P 2** tab, the 16th of every month setting is duplicated on the two tabs and the ON settings for the **P 1** tab are disabled. In this situation, the output is on from the 8th to the 16th of every month.


Month
Year
2012
9
10
P 1 tab settings
ON
OFF
ON date 16th of every month
OFF date 22nd of every month
P 2 tab settings
ON
OFF
ON date 8th of every month
OFF date 16th of every month
Output D1
ON
OFF

- **Preview**



YEAR (Yearly Timer)

Devices | P1

☑ Yearly ☐ Monthly

ON settings:

Year: 2000

Month and Day Settings:

Month: January

● Day: 1

○ Day of the week: 1st Sunday

○ End of month

OFF settings:

Year: 2099

Month and Day Settings:

Month: January

● Day: 2

○ Day of the week: 1st Monday

○ End of month

☐ Specify ON duration: 1

Preview:

Year: 2013

January 2013

Su	Mo	Tu	We	Th	Fr	Sa
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

February 2013

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28		

March 2013

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

The preview shows the ON/OFF state for the output based on the settings configured on the parameter tabs in a calendar. The dates that are set to ON are highlighted in orange. Three months are shown at one time.

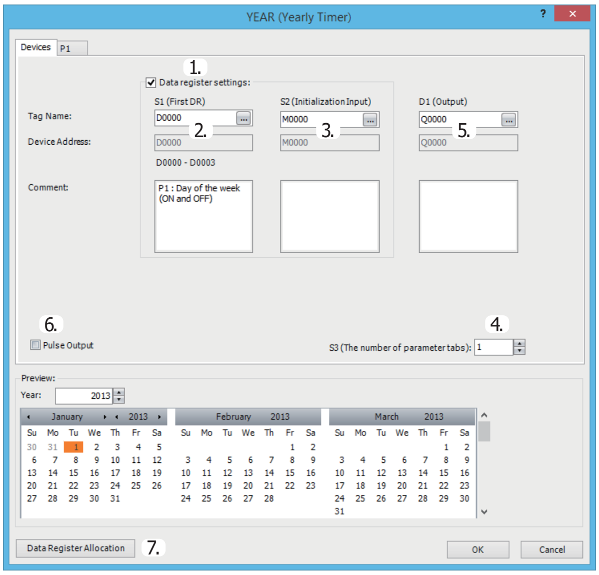
Setting	Description
Year	Specifies the year to show in the preview.
Scrollbar	You can change the month shown in the preview by moving the scrollbar.

To configure the dates by specifying data registers

The dates for the ON/OFF settings are configured according to the values stored in the specified data registers.
The dates for the ON/OFF settings can be changed while the FC6A Series MICROSmart is running.

Note: When a special date in the ON/OFF settings is modified with the YEAR instruction input turned on, it is not reflected in the output operation until the current date and time match the changed special date.
To immediately reflect this in the output operation, turn the YEAR instruction off and then on again after changing the special date.

- **Devices tab**



☑ Data register settings:

	S1 (First DR)	S2 (Initialization Input)	D1 (Output)
Tag Name:	D0000	M0000	Q0000
Device Address:	D0000	M0000	Q0000
	D0000 - D0003		
Comment:	P1 : Day of the week (ON and OFF)		

□ Pulse Output

S3 (The number of parameter tabs): 1

Preview:

Year: 2013

```
January      2013     February      2013      March         2013
Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa   Su Mo Tu We Th Fr Sa
30 31  1  2  3  4  5                  1  2                  1  2
 6  7  8  9 10 11 12    3  4  5  6  7  8  9    3  4  5  6  7  8  9
13 14 15 16 17 18 19   10 11 12 13 14 15 16   10 11 12 13 14 15 16
20 21 22 23 24 25 26   17 18 19 20 21 22 23   17 18 19 20 21 22 23
27 28 29 30 31         24 25 26 27 28         24 25 26 27 28 29 30
                                              31
```

1. Data register settings

This setting selects whether to configure YEAR instruction dates as fixed settings or to indirectly specify them with data registers.

To configure the dates by specifying data registers, select this check box.

☑ **Selected check box**

The date settings are indirectly specified by data registers. Configure the dates by using data registers. The dates can be changed while the FC6A Series MICROSmart is running. For allocating the data register region, see "Data Register Allocation" on page 11-23.

The data registers can be initialized with the days of the week and the times configured on the parameter tabs by turning on the initialization input. For initialization, see "3. S2 (source 2): Initialization Input" on page 11-21.

Note: When this check box is not selected, the dates are fixed settings. The dates are configured on the parameter tabs. The dates cannot be changed while the FC6A Series MICROSmart is running. For fixed settings, see "To configure the dates as a fixed setting" on page 11-17.

2. S1 (source 1): First Data Register

This setting specifies the start of the data register region to store the dates for the YEAR instruction.

This setting is only used when indirectly specifying the dates with data registers.

Setting	Description
Tag Name	Specifies the tag name or the device address for the device.
Device Address	Shows the device address that corresponds to the tag name.
Used Data Registers	Shows the range of data registers used to store the settings. This item changes when the device address or the number of parameter tabs changes.
Comment	Shows the comment for the device address. This item can be edited.

For allocating the data register region, see "Data Register Allocation" on page 11-23.

3. S2 (source 2): Initialization Input

This setting specifies the device to initialize the dates stored in the data register region that starts from S1 (source 1).

The values configured on the parameter tabs are stored in the data registers by turning on the initialization input.

This setting is only used when indirectly specifying the settings for the YEAR instruction with data registers.

4. S3 (source 3): The number of parameter tabs

This setting configures the number of parameter tabs.
This setting is shared in common with "To configure the dates as a fixed setting". See "4. S3 (source 3): Number of parameter tabs" on page 11-17.

5. D1 (destination 1): Output

This setting configures the output for the results when the configured dates and the current date are compared.
This setting is shared in common with "To configure the dates as a fixed setting". See "5. D1 (destination 1): Output" on page 11-18.

6. Pulse Output

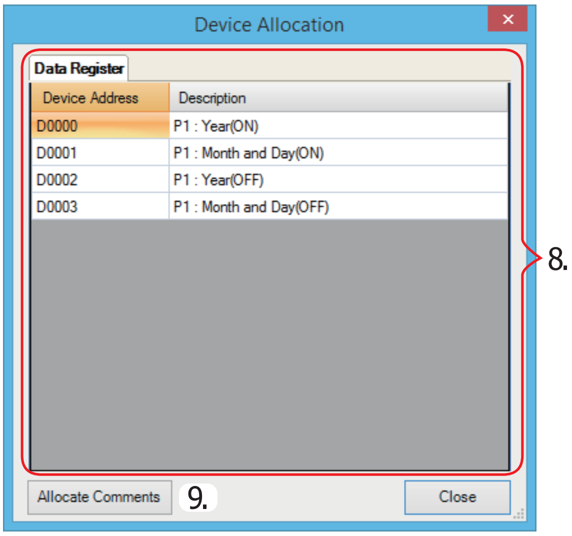
This setting configures the operation for D1 (output). This setting is applied to all parameter tabs.

This setting is shared in common with "To configure the dates as a fixed setting". See "6. Pulse Output" on page 11-18.

7. Data Register Allocation

Click this button to display the **Device Allocation** dialog box. As shown below, a table of the data registers and their corresponding YEAR instruction settings is displayed on the dialog box (8). Click **Allocate Comments** (9) and you can configure the comments for the data registers that correspond to the names of the settings.
This button is only used when indirectly specifying the settings for the YEAR instruction with data registers.

- **Device Allocation dialog box**



Device Address	Description
D0000	P1 : Year(ON)
D0001	P1 : Month and Day(ON)
D0002	P1 : Year(OFF)
D0003	P1 : Month and Day(OFF)

- **Parameter tab**

This tab configures the settings for the output. A maximum of 20 parameter tabs can be configured for 1 YEAR instruction.
If indirectly specifying the settings for the YEAR instruction with data registers, the settings configured on the parameter tabs are stored in the data registers when the initialization input is turned on.

This setting is shared in common with "To configure the dates as a fixed setting". See "Parameter tab" on page 11-18.

- **Preview**

The preview shows the ON/OFF state for the output based on the settings configured on the parameter tabs in a calendar.
This function is shared in common with "To configure the dates as a fixed setting". See "Preview" on page 11-20.

Data Register Allocation

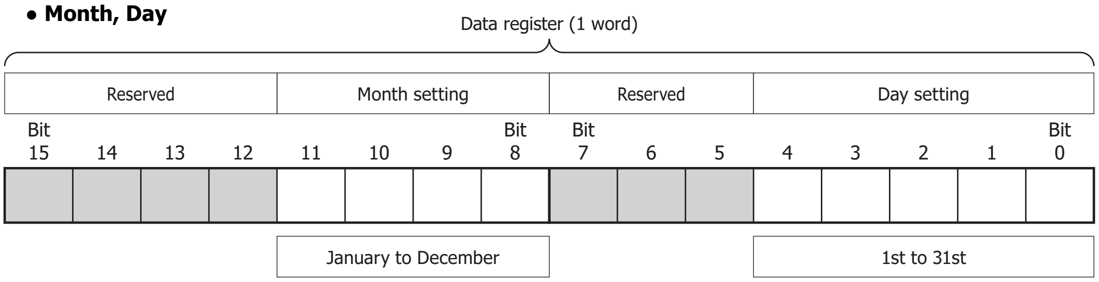
The settings configured on the **Parameter** tabs are allocated to the data registers as follows.

Storage destination	Data size (word)	R (Read)/W (Write)	Setting		
Start address+0	1	R/W	P 1 tab	ON setting	Year
Start address+1	1	R/W			Month, Day or Day of the week
Start address+2	1	R/W		OFF setting	Year
Start address+3	1	R/W			Month, Day or Day of the week (number of days when specified to remain on)
Start address+4	1	R/W	P 2 tab	ON setting	Year
Start address+5	1	R/W			Month, Day or Day of the week
Start address+6	1	R/W		OFF setting	Year
Start address+7	1	R/W			Month, Day or Day of the week (number of days when specified to remain on)
• • •	• • •	• • •	• • •		
Start address+76	1	R/W	P 20 tab	ON setting	Year
Start address+77	1	R/W			Month, Day or Day of the week
Start address+78	1	R/W		OFF setting	Year
Start address+79	1	R/W			Month, Day or Day of the week (number of days when specified to remain on)

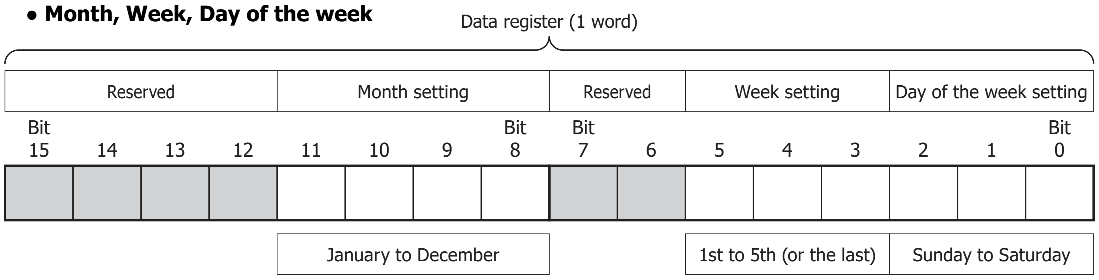
- **Month, Day or Day of the week data register allocation**

The Month, Day or Day of the week are allocated as bits in 1 data register as follows.

- **Month, Day**



- **Month, Week, Day of the week**



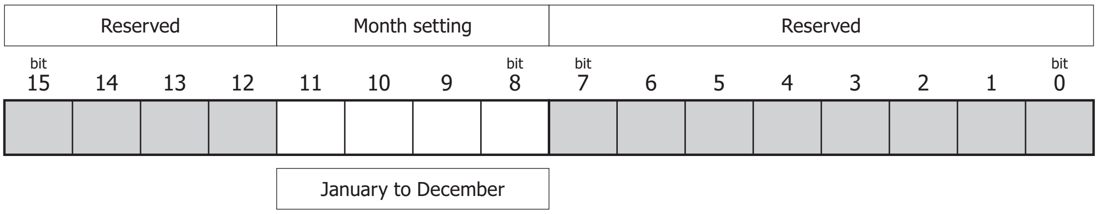
Week setting assignments

(Binary)		(Decimal)	
001000	=	8	: 1st
010000	=	16	: 2nd
011000	=	24	: 3rd
100000	=	32	: 4th
101000	=	40	: 5th
110000	=	48	: Last

Day of the week assignments

(Binary)		(Decimal)	
000	=	0	: Sunday
001	=	1	: Monday
010	=	2	: Tuesday
011	=	3	: Wednesday
100	=	4	: Thursday
101	=	5	: Friday
110	=	6	: Saturday

- **Month, End of month**



Example day of the week settings

[When configured to turn on the output on January 1st]

January 1st

Reserved				Month setting				Reserved			Day setting				
Bit 15	14	13	12	11	10	9	Bit 8	Bit 7	6	5	4	3	2	1	Bit 0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1

Month setting: 0001 = 1
Day setting: 00001 = 1
The value of the data register is 100000001 (binary) = 257 (decimal).

[When configured to turn on the output on December 31st]

December 31st

Reserved				Month setting				Reserved			Day setting				
Bit 15	14	13	12	11	10	9	Bit 8	Bit 7	6	5	4	3	2	1	Bit 0
0	0	0	0	1	1	0	0	0	0	0	1	1	1	1	1

Month setting: 1100 = 12
Day setting: 11111 = 31
The value of the data register is 110000011111 (binary) = 3103 (decimal).

[When configured to turn on the output on the 1st Monday of January]

1st Monday of January

Reserved				Month setting				Reserved		Week setting			Day of the week setting		
Bit 15	14	13	12	11	10	9	Bit 8	Bit 7	6	5	4	3	2	1	Bit 0
0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1

Month setting: 0001 = 1 (January)
Week setting: 001 = 1 (1st)
Day of the week setting: 001 = 1 (Monday)
The value of the data register is 100001001 (binary) = 265 (decimal).

[When configured to turn on the output on the 4th Thursday of June]

4th Thursday of June

Reserved				Month setting				Reserved		Week setting			Day of the week setting		
Bit 15	14	13	12	11	10	9	Bit 8	Bit 7	6	5	4	3	2	1	Bit 0
0	0	0	0	0	1	1	0	0	0	1	0	0	1	0	0

Month setting: 0110 = 6 (June)
Week setting: 100 = 4 (4th)
Day of the week setting: 100 = 4 (Thursday)
The value of the data register is 11000100100 (binary) = 1572 (decimal).

[When configured to turn on the output on the Final]

Final

Reserved				Month setting				Reserved		Week setting			Day of the week setting		
Bit 15	14	13	12	11	10	9	Bit 8	Bit 7	6	5	4	3	2	1	Bit 0
0	0	0	0	0	1	1	0	0	1	1	0	0	1	0	0

Month setting: 0110 = 6 (June)
Week setting: 110 = 6 (final)
Day of the week setting: 100 = 4 (Thursday)
The value of the data register is 11000110100 (binary) = 1588 (decimal).

Timing Chart when the Input Turns On during the Configured Interval

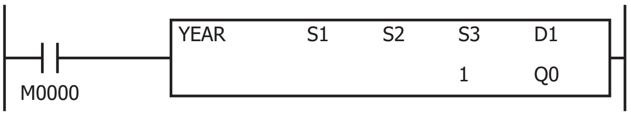
When the input turns on or off during the interval between the ON settings and the OFF settings, and when the input turns on or off after 0:00 on the date configured by the ON settings when pulse output is enabled, the timing chart is as follows.

- **When pulse output is disabled**

When the input is turned on, the current date and the ON and OFF settings are compared and the output turns on or off.

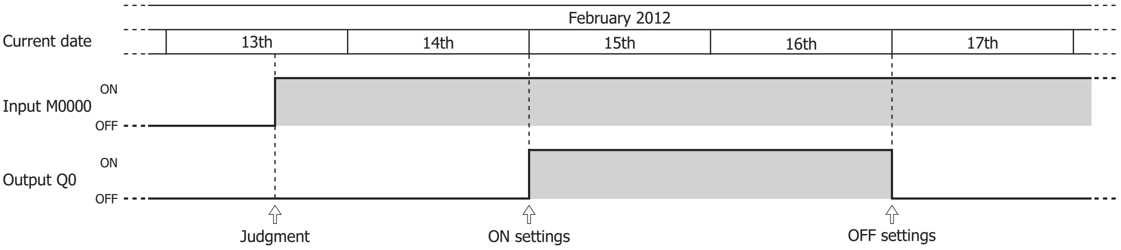
Settings

ON settings	February 15, 2012
OFF settings	February 17, 2012
Output port	Q0



[When the input is on before the day specified by the ON settings]

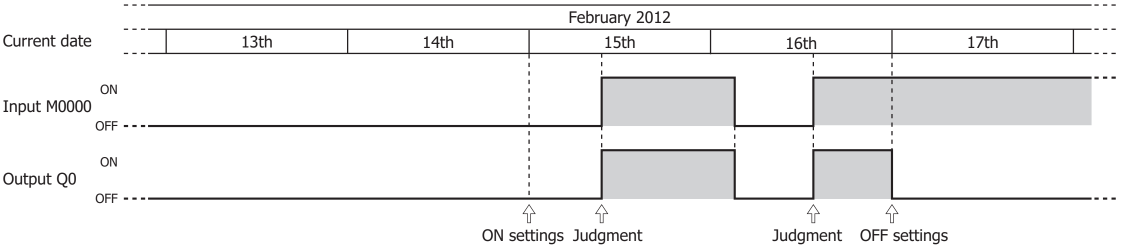
On February 13, 2012, when the input is turned on, the result of the current date compared with the ON and OFF settings is not within the ON and OFF settings (February 15 to February 17, 2012). Therefore, the output remains off.



[When the input is turned on or off during the interval between the ON and OFF settings]

On February 15, 2012, when the input is turned on, the result of the current date compared with the ON and OFF settings is within the ON and OFF settings (February 15 to February 17, 2012). Therefore, the output turns on. The output turns off when the input is turned off.

On February 16, 2012, when the input is turned on, the result of the current date compared with the ON and OFF settings is that the output is turned on.

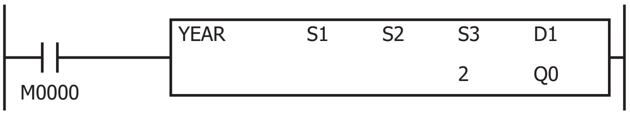


• When pulse output is enabled

The input is determined to be on or off at 0:00 on the date of the ON settings and the output is turned on. The current date is not compared with the ON settings when the input is turned on.

Settings

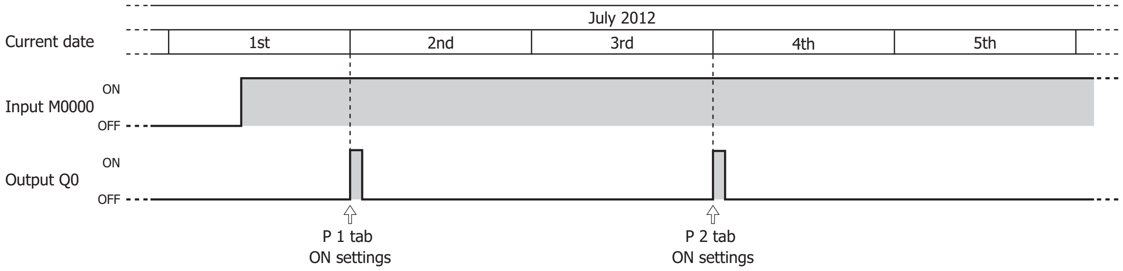
P 1 tab ON settings	July 2, 2012
P 2 tab ON settings	July 4, 2012
Output port	Q0



[When the input turns on before the day specified by the ON settings]

The result of the input being determined at 0:00 on July 2, 2012 is to turn on the output for one scan.

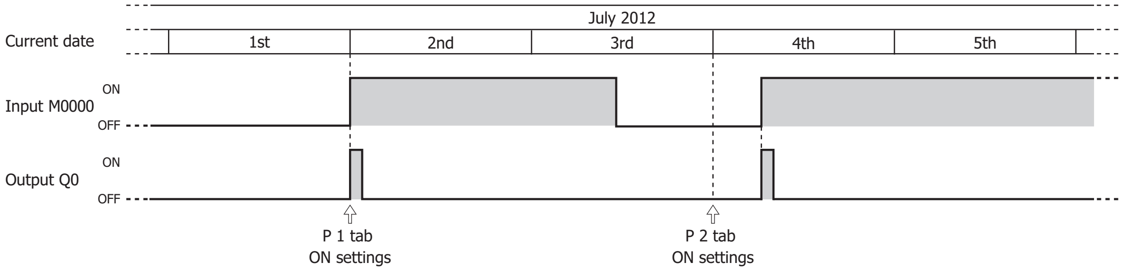
The result of the input being determined at 0:00 on July 4, 2012 is to turn on the output for one scan.



[When the input turns on exactly on the day specified by the ON settings]

When the input turns on at 0:00 on July 2, 2012, the output turns on for one scan.

When the input turns on after 0:00 on July 4, 2012, the output turns on for one scan.

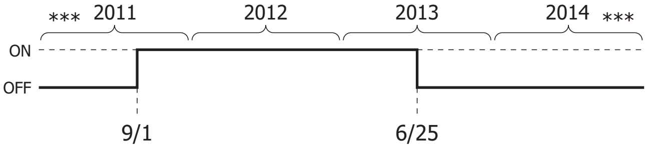


Examples: YEAR

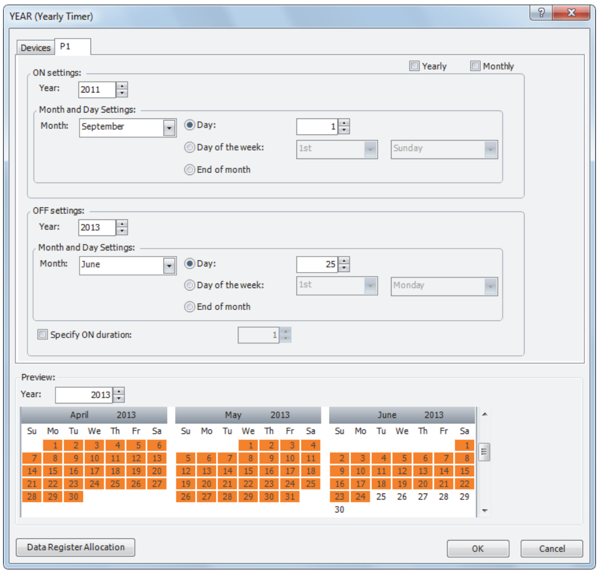
- **To configure the dates as fixed settings**

[To turn on Q0 from 0:00 on September 1, 2011, to 0:00 on June 25, 2013]

ON date : 2011/09/01
OFF date : 2013/06/25
Yearly : OFF
Monthly : OFF



Parameter tab



YEAR (Yearly Timer)

Devices | P1

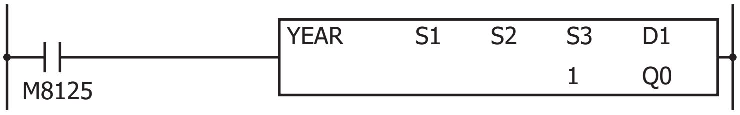
☐ Yearly ☐ Monthly

ON settings:
Year: 2011
Month and Day Settings:
Month: September
● Day: 1
○ Day of the week: 1st Sunday
○ End of month

OFF settings:
Year: 2013
Month and Day Settings:
Month: June
● Day: 25
○ Day of the week: 1st Monday
○ End of month
☐ Specify ON duration: 1

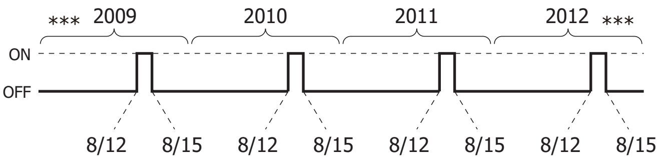
Preview:
Year: 2013

Configure the tab as shown above and set **D1** to Q0.

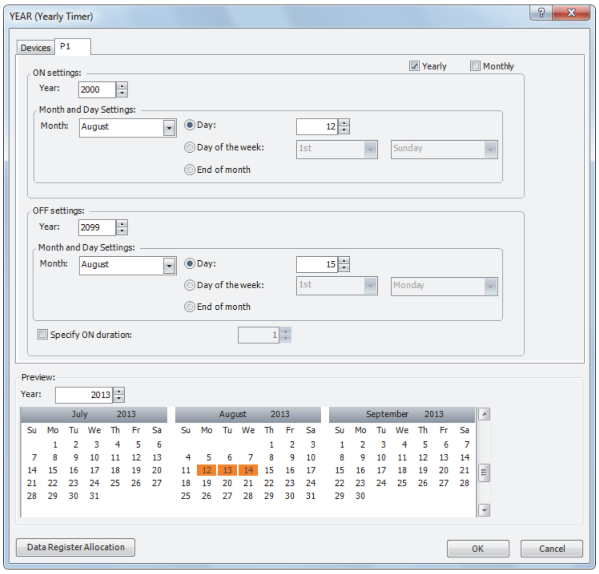


[To turn on output Q0 from 0:00 on August 12 to 0:00 on August 15 every year]

ON date : 2000/08/12
OFF date : 2099/08/15
Yearly : ON
Monthly : OFF



Parameter tab



YEAR (Yearly Timer)

Devices | P1

☑ Yearly ☐ Monthly

ON settings:
Year: 2000
Month and Day Settings:
Month: August
● Day: 12
○ Day of the week: 1st Sunday
○ End of month

OFF settings:
Year: 2099
Month and Day Settings:
Month: August
● Day: 15
○ Day of the week: 1st Monday
○ End of month

☐ Specify ON duration: 1

Preview:
Year: 2013

July 2013

Su	Mo	Tu	We	Th	Fr	Sa
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

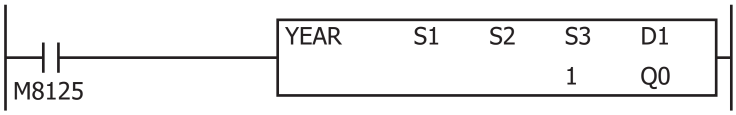
August 2013

Su	Mo	Tu	We	Th	Fr	Sa
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

September 2013

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

Configure the tab as shown above and set **D1** to Q0.



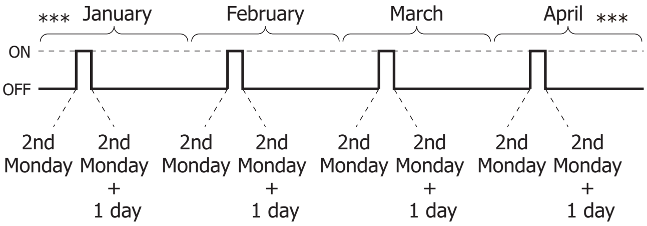
[To turn on output Q0 only on the 2nd Monday of each month from 2000 to 2099]

ON date : 2000/**/2nd Monday

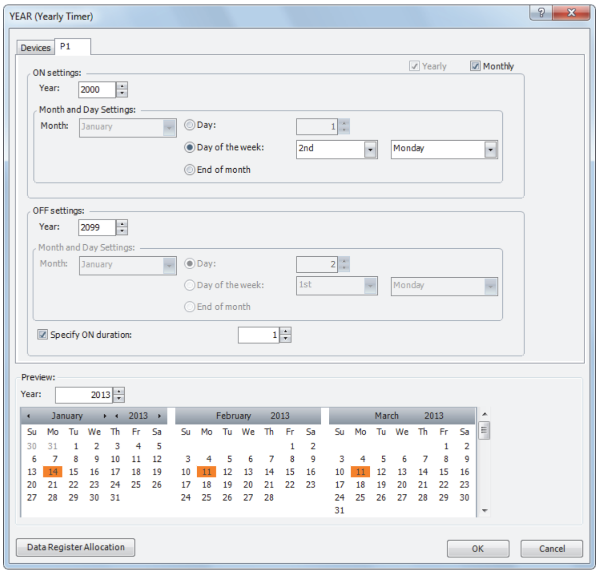
OFF date : 2099/**/2nd Monday + 1 day

Yearly : ON

Monthly : ON



Parameter tab



YEAR (Yearly Timer)

Devices | P1

☑ Yearly ☑ Monthly

ON settings:

Year: 2000

Month and Day Settings:

Month: January

○ Day: 1

● Day of the week: 2nd Monday

○ End of month

OFF settings:

Year: 2099

Month and Day Settings:

Month: January

● Day: 2

○ Day of the week: 1st Monday

○ End of month

☑ Specify ON duration: 1

Preview:

Year: 2013

January 2013

Su	Mo	Tu	We	Th	Fr	Sa
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

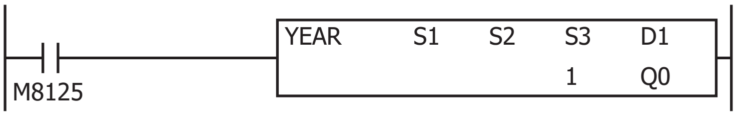
February 2013

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28		

March 2013

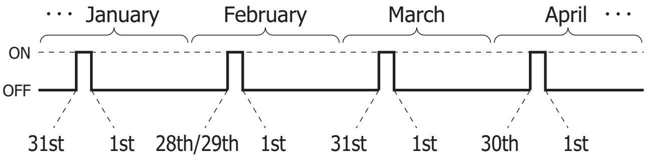
Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Configure the tab as shown above and set **D1** to Q0.

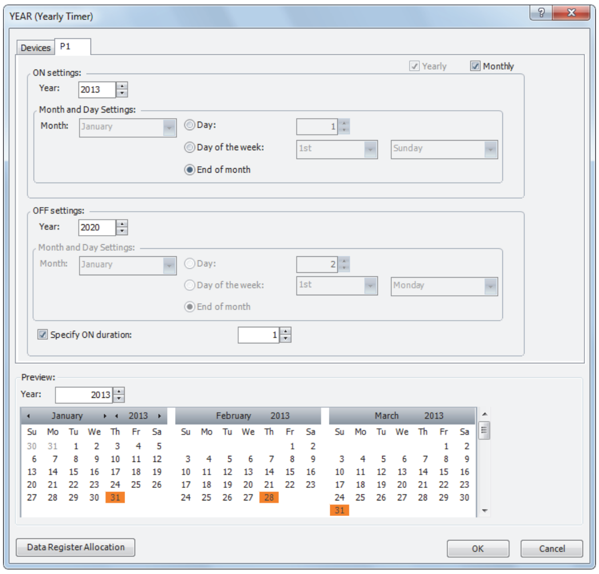


[To turn on output Q0 on the last day of every month between 2013 and 2020]

ON date : 2013/**/end of month
OFF date : 2020/**/end of month
+ 1 day
Yearly : ON
Monthly : ON



Parameter tab



YEAR (Yearly Timer)

Devices | P1

☑ Yearly ☑ Monthly

ON settings:
Year: 2013
Month and Day Settings:
Month: January
○ Day: 1
○ Day of the week: 1st Sunday
● End of month

OFF settings:
Year: 2020
Month and Day Settings:
Month: January
○ Day: 2
○ Day of the week: 1st Monday
● End of month

☑ Specify ON duration: 1

Preview:
Year: 2013

January 2013

Su	Mo	Tu	We	Th	Fr	Sa
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

February 2013

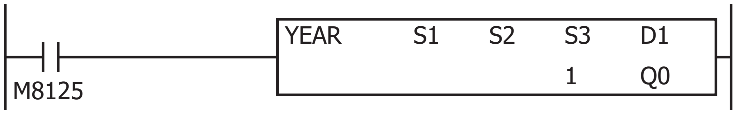
Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28		

March 2013

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Data Register Allocation | OK | Cancel

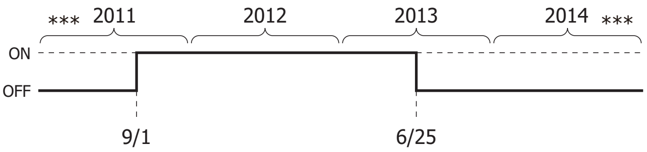
Configure the tab as shown above and set **D1** to Q0.



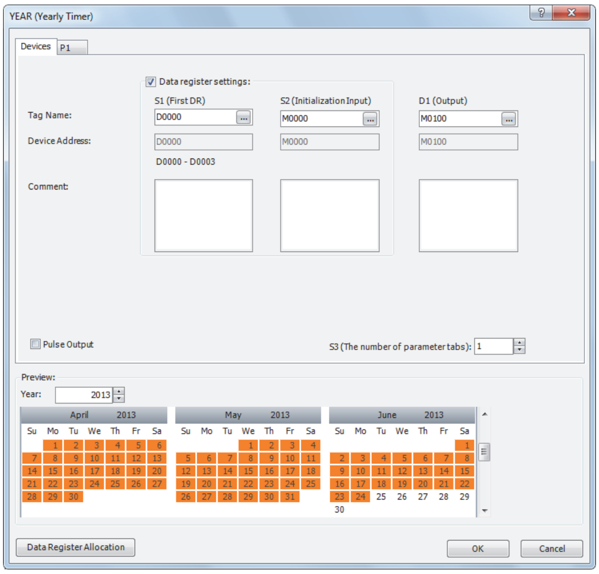
- **To configure the dates by specifying data registers**

[To turn on M100 from 0:00 on September 1, 2011, to 0:00 on June 25, 2013]

ON date : 2011/09/01
OFF date : 2013/06/25
Yearly : OFF
Monthly : OFF



Devices tab


YEAR (Yearly Timer)

Devices | P1

☑ Data register settings:

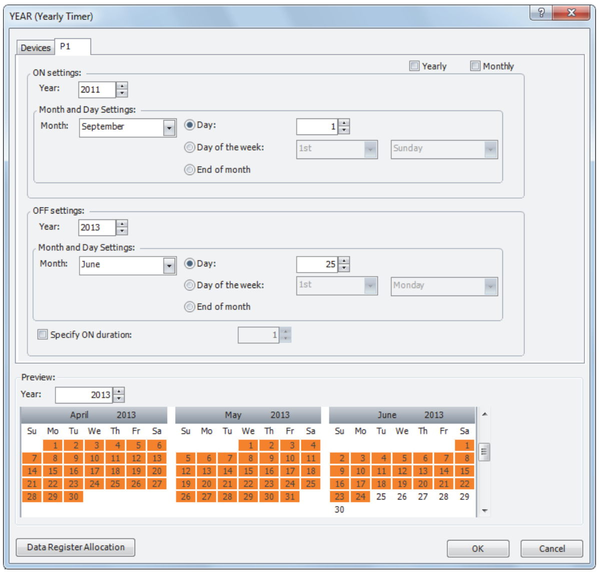
	S1 (First DR)	S2 (Initialization Input)	D1 (Output)
Tag Name:	D0000	M0000	M0100
Device Address:	D0000 D0000 - D0003	M0000	M0100
Comment:			

□ Pulse Output

S3 (The number of parameter tabs): 1

Preview:
Year: 2013

Parameter tab


YEAR (Yearly Timer)

Devices | P1

□ Yearly □ Monthly

ON settings:
Year: 2011
Month and Day Settings:
Month: September
● Day: 1
○ Day of the week: 1st Sunday
○ End of month

OFF settings:
Year: 2013
Month and Day Settings:
Month: June
● Day: 25
○ Day of the week: 1st Monday
○ End of month

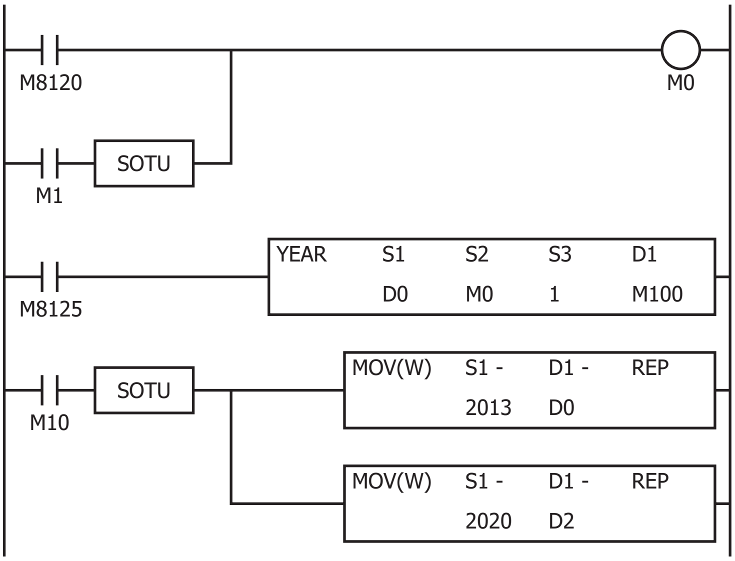
□ Specify ON duration: 1

Preview:
Year: 2013

Data register allocation

The settings on the **P 1** tab are allocated to data registers D0 to D3 as shown in the table below. The settings configured on the **P 1** tab are stored in D0 to D3 by turning on initialization input S2.

Data register	Setting			Initial setting
D0	P 1 tab	ON settings	Year	2011
D1			Month, Day	2305 (September 1st)
D2		OFF settings	Year	2013
D3			Month, Day	1561 (June 25th)



- The initial settings configured on the **P 1** tab are stored in D0 to D3 at the first scan.
- When M1 turns on, all of the YEAR instruction settings (D0 to D3) return to the initial settings.
- The YEAR instruction starts operating according to the values of data registers D0 to D3.
- When M10 turns on, the ON settings year changes to 2013 (D0) and the OFF settings year changes to 2020 (D2).

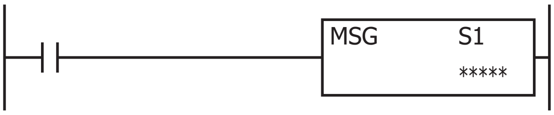
12: Display Instructions

Introduction

This chapter describes the display instructions that display specified data on the HMI module LCD and external devices.

MSG (Message)

Displays the specified data on the HMI module LCD.



When the input is on, a message is displayed on the HMI module LCD according to the content configured in the **MSG (Message)** dialog box.

The MSG instruction functions are as follows.

Device values can be displayed.

- Word device values can be displayed as numeric values according to the specified data type.
  For details, see "Insert Word Device" on page 12-4.
- Word device values can be displayed as bar graphs.
  For details, see "Insert Bar Graph" on page 12-8.
- Text can be displayed and switched according to the value of a bit device (input/output, internal relay, shift register, timer contact, or counter contact).
  For details, see "Insert Bit Device" on page 12-6.

Arbitrary text can be displayed.

- The specified text can be displayed.
  For details, see "MSG (Message) dialog box" on page 12-2.

The text display effects can be configured.

- Text can be scrolled, blinked, or inverted.
  For details, see "Insert Text with Effect" on page 12-7.

Date/time data can be displayed.

- The current date/time and the date/time when the input to the MSG instruction is turned on can be displayed on the LCD.
  For details, see "7. Special Data" on page 12-3.

The language for displayed text can be selected from 9 languages.

- Text can be displayed in 9 languages using 4 types of character sets.
  For details, see "MSG Instruction Common Settings" on page 12-9.

The text display settings can be configured.

- The scroll unit, scroll speed, and blinking speed can be configured.
  For details, see "MSG Instruction Common Settings" on page 12-9.

Device values can be modified.

- Device values displayed with the MSG instruction can be modified on the HMI module.
  For details, see "Modifying Device Values on the HMI Module" on page 12-17.

Notes:

- A maximum of 50 MSG instructions can be entered in a user program, but only one MSG instruction message can be displayed on the HMI module LCD. S1 for the MSG instruction stores the MSG instruction priority. When the display conditions for multiple MSG instructions are satisfied, the messages are displayed according to the priority set for the MSG instruction.
  For MSG instruction priority, see "10. Priority" on page 12-3.
- The MSG instruction cannot be used during an interrupt program. If it is used during an interrupt program, a user program execution error occurs, the execution of the instruction is canceled, and the next instruction is executed.
  For user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Priority	—	—	—	—	—	—	—	—	X	—

Settings

For the MSG instruction settings, there are settings for the individual MSG instruction and settings that are common to all the MSG instructions.

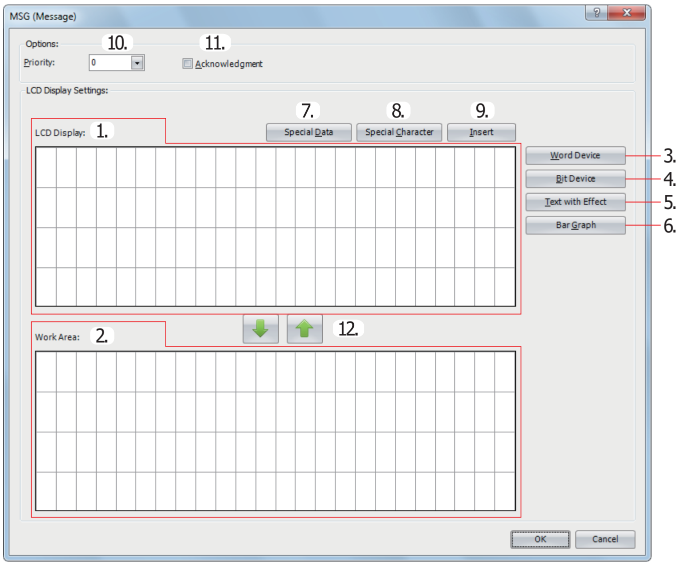
Note: Settings that are common to all the MSG instructions are modified in the **Function Area Settings** dialog box of WindLDR. For details, see "MSG Instruction Common Settings" on page 12-9.

■ MSG Instruction Individual Settings

The individual settings configure items such as the text to display, the devices, and the priority.

The individual settings are configured in the **MSG (Message)** dialog box.

MSG (Message) dialog box



1. LCD Display

Configures the screen that is displayed on the HMI module LCD by placing text and devices in the selected area with the mouse cursor. Enter the text with the keyboard at the cursor position. The character input method (insert/overwrite) is changed by clicking **Insert** (9).

Note: Text entered on the LCD Display (1) with the keyboard cannot be set to scroll, blink, or invert. To scroll, blink, or invert the text, enter the text with **Text with Effect** (5) and configure the display options. For details on inserting text with effects, see "Insert Text with Effect" on page 12-7.

2. Work Area

This area is used when editing the LCD display. This area is for temporarily relocating text and device data.

Text and device data at the cursor position can be moved between the LCD Display and the Work Area with ⬇ or ⬆ (12).

When the dialog is closed, the work area data is discarded.

3. Word Device

Inserts a word device at the cursor position. The value of the specified word device is displayed on the HMI module LCD. For details, see "Insert Word Device" on page 12-4.

4. Bit Device

Inserts a bit device in the area specified by the cursor. Two different items of text can be switched between and displayed on the HMI module LCD according to the value of the specified bit device. For details, see "Insert Bit Device" on page 12-6.

5. Text with Effect

Inserts text in the area specified by the cursor. The specified text is displayed on the HMI module LCD. For details, see "Insert Text with Effect" on page 12-7.

6. Bar Graph

Inserts a bar graph in the area specified by the cursor. The value of the specified device is displayed as a bar graph on the HMI module LCD. For details, see "Insert Bar Graph" on page 12-8.

7. Special Data

Special data such as the current date and time can be entered at the cursor position. Select the data to enter on the special data list window popped up when **Special Data** is pressed. Size of the area used on the LCD display varies based on the selected special data.

Special Data	Display		Occupied Area (Rows x columns)
	Display Type	Display Example (December 1, 2015, 13:30)	
Current date	YYYY/MM/DD	2015/12/01	1 x 10
Current time	HH:MM	13:30	1 x 5
Date the input to the MSG instruction is turned on	YYYY/MM/DD	2015/12/01	1 x 10
Time the input to the MSG instruction is turned on	HH:MM	13:30	1 x 5

8. Special Character

A special character can be entered at the cursor position. Select the character to enter on the special characters list window popped up when **Special Character** is pressed. The special characters that can be used are as follows.

Special Characters List							
▼	▲	◀	▶	°C	°F	°	±

9. Insert/Overwrite

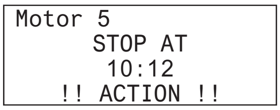
Selects insert or overwrite as the character input mode. Click this button to switch the input mode.

10. Priority

The priority of the MSG instruction can be configured between 0 to 49. 0 is the highest priority and 49 is the lowest priority.

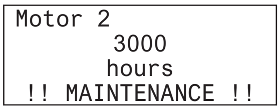
- The same priority cannot be set for multiple MSG instructions.
- When inputs to two or more MSG instructions turns on, the message of the MSG instruction with the highest priority is displayed, out of all the MSG instructions with inputs that are on.
- When the MSG instruction input with the highest priority changes from on to off, the message for the MSG instruction with the next highest priority is displayed (the priority is checked when the input changes).

Note: When you press the ⇧ (up) or ⇩ (down) button on the HMI module, the messages are switched between the MSG instructions that have inputs that are on. Messages are also switched when the acknowledgement for the MSG instruction is enabled.



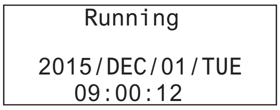
```
Motor 5
     STOP AT
      10:12
  !! ACTION !!
```

Example: Message output in Priority 30



```
Motor 2
       3000
      hours
!! MAINTENANCE !!
```

Example: Message output in Priority 10



```
      Running

  2015/DEC/01/TUE
     09:00:12
```

Date and current time

11. Acknowledgment

When acknowledgment is enabled, the message continues to be displayed even when the MSG instruction input turns off. The message is closed when the OK (OK) button on the HMI module is pressed, and then the message for the MSG instruction with the next highest priority is displayed, out of all the MSG instructions with inputs that are on at that time.

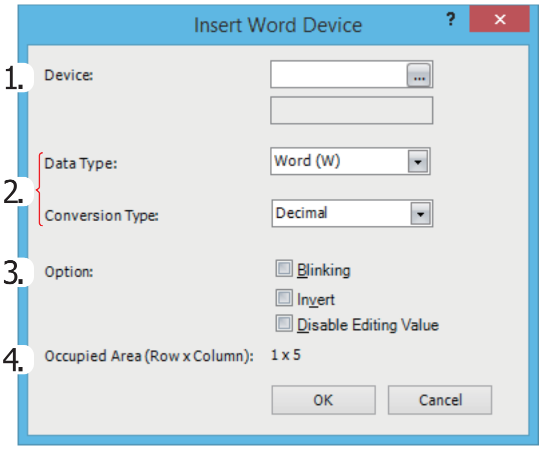
If the input is on for the MSG instruction that had acknowledgment enabled, the message is not closed even when the ⇧ (up) button is pressed.

12. ⬇ / ⬆ buttons

Text and device data at the cursor position are moved between the LCD Display and the Work Area.

Insert Word Device

The value of the specified word device can be displayed on the HMI module LCD.



1. Device

Enter the device to display.

Valid Devices

W (word)	TC, TP, CC, CP, D
I (integer)	D
D (double word)	CC, CP, D
L (long)	D
F (float)	D

2. Data Type and Conversion Type

Select the display type for the specified device. The size of the area used on the LCD display varies based on the specified data type and conversion type.

Data Type	Conversion Type	Occupied Area	Example on LCD
W (word)	Decimal	5	65535
	Hexadecimal	4	FFFF
I (integer)	Decimal	6	-32768
D (double word)	Decimal	10	4294967295
	Hexadecimal	8	FFFFFFFF
L (long)	Decimal	11	-2147483648
F (float)	Decimal	13	1.234567E-01

3. Option

Configure the options for the specified device to blink, invert the text, or allow/prohibit value editing. For the blinking speed, see "MSG Instruction Common Settings" on page 12-9.

Display Options	Description
Blinking	Blinks the value of the specified device.
Invert	Inverts the display of the specified value.
Disable Editing Value	Prohibits editing the specified device value using the HMI module.

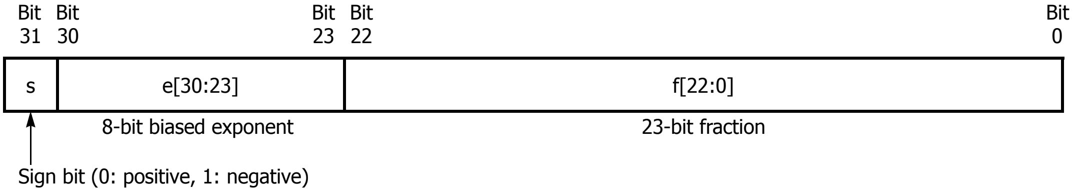
4. Occupied Area

Shows the size of the area to be used on the LCD display (rows: 1, columns: 4 to 13). The area is determined by the selected data type and conversion type.

Floating Point Value Notation on LCD

Whether or not F (float) LCD notation uses decimal notation or exponent notation is determined according to the single-precision floating point definition in IEEE 754 and the internal system function included in the CPU module.

The IEEE 754 single precision format consists of three fields: a 23-bit fraction, f; an 8-bit biased exponent, e; and 1-bit sign, s. These fields are stored contiguously in one 32-bit word, as shown in the figure below. Bits 0:22 contain the 23-bit fraction, f, with bit 0 being the least significant bit of the fraction and bit 22 being the most significant; bits 23:30 contain the 8-bit biased exponent, e, with bit 23 being the least significant bit of the biased exponent and bit 30 being the most significant; and the highest-order bit 31 contains the sign bit, s.



The table below shows the correspondence between the values of the three constituent fields s, e, and f and the value represented by the single precision format bit pattern.

When the value is ±0, denormalized number, or normalized number, the value is displayed according to the internal system function of the CPU module.

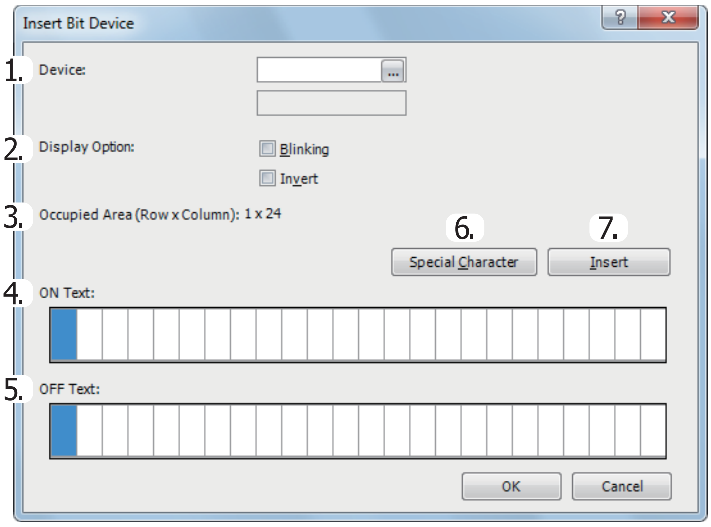
Value	Exponent Field e	Fraction Field f	Representation
±0	e=0	f=0	0
Denormal numbers	e=0	f≈0	• Displayed in decimal notation 0.0001 ≤ | Value | < 9,999,999.5 (Number of significant digits is 7 digits, 8th digit is rounded) • Displayed in exponent notation | Value | < 0.0001 | Value | ≥ 9,999,999.5
Normal numbers	0<e<255	Arbitrary	
±∞ (± infinities)	e=255	f=0	INF
NaNs (Not a number)		f≈0	NAN

Example: This table shows the representation on the LCD for the following floating-point values.

Value	Representation in LCD
1234567	1234567
12345678	1.234568E+07
1234567.8	1234568
0.0001	0.0001
0.00001	1E-05
0.000001	1E-06
0.0000001	1E-07
0.123456	0.123456
0.1234567	0.1234567
0.12345678	0.1234568
0.0000012	1.2E-06
1.2345678	1.234568
0	0
0.0001234568	0.0001234568

Insert Bit Device

Two different items of text can be switched between and displayed on the HMI module LCD according to the value of the specified bit device (when on/when off).



1. Device

Enter the device to display.

Valid Devices

I	Q	M	R	T	C	D	P
X	X	X	X	X	X	—	—

2. Display Option

Configure the options to blink or invert the text. For the blinking speed, see "MSG Instruction Common Settings" on page 12-9.

Display Option	Description
Blinking	Blinks the specified text.
Invert	Inverts the display of the specified text.

3. Occupied Area

Shows the size of the area to be used on the LCD display (rows: 1, columns: 1 to 24). The display size is determined by the selected range on the LCD display area (or work area). If multiple rows are selected, the area at the top line in the selected range is used.

4. ON Text

Enter the text to display when the specified device is on. Up to 24 single-byte characters can be entered. A space is also counted as one character.

5. OFF Text

Enter the text to display when the specified device is off. Up to 24 single-byte characters can be entered. A space is also counted as one character.

6. Special Character

Enters a special character at the cursor position. Select the character to enter on the special characters list window popped up when **Special Character** is pressed. For the special characters, see "8. Special Character" on page 12-3.

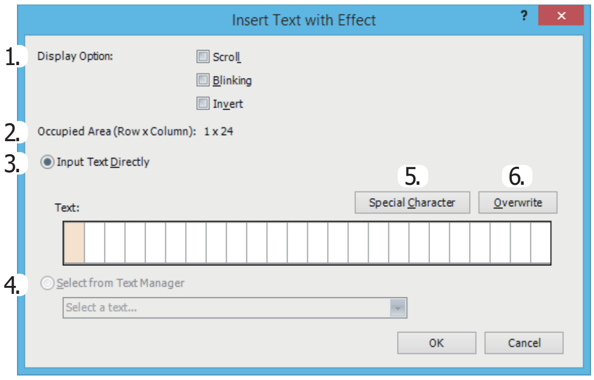
7. Insert/Overwrite

Selects whether to insert or overwrite characters when entering new characters.

Insert Text with Effect

The specified text can be displayed on the HMI module LCD.

[When scrolling is disabled]



Display Option: □ Scroll □ Blinking □ Invert

Occupied Area (Row x Column): 1 x 24

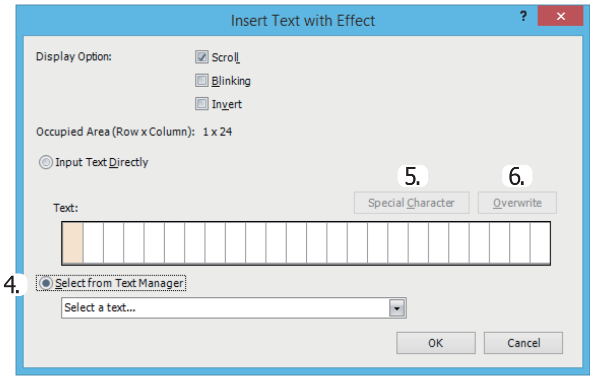
● Input Text Directly

Text:

○ Select from Text Manager

Select a text...

[When scrolling is enabled]



Display Option: ☑ Scroll □ Blinking □ Invert

Occupied Area (Row x Column): 1 x 24

○ Input Text Directly

Text:

● Select from Text Manager

Select a text...

1. Display Option

Configure the options to scroll, blink, or invert the specified text. For the scroll unit, scroll speed, and blinking speed, see "MSG Instruction Common Settings" on page 12-9.

Display Options	Description
Scroll	Scrolls the specified text.
Blinking	Blinks the specified text.
Invert	Invert the display of the specified text.

2. Occupied Area

Shows the size of the area to be used on the LCD display (rows: 1, columns: 1 to 24). The occupied area is determined by the selected range on the LCD display area (or work area). If multiple rows are selected, the area at the top line in the selected range is used.

3. Input Text Directly

Directly enter the characters to display. A space is also counted as one character. If scrolling is disabled, the message can be entered up to 24 single-byte characters. However, you cannot enter a number of characters that exceeds the occupied area. If scrolling is enabled, the message can be entered up to 48 single-byte characters.

4. Select from Text Manager

Select the text to display from the text manager. This can be selected only when the scroll is enabled.

5. Special Character

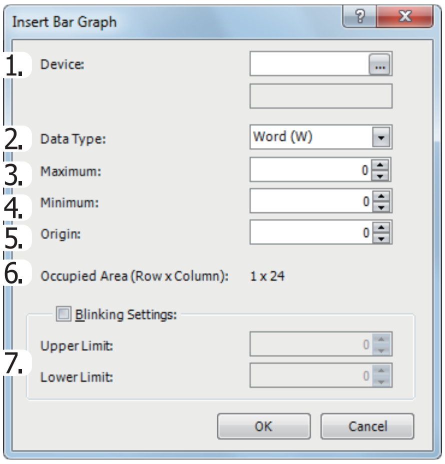
Enters a special character at the cursor position. Select the character to enter on the special characters list window popped up when **Special Character** is pressed. For the special characters, see "8. Special Character" on page 12-3.

6. Insert/Overwrite

Selects whether to insert or overwrite characters when entering new characters.

Insert Bar Graph

The value of the specified device can be displayed as a bar graph on the HMI module LCD.



Insert Bar Graph

1. Device:

2. Data Type: Word (W)

3. Maximum: 0

4. Minimum: 0

5. Origin: 0

6. Occupied Area (Row x Column): 1 x 24

☐ Blinking Settings:

7. Upper Limit: 0

Lower Limit: 0

1. Device

Enter the device to display as a bar graph.

Valid Devices

W (word)	TC, TP, CC, CP, D
I (integer)	D
D (double word)	CC, CP, D
L (long)	D
F (float)	—

2. Data Type

Selects the data type for the specified device.

Data Type

W (word)	X
I (integer)	X
D (double word)	X
L (long)	X
F (float)	—

For data types, see "Data Types for Advanced Instructions" on page 3-7.

3. Maximum

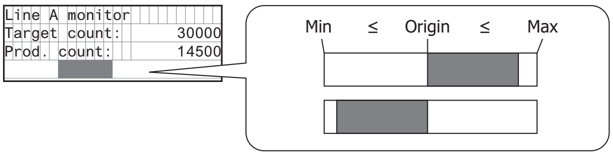
Enter the maximum value for the bar graph. If the device value is larger than the maximum value, the bar graph is displayed as the maximum value. The valid range varies based on the data type. For data types, see "Data Types for Advanced Instructions" on page 3-7.

4. Minimum

Enter the minimum value for the bar graph. If the device value is smaller than the minimum value, the bar graph is displayed as the minimum value. The valid range varies based on the data type. For data types, see "Data Types for Advanced Instructions" on page 3-7.

5. Origin

Enter the value to be the origin of the bar graph. If the device value is larger than the origin value, the bar graph is displayed on the right side of the origin value. If the device value is smaller than the origin value, the bar graph is displayed on the left side of the origin value. The valid range varies based on the data type. For data types, see "Data Types for Advanced Instructions" on page 3-7. The origin value must satisfy the condition of Minimum ≤ Origin ≤ Maximum.



6. Occupied Area

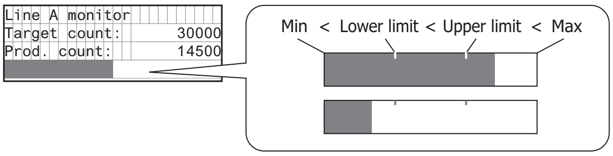
The occupied area is determined by the selected range on the LCD display area (or work area). If multiple rows are selected, the area at the top line in the selected range is used (rows: 1, columns: 1 to 24).

7. Blinking Settings

Blink the bar graph when the value of the specified device exceeds the upper or lower limit.
For the blinking speed, see "MSG Instruction Common Settings" on page 12-9.

Blinking Settings	Description
Upper limit	The bar graph is blinked when the value of the specified device is larger than the upper limit.
Lower limit	The bar graph is blinked when the value of the specified device is smaller than the lower limit.

Maximum, minimum, upper limit, and lower limit values must satisfy the following condition.



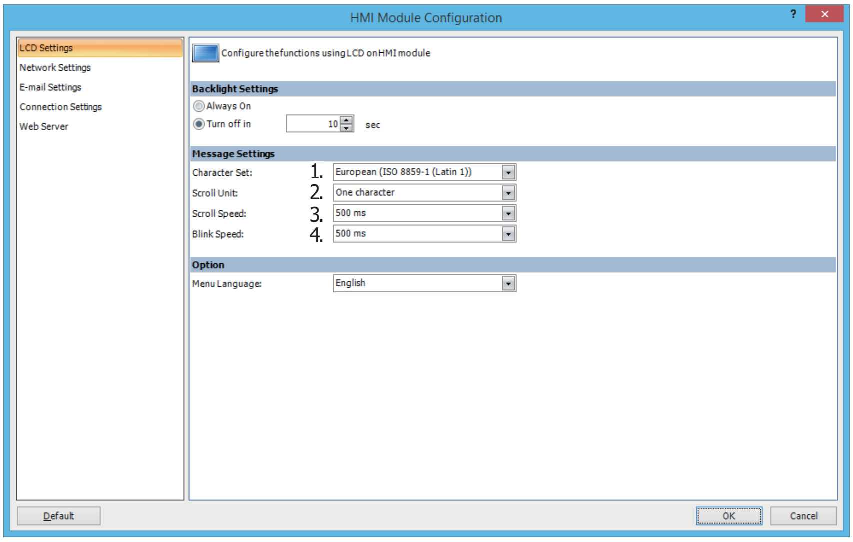
Min < Lower limit < Upper limit < Max

■ MSG Instruction Common Settings

The common settings, character set, scroll unit, scroll speed, and blinking speed, for the message to display can be configured. The common settings are configured in the **LCD** tab of the **HMI Module Configuration** dialog box that is displayed from the WindLDR **Expansion Modules Configuration**.

Note: The common settings for the MSG instructions are applicable for all MSG (message) instructions in the user program. For the MSG instruction individual settings, see "MSG Instruction Individual Settings" on page 12-2.

HMI Module Configuration dialog box LCD tab



Configure the functions using LCD on HMI module

Backlight Settings
○ Always On
● Turn off in 10 sec

Message Settings
Character Set: 1. European (ISO 8859-1 (Latin 1))
Scroll Unit: 2. One character
Scroll Speed: 3. 500 ms
Blink Speed: 4. 500 ms

Option
Menu Language: English

1. Character Set

The character set used for the messages can be configured from the following.

Selection	Character Set	Languages Usable in the MSG Instruction
European	ISO-8859-1 (Latin 1)	Italian, English, Dutch, Spanish, German, French
Japanese	Shift-JIS	Japanese
Chinese	GB2312	Chinese (simplified)
Cyrillic	ANSI 1251	Russian

2. Scroll Unit

The unit to scroll the texts can be configured.

One character	Scrolls the text in 1 character units.
One dot	Scrolls the text in 1 dot units.

3. Scroll Speed

The speed to scroll the texts can be configured. The setting range is 500 to 1,000 ms.

4. Blink Speed

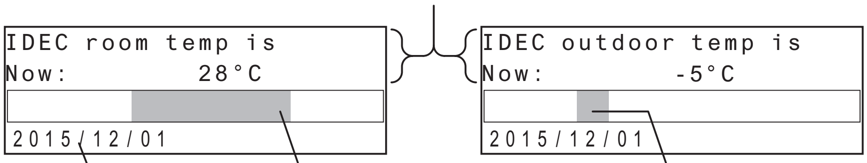
The speed to blink the texts can be configured. The setting range is 500 to 1,000 ms.

Note: The MSG instruction character set can be configured irrespective of the HMI module system menu language.

MSG Instruction Configuration Example

[Display the room temperature when M0000 is on and the outdoor temperature when M0000 is off]

The room temperature/outdoor temperature value is stored in D0002. Create a ladder program that stores the room temperature value when M0000 is on and the outside temperature value when M0000 is off.

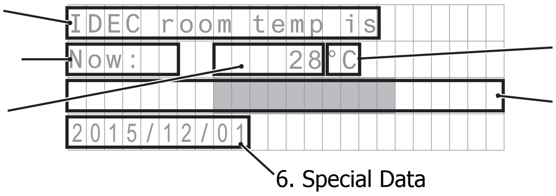


Bar graph is used to Show the rise and fall in temperature visually.

Displays the current date.

■ Settings

Configure the following items.



1. Bit Device
2. Text with Effect
3. Word Device
4. Special Character
5. Bar Graph
6. Special Data

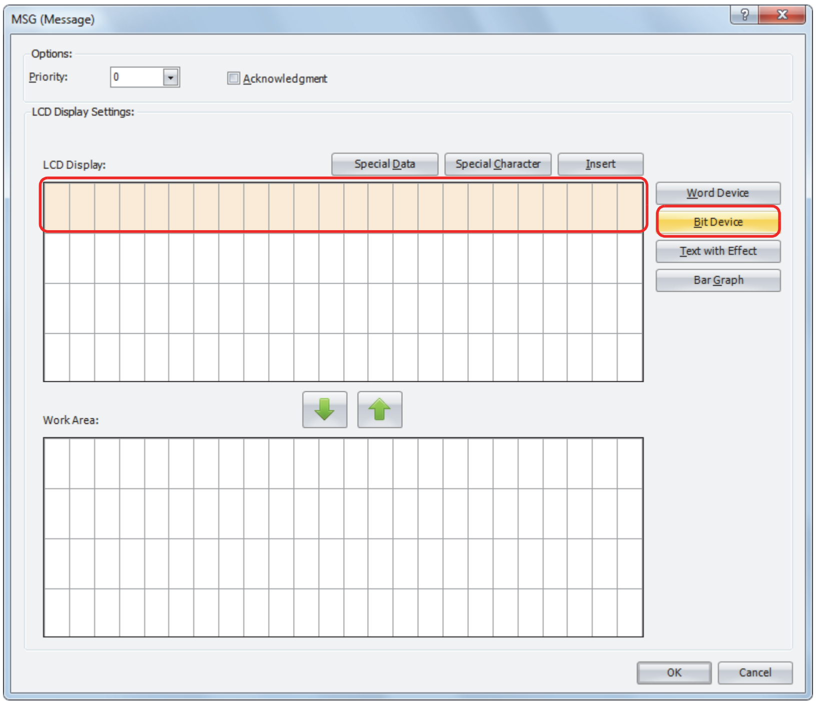
Setting Items		Setting Details
1. Bit Device	Device	M0000
	Display Options	All disabled (scroll, blink, invert)
	ON Text	IDEC room temp is
	OFF Text	IDEC outdoor temp is
2. Text with Effect	Text	Now:
	Display Options	All disabled (scroll, blink, invert)
3. Word Device	Device	D0002
	Data Type	I (integer)
	Conversion Type	Decimal
4. Special Character		°C
5. Bar Graph	Device	D0002
	Data Type	I (integer)
	Max	50
	Min	-20
	Origin	0
	Blinking Settings	Disabled
6. Special Data		Current date

■ Operation Procedure

1. Right-click at the location to insert the MSG instruction, and click **Advanced Instructions** > **Display** > **MSG (Message)**. The **MSG (Message)** dialog box is displayed.

Configuring the bit device

2. Select the area to insert the parameter and click **Bit Device**.



MSG (Message)

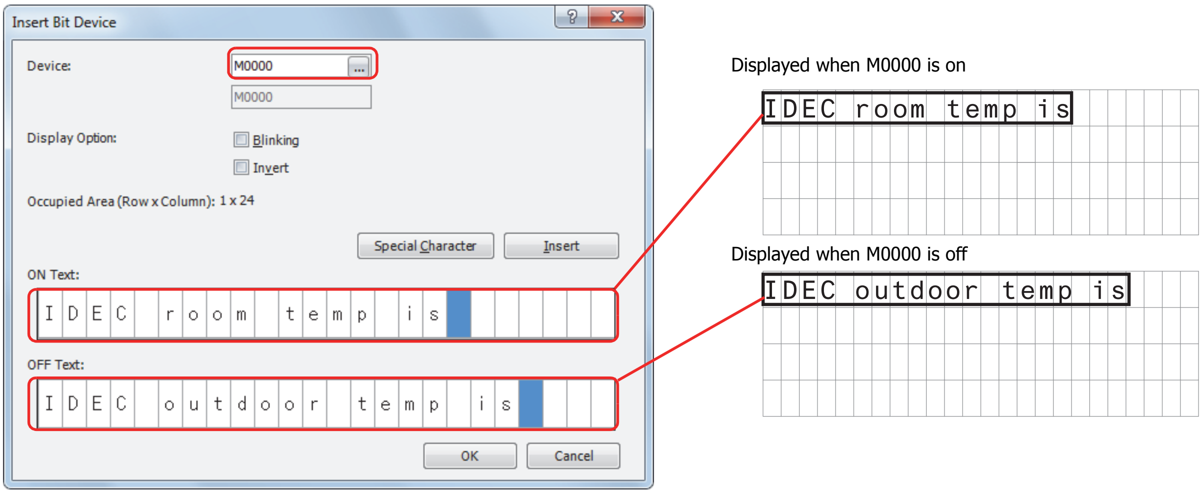
Options:

Priority: 0

□ Acknowledgment

The **Insert Bit Device** dialog box opens.

3. Configure the parameters so that the room temperature is displayed when M0000 is on and the outdoor temperature is displayed when M0000 is off. Set **Device** to "M0000". Using the keyboard, enter "IDEC room temp is" as the ON Text and "IDEC outdoor temp is" as the OFF Text. Under **Display Option**, disable all of the options.



Insert Bit Device

Device: M0000

M0000

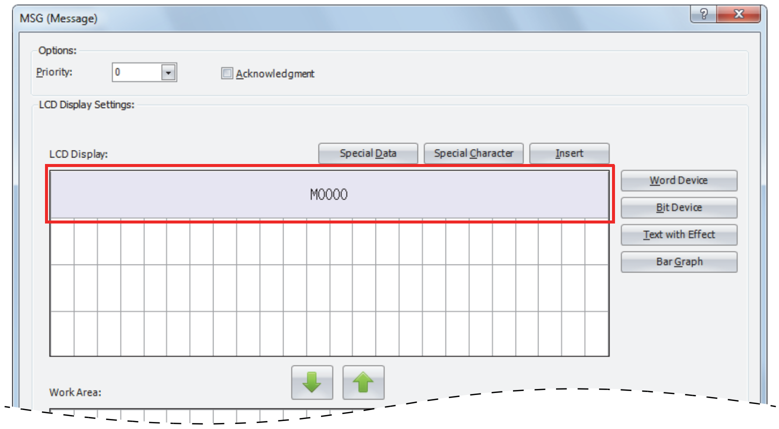
Display Option: □ Blinking □ Invert

Occupied Area (Row x Column): 1 x 24

ON Text: IDEC room temp is

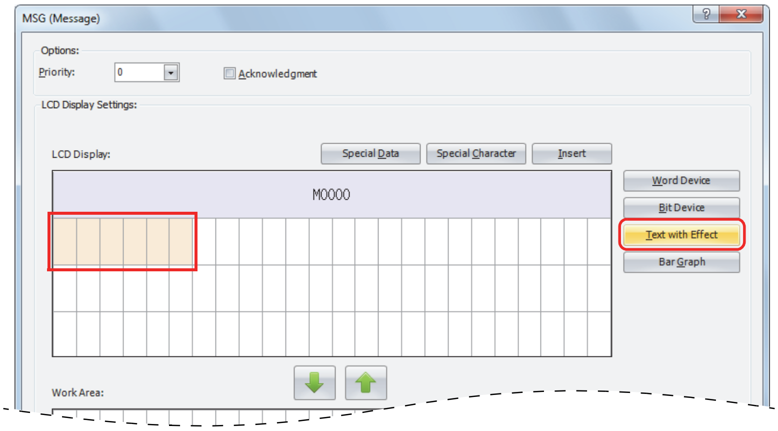
OFF Text: IDEC outdoor temp is

4. After the settings are configured, click **OK**.
   The configured content is displayed on the LCD display area.



Configuring the text with effect

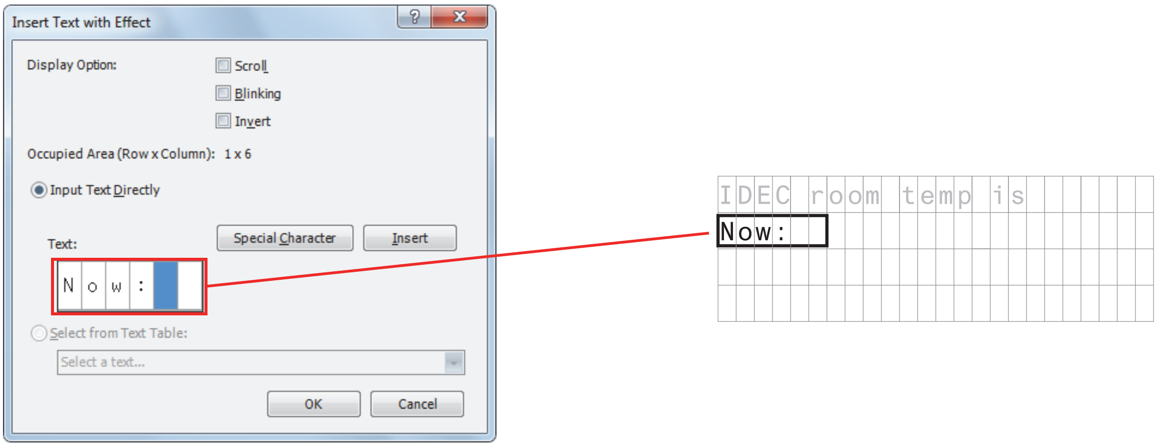
5. Select the six-column area from the start of the second row and click **Text with Effect**.



Note: The text can also be entered directly on the LCD display area. In the sample above, move the cursor to the start of the second line and enter "Now:" with the keyboard. When you directly enter the text, proceed to "Configuring the word device" on page 12-13, step 8.

The **Insert Text with Effect** dialog box is displayed.

6. In Text, enter "Now:" with the keyboard. Under **Display Option**, disable all of the options.


Insert Text with Effect

Display Option: □ Scroll □ Blinking □ Invert

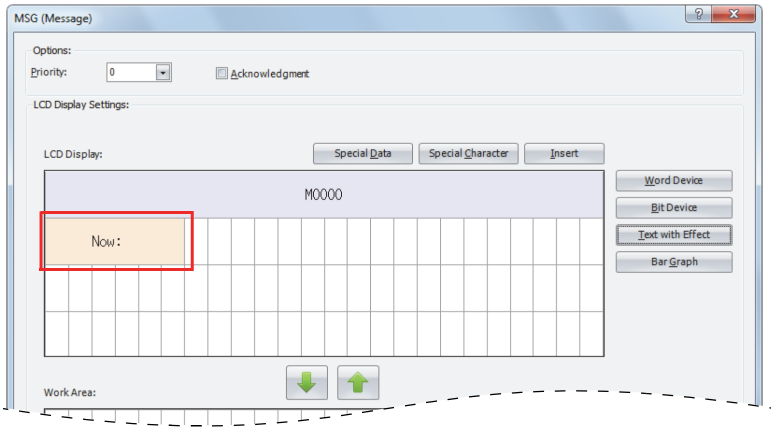
Occupied Area (Row x Column): 1 x 6

● Input Text Directly

Text: N o w :

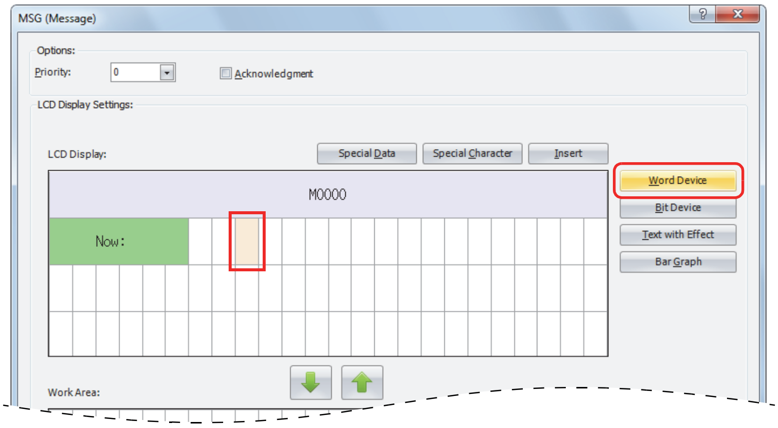
○ Select from Text Table: Select a text...

7. After the settings are configured, click **OK**.
The configured content is displayed on the LCD display area.



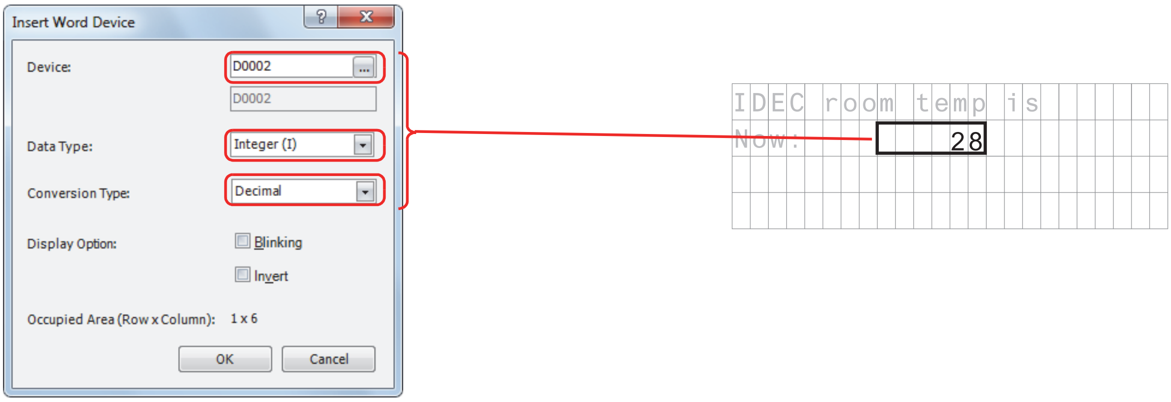
Configuring the word device

8. Select the area at the ninth column on the second row and click **Word Device**.



The **Insert Word Device** dialog box is displayed.

9. Set **Device** to "D0002", **Data Type** to "Integer (I)", and **Conversion Type** to "Decimal".
Under **Display Option**, disable all of the options.


Insert Word Device

Device: D0002

D0002

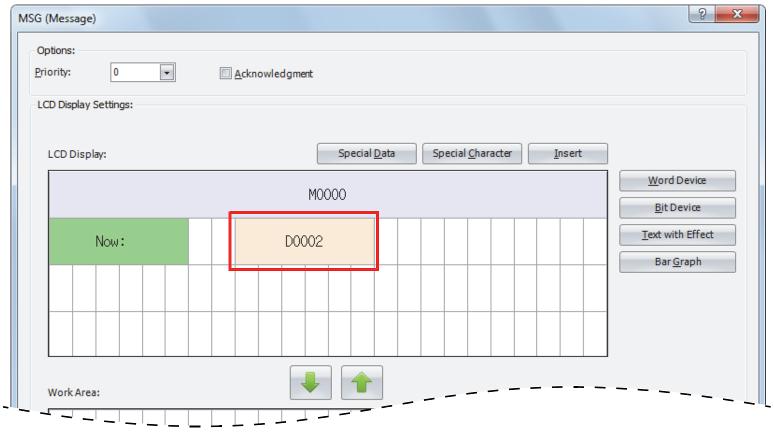
Data Type: Integer (I)

Conversion Type: Decimal

Display Option: □ Blinking □ Invert

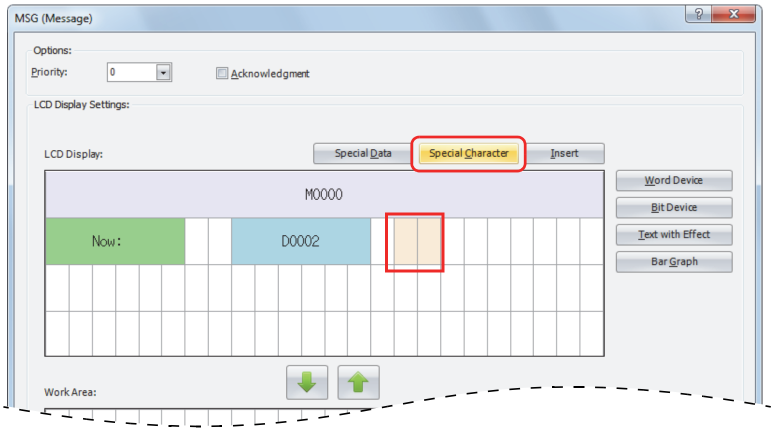
Occupied Area (Row x Column): 1 x 6

10. After the settings are configured, click **OK**.
The configured content is displayed on the LCD display area.



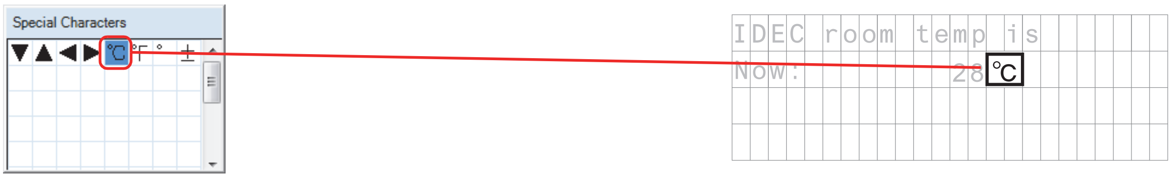
Configuring the special character

11. Select the area at the 15th column on the second row and click **Special Character**.

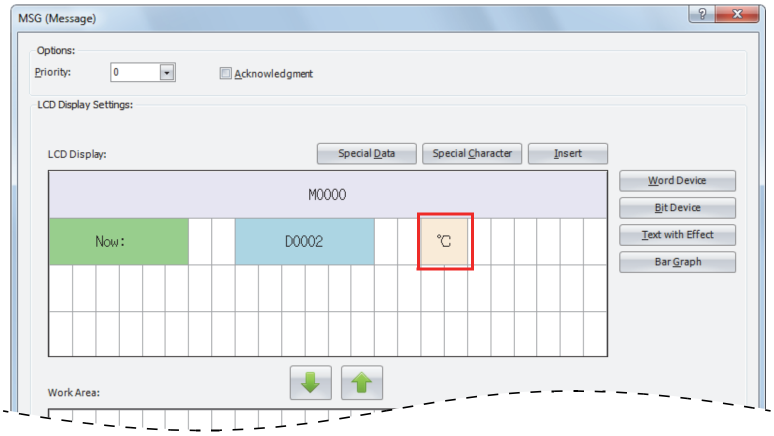


The **Special Characters** popup window is displayed.

12. Double-click on **°C**.

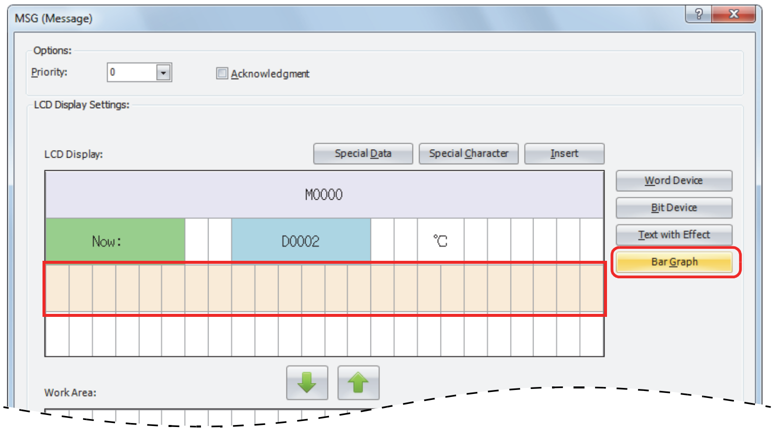


The configured content is displayed on the LCD display area.



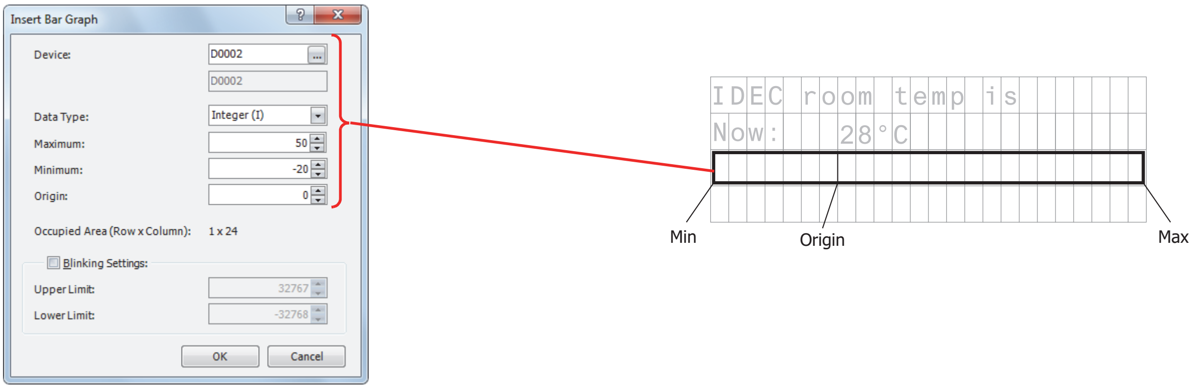
Configuring the bar graph

13. Select the entire area on the third line and click **Bar Graph**.



The **Insert Bar Graph** popup window is displayed.

14. Set **Device** to "D0002", **Data Type** to "Integer (I)", **Max** to "50", **Min** to "-20", and **Origin** to "0". Disable the blinking settings.


Insert Bar Graph

Device: D0002

D0002

Data Type: Integer (I)

Maximum: 50

Minimum: -20

Origin: 0

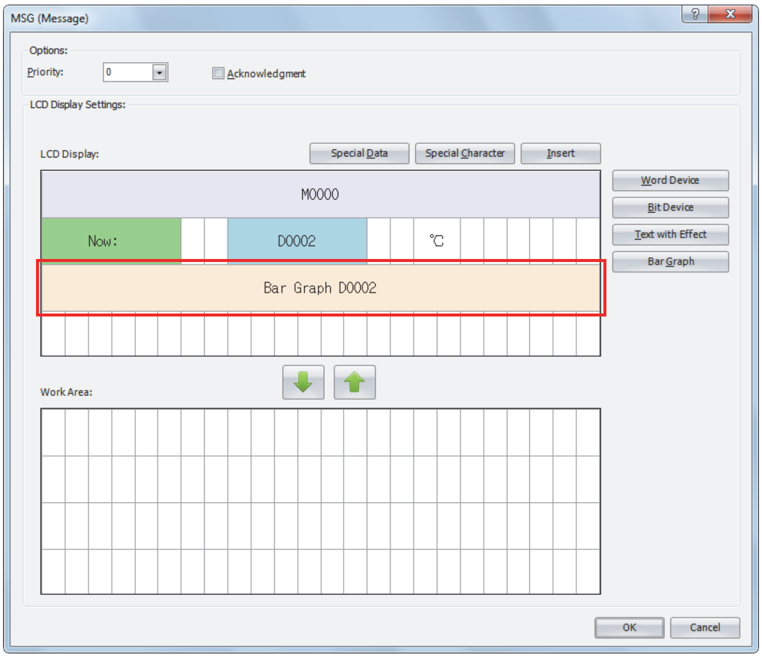
Occupied Area (Row x Column): 1 x 24

□ Blinking Settings:

Upper Limit: 32767

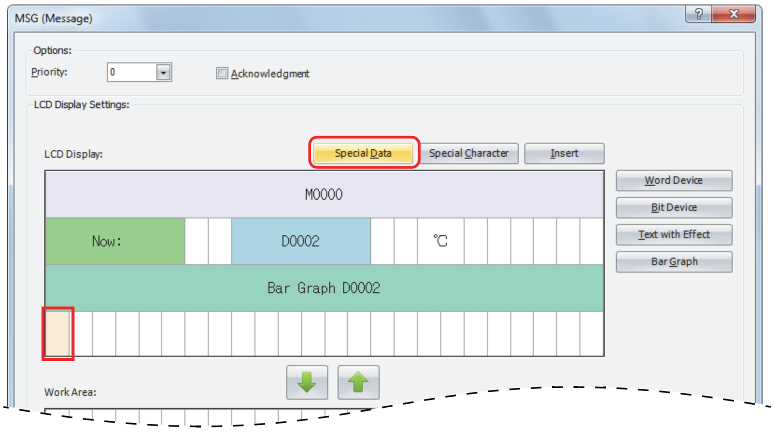
Lower Limit: -32768

15. After the settings are configured, click **OK**.
The configured content is displayed on the LCD screen area.



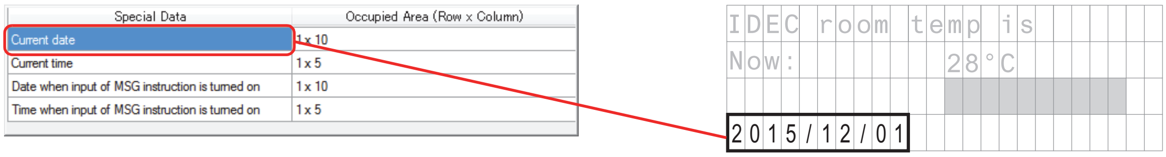
Configuring the special data

16. Select the left edge of the fourth line and click **Special Data**.



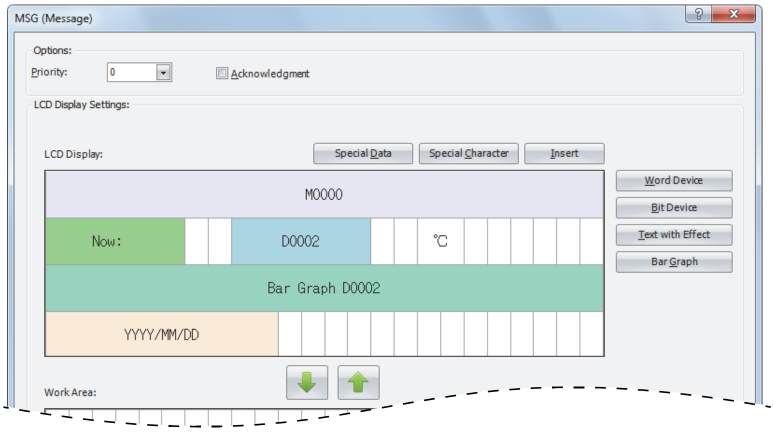
The **Special Data** window is poped up.

17. Double-click on the **Current date**.



Special Data	Occupied Area (Row x Column)
Current date	1 x 10
Current time	1 x 5
Date when input of MSG instruction is turned on	1 x 10
Time when input of MSG instruction is turned on	1 x 5

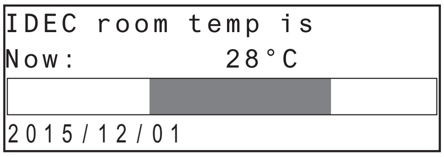
18. After the settings are configured, click **OK**.
The configured content is displayed on the LCD display area.



This concludes configuring the settings.

LCD display

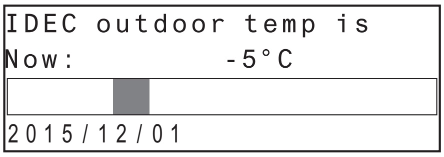
When M0000 is on, the room temperature stored in D0002 is displayed as the numeric value (°C) and bar graph.



```
IDEC room temp is
Now:        28°C

2015/12/01
```

When M0000 is off, the outdoor temperature stored in D0002 is displayed as the numeric value (°C) and bar graph.



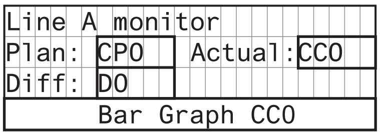
```
IDEC outdoor temp is
Now:        -5°C

2015/12/01
```

Modifying Device Values on the HMI Module

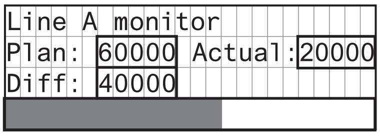
The values of the word devices displayed on the HMI module LCD can be modified using the HMI module operation buttons. The values cannot be modified when the user program is stopped.

[To modify the value of word device CP0]



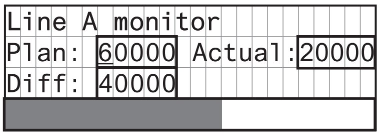
```
Line A monitor
Plan: CP0    Actual:CC0
Diff: D0
       Bar Graph CC0
```

When the user program is running and the input to the MSG instruction is turned on, the following screen will be displayed on the LCD.



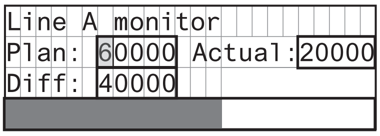
```
Line A monitor
Plan: 60000 Actual:20000
Diff: 40000
```

While the above message is displayed, press and hold the 🆗 (OK) button and the cursor is displayed on the devices that can be modified.



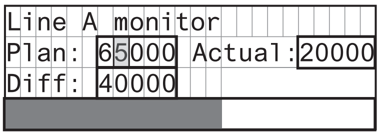
```
Line A monitor
Plan: 60000 Actual:20000
Diff: 40000
```

Move the cursor to the device you wish to edit using the ▲ (up) ▼ (down) ◀ (left) ▶ (right) buttons and press the 🆗 (OK) button to change the device in the editable state.



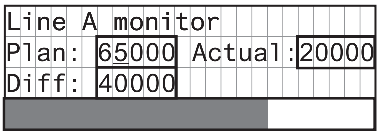
```
Line A monitor
Plan: 60000 Actual:20000
Diff: 40000
```

Press the ▶ (right) button to move the cursor to the fourth digit, and then use the ▲ (up) button to modify the target value to "65000".



```
Line A monitor
Plan: 65000 Actual:20000
Diff: 40000
```

Press the 🆗 (OK) button to confirm the modifications.



```
Line A monitor
Plan: 65000 Actual:20000
Diff: 40000
```

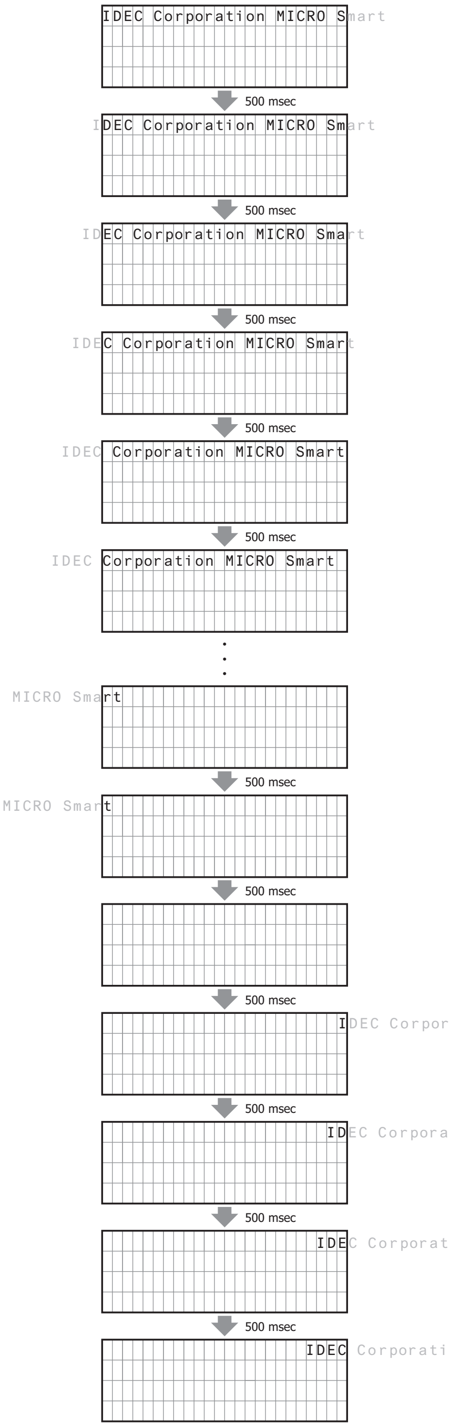
Scrolling text example

[Character Set: European scroll speed: 500 ms]

When a text with effect is set to scroll, the text is displayed on the HMI module LCD as follows.

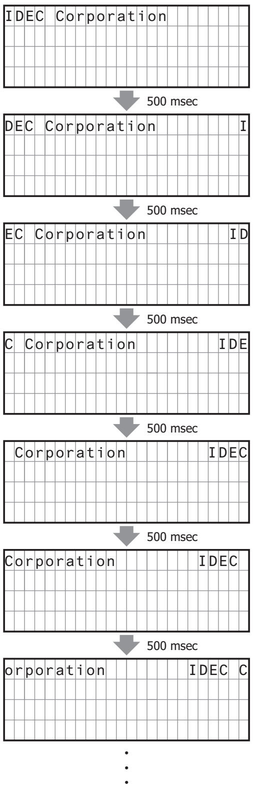
When scrolling a text longer than the specified area on the LCD

Text: "IDEC Corporation MICRO Smart"


IDEC Corporation MICRO Smart
500 msec
IDEC Corporation MICRO Smart
500 msec
IDEC Corporation MICRO Smart
500 msec
IDEC Corporation MICRO Smart
500 msec
IDEC Corporation MICRO Smart
500 msec
IDEC Corporation MICRO Smart
MICRO Smart
500 msec
MICRO Smart
500 msec
500 msec
IDEC Corpor
500 msec
IDEC Corpora
500 msec
IDEC Corporat
500 msec
IDEC Corporati

When scrolling a text shorter than or equal to the specified area on the LCD

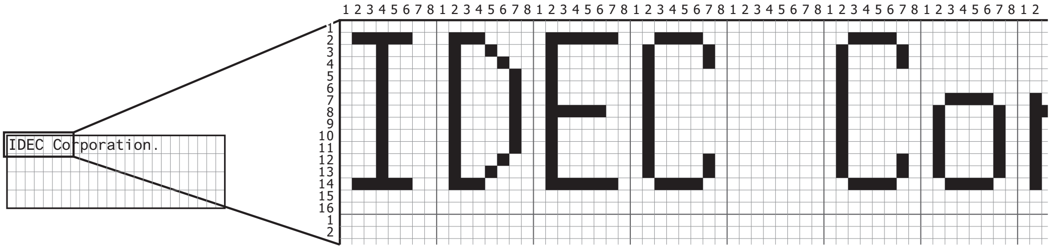
Text: "IDEC Corporation"

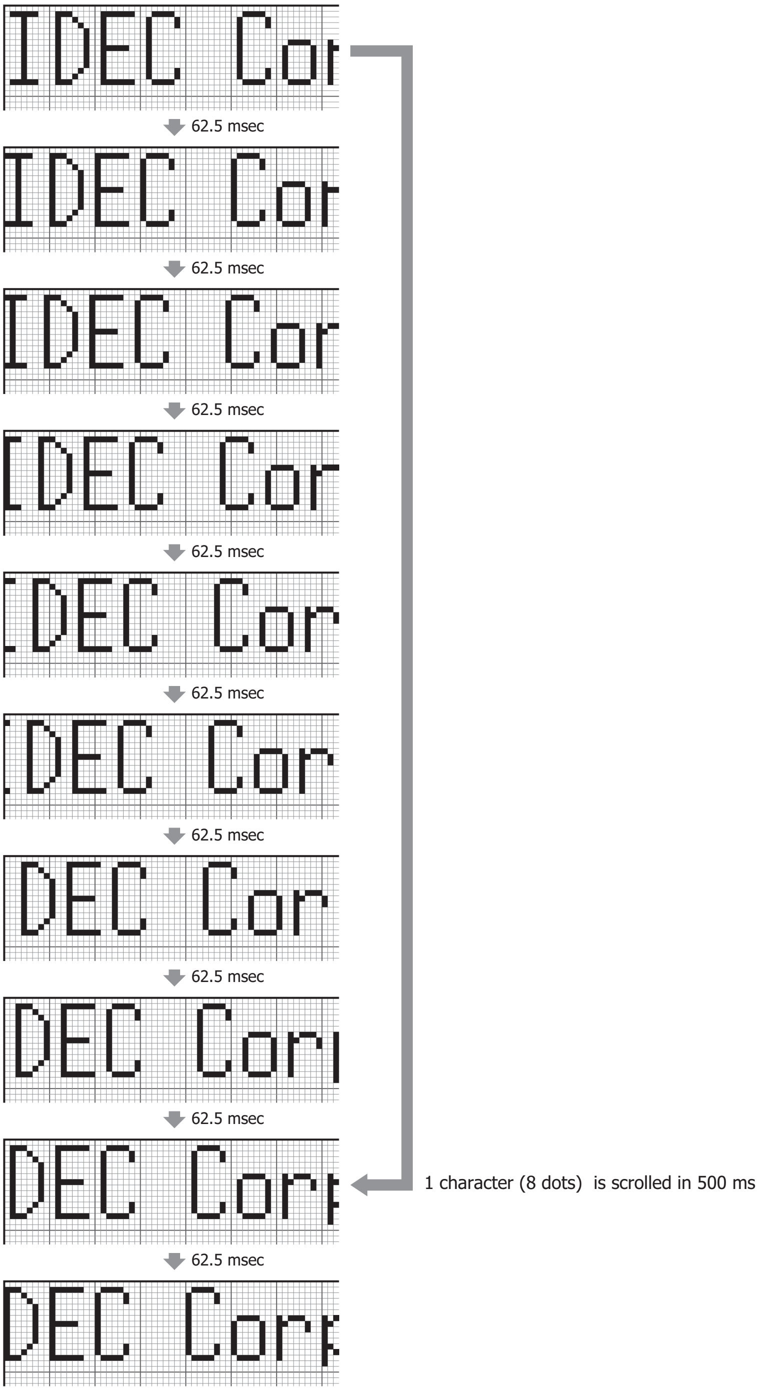


[Character Set: European, Scroll speed: 500 ms, Scroll unit: 1-dot]

When a text with effect is set to scroll, the text is displayed on the LCD of the HMI module as follows.

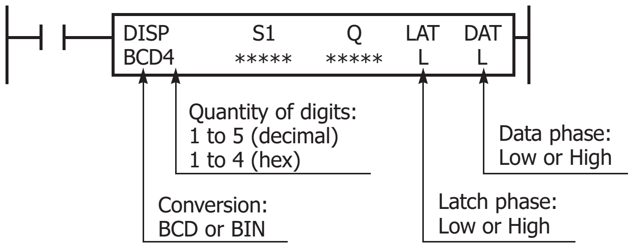
Text: "IDEC Corporation"





1 character (8 dots) is scrolled in 500 ms

DISP (Display)



Displays the specified data on a 7-segment display.

When the input is on, the data specified by S1 is displayed on the 7-segment display.

Display data can be 0 through 65535 (FFFFh).

Note: Always use transistor output to display data using the DISP instruction.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Data to display	—	—	—	—	X[*1]	X[*1]	X	—	—	—
Q (Output)	Display data output	—	X	X[*2]	—	—	—	—	—	—	—

For the valid device address range, see "Device Addresses" on page 2-1.

*1 When T (timer) or C (counter) is used as S1, the timer/counter current value (TC or CC) is read out.

*2 Special internal relays cannot be designated as Q.

Conversion

BCD: To connect BCD (decimal) display units

BIN: To connect BIN (hexadecimal) display units

Latch Phase[*1] and Data Phase[*1]

Select the latch and data phases to match the phases of the display units in consideration of sink or source output of the output module.

*1 The LAT (latch phase) and DAT (data phase) settings will differ depending on the 7-segment display sink output specification and source output specification. Configure these settings according to the specifications of the 7-segment display.

Specify the data to display on the 7-segment display with S1 (the display data). Specify the output for the display data with Q. Starting from the device specified with Q, the specified data (4 points + number of digits) is sequentially allocated by the display data and the number of display digits.

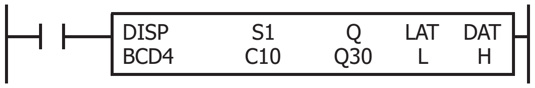
For example, if the number of display digits is 4 digits and the display data output is set to Q0, Q0 to Q7 are allocated (Q0 to Q3 are allocated to the data signal to the display, Q4 to Q7 are allocated to the digit selection signal).

Notes:

- A maximum of 8 DISP instructions can be entered in a user program.
- The displayable range is 0 to 65535 (FFFFh).
- Displaying one digit of data requires 3 scan times after the input to the DISP instruction is turned on. Keep the input to the DISP instruction for the period of time shown below to process all digits of the display data.
  Display Processing Time
  3 scan times × Quantity of digits
  When the scan time is less than 2 ms, the data cannot be displayed correctly. When the scan time is too short to ensure normal display, set a value of 3 or more (in ms) to data register D8022 (constant scan time preset value).

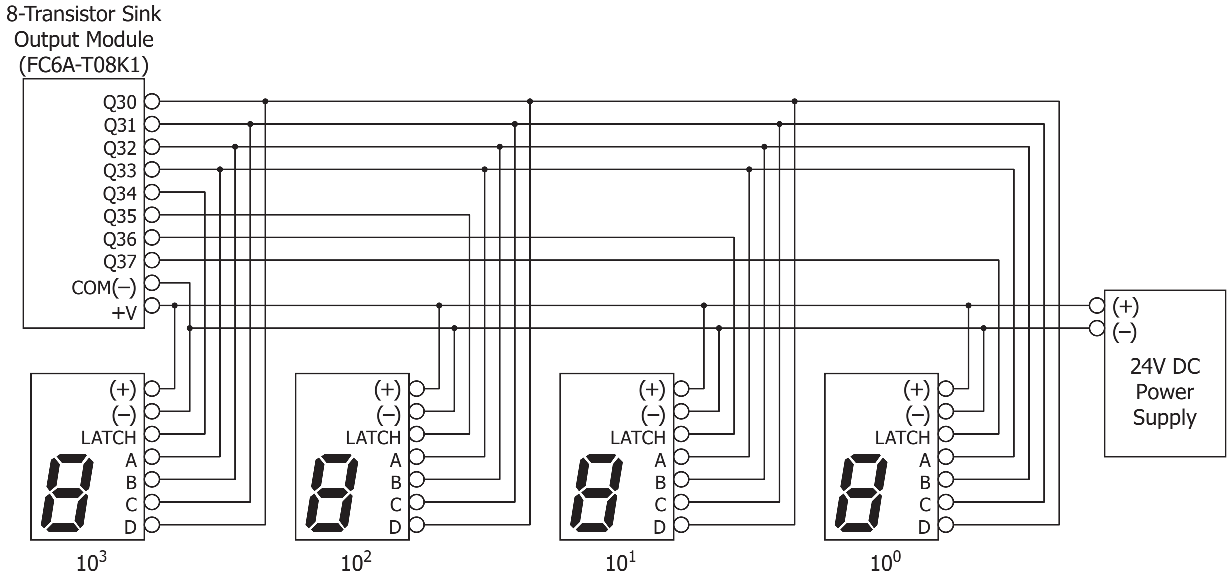
Operation Example

To display the 4-digit current value of counter C010 on a 7-segment display (IDEC DD3S-F31N) connected to the transistor sink output module.

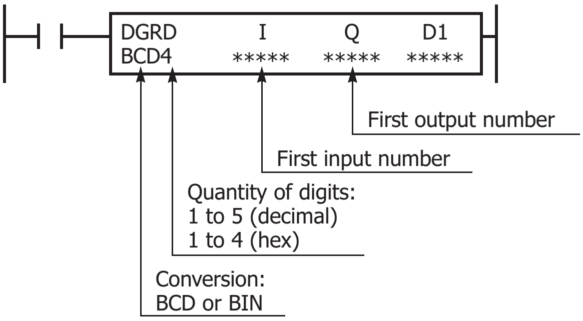


When input I0 is on, the 4-digit current value of counter C10 is displayed on 7-segment digital display units.

Output Wiring Diagram



DGRD (Digital Read)



When input is on, the setting values for the connected digital switch are stored in the device specified by D1.

This instruction can be used to change preset values for timer and counter instructions using digital switches.

Note: Always use transistor output to display data using the DGRD instruction.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
I	First input number to read	X	—	—	—	—	—	—	—	—	—
Q	First output number for digit selection	—	X	—	—	—	—	—	—	—	—
D1 (Destination 1)	Destination to store results	—	—	—	—	—	—	X	—	—	—

For the valid device address range, see "Device Addresses" on page 2-1.

The DGRD instruction can read 65535 (5 digits) at the maximum. When the read value exceeds 65,535 with the quantity of digits set to 5, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the CPU module.

Note: When transferring a user program containing more than 16 DGRD instructions to the CPU, a user program syntax error occurs, turning on the ERR LED. The user program cannot be executed.

Conversion

BCD: To connect BCD (decimal) digital switches
BIN: To connect BIN (hexadecimal) digital switches

Four points from the input specified by I are allocated as the reading inputs.

For example, when I is set to I0, I0 to I3 are allocated.

The number of digits specified by the number of display digits from the output specified by Q are allocated as the digit selection outputs.

For example, if the number of display digits is three digits, and Q is set to Q0, Q0 to Q2 are allocated.

Notes:

- A maximum of 16 DGRD instructions can be entered in a user program.
- Reading digital switch data requires the following time after the input to the DGRD instruction is turned on. Keep the input to the DGRD instruction for the period of time shown below to read the digital switch data. For example, when reading data from 5 digital switches to the destination device, 14 scans are required.

  Digital Switch Data Reading Time
  2 scan times × (Quantity of digits + 2)

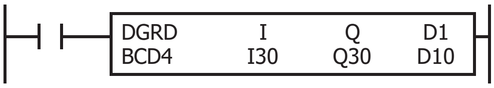
  The DGRD instruction requires a scan time longer than the filter time plus 6 ms.

  Minimum Required Scan Time
  (Scan time) ≥ (Filter time) + 6 ms
- As necessary, adjust the scan time using the constant scan time (D8022) function. The filter time differs according to the input that is used as the reading input.

  When specifying CPU module inputs as the reading inputs: I0 to I27
  See the input filter time configured in the WindLDR **Function Area Settings**. The initial value is 3 ms.
  When specifying input module inputs as the reading inputs: From I30
  There is no filter time, so calculate the value by replacing the filter time with the input delay time.
- The data that can be read using this instruction is 0 through 65535 (5 digits), or FFFFh. A user program execution error will occur when a value of 65,536 or greater is specified as a five digit decimal code and the processing will not be executed.
  For user program execution errors, see "User Program Execution Errors" on page 3-10.

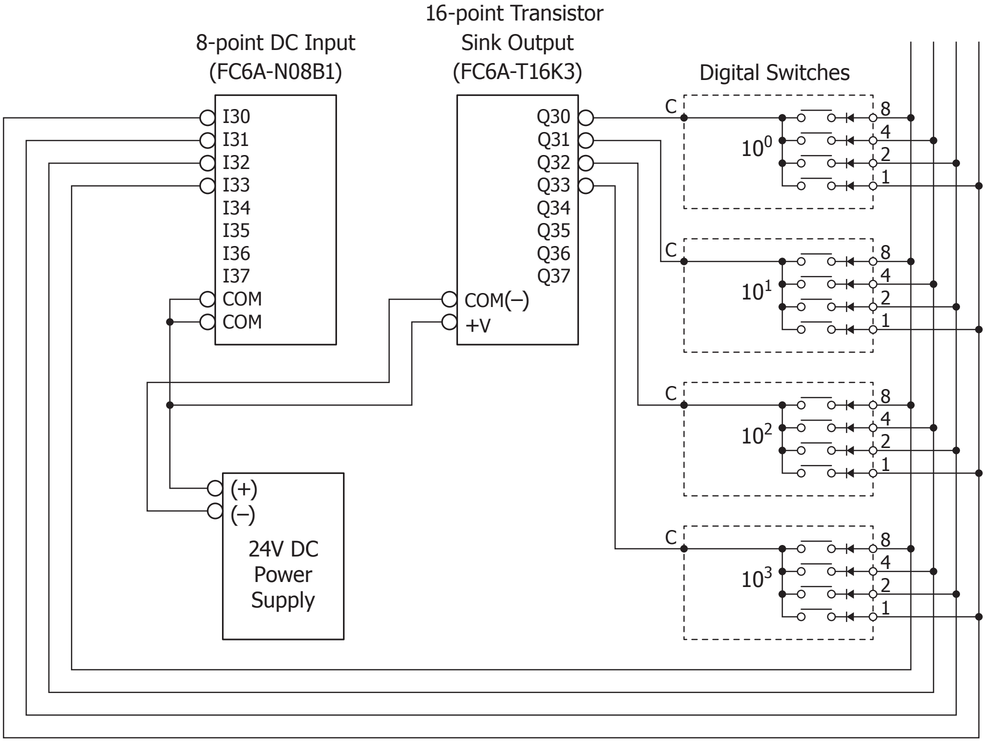
Operation Example

The following example demonstrates a program where digital switches (IDEC DFBN-031D-B) have been connected to the 16-I/O type transistor sink output module and those setting values are read into D0010.



When input I5 is on, the 4-digit value from BCD digital switches is read to data register D10.

I/O Wiring Diagram



13: Program Branching Instructions

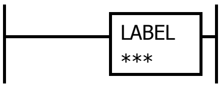
Introduction

This chapter describes the ladder program branching instructions that are used for branching program execution, looping ladder program execution, and creating ladder program subroutines.

Ladder program instruction execution branches from the LJMP (label jump) instruction and the DJNZ (decrement jump non-zero) instruction to the location where the LABEL instruction has been entered. Programming tools include "either/or" options between numerous portions of a program and the ability to call one of several subroutines which return execution to where the normal program stopped.

For the differences between user-defined macros and subroutines, see "Differences between User-defined Macros and Subroutines" on page A-6.

LABEL (Label)



This instruction sets the label number.

The label number that can be specified is 0 to 255.

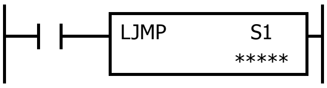
An END instruction may be used to separate a tagged portion of the program from the main program. In this way, scan time is minimized by ***not*** executing the program branch unless input conditions are satisfied.

Note: A single label number cannot be used more than once.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
Label number	Tag for LJMP and LCAL	—	—	—	—	—	—	—	—	0-255	—

LJMP (Label Jump)



When input is on, jump to the address with label 0 through 255 assigned by S1.

When input is off, no jump takes place, and program execution proceeds with the next instruction.

The LJMP instruction is used as an "either/or" choice between two portions of a program. Program execution does *not* return to the instruction following the LJMP instruction, after the program branch.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Label number	—	—	—	—	—	—	X	—	0-255	—

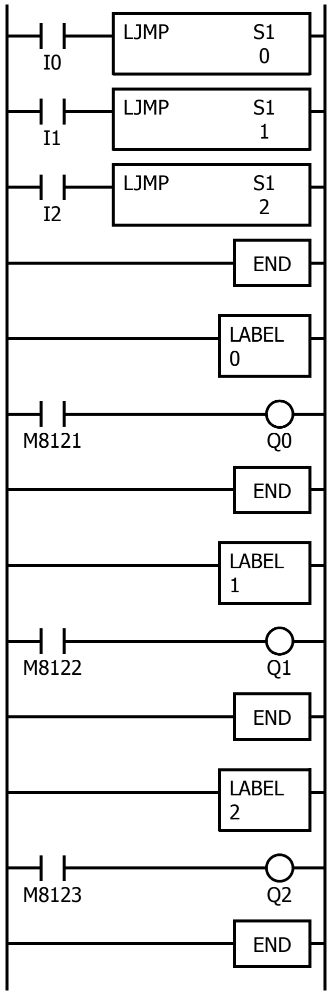
For valid device address ranges, see "Device Addresses" on page 2-1.

Since the LJMP instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Note: Make sure that a LABEL instruction of the label number used for a LJMP instruction is programmed. When designating S1 using other than a constant, the value for the label is a variable. When using a variable for a label, make sure that all probable LABEL numbers are included in the user program. If a matching label does not exist, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Example: LJMP and LABEL

The following example demonstrates a program that jumps to three different portions of the program depending on the input.



When input I0 is on, program execution jumps to label 0.

When input I1 is on, program execution jumps to label 1.

When input I2 is on, program execution jumps to label 2.

M8121 is the 1-s clock special internal relay.

When jump occurs to label 0, output Q0 oscillates in 1-s increments.

M8122 is the 100-ms clock special internal relay.

When jump occurs to label 1, output Q1 oscillates in 100-ms increments.

M8123 is the 10-ms clock special internal relay.

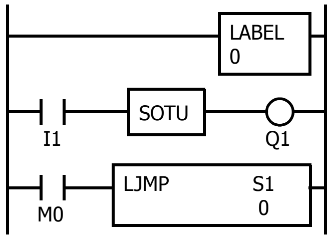
When jump occurs to label 2, output Q2 oscillates in 10-ms increments.

Using the Timer Instruction with Program Branching

When the timer start input of the TML, TIM, TMH or TMS instruction is already on, countdown begins immediately at the location jumped to, starting with the timer current value. When using a program branch, it is important to make sure that timers are initialized when desired, after the jump. If it is necessary to initialize the timer instruction (set to the preset value) after the jump, the timer's start input should be kept off for one or more scan cycles before initialization. Otherwise, the timer input on will not be recognized.

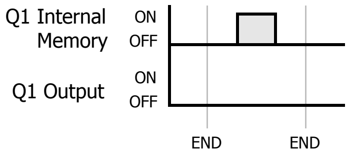
Using the SOTU/SOTD Instructions with Program Branching

Check that pulse inputs of counters and shift registers, and inputs of single outputs (SOTU and SOTD) are maintained during the jump, if required. Hold the input off for one or more scan cycles after the jump for the rising or falling edge transition to be recognized.



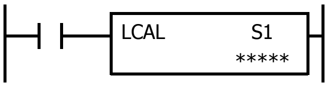
Although normally, the SOTU instruction produces a pulse for one scan, when used in a program branch the SOTU pulse will last only until the next time the same SOTU instruction is executed.

In the example on the left, the program branch will loop as long as internal relay M0 remains on. However, the SOTU produces a pulse output only during the first loop.



Since the END instruction is not executed as long as M0 remains on, output Q1 is not turned on even if input I1 is on.

LCAL (Label Call)



When input is on, the address with label 0 through 255 assigned by S1 is called. When input is off, no call takes place, and program execution proceeds with the next instruction.

The LCAL instruction calls a subroutine, and returns to the main program after the branch is executed. A LRET instruction (see below) must be placed at the end of a program branch which is called, so that normal program execution resumes by returning to the instruction following the LCAL instruction.

Note: The END instruction must be used to separate the main program from any subroutines called by the LCAL instruction.

A maximum of four LCAL instructions can be nested. When more than four LCAL instructions are nested, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

Valid Devices

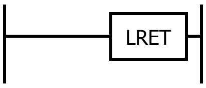
Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Label number to call	—	—	—	—	—	—	X	—	0-255	—

For valid device address ranges, see "Device Addresses" on page 2-1.

Since the LCAL instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Note: Make sure that a LABEL instruction of the label number used for a LCAL instruction is programmed. When designating S1 using other than a constant, the value for the label is a variable. When using a variable for a label, make sure that all probable LABEL numbers are included in the user program. If a matching label does not exist, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

LRET (Label Return)



This instruction returns to the address of the LCAL (label call) instruction.

This instruction is placed at the end of a subroutine called by the LCAL instruction. When the subroutine is completed, normal program execution resumes by returning to the instruction following the LCAL instruction.

The LRET must be placed at the end of the subroutine starting with a LABEL instruction. When the LRET is programmed at other places, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

Valid Devices

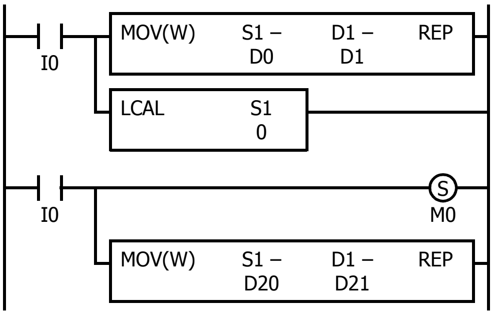
Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
—	—	—	—	—	—	—	—	—	—	—	—

Correct Structure for Calling Subroutine

When a LCAL instruction is executed, the remaining program instructions on the same rung may not be executed upon return, if input conditions are changed by the subroutine. After the LRET instruction of a subroutine, program execution begins with the instruction following the LCAL instruction, depending on current input condition.

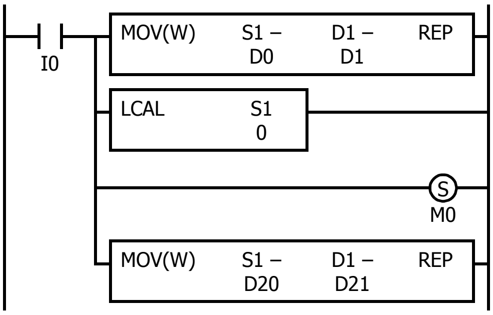
When instructions following a LCAL instruction must be executed after the subroutine is called, make sure the subroutine does not change input conditions unfavorably. In addition, include subsequent instructions in a new ladder line, separated from the LCAL instruction.

Correct



Separate the ladder line for each LCAL instruction.

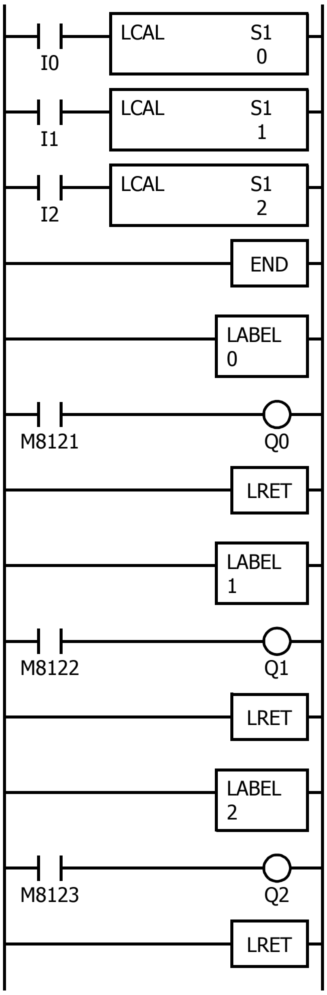
Incorrect



I0 status may be changed by the subroutine upon return.

Example: LCAL and LRET

The following example demonstrates a program that calls three different portions of program depending on the input. When the subroutine is complete, program execution returns to the instruction following the LCAL instruction.



When input I0 is on, program execution jumps to label 0.

When input I1 is on, program execution jumps to label 1.

When input I2 is on, program execution jumps to label 2.

M8121 is the 1-s clock special internal relay.

When jump occurs to label 0, output Q0 oscillates in 1-s increments.

Program execution returns to the address of input I1.

M8122 is the 100-ms clock special internal relay.

When jump occurs to label 1, output Q1 oscillates in 100-ms increments.

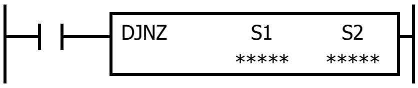
Program execution returns to the address of input I2.

M8123 is the 10-ms clock special internal relay.

When jump occurs to label 2, output Q2 oscillates in 10-ms increments.

Program execution returns to the address of END.

DJNZ (Decrement Jump Non-zero)



When input is on, the value stored in the data register assigned by S1 is decremented by one and is checked. If the resultant value is not 0, program execution jumps to address with label 0 through 255 assigned by S2. If the decrement results in 0, no jump takes place, and program execution proceeds with the next instruction.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Decrement value	—	—	—	—	—	—	X	—	—	—
S2 (Source 2)	Label number to jump to	—	—	—	—	—	—	X	—	0-255	—

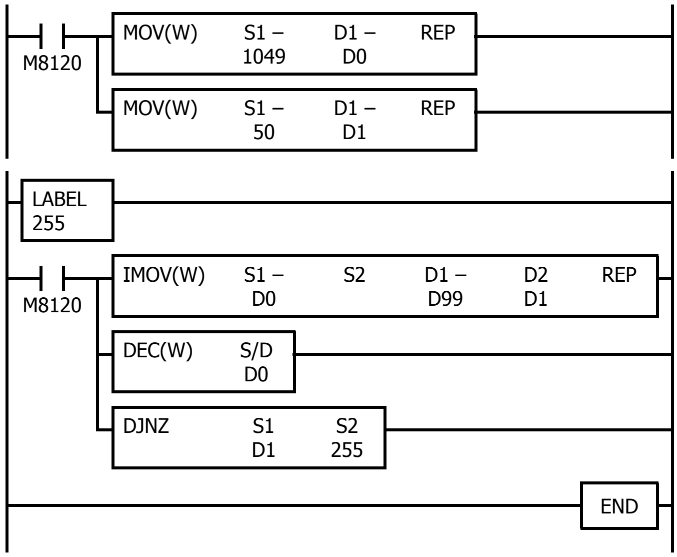
For valid device address ranges, see "Device Addresses" on page 2-1.

Since the DJNZ instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

The label number can be 0 through 255. Make sure that a LABEL instruction of the label number used for a DJNZ instruction is programmed. When designating S2 using a data register, the value for the label is a variable. When using a variable for a label, make sure that all probable LABEL numbers are included in the user program. If a matching label does not exist, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

Example: DJNZ and LABEL

The following example demonstrates a program to store consecutive values 1,000 through 1,049 to data registers D100 through D149, respectively.


M8120
MOV(W) S1 – 1049 D1 – D0 REP
MOV(W) S1 – 50 D1 – D1 REP
LABEL 255
M8120
IMOV(W) S1 – D0 S2 D1 – D99 D2 D1 REP
DEC(W) S/D D0
DJNZ S1 D1 S2 255
END

M8120 is the initialize pulse special internal relay.

At start-up, MOV instructions store initial data.

1049 → D0 to store the value for the first cycle.

50 → D1 to determine the jump cycles.

IMOV moves D0 data 1049 to D1049 in the first cycle.

DEC decrements D0 data to 1048.

DJNZ jumps to label 255 until D1 value reduces to 0.

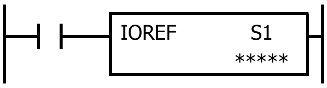
1st cycle:	D1	50	Destination: D99 + 50 = D149	D0	1049	→	D149	1049
2nd cycle:	D1	49	Destination: D99 + 49 = D148	D0	1048	→	D148	1048
3rd cycle:	D1	48	Destination: D99 + 48 = D147	D0	1047	→	D147	1047
4th cycle:	D1	47	Destination: D99 + 47 = D146	D0	1046	→	D146	1046
			⋮					
49th cycle:	D1	2	Destination: D99 + 2 = D101	D0	1001	→	D101	1001
50th cycle:	D1	1	Destination: D99 + 1 = D100	D0	1000	→	D100	1000

14: REFRESH INSTRUCTIONS

Introduction

This chapter describes the refresh instructions that update the status of input contacts, external output , and the current value of the high-speed counter, to the latest value during a ladder scan.

IOREF (I/O Refresh)



When input is on, 1-bit I/O data assigned by source device S1 is refreshed immediately regardless of the scan time.

When I (input) is used as S1, the actual input status is immediately read into an internal relay starting with M300 allocated to each input available on the FC6A Series MICROSmart.

When Q (output) is used as S1, the output data in the RAM is immediately written to the actual output available on the FC6A Series MICROSmart.

Refresh instructions are useful when a real-time response is required in a user program which has a long scan time. The refresh instruction is most effective when using the refresh instruction at a ladder step immediately before using the data.

The IOREF instruction can be used with an interrupt input or timer interrupt to refresh data.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	I/O for refresh	X	X	—	—	—	—	—	—	—	—

Only inputs and outputs available on the CPU module can be specified for S1 of the IOREF instruction. However, I14 to I17 and Q14 to Q17 of Plus CPU module cannot be specified for S1. Input and output numbers for expansion modules or cartridges cannot be designated as S1. For valid device address ranges, see "Device Addresses" on page 2-1.

Input Device Addresses and Allocated Internal Relays

Internal relays for storage are allocated by input device. For example, when I10 is designated as S1, the latest value of I10 is set to M310.

Input Device	Internal Relay	Input Device	Internal Relay	Input Device	Internal Relay
I0	M300	I10	M310	I20	M320
I1	M301	I11	M311	I21	M321
I2	M302	I12	M312	I22	M322
I3	M303	I13	M313	I23	M323
I4	M304	I14	M314	I24	M324
I5	M305	I15	M315	I25	M325
I6	M306	I16	M316	I26	M326
I7	M307	I17	M317	I27	M327

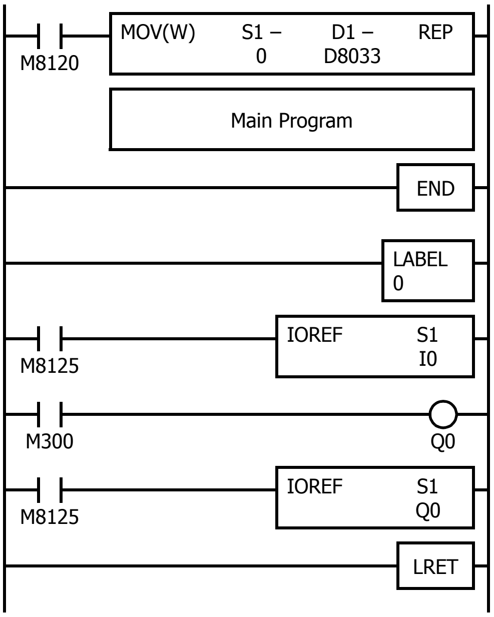
During normal execution of a user program, I/O statuses are refreshed simultaneously when the END instruction is executed at the end of a scan. When a real-time response is needed to execute an interrupt, the IOREF instruction can be used. When the input to the IOREF instruction is turned on, the status of the designated input or output is read or written immediately.

When the IOREF instruction is executed for an input, the filter does not take effect and the input status at the moment is read to a corresponding internal relay.

The actual input status of all inputs are read regardless of whether IOREF instruction is programmed when the END instruction is executed as in the normal scanning, then the filter value has effect as designated in the **Function Area Settings**. See Chapter 5 "Functions and Settings" - "Input Filter" in the "FC6A Series MICROSmart User's Manual".

Example: IOREF

The following example demonstrates a program that transfers the input I0 status to output Q0 using the IOREF instruction. Input I3 is designated as an interrupt input. For the interrupt input function, see Chapter 5 "Functions and Settings" - "Interrupt Input" in the "FC6A Series MICROSmart User's Manual".



M8120 is the initialize pulse special internal relay.

D8033 stores 0 to designate jump destination label 0 for interrupt input I3.

When input I3 is on, program execution jumps to label 0.

M8125 is the in-operation output special internal relay.

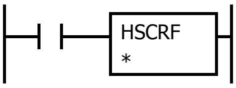
IOREF immediately reads input I0 status to internal relay M300.

M300 turns on or off the output Q0 internal memory.

Another IOREF immediately writes the output Q0 internal memory status to actual output Q0.

Program execution returns to the main program.

HSCRF (High-speed Counter Refresh)



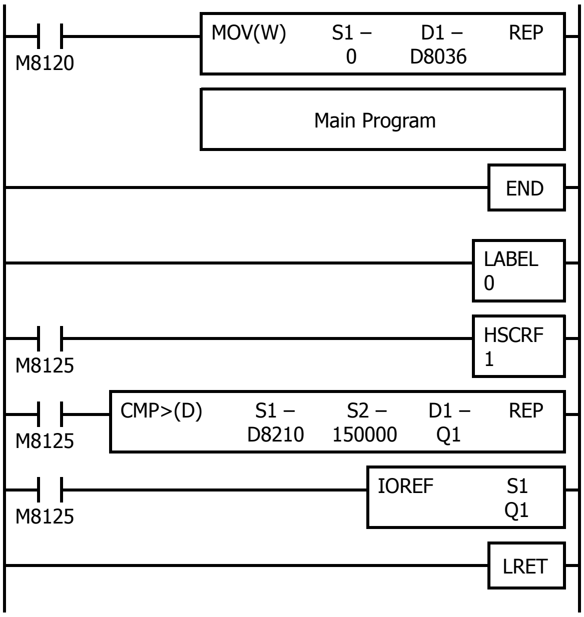
When the input is on, the current values of the high-speed counters (select from group 1 to group 6) allocated to special data registers are updated to the latest values. (The group number of the selected high-speed counter is displayed under the advanced instruction symbol.)

The current values of six high-speed counters HSC1 through HSC6 are usually updated in every scan. The HSCRF can be used in any place in the ladder diagram where you want to read the updated high-speed counter current value.

For the high-speed counter function, see Chapter 5 "Functions and Settings" - "High-Speed Counter" in the "FC6A Series MICROSmart User’s Manual".

Example: HSCRF

The following example demonstrates a program that updates the current value of high-speed counter HSC1 using the HSCRF instruction. For the timer interrupt, see Chapter 5 "Functions and Settings" - "Timer Interrupt" in the "FC6A Series MICROSmart User’s Manual".



M8120 is the initialize pulse special internal relay.

D8036 stores 0 to designate jump destination label 0 for timer interrupt.

The interrupt program is separated from the main program by the END instruction.

While the CPU is running, program execution jumps to label 0 repeatedly at intervals selected in the **Function Area Settings**.

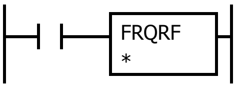
M8125 is the in-operation output special internal relay.
HSCRF updates the HSC1 current value in data registers D8210 and D8211.

When D8210/D8211 exceeds 150,000, Q1 is turned on.

IOREF immediately writes the output Q1 internal memory status to actual output Q1.

Each time the interrupt program is completed, program execution returns to the main program at the address where timer interrupt occurred.

FRQRF (Frequency Measurement Refresh)



When the input is on, the frequency measurement values (select from group 1 to group 6) allocated to special data registers are updated to the latest values. (The group number of the selected high-speed counter is displayed under the advanced instruction symbol.)

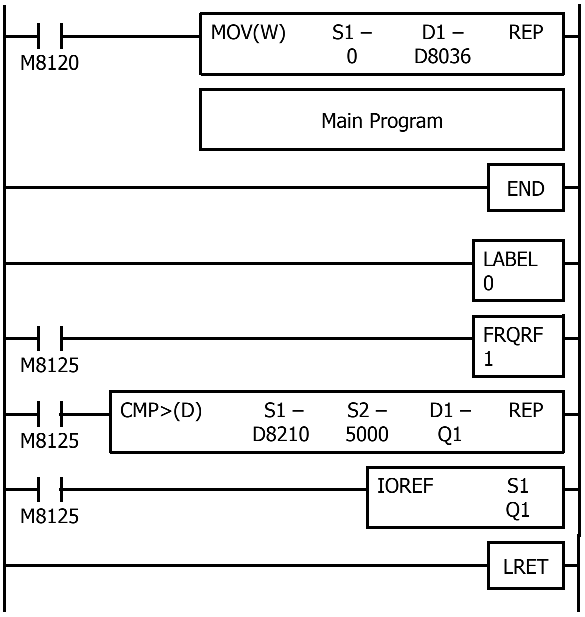
Before the measured results are reflected in special data registers, it takes a maximum of approximately 250 ms + the scan time, but by using the FRQRF instruction, control processing using the latest current values of the frequency measurements can be performed in any location in the user program.

For the frequency measurement function, see Chapter 5 "Frequency Measurement" in the "FC6A Series MICROSmart User's Manual".

Example: FRQRF

Periodically update the current value of the frequency measurement value (D8210, D8211) using the FRQRF instruction in an interrupt program.

When the frequency measurement value becomes 5,000 or greater, turn on the error signal (Q1).



Store 0 in D8036 and specify LABEL 0 as the interrupt program to execute when the timer interrupt occurs.

The interrupt program is separated from the main program by the END instruction.

Interrupt program starts.

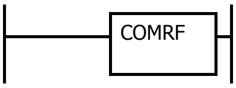
Update the current value (D8210, D8211) of the frequency measurement value.

When the frequency measurement value becomes 5,000 or greater, turn on Q1 as the error signal.

The status of output Q1 is reflected in the external output.

Interrupt program ends.

COMRF (Communication Refresh)



Execute reading of received data and writing of send data for port 2 and 3 (communication port refresh) during user program processing.

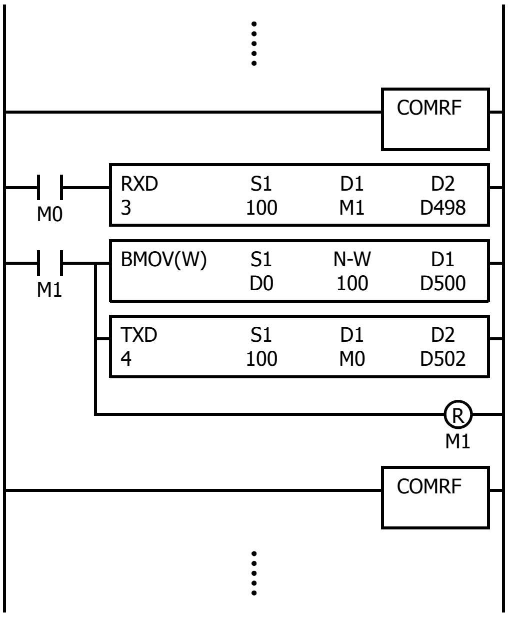
When implementing an application that requires a communication response from the communication cartridge in a time interval that is shorter than the scan time of FC6A Series MICROSmart, program it with the COMRF instruction.

Regardless of the presence of the COMRF instruction, reading received data and writing send data for expanded communication port is executed in END processing.

Note: This instruction cannot be used in interrupt programs. If this instruction is used in an interrupt program, a user program execution error will occur.
For user program execution errors, see "User Program Execution Errors" on page 3-10.

Example: COMRF

The following example demonstrates a program to execute COMRF instructions among other instructions.



Before executing the RXD instruction, COMRF refreshes send and receive data of the expansion communication ports.

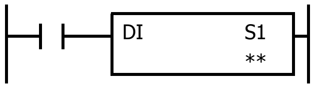
After executing the TXD instruction, COMRF refreshes send and receive data of the expansion communication ports.

15: INTERRUPT CONTROL INSTRUCTIONS

Introduction

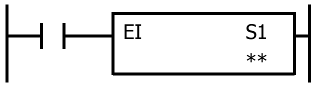
This chapter describes interrupt control instructions that prohibit and allow user interrupt operations (interrupt input, timer interrupt).

DI (Disable Interrupt)



When input is on, interrupt inputs and timer interrupt assigned by source device S1 are disabled.

EI (Enable Interrupt)



When input is on, interrupt inputs and timer interrupt assigned by source device S1 are enabled.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Interrupt inputs and timer interrupt	—	—	—	—	—	—	—	—	X[*1]	—

[*1] Group 1 through Group 6 or Timer Interrupt can be set.

Interrupt inputs I0 through I7 and timer interrupt selected in the Function Area Settings are normally enabled when the CPU starts. When the DI instruction is executed, interrupt inputs and timer interrupt designated as source device S1 are disabled even if the interrupt condition is met in the user program area subsequent to the DI instruction. When the EI instruction is executed, disabled interrupt inputs and timer interrupt designated as source device S1 are enabled again in the user program area subsequent to the EI instruction. Different interrupts can be selected for the DI and EI instructions to disable and enable interrupt inputs selectively. For Interrupt Input and Timer Interrupt, see Chapter 5 "Functions and Settings" - "Interrupt Input" and "Timer Interrupt" in the "FC6A Series MICROSmart User's Manual".

Make sure that interrupt inputs and timer interrupt designated as source device S1 are selected in the Function Area Settings. Otherwise, when the DI or EI instruction is executed, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

The DI and EI instructions cannot be used in an interrupt program. If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

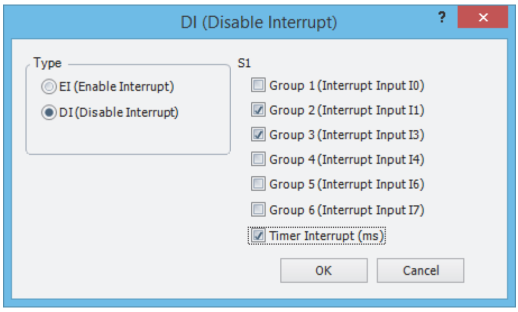
Special Internal Relays M8137-M8143, M8167, and M8144: Interrupt Status

Special internal relays M8137 through M8143, M8167, and M8144 are provided to indicate whether interrupt inputs and timer interrupt are enabled or disabled.

Interrupt	Interrupt Enabled	Interrupt Disabled
Group 1 (I0)	M8137 ON	M8137 OFF
Group 2 (I1)	M8140 ON	M8140 OFF
Group 3 (I3)	M8141 ON	M8141 OFF
Group 4 (I4)	M8142 ON	M8142 OFF
Group 5 (I6)	M8143 ON	M8143 OFF
Group 6 (I7)	M8167 ON	M8167 OFF
Timer Interrupt	M8144 ON	M8144 OFF

Programming WindLDR

In the DI (Disable Interrupt) or EI (Enable Interrupt) dialog box, select the check box on the left of Interrupt Groups 1 through 6 or Timer Interrupt to select source device S1. The example below selects interrupt groups 2, 3, and timer interrupt for the DI instruction, and a 22 will be shown as source device S1.



DI (Disable Interrupt)

Type

○ EI (Enable Interrupt)

● DI (Disable Interrupt)

S1

□ Group 1 (Interrupt Input I0)

☑ Group 2 (Interrupt Input I1)

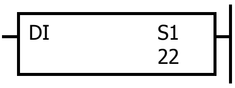
☑ Group 3 (Interrupt Input I3)

□ Group 4 (Interrupt Input I4)

□ Group 5 (Interrupt Input I6)

□ Group 6 (Interrupt Input I7)

☑ Timer Interrupt (ms)

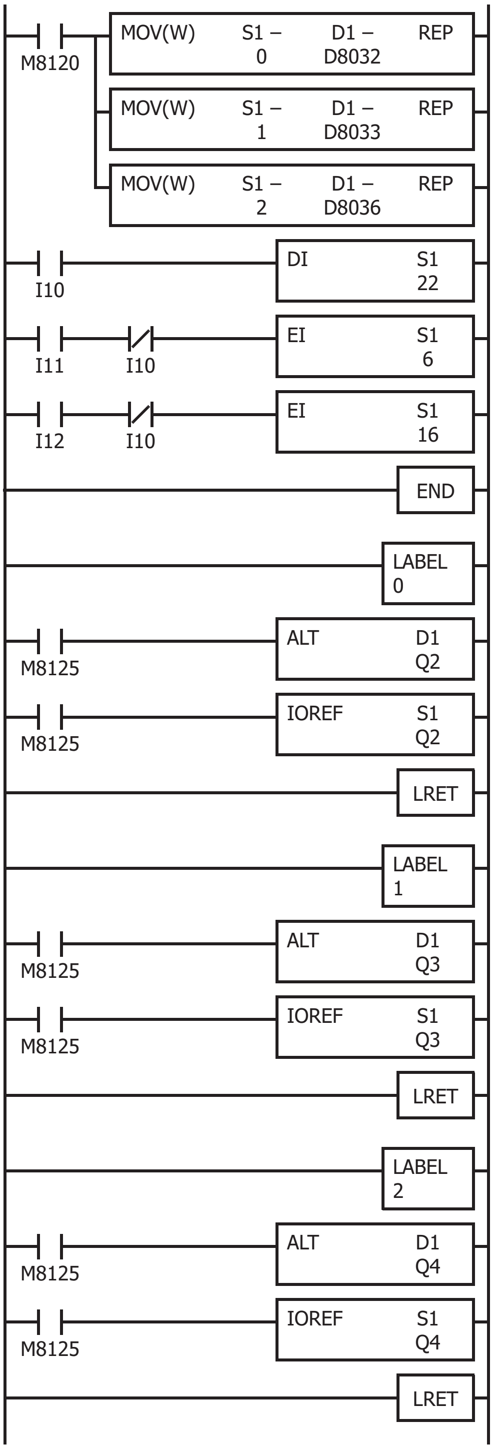


The total of selected interrupt inputs and timer interrupt is shown as source device S1.

Interrupt	S1 Value
Group 1 (I0)	1
Group 2 (I1)	2
Group 3 (I3)	4
Group 4 (I4)	8
Group 5 (I6)	32
Group 6 (I7)	64
Timer Interrupt	16

Example: DI and EI

The following example demonstrates a program that will disable and enable interrupt inputs and timer interrupt selectively. For the interrupt input and timer interrupt functions, see Chapter 5 "Functions and Settings" - "Interrupt Input" and "Timer Interrupt" in the "FC6A Series MICROSmart User’s Manual". In this example, inputs I1 and I3 are designated as interrupt inputs and timer interrupt is used with interrupt intervals of 100 ms.



M8120 is the initialize pulse special internal relay.

D8032 stores jump destination label number 0 for interrupt input I1.

D8033 stores jump destination label number 1 for interrupt input I3.

D8036 stores jump destination label number 2 for timer interrupt.

When input I10 is on, DI disables interrupt inputs I1, I3, and timer interrupt, then M8140, M8141, and M8144 turn off.

When input I11 is on and I10 is off, EI enables interrupt inputs I1 and I3, then M8140 and M8141 turn on.

When input I12 is on and I10 is off, EI enables timer interrupt, then M8144 turns on.

End of the main program.

When input I1 is on, program execution jumps to label 0.

M8125 is the in-operation output special internal relay.

ALT turns on or off the output Q2 internal memory.

IOREF immediately writes the output Q2 internal memory status to actual output Q2.

Program execution returns to the main program.

When input I3 is on, program execution jumps to label 1.

M8125 is the in-operation output special internal relay.

ALT turns on or off the output Q3 internal memory.

IOREF immediately writes the output Q3 internal memory status to actual output Q3.

Program execution returns to the main program.

Timer interrupt occurs every 100 ms, then program execution jumps to label 2.

M8125 is the in-operation output special internal relay.

ALT turns on or off the output Q4 internal memory.

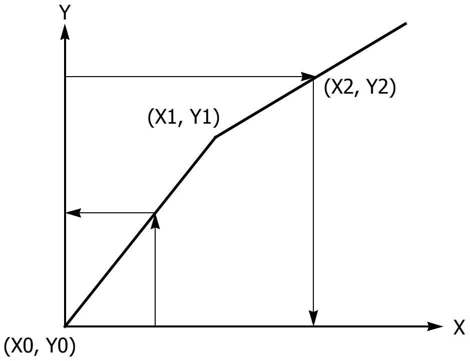
IOREF immediately writes the output Q4 internal memory status to actual output Q4.

Program execution returns to the main program.

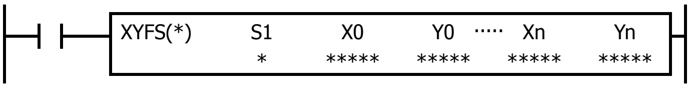
16: COORDINATE CONVERSION INSTRUCTIONS

Introduction

This chapter describes the coordinate conversion instructions that convert one data point to another value using the linear relationship between the values of X and Y.



XYFS (XY Format Set)



When input is on, the format for XY conversion is set. The XY coordinates define the linear relationship between X and Y.

No. of XY Coordinates	n
2 to 32	0 ≤ n ≤ 31

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Format number	—	—	—	—	—	—	—	—	0 to 7	—
X0 through Xn	X value	X	X	X	X	X	X	X	—	0 to 65,535	—
Y0 through Yn	Y value	X	X	X	X	X	X	X	—	0 to 65,535 –32,768 to 32,767	—

For valid device address ranges, see "Device Addresses" on page 2-1.

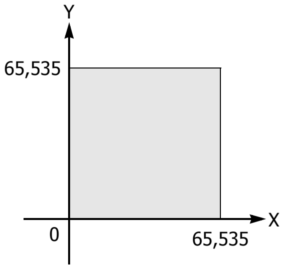
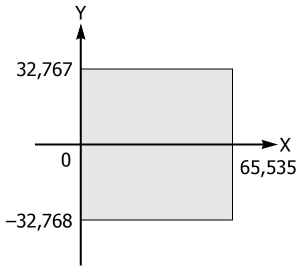
When T (timer) or C (counter) is used as X0 through Xn or Y0 through Yn, the timer/counter current value (TC or CC) is displayed.

S1 (Format number)

Select a format number 0 through 7. A maximum of 8 formats for XY conversion can be set.

Xn (X value), Yn (Y value)

Enter values for the X and Y coordinates. Two different data ranges are available depending on the data type.

Data Type	Word	Integer
Xn (X value)	0 to 65,535	0 to 65,535
Yn (Y value)	0 to 65,535	–32,768 to 32,767
Valid Coordinates		

Valid Data Types

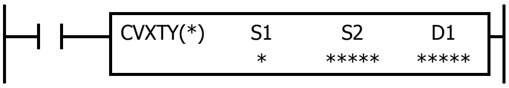
W (word)	X
I (integer)	X
D (double word)	—
L (long)	—
F (float)	—

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as Xn or Yn, 16 points are used.

When a word device such as T (timer), C (counter), or D (data register) is assigned as Xn or Yn, 1 point is used.

Note: The XYFS instruction cannot be used in an interrupt program.
If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

CVXTY (Convert X to Y)



When input is on, the X value assigned by device S2 is converted into the corresponding Y value according to the linear relationship defined in the XYFS instruction. Device S1 selects a format from a maximum of 8 XY conversion formats. The conversion result is set to the device assigned by D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Format number	—	—	—	—	—	—	—	—	0 to 7	—
S2 (Source 2)	X value	X	X	X	X	X	X	X	—	0 to 65,535	—
D1 (Destination 1)	Destination to store results	—	X	▲	X	X	X	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S2, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP) which can be 0 through 65,535.

S1 (Format number)

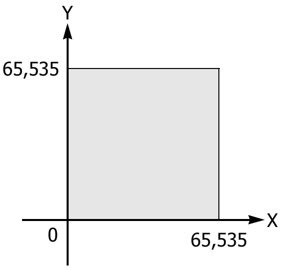
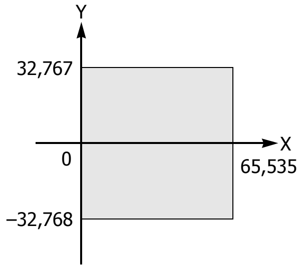
Select a format number 0 through 7 which have been set using the XYFS instruction. When an XYFS instruction with the corresponding format number is not programmed, or when XYFS and CVXTY instructions of the same format number have different data type designations, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

S2 (X value)

Enter a value for the X coordinate to convert, within the range specified in the XYFS instruction.

D1 (Destination to store results)

The conversion result of the Y value is stored to the destination.

Data Type	Word	Integer
S2 (X value)	0 to 65,535	0 to 65,535
D1 (Y value)	0 to 65,535	–32,768 to 32,767
Valid Coordinates		

Valid Data Types

W (word)	X
I (integer)	X
D (double word)	—
L (long)	—
F (float)	—

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as S2 or D1, 16 points are used.

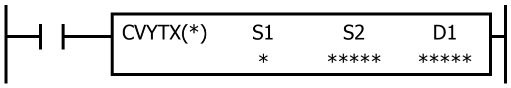
When a word device such as T (timer), C (counter), or D (data register) is assigned as S2 or D1, 1 point is used.

Data Conversion Error

The data conversion error is ±0.5.

Note: The CVXTY instruction cannot be used in an interrupt program.
If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

CVYTX (Convert Y to X)



When input is on, the Y value assigned by device S2 is converted into the corresponding X value according to the linear relationship defined in the XYFS instruction. Device S1 selects a format from a maximum of 8 XY conversion formats. The conversion result is set to the device assigned by D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Format number	—	—	—	—	—	—	—	—	0 to 7	—
S2 (Source 2)	Y value	X	X	X	X	X	X	X	—	0 to 65,535 –32,768 to 32,767	—
D1 (Destination 1)	Destination to store results	—	X	▲	X	X	X	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D1.

When T (timer) or C (counter) is used as S2, the timer/counter current value (TC or CC) is displayed. When T (timer) or C (counter) is used as D1, the data is written in as a preset value (TP or CP) which can be 0 through 65,535.

S1 (Format number)

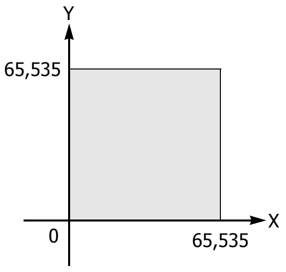
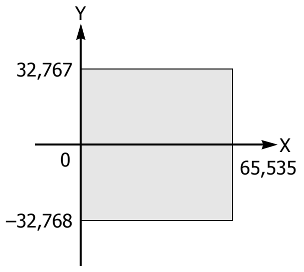
Select a format number 0 through 7 which have been set using the XYFS instruction. When an XYFS instruction with the corresponding format number is not programmed, or when XYFS and CVYTX instructions of the same format number have different data type designations, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

S2 (Y value)

Enter a value for the Y coordinate to convert, within the range specified in the XYFS instruction. Two different data ranges are available depending on the data type.

D1 (Destination to store results)

The conversion result of the X value is stored to the destination.

Data Type	Word	Integer
S2 (Y value)	0 to 65,535	–32,768 to 32,767
D1 (X value)	0 to 65,535	0 to 65,535
Valid Coordinates		

Valid Data Types

W (word)	X
I (integer)	X
D (double word)	—
L (long)	—
F (float)	—

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as S2 or D1, 16 points are used.

When a word device such as T (timer), C (counter), or D (data register) is assigned as S2 or D1, 1 point (integer data) is used.

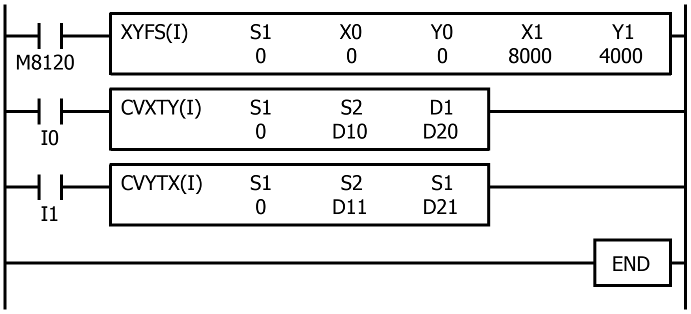
Data Conversion Error

The data conversion error is ±0.5.

Note: The CVYTX instruction cannot be used in an interrupt program.
If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Example: Linear Conversion

The following example demonstrates setting up two coordinate points to define the linear relationship between X and Y. The two points are (X0, Y0) = (0, 0) and (X1, Y1) = (8,000, 4,000). Once these are set, there is an X to Y conversion, as well as a Y to X conversion.

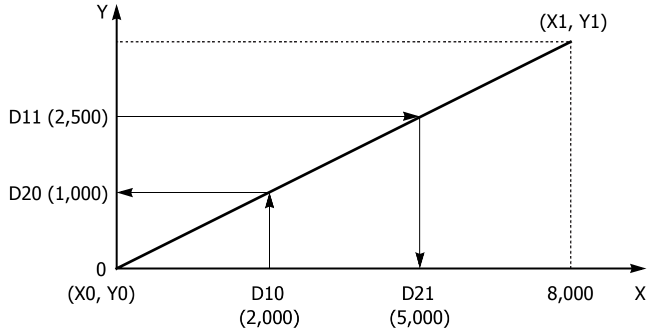


M8120 is the initialize pulse special internal relay.

At startup, XYFS specifies two points.

When input I0 is on, CVXTY converts the value in D10 and stores the result in D20.

When input I1 is on, CVYTX converts the value in D11 and stores the result in D21.



The graph shows the linear relationship that is defined by the two points:

$$Y = \frac{1}{2}X$$

If the value in data register D10 is 2,000, the value assigned to D20 is 1,000.

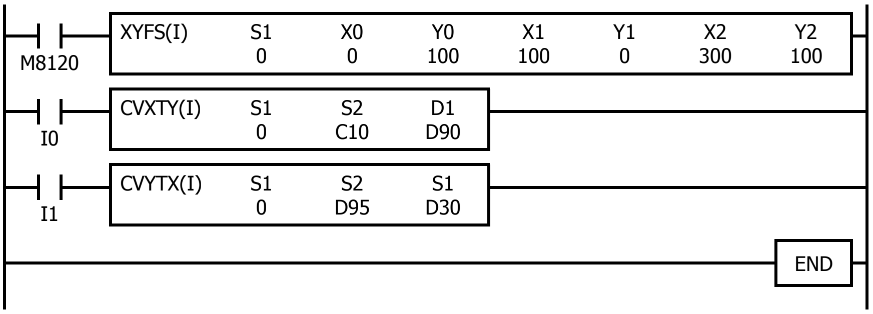
For Y to X conversion, the following equation is used:

$$X = 2Y$$

If the value in data register D11 is 2,500, the value assigned to D21 is 5,000.

Example: Overlapping Coordinates

In this example, the XYFS instruction sets up three coordinate points, which define two different linear relationships between X and Y. The three points are: (X0, Y0) = (0, 100), (X1, Y1) = (100, 0), and (X2, Y2) = (300, 100). The two line segments define overlapping coordinates for X. That is, for each value of Y within the designated range, there would be two X values assigned.

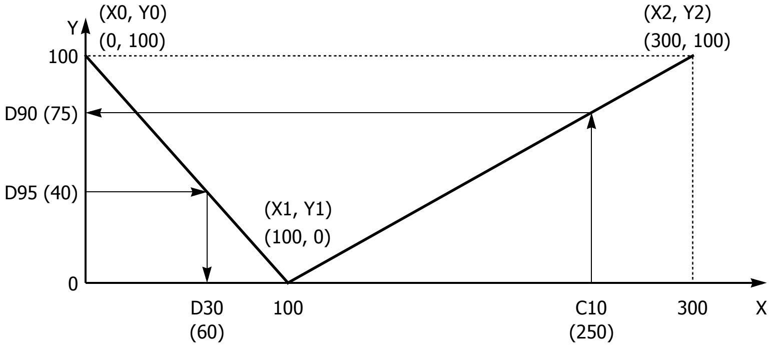


M8120 is the initialize pulse special internal relay.

At startup, XYFS specifies three points.

CVXTY converts the current value in C10 and stores the result in D90.

CVYTX converts the value in D95 and stores the result in D30.



The first line segment defines the following relationship for X to Y conversion:

$$Y = -X + 100$$

The second line segment defines another relationship for X to Y conversion:

$$Y = \frac{1}{2}X - 50$$

For X to Y conversion, each value of X has only one corresponding value for Y. If the current value of counter C10 is 250, the value assigned to D90 is 75.

For Y to X conversion, the XYFS instruction assigns two possible values of X for each value of Y. The relationship defined by the first two points has priority in these cases. The line between points (X0, Y0) and (X1, Y1), that is, the line between (0, 100) and (100, 0), has priority in defining the relationship for Y to X conversion (X = –Y + 100).

Therefore, if the value in data register D95 is 40, the value assigned to D30 is 60, not 180.

Exactly the same two line segments might also be defined by the XYFS instruction, except that the point (300, 100) could be assigned first, as (X0, Y0), and the point (100, 0) could be defined next, as (X1, Y1). In this case, this linear relationship would have priority.

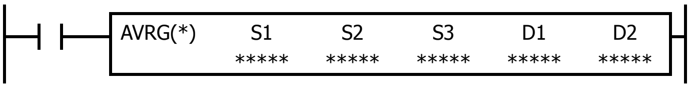
In this case, if the value in data register D95 is 40, the value assigned to D30 is 180, not 60.

17: AVERAGE INSTRUCTIONS

Introduction

This chapter describes the average instructions that average the specified data.

AVRG (Average)



When input is on, sampling data assigned by device S1 is processed according to sampling conditions assigned by devices S2 and S3.

When sampling is complete, average, maximum, and minimum values are stored to 3 consecutive devices starting with device assigned by D1, then sampling completion output assigned by device D2 is turned on.

Data Type	W, I	D, L, F
Average	D1	D1·D1+1
Maximum value	D1+1	D1+2·D1+3
Minimum value	D1+2	D1+4·D1+5

The AVRG instruction is effective for data processing of analog input values. A maximum of 32 AVRG instructions can be programmed in a user program.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Sampling data	X	X	X	X	X	X	X	—	—	—
S2 (Source 2)	Sampling end input	X	X	X	X	—	—	—	—	—	—
S3 (Source 3)	Sampling cycles (scan times)	X	X	X	X	X	X	X	—	0-65,535	—
D1 (Destination 1)	First device address to store results	—	—	—	—	—	—	X	—	—	—
D2 (Destination 2)	Sampling completion output	—	X	▲	—	—	—	—	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D2.

When T (timer) or C (counter) is used as S1 or S3, the timer/counter current value (TC or CC) is displayed.

When F (float) data is selected, only data registers can be designated as S1 and D1.

While input is on, the AVRG instruction is executed in each scan. When the quantity of sampling cycles (scan times) designated by device S3 is 1 through 65,535, sampling data designated by device S1 is processed in each scan. When the designated sampling cycles have been completed, the average value of the sampling data is set to the device designated by D1 (data type W or I) or D1·D1+1 (data type D, L, or F). The maximum value of the sampling data is set to the next device, D1+1 (data type W or I) or D1+2·D1+3 (data type D, L, or F). The minimum value of the sampling data is set to the next device, D1+2 (data type W or I) or D1+4·D1+5 (data type D, L, or F). The sampling completion output designated by device D2 is turned on.

When the quantity of sampling cycles designated by device S3 is 0, sampling is started when the input to the AVRG instruction is turned on, and stopped when the sampling end input designated by device S2 is turned on. Then, the average, maximum, and minimum values are set to 3 devices starting with device designated by D1.

When the sampling exceeds 65,535 cycles, the average, maximum, and minimum values at this point are set to 3 devices starting with device designated by D1, and sampling continues.

When the sampling end input is turned on before the sampling cycles designated by device S3 have been completed, sampling is stopped and the results at this point are set to 3 devices starting with device designated by D1.

The average value is calculated to units, rounding the fractions of one decimal place.

When the sampling end input is not used, designate an internal relay or another valid device as a dummy for source device S2.

When F (float) data is selected and S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. When an error occurs, incorrect S1 data are skipped. Average, maximum, and minimum values are calculated from correct S1 data, and set to 3 devices starting with the device designated by D1. For details about user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Data Types

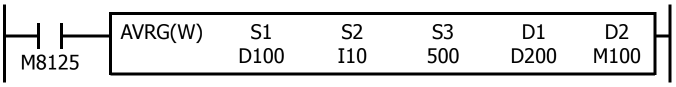
W (word)	X
I (integer)	X
D (double word)	X
L (long)	X
F (float)	X

When a bit device such as I (input), Q (output), M (internal relay), or R (shift register) is assigned as the source, 16 points (word or integer data) or 32 points (double-word or long data) are used.

When a word device such as T (timer), C (counter), or D (data register) is assigned as the source, 1 point (word or integer data) or 2 points (double-word, long, or float data) are used.

Example: AVRG

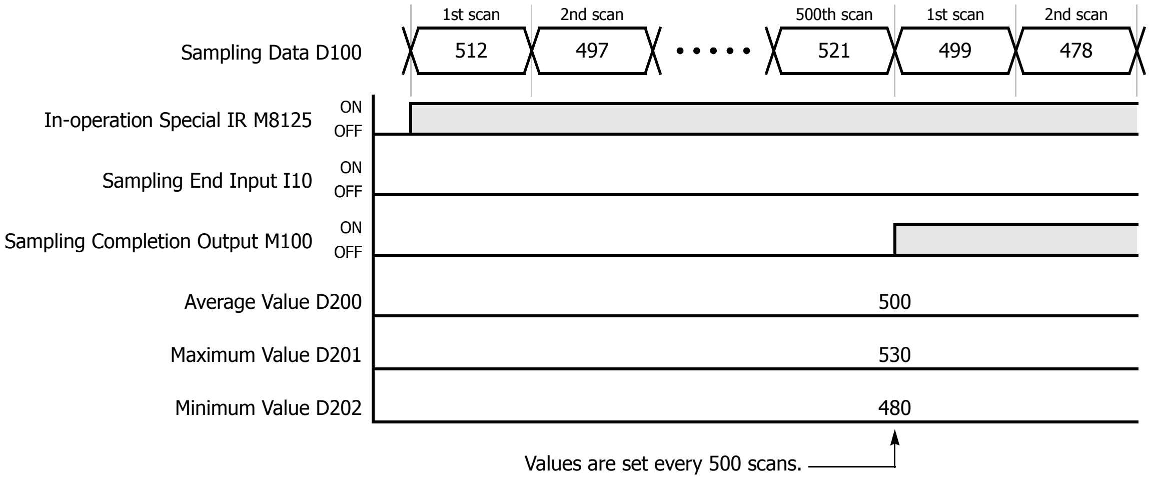
The following example demonstrates a program that calculates the average values of data register D100 and store the result to data register D200 every 500 scans.



M8125 is the in-operation output special internal relay.

When the sampling end input does not turn on

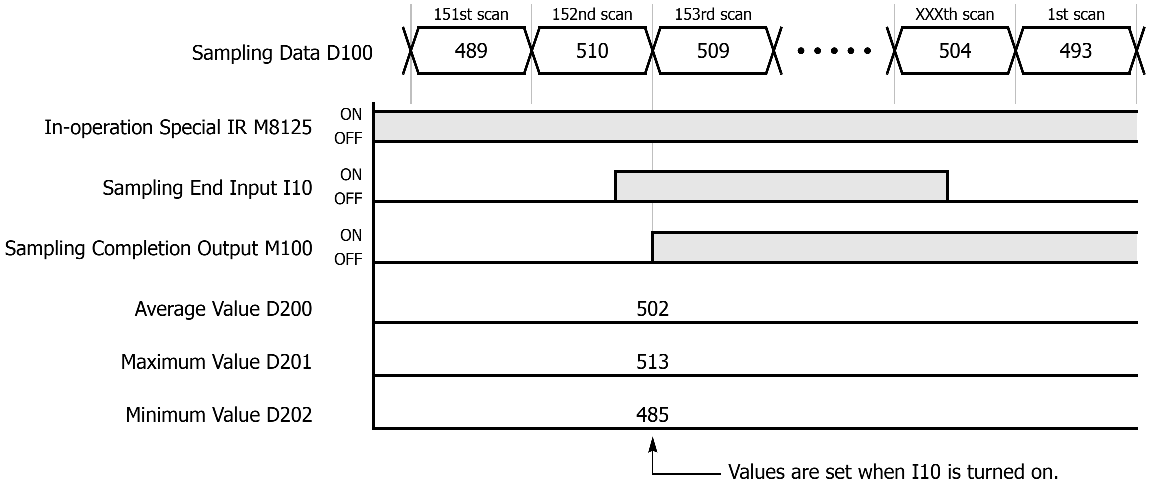
While sampling end input I10 is off, the average, maximum, and minimum values are calculated every 500 scans and stored to data registers D200, D201, and D202, respectively. Sampling completion output M100 is set every 500 scans.



Values are set every 500 scans.

When the sampling end input turns on

When sampling end input I10 turns on, the average, maximum, and minimum values at this point are stored to data registers D200, D201, and D202, respectively. Sampling completion output M100 is also set. When sampling end input I10 turns off, sampling resumes starting at the first scan.



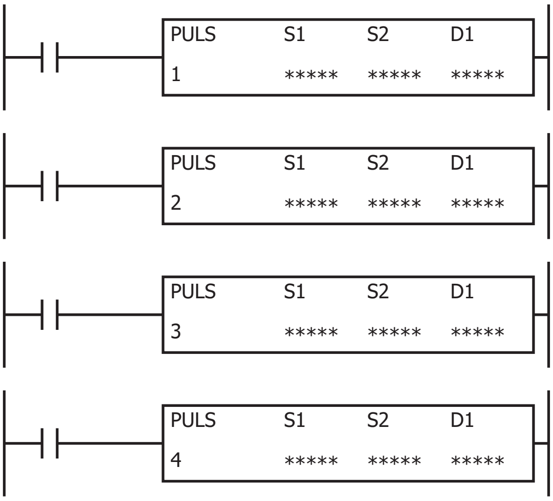
Values are set when I10 is turned on.

18: Pulse Output Instructions

Introduction

This chapter describes the pulse output instructions that output pulses of a specified frequency from the pulse outputs.

PULS (Pulse Output)



The PULS instruction outputs pulses of the specified frequency from the pulse outputs with a fixed duty cycle.

When the input is on, pulses are output with a fixed duty cycle according to the control register settings specified by S1.

When the initialization input specified by S2 is turned on, the initial values configured in the WindLDR **PULS** dialog box, on the **Settings** tab, are stored in the control registers.

The pulse control information (output on/output complete/error) is stored in the internal relays specified by D1 as the operation status.

Note: Configure multiple PULS (pulse output), PWM (pulse width modulation), RAMP (trapezoidal control), and ARAMP (RAMP with table) instructions so that they do not share the same output.
However, the ZRN (zero return) instruction can be configured with the same output as the PULS (pulse output), PWM (pulse width modulation), RAMP (trapezoidal control), and ARAMP (RAMP with table) instructions.

Notes:

- If a pulse output instruction is simultaneously executed with the same output, a user program execution error will occur.
  Error code 48 will be stored in D8006 and instructions that are executed later will be canceled.
- The PULS instruction cannot be used in an interrupt program. If used in an interrupt program, a user program execution error will occur.
  Error code 18 will be stored in D8006 and instruction execution will be canceled.
- If a pulse output instruction is executed with the relay output type, a user program execution error will occur.
  Error code 19 is stored in D8006 and instruction execution is canceled.
- For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Devices

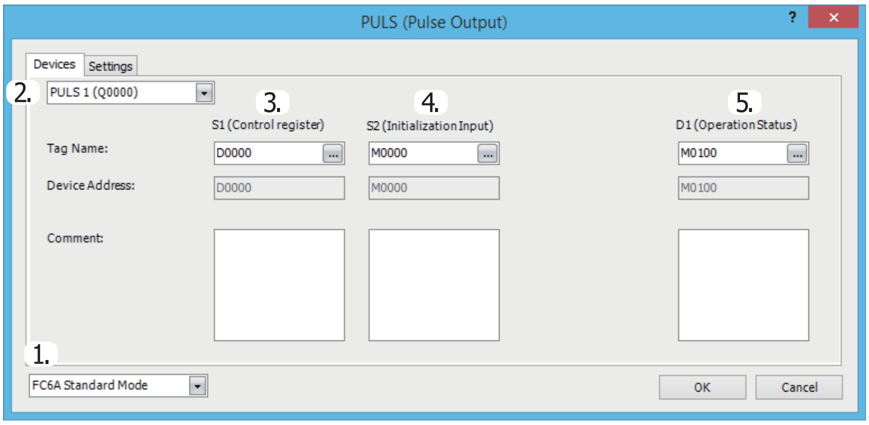
Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Control register	—	—	—	—	—	—	X[*1]	—	—	—
S2 (Source 2)	Initialization input	X	—	X	—	—	—	—	—	—	—
D1 (Destination 1)	Operation status	—	—	X[*2]	—	—	—	—	—	—	—

*1 Special data registers cannot be used.

*2 Special internal relays cannot be used. Only 0 can be specified as the first digit of the internal relay number. 1 to 7 cannot be specified.

Settings

■ Devices tab



	S1 (Control register)	S2 (Initialization Input)	D1 (Operation Status)
Tag Name:	D0000	M0000	M0100
Device Address:	D0000	M0000	M0100

1. Select Mode

Selects the configuration mode. **FC6A Standard Mode** or **FC5A (except FC5A-D12X1E) Compatible Mode** can be selected.

Select **FC5A (except FC5A-D12X1E) Compatible Mode** to use the FC5A Series MICROSmart PULS instruction specification. When changing the PLC type from the FC5A/FC4A Series MICROSmart, **FC5A (except FC5A-D12X1E) Compatible Mode** is automatically selected.

The lowest frequency that can be used with **FC5A (except FC5A-D12X1E) Compatible Mode** is 20 Hz*1.

*1 The lower limit value for the frequency that the FC5A Series MICROSmart can output is 10 Hz, but in compatibility mode this is 20 Hz.

Notes:

- The lowest frequency of pulses that can be output in FC5A (except FC5A-D12X1E) Compatible Mode is 20 Hz. A frequency lower than 20 Hz cannot be output. If such a value is specified, the pulse frequency designation error will occur.
- The highest frequency of pulses that can be output with PULS3 in FC5A (except FC5A-D12X1E) Compatible Mode is 5 kHz. A frequency higher than 5 kHz cannot be output. If such a value is specified, the pulse frequency error will occur.

The rest of this section is written under the assumption that **FC6A Standard Mode** has been selected.

Note: For details on the FC5A (except FC5A-D12X1E) Compatible Mode settings, refer to the PULS instruction in Chapter 15 "Pulse Output Instructions" in the "FC5A Series MICROSmart Pentra User's Manual Advanced Volume".

2. Select instruction

This item selects which PULS instruction to use ("PULS1", "PULS2", "PULS3", or "PULS4").

The pulse output and the frequency that can be set are determined by the instruction and the CPU module type.

All-in-One CPU module

Instruction	Pulse Output	Configurable Frequency
PULS1	Q0	15 Hz to 100 kHz (increments of 1 Hz)
PULS2	Q1	
PULS3	Q2	15 Hz to 5 kHz (increments of 1 Hz)
PULS4	Q3	

CAN J1939 All-in-One CPU module/Plus CPU module

Instruction	Pulse Output	Configurable Frequency
PULS1	Q0	15 Hz to 100 kHz (increments of 1 Hz)
PULS2	Q2	
PULS3	Q4	
PULS4	Q6	

3. S1 (source 1): Control register

S1 specifies the first data register of the data registers to use with PULS1, PULS2, PULS3, or PULS4 instructions.
Starting from the specified data register, 8 consecutive data registers are used.
Specify the first data register so that the device range is not exceeded.
The operation mode and pulse frequency settings differ by the CPU module type and the instruction.

Storage destination	Function	Setting		Reference
		All-in-One CPU Module	CAN J1939 All-in-One CPU Module/Plus CPU Module	
Starting number+0	Output pulse frequency (high word)*1	PULS1, PULS2: 15 to 100,000 (increments of 1 Hz) PULS3, PULS4: 15 to 5,000 (increments of 1 Hz)	PULS1 to PULS4: 15 to 100,000 (increments of 1 Hz)	"6. Output pulse frequency" on page 18-4
Starting number+1	Output pulse frequency (low word)*1			
Starting number+2	Preset value (high word)*1	1 to 100,000,000 pulses		"8. Preset value" on page 18-4
Starting number+3	Preset value (low word)*1			
Starting number+4	Current value (high word)*1	1 to 100,000,000 pulses		"9. Current value" on page 18-4
Starting number+5	Current value (low word)*1			
Starting number+6	Error status	0 to 4		"10. Error status" on page 18-4
Starting number+7	**Reserved**			

*1 The upper and lower data registers change according to the 32-bit data storage method specified.
For details, see "32-bit Data Storage" on page 3-9.

4. S2 (source 2): Initialization Input

S2 specifies the initialization input. When the initialization input is turned on, the initial values configured in the WindLDR **PULS** dialog box, on the **Settings** tab, are stored in the control registers. An external input or an internal relay can be specified.

When the initialization input is on, the initial values are written to the data registers with each scan. (Even when the PULS instruction is not executed (when not on), if the initialization input is turned on, the initial values are stored in the data registers.) To only initialize the values one time, use the initialization input in combination with the SOTU (single output up) instruction or the SOTD (single output down) instruction.

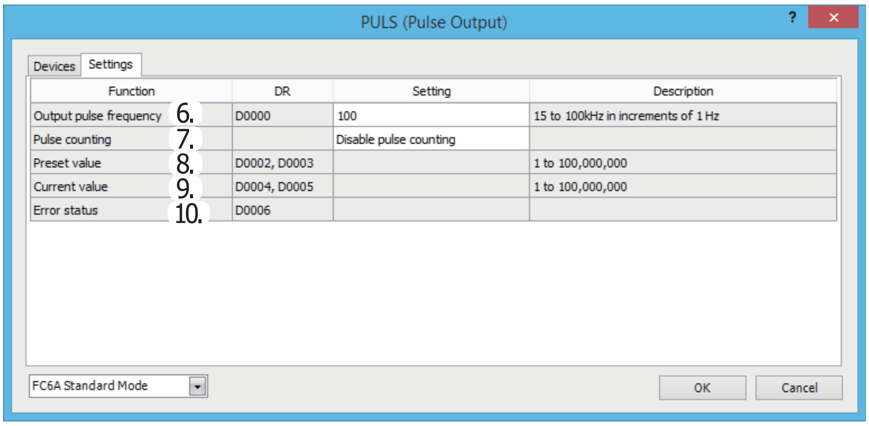
5. D1 (destination 1): Operation Status

D1 specifies the first internal relay of the internal relays that store the operation status.
Starting from the specified internal relay, 3 sequential internal relays are used.
Specify the first internal relay so that the device range is not exceeded.

Storage Destination	Function	Setting	
Starting number+0	Pulse output ON	0: Pulse output OFF 1: Pulse output ON	This relay turns on during pulse output. This relay turns off when pulse output stops. This relay turns off when the specified number of pulses are output and output ends.
Starting number+1	Pulse output complete	0: Pulse output not complete 1: Pulse output complete	This relay turns on when pulse output is complete. This relay turns off when pulse output starts.
Starting number+2	Overflow	0: None 1: An overflow has occurred	When pulse counting is enabled, this relay turns on when a pulse is output that exceeds the configured preset value.

■ Settings tab

This tab configures the operation of the PULS instruction.



Function	DR	Setting	Description
Output pulse frequency	D0000	100	15 to 100kHz in increments of 1 Hz
Pulse counting		Disable pulse counting	
Preset value	D0002, D0003		1 to 100,000,000
Current value	D0004, D0005		1 to 100,000,000
Error status	D0006		

6. Output pulse frequency

Specifies the pulse frequency. The output frequency error is ±5%.

The setting differs by the CPU module type and the instruction.

CPU Module Type	Instruction	Configurable Range	
		Setting Value	Frequency
All-in-One CPU module	PULS1, PULS2	15 to 100,000	15 Hz to 100 kHz (increments of 1 Hz)
	PULS3, PULS4	15 to 5,000	15 Hz to 5 kHz (increments of 1 Hz)
CAN J1939 All-in-One CPU module/ Plus CPU module	PULS1 to PULS4	15 to 100,000	15 Hz to 100 kHz (increments of 1 Hz)

7. Pulse counting

Enables or disables pulse counting.

If **Enable pulse counting** is specified, the number of pulses specified by the preset value (8) is output.

If **Disable pulse counting** is specified, pulses are continuously output while the PULS instruction input is on.

8. Preset value

When the setting for 7. Pulse counting is **Enable pulse counting**, this setting specifies the number of pulses to output.

9. Current value

Specifies the device (data register) that will store the number of pulses that were output.

The current value is updated when the PULS instruction is executed at each scan.

10. Error status

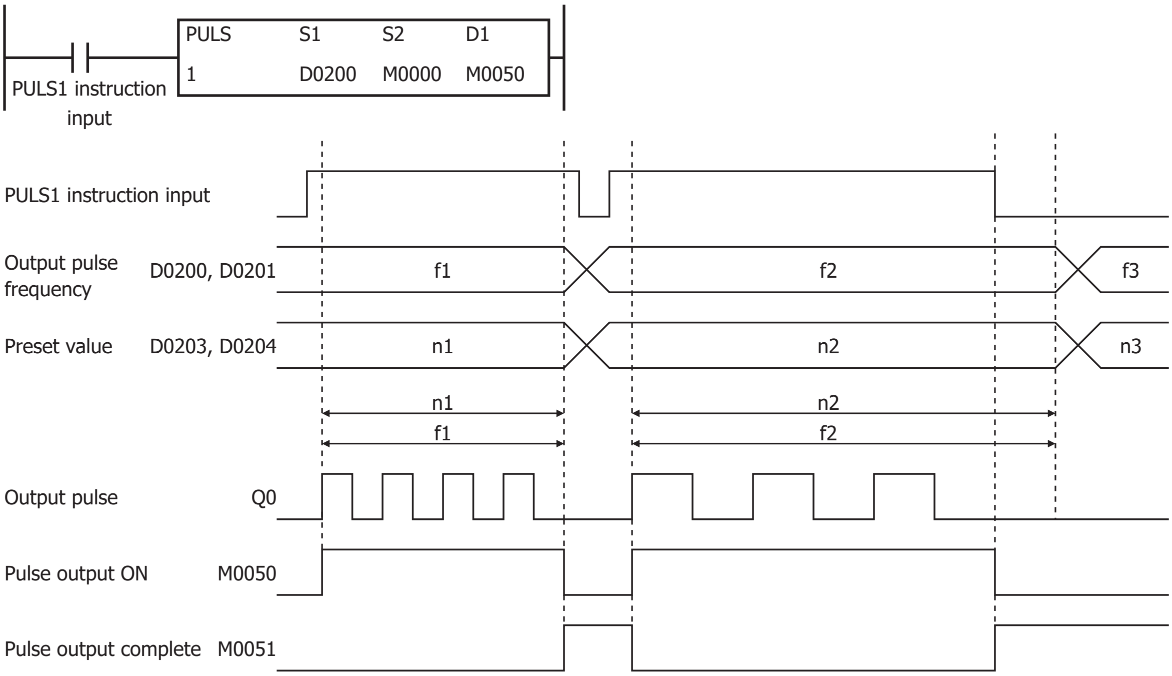
Outputs the error code that corresponds to the content of an error when there is an error in the settings. If a configuration error occurs when the PULS instruction input changes from off to on, a user program execution error will occur and error code 20 will be stored in D8006.

Error Code	Status	Description		
0	Normal	—		
2	Pulse frequency designation error	All-in-One CPU module	PULS1, PULS2:	The pulse frequency was not set between 15 and 100,000.
			PULS3, PULS4:	The pulse frequency was not set between 1 and 5,000.
		CAN J1939 All-in-One CPU module/Plus CPU module	PULS1 to PULS4:	The pulse frequency was not set between 15 and 100,000.
4	Preset value designation error	If **Enable pulse counting** is specified, the preset value was not set between 1 and 100,000,000.		

Examples: PULS

PULS1 instruction (pulse counting enabled) timing chart

[PULS1 instruction, S1 is specified as D0200, D1 is specified as internal relay M0050]



f1, f2, f3 = frequency
n1, n2, n3 = output pulse count

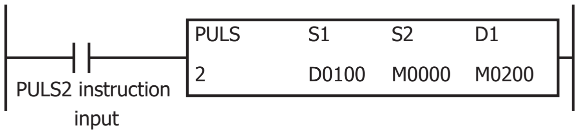
When the PULS1 instruction input changes from off to on, M0050 turns on and pulses with the frequency configured by D0200 and D0201 are output. When the number of pulses configured by D202 and D203 are output, pulse output stops. If the value of D0200 and D0201 changes during pulse output, pulses are output with the frequency based on that value.

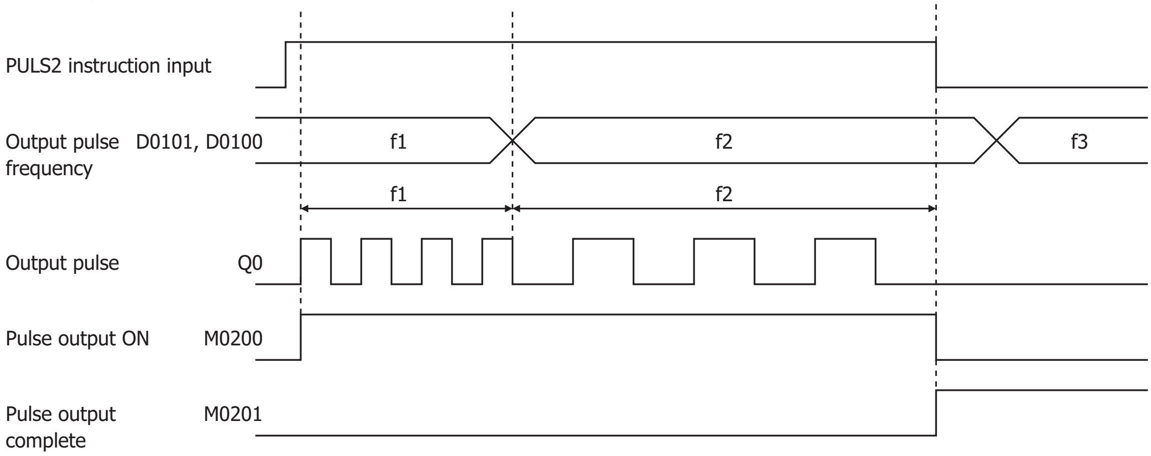
When the PULS1 instruction input changes from on to off, M0050 turns off and M0051 turns on at the same time.

The changes from the initialization input are not reflected while the PULS1 instruction input is on. If you wish to initialize the data registers with the initialization input, turn the initialization input on after turning off the input.

PULS2 instruction (pulse counting disabled) timing chart

[PULS2 instruction, S1 is specified as D0100, D1 is specified as internal relay M0200]





f1, f2, f3 = pulse frequency

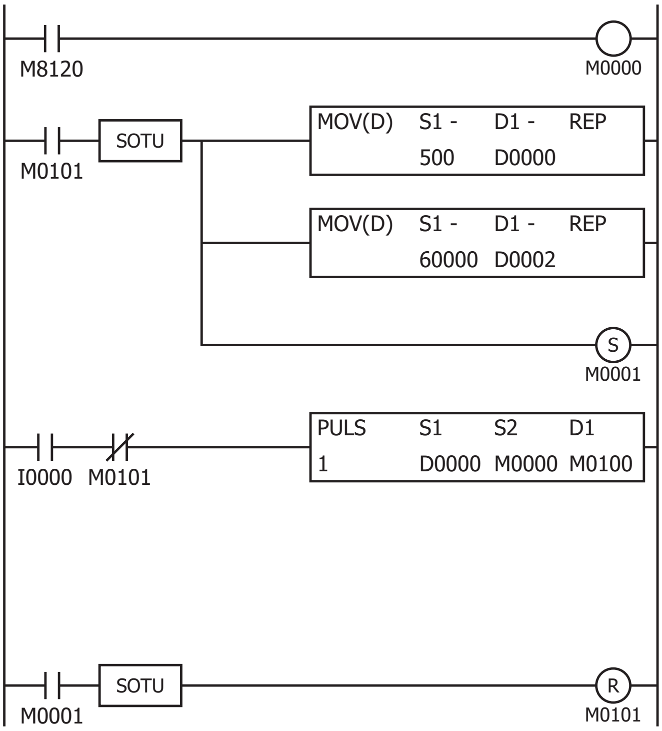
When the PULS2 instruction input changes from off to on, M0200 turns on and pulses with the frequency configured by D0101 and D0100 are output. If the value of D0101 and D0100 changes during pulse output, pulses are output with the frequency based on that value.

When the PULS2 instruction input changes from on to off, M0200 turns off and M0201 turns on at the same time.

The changes from the initialization input are not reflected while the PULS2 instruction input is on. If you wish to initialize the data registers with the initialization input, turn the initialization input on after turning off the input.

Sample program

This section describes an example program that outputs 5,000 pulses (each at 200 Hz) and then 60,000 pulses (each at 500 Hz) from output Q0



Turn on initialization input (M0000)

Store 500 (500 Hz) in frequency (D0000, D0001)

60,000 is stored in the preset value (D0002, D0003).

Turn on M0001 to output next pulse

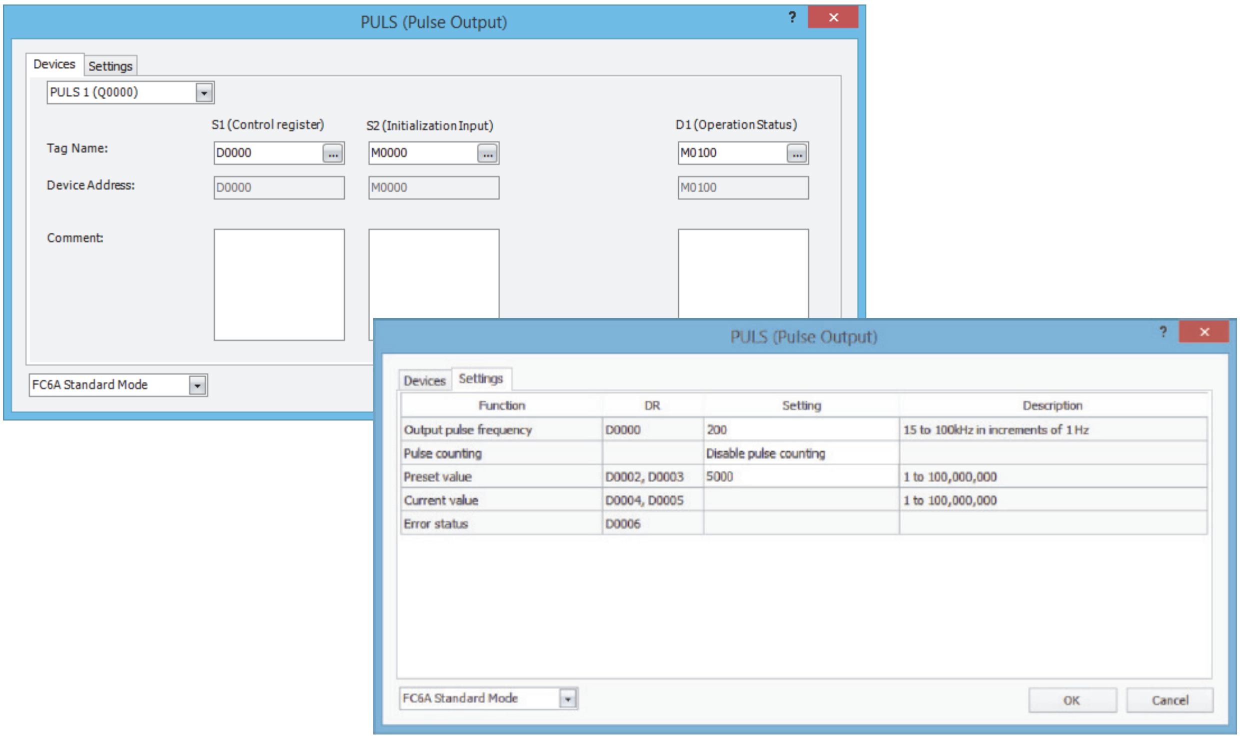
Ladder (1)

When the PULS1 instruction input (I0) turns on, a 200 Hz frequency pulse is output, and after 5,000 pulses, pulse output complete (M0101) turns on and pulse output stops.

Then ladder (1) is executed, and after the frequency and preset value are changed, a 500 Hz pulse is output, and after 60,000 pulses, pulse output complete (M0101) turns off and pulse output stops.

Pulse output complete (M0101) turns off (pulse output not complete)

Settings



PULS (Pulse Output)

Devices | Settings

PULS 1 (Q0000)

	S1 (Control register)	S2 (Initialization Input)	D1 (Operation Status)
Tag Name:	D0000	M0000	M0100
Device Address:	D0000	M0000	M0100
Comment:			

FC6A Standard Mode

PULS (Pulse Output)

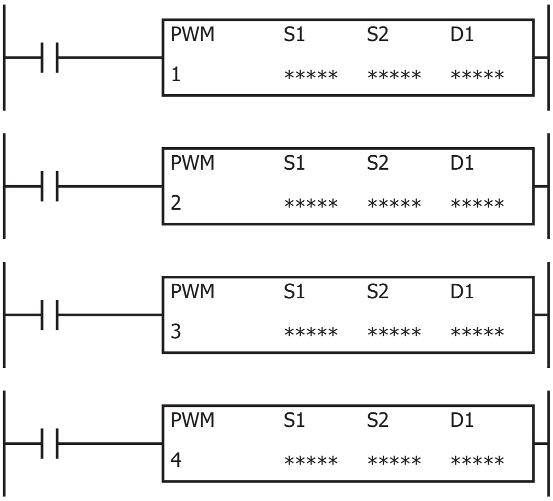
Devices | Settings

Function	DR	Setting	Description
Output pulse frequency	D0000	200	15 to 100kHz in increments of 1 Hz
Pulse counting		Disable pulse counting	
Preset value	D0002, D0003	5000	1 to 100,000,000
Current value	D0004, D0005		1 to 100,000,000
Error status	D0006		

FC6A Standard Mode

Function	Device Address	Setting Value	Details
Output pulse frequency	D0000	200	200 Hz
Pulse counting	—	1	Enable pulse counting
Preset value	D0002, D0003	5000	Preset value=5,000

PWM (Variable Duty Cycle Pulse Output)



The PWM instruction outputs pulses of the specified frequency and duty cycle from the output.

When the input is on, pulses are output according to the control register settings specified by S1.

The pulse control information (output on/output complete/error) is stored in the internal relays specified by D1 as the operation status.

When the initialization input specified by S2 is turned on, the initial values configured in the WindLDR **PWM (Pulse Width Modulation)** dialog box, on the **Settings** tab, are stored in the control registers.

Notes:

- If a pulse output instruction is simultaneously executed with the same output, a user program execution error will occur.
  Error code 48 will be stored in D8006 and instructions that are executed later will be canceled.
- The PWM instruction cannot be used in an interrupt program. If used in an interrupt program, a user program execution error will occur.
  Error code 18 will be stored in D8006 and instruction execution will be canceled.
- If a pulse output instruction is executed with the relay output type, a user program execution error will occur.
  Error code 19 is stored in D8006 and instruction execution is canceled.
- For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Devices

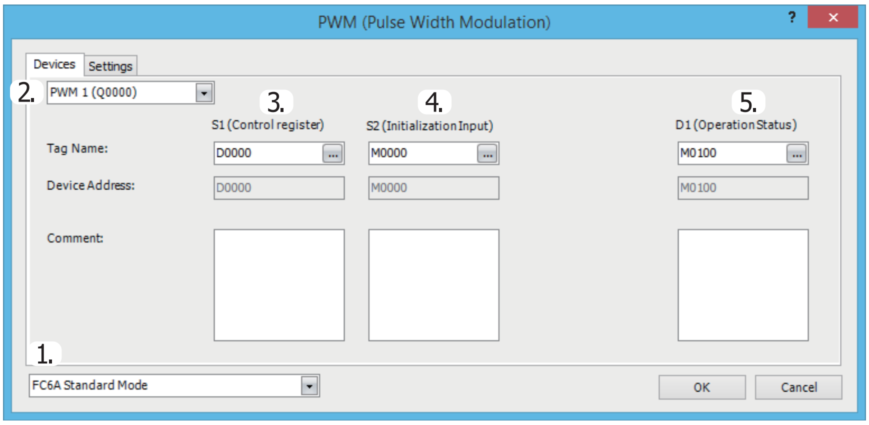
Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Control register	—	—	—	—	—	—	X*1	—	—	—
S2 (Source 2)	Initialization input	X	—	X	—	—	—	—	—	—	—
D1 (Destination 1)	Operation status	—	—	X*2	—	—	—	—	—	—	—

*1 Special data registers cannot be used.

*2 Special internal relays cannot be used. Only 0 can be specified as the first digit of the internal relay number. 1 to 7 cannot be specified.

Settings

■ Devices tab


PWM (Pulse Width Modulation)

Devices | Settings

PWM 1 (Q0000)

	S1 (Control register)	S2 (Initialization Input)	D1 (Operation Status)
Tag Name:	D0000	M0000	M0100
Device Address:	D0000	M0000	M0100
Comment:			

FC6A Standard Mode

1. Select Mode

Selects the configuration mode. **FC6A Standard Mode**, **FC4A Compatible Mode**, **FC5A (except FC5A-D12X1E) Compatible Mode**, or **FC5A-D12X1E Compatible Mode** can be selected.

To use the PWM instruction with the FC4A Series MICROSmart, the FC5A Series MICROSmart, or the FC5A-D12 PWM instruction specification, select one of the FC compatibility modes. When changing the PLC type from the FC4A Series MICROSmart, **FC4A Compatible Mode** is automatically selected. When changing the PLC type from the FC5A or FC5A-D12, **FC5A (except FC5A-D12X1E) Compatible Mode** or **FC5A-D12X1E Compatible Mode** is automatically selected.

The frequency that can be specified in FC compatibility mode is an approximate value of the FC4A, FC5A, and FC5A-D12 PWM instruction frequencies.

Note: The pulse frequencies that can be output in FC compatibility mode are as follows.

Operation Mode	FC4A Compatible Mode	FC5A (except FC5A-D12X1E) Compatible Mode	FC5A-D12X1E Compatible Mode
Operation mode 0	15 Hz (fixed)	15 Hz (fixed)	15 Hz (fixed)
Operation mode 1	27 Hz (fixed)	46 Hz (fixed)	61 Hz (fixed)
Operation mode 2	218 Hz (fixed)	366 Hz (fixed)	488 Hz (fixed)

The rest of this section is written under the assumption that **FC6A Standard Mode** has been selected.

Notes:

- For details on the FC4A Compatible Mode settings, refer to the PWM instruction in Chapter 2 "Instructions" in the "FC4A Series MICROSmart User's Manual".
- For details on the settings in FC5A (except FC5A-D12X1E) Compatible Mode and FC5A-D12X1E Compatible Mode, refer to the PWM instruction in Chapter 13 "Pulse Instructions" in the FC5A Series MICROSmart Pentra User's Manual Advanced Volume.

2. Select instruction

This item selects which PWM instruction to use ("PWM1", "PWM2", "PWM3", or "PWM4").

The pulse output that can be set is determined by the instruction and the CPU module type.

All-in-One CPU module

Instruction	Pulse Output	Configurable Range	
		Frequency	Duty Cycle
PWM1	Q0[*1*3]	15 Hz to 5 kHz (increments of 1 Hz)	0.1 to 100.0 (increments of 0.1%)
PWM2	Q1[*1*3]		
PWM3	Q2[*2*3]		
PWM4	Q3[*2*3]		

*1 When the output is Q0 or Q1 and the calculated OFF time is shorter than 15 μs, the pulse duty cycle is adjusted so that the OFF time becomes 15 μs, and then the pulse is output.

*2 When the output is Q2 or Q3, configure the output pulse frequency and pulse duty cycle so that ON time and OFF time are greater than or equal to 100 μs.

*3 For details on the output delay time of each output, see Chapter 2 "Product Specifications" in the "FC6A Series MICROSmart User's Manual".

CAN J1939 All-in-One CPU module/Plus CPU module

Instruction	Pulse Output	Configurable Range	
		Frequency	Duty Cycle
PWM1	Q0[*1][*2]	15 Hz to 5 kHz (increments of 1 Hz)	0.1 to 100.0 (increments of 0.1%)
PWM2	Q2[*1][*2]		
PWM3	Q4[*1][*2]		
PWM4	Q6[*1][*2]		

[*1] When the output is Q0 or Q1 and the calculated OFF time is shorter than 15 µs, the pulse duty cycle is adjusted so that the OFF time becomes 15 µs, and then the pulse is output.

[*2] For details on the output delay time of each output, see Chapter 2 "Product Specifications" in the "FC6A Series MICROSmart User's Manual".

3. S1 (source 1): Control register

S1 specifies the first data register of the data registers to use with PWM1, PWM2, PWM3, or PWM4 instructions.
Starting from the specified data register, 8 consecutive data registers are used.
Specify the first data register so that the device range is not exceeded.

Storage Destination	Function	Setting	Reference
Starting number+0	Output pulse frequency	15 to 5,000 (increments of 1 Hz)	"6. Output pulse frequency" on page 18-11
Starting number+1	Pulse duty cycle (ON ratio)	1 to 1,000 (increments of 0.1%)	"7. Pulse width ratio" on page 18-11
		1 to 100 (increments of 1%)	
Starting number+2	Preset value (high word)[*1]	1 to 100,000,000 pulses	"9. Preset value" on page 18-11
Starting number+3	Preset value (low word)[*1]		
Starting number+4	Current value (high word)[*1]	1 to 100,000,000 pulses	"10. Current value" on page 18-11
Starting number+5	Current value (low word)[*1]		
Starting number+6	Error status	0 to 4	"11. Error status" on page 18-11
Starting number+7	**Reserved**		

[*1] The upper and lower data registers change according to the 32-bit data storage method specified.
For details, see "32-bit Data Storage" on page 3-9.

4. S2 (source 2): Initialization Input

S2 specifies the initialization input. When the initialization input S2 is turned on, the initial values configured in the WindLDR **PWM (Pulse Width Modulation)** dialog box, on the **Settings** tab, are stored in the control registers. An external input or an internal relay can be specified.

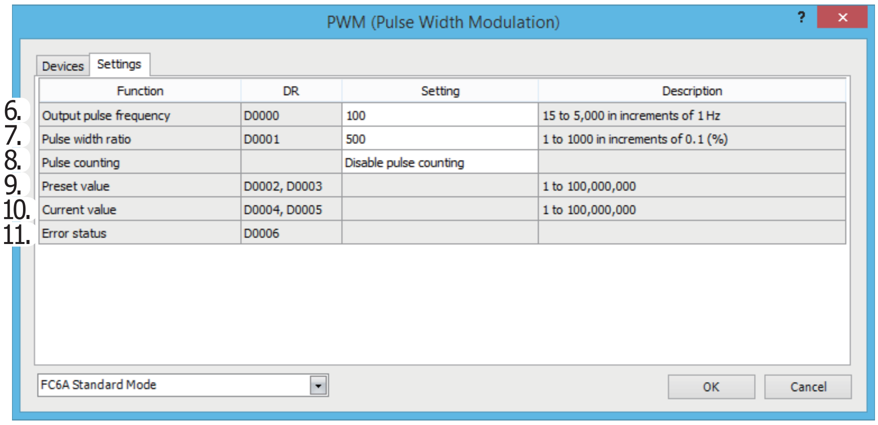
When the initialization input is on, the initial values are written to the data registers with each scan. To only initialize the values one time, use the initialization input in combination with the SOTU (single output up) instruction or the SOTD (single output down) instruction.

5. D1 (destination 1): Operation Status

D1 specifies the first internal relay of the internal relays to use with PWM instructions.
Starting from the specified internal relay, 3 sequential internal relays are used.
Specify the first internal relay so that the device range is not exceeded.

Storage Destination	Function	Setting	
Starting number+0	Pulse output ON	0: Pulse output OFF 1: Pulse output ON	This relay turns on during pulse output. This relay turns off when pulse output stops. This relay turns off when the specified number of pulses are output and output ends.
Starting number+1	Pulse output complete	0: Pulse output not complete 1: Pulse output complete	This relay turns on when pulse output is complete. This relay turns off when pulse output stops.
Starting number+2	Overflow	0: None 1: An overflow has occurred	When pulse counting is enabled, this relay turns on when a pulse is output that exceeds the configured preset value.

■ Settings tab



Function	DR	Setting	Description
Output pulse frequency	D0000	100	15 to 5,000 in increments of 1 Hz
Pulse width ratio	D0001	500	1 to 1000 in increments of 0.1 (%)
Pulse counting		Disable pulse counting	
Preset value	D0002, D0003		1 to 100,000,000
Current value	D0004, D0005		1 to 100,000,000
Error status	D0006		

6. Output pulse frequency

Specifies the frequency of the pulses to output.

The frequency is specified between 15 Hz and 5 kHz in 1 Hz increments.

The pulse frequency error is within ±5%.

7. Pulse width ratio

This setting specifies the ON ratio (duty cycle) for the pulse frequency to output.

If the configuration mode is **FC6A Standard Mode**, the pulse width ratio is specified between 0.1% and 100.0% in 0.1% increments. If the configuration mode is **FC4A Compatible Mode, FC5A (except FC5A-D12X1E) Compatible Mode**, or **FC5A-D12X1E Compatible Mode**, the pulse width ratio is specified between 1% and 100% in 1% increments.

8. Pulse counting

Enables or disables pulse counting.

If **Enable pulse counting** is specified, the number of pulses specified by the preset value (9) is output.

If **Disable pulse counting** is specified, pulses are continuously output while the PULS instruction input is on.

9. Preset value

When the setting for 8. Pulse counting is **Enable pulse counting**, this setting specifies the number of pulses to output.

10. Current value

Specifies the device (data register) that will store the number of pulses that were output.

The current value is updated when the PWM instruction is executed at each scan.

11. Error status

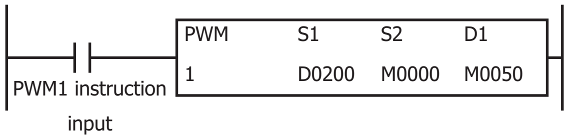
Outputs the error code that corresponds to the content of an error when there is an error in the settings. If a configuration error occurs when the PWM instruction input changes from off to on, a user program execution error will occur and error code 20 will be stored in D8006.

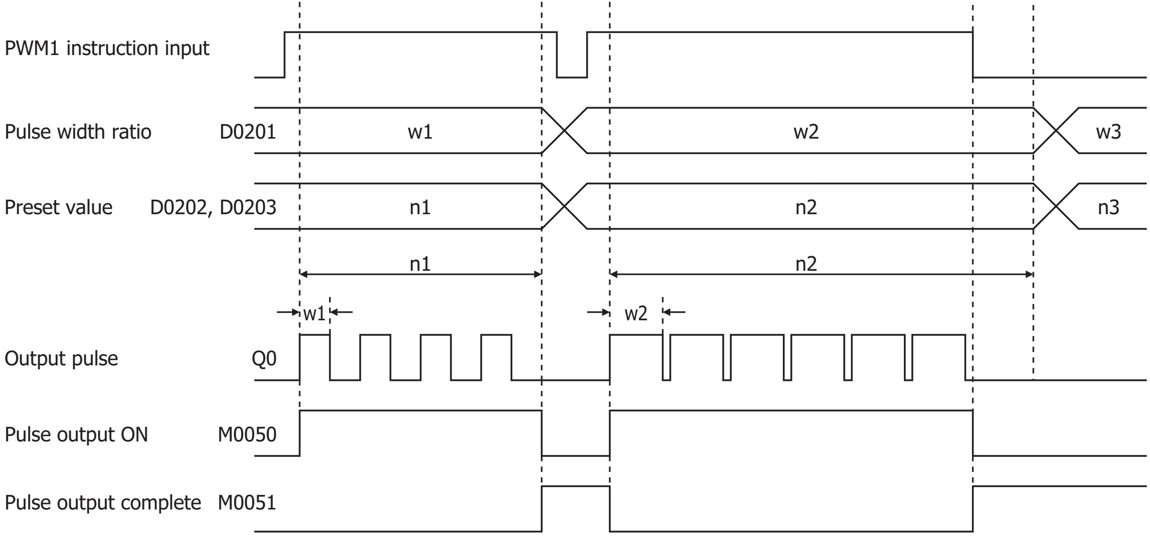
Error Code	Status	Description
0	Normal	—
1	Output frequency designation error	The pulse frequency was not set between 15 and 5,000.
2	Pulse width ratio designation error	The pulse width ratio was not set between 1 and 1,000.
4	Preset value designation error	If **Enable pulse counting** is specified, the preset value was not set between 1 and 100,000,000.

Examples: PWM

PWM1 instruction (pulse counting enabled) timing chart

[PWM1 instruction, S1 is specified as D200, D1 is specified as internal relay M0050]





w1, w2, w3 = pulse width ratio
n1, n2, n3 = output pulse count

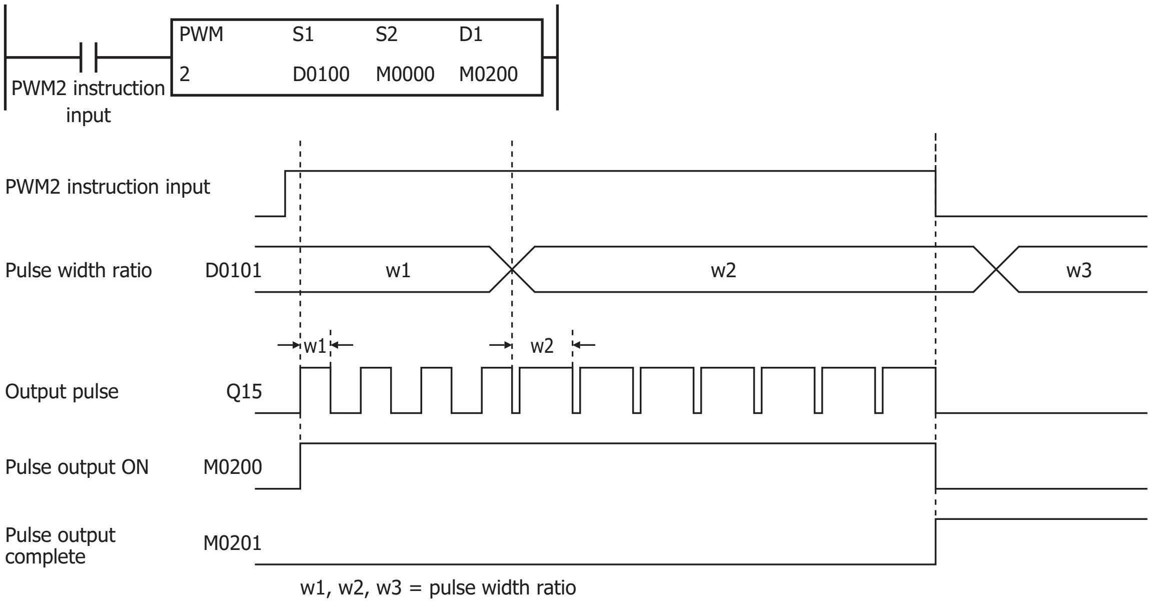
When the PWM1 instruction input changes from off to on, M0050 turns on and pulses with the width ratio configured by D0201 are output. When the number of pulses configured by D0202 and D0203 are output, pulse output stops. If the value of D0201 changes during pulse output, pulses are output with the width ratio based on that value.

When the PWM1 instruction input changes from on to off, M0050 turns off and M0051 turns on at the same time.

The changes from the initialization input are not reflected while the PWM1 instruction input is on. If you wish to initialize the data registers with the initialization input, turn the initialization input on after turning off the input.

PWM2 instruction (pulse counting disabled) timing chart

[PWM2 instruction, S1 is specified as D0100, D1 is specified as internal relay M0200]



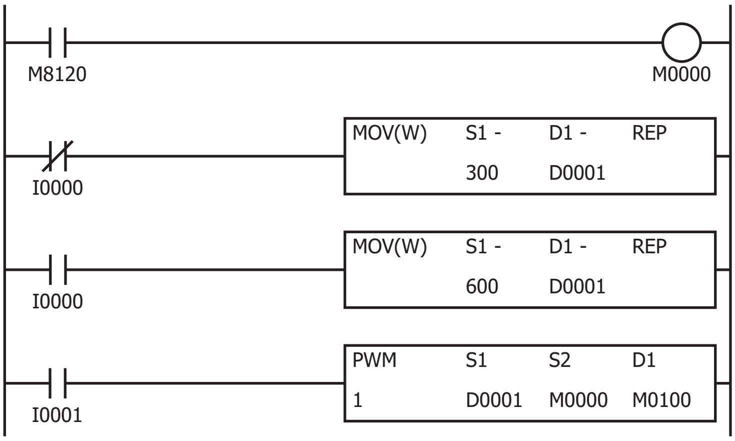
When the PWM2 instruction input changes from off to on, M0200 turns on and pulses with the width ratio configured by D0101 are output. If the value of D0101 changes during pulse output, pulses are output with the width ratio based on that value.

When the PWM2 instruction input changes from on to off, M0200 turns off and M0201 turns on at the same time.

The changes from the initialization input are not reflected while the PWM2 instruction input is on. If you wish to initialize the data registers with the initialization input, turn the initialization input on after turning off the input.

Sample program

This section describes an example program that outputs a pulse with the pulse width ratio of 30% when I0 is off, and a pulse with the pulse width ratio of 60% when I0 is on.



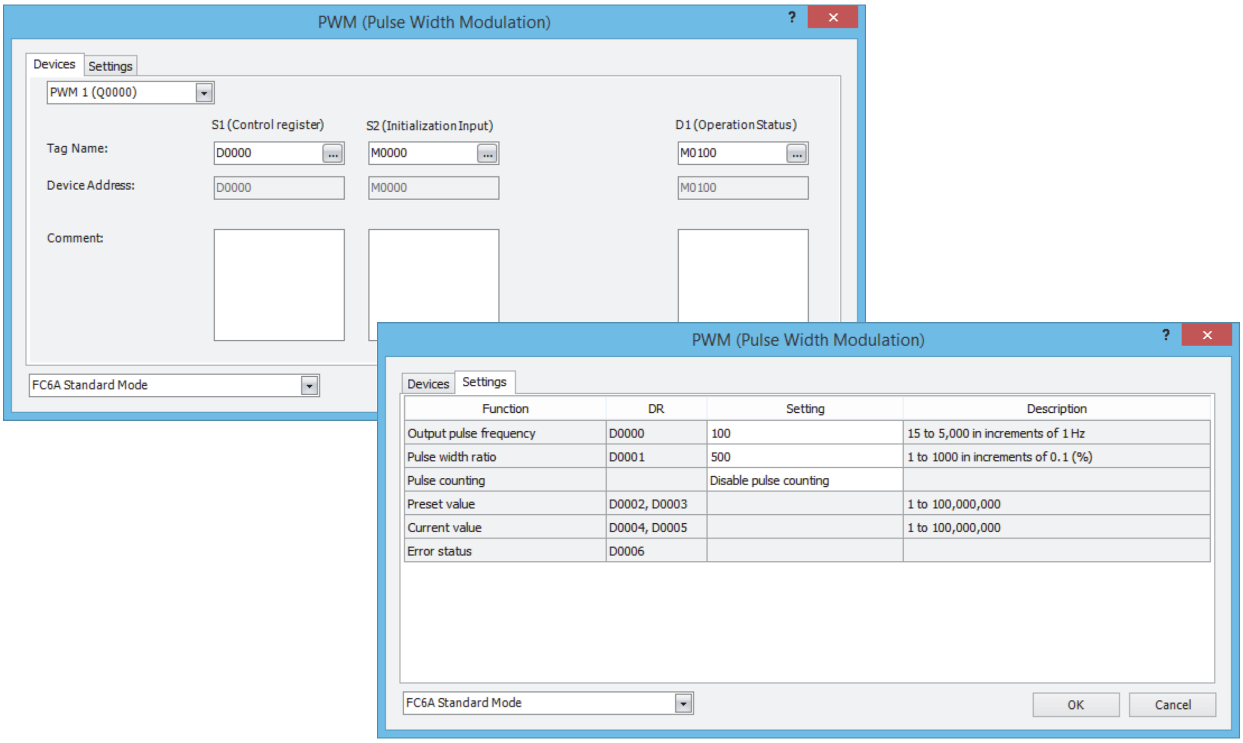
Turn on initialization input (M0000)

Set pulse width ratio (D0001) to 30 (30% ON)

Set pulse width ratio (D0001) to 60 (60% ON)

When PWM1 instruction input (I0001) turns on, start output of a pulse with the pulse width ratio of 30% or 60%

Settings



PWM (Pulse Width Modulation)

Devices | Settings

PWM 1 (Q0000)

	S1 (Control register)	S2 (Initialization Input)	D1 (Operation Status)
Tag Name:	D0000	M0000	M0100
Device Address:	D0000	M0000	M0100
Comment:			

FC6A Standard Mode

PWM (Pulse Width Modulation)

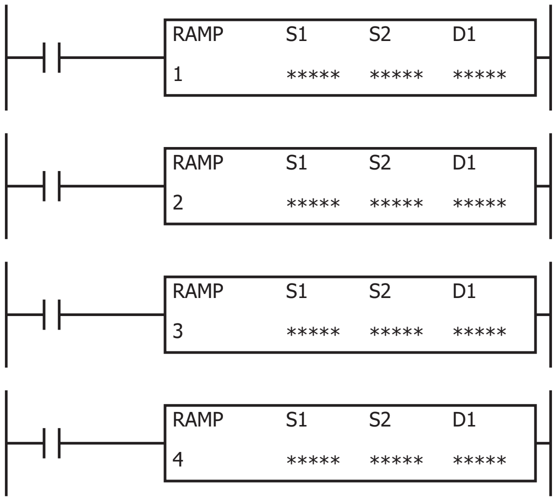
Devices | Settings

Function	DR	Setting	Description
Output pulse frequency	D0000	100	15 to 5,000 in increments of 1 Hz
Pulse width ratio	D0001	500	1 to 1000 in increments of 0.1 (%)
Pulse counting		Disable pulse counting	
Preset value	D0002, D0003		1 to 100,000,000
Current value	D0004, D0005		1 to 100,000,000
Error status	D0006		

FC6A Standard Mode

Function	Device address	Setting value	Details
Output pulse frequency	D0000	50	50 Hz
Pulse width ratio	D0001	300	30%
Pulse counting	—	Disable pulse counting	—
Preset value	D0002, D0003	—	—

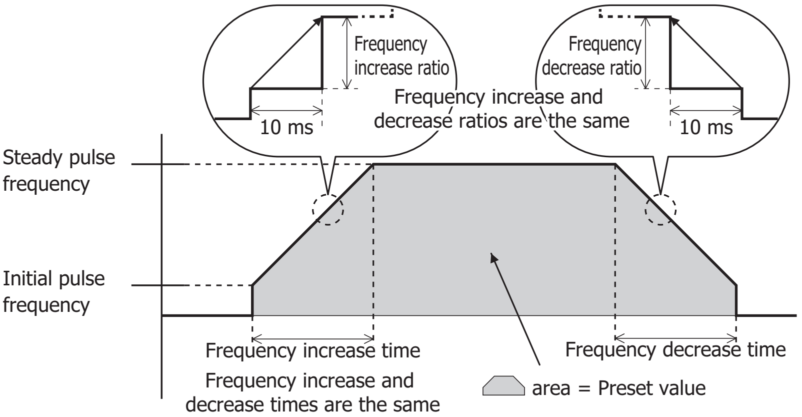
RAMP (Trapezoidal Control)



The RAMP instruction outputs pulses with a frequency change function.

When the input is on, pulses of the initial pulse frequency specified by S1 are output, and then the pulse frequency is increased by a fixed ratio until it reaches the steady pulse frequency.

After steady pulse output at the steady pulse frequency, the pulse frequency is decreased before the pulse count reaches the preset value specified by S1, and then pulse output stops when the preset value is reached.



When the initialization input specified by S2 is turned on, the initial values configured in the WindLDR **RAMP (Trapezoidal Control)** dialog box, on the **Settings** tab, are stored in the control registers.

The pulse control information (output on/output complete/error) is stored in the internal relays specified by D1 as the operation status.

Notes:

- If a pulse output instruction is simultaneously executed with the same output, a user program execution error will occur. Error code 48 will be stored in D8006 and instructions that are executed later will be canceled.
- The PULS instruction cannot be used in an interrupt program. If used in an interrupt program, a user program execution error will occur. Error code 18 will be stored in D8006 and instruction execution will be canceled.
- If a pulse output instruction is executed with the relay output type, a user program execution error will occur. Error code 19 is stored in D8006 and instruction execution is canceled.
- For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Devices

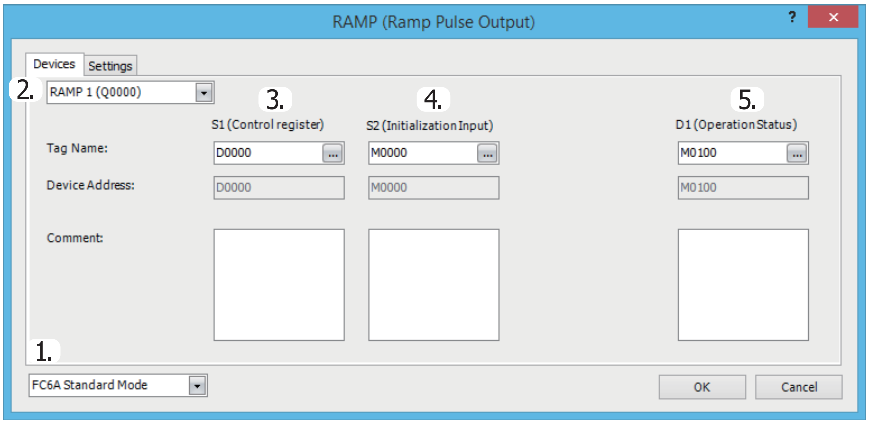
Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Control register	—	—	—	—	—	—	X*1	—	—	—
S2 (Source 2)	Initialization input	X	—	X	—	—	—	—	—	—	—
D1 (Destination 1)	Operation status	—	—	X*2	—	—	—	—	—	—	—

*1 Special data registers cannot be used.

*2 Special internal relays cannot be used. Only 0 can be specified as the first digit of the internal relay number. 1 to 7 cannot be specified.

Settings

■ Devices tab



RAMP (Ramp Pulse Output)

RAMP 1 (Q0000)

	S1 (Control register)	S2 (Initialization Input)	D1 (Operation Status)
Tag Name:	D0000	M0000	M0100
Device Address:	D0000	M0000	M0100
Comment:			

FC6A Standard Mode

1. Select Mode

Selects the configuration mode. **FC6A Standard Mode** or **FC5A (except FC5A-D12X1E) Compatible Mode** can be selected.

Select **FC5A (except FC5A-D12X1E) Compatible Mode** to use the FC5A Series MICROSmart PULS instruction specification.

When changing the PLC type from the FC5A/FC4A Series MICROSmart, **FC5A (except FC5A-D12X1E) Compatible Mode** is automatically selected.

The lowest frequency that can be used with **FC5A (except FC5A-D12X1E) Compatible Mode** is 20 Hz*1.

*1 The lower limit value for the frequency that the FC5A Series MICROSmart can output is 10 Hz, but in compatibility mode this is 20 Hz.

Notes:

- The lowest frequency of pulses that can be output in FC5A (except FC5A-D12X1E) Compatible Mode is 20 Hz. A frequency lower than 20 Hz cannot be output. If such a value is specified, the pulse frequency designation error will occur.
- The highest frequency of pulses that can be output with RAMP3 in FC5A (except FC5A-D12X1E) Compatible Mode is 5 kHz. A frequency higher than 5 kHz cannot be output. If such a value is specified, the pulse frequency error will occur.
- FC5A compatible mode does not support the S-shaped frequency change curve.

The rest of this section is written under the assumption that **FC6A Standard Mode** has been selected.

Note: For details on the FC5A (except FC5A-D12X1E) Compatible Mode settings, refer to the RAMP3 instruction in Chapter 15 "Pulse Output Instructions" in the "FC5A Series MICROSmart Pentra User's Manual Advanced Volume".

2. Select instruction

This item selects which RAMP instruction to use ("RAMP1", "RAMP2", "RAMP3" or "RAMP4").

The output and the reversible control mode and frequency that can be set differ by the instruction and CPU module type.

For limitations based on the combination of instruction, reversible control mode, and the pulse output mode, see "10. Reversible control enable" on page 18-19.

3. S1 (source 1): Control register

S1 specifies the first data register of the data registers to use with RAMP1, RAMP2, RAMP3 or RAMP4 instructions.
Starting from the specified data register, 12 consecutive data registers are used.
Specify the first data register so that the device range is not exceeded.

Storage Destination	Function	Setting		Reference
		All-in-One CPU Module	CAN J1939 All-in-One CPU Module/ Plus CPU Module	
Starting number+0	Steady pulse frequency (high word)*1	RAMP1, RAMP2: 15 to 100,000 (increments of 1 Hz) RAMP3, RAMP4: 15 to 5,000 (increments of 1 Hz)	RAMP1 to RAMP4: 15 to 100,000 (increments of 1 Hz)	"6. Steady pulse frequency" on page 18-18
Starting number+1	Steady pulse frequency (low word)*1			
Starting number+2	Initial pulse frequency (high word)*1	RAMP1, RAMP2: 15 to 100,000 (increments of 1 Hz) RAMP3, RAMP4: 15 to 5,000 (increments of 1 Hz)	RAMP1 to RAMP4: 15 to 100,000 (increments of 1 Hz)	"7. Initial pulse frequency" on page 18-18
Starting number+3	Initial pulse frequency (low word)*1			
Starting number+4	Frequency change time	10 to 10,000 ms		"8. Frequency change time" on page 18-18
Starting number+5	Control direction	0: Forward 1: Reverse		"11. Control direction" on page 18-19
Starting number+6	Preset value (high word)*1	When specify absolute position mode is disabled: 1 to 100,000,000 pulses When specify absolute position mode is enabled: -2,147,483,648 to 2,147,483,647 pulses		"13. Preset value" on page 18-20
Starting number+7	Preset value (low word)*1			
Starting number+8	Current value (high word)*1	1 to 100,000,000 pulses*2		"14. Current value" on page 18-20
Starting number+9	Current value (low word)*1			
Starting number+10	Error status	0 to 4		"15. Error status" on page 18-20
Starting number+11	**Reserved**			

*1 The upper and lower data registers change according to the 32-bit data storage method specified.
For details, see "32-bit Data Storage" on page 3-9.

*2 Stores the number of pulses that were output, regardless of whether specify absolute position mode is enabled or disabled. For details on specify absolute position mode, see "ABS (Set Absolute Position)" on page 18-68.

4. S2 (source 2): Initialization Input

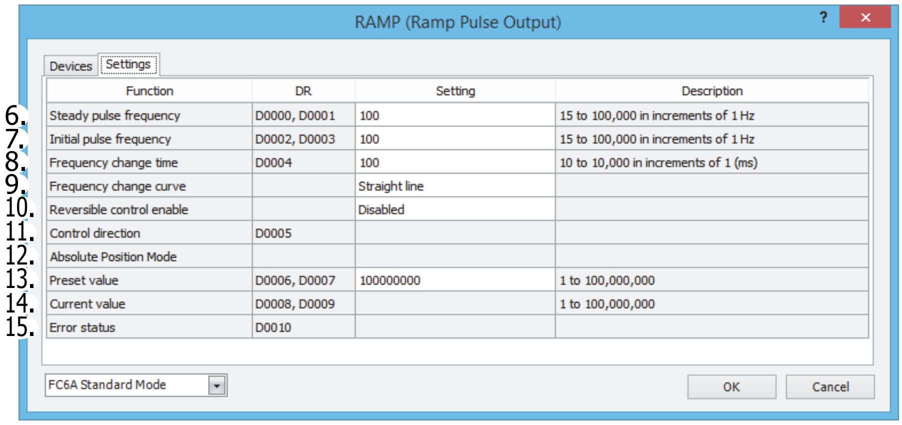
S2 specifies the initialization input.
When the initialization input S2 is turned on, the initial values configured in the WindLDR **RAMP (Ramp Pulse Output)** dialog box, on the **Settings** tab, are stored in the control registers.
An external input or an internal relay can be specified.
When the initialization input is on, the initial values are written to the data registers with each scan. To only initialize the values one time, use the initialization input in combination with the SOTU (single output up) instruction or the SOTD (single output down) instruction.

5. D1 (destination 1): Operation Status

D1 specifies the first internal relay of the internal relays to use with the RAMP instructions. Starting from the specified internal relay, 4 sequential internal relays are used. Specify the first internal relay so that the device range is not exceeded.

Storage Destination	Function	Setting	
Starting number+0	Pulse output ON	0: Pulse output OFF 1: Pulse output ON	This relay turns on during pulse output. This relay turns off when pulse output stops. This relay turns off when the specified number of pulses are output and output ends.
Starting number+1	Pulse output complete	0: Pulse output not complete 1: Pulse output complete	This relay turns on when pulse output is complete. This relay turns off when pulse output starts.
Starting number+2	Pulse output status	0: Steady pulse output 1: Changing output pulse frequency	This relay turns off when the pulse output status is steady. This relay turns on when the pulse output is changing.
Starting number+3	Overflow	0: None 1: An overflow has occurred	When pulse counting is enabled, this relay turns on when a pulse is output that exceeds the configured preset value. Pulse output continues even if an the overflow occurs during steady output or while the pulse frequency is changing. However, pulse counting is suspended at the point when the overflow occurred.

■ Settings tab



RAMP (Ramp Pulse Output)

	Function	DR	Setting	Description
6.	Steady pulse frequency	D0000, D0001	100	15 to 100,000 in increments of 1 Hz
7.	Initial pulse frequency	D0002, D0003	100	15 to 100,000 in increments of 1 Hz
8.	Frequency change time	D0004	100	10 to 10,000 in increments of 1 (ms)
9.	Frequency change curve		Straight line	
10.	Reversible control enable		Disabled	
11.	Control direction	D0005		
12.	Absolute Position Mode			
13.	Preset value	D0006, D0007	100000000	1 to 100,000,000
14.	Current value	D0008, D0009		1 to 100,000,000
15.	Error status	D0010		

FC6A Standard Mode

6. Steady pulse frequency

This setting specifies the steady pulse frequency after the pulse frequency increases. The output frequency error is ±5%.

The corresponding instruction and frequency differ by the CPU module type.

CPU Module Type	Instruction	Configurable Range	
		Setting Value	**Frequency**
All-in-One CPU Module	RAMP1, RAMP2	15 to 100,000	15 Hz to 100 kHz (increments of 1 Hz)
	RAMP3, RAMP4	15 to 50,000	15 Hz to 5 kHz (increments of 1 Hz)
CAN J1939 All-in-One CPU Module/Plus CPU Module	RAMP1 to RAMP4	15 to 100,000	15 Hz to 100 kHz (increments of 1 Hz)

7. Initial pulse frequency

This setting specifies the frequency when pulse output starts. The output frequency error is ±5%.

The corresponding instruction and frequency differ by the CPU module type.

CPU Module Type	Instruction	Configurable Range	
		Setting Value	**Frequency**
All-in-One CPU Module	RAMP1, RAMP2	15 to 100,000	15 Hz to 100 kHz (increments of 1 Hz)
	RAMP3, RAMP4	15 to 50,000	15 Hz to 5 kHz (increments of 1 Hz)
CAN J1939 All-in-One CPU Module/Plus CPU Module	RAMP1 to RAMP4	15 to 100,000	15 Hz to 100 kHz (increments of 1 Hz)

8. Frequency change time

This setting specifies the time to increase and decrease the pulse frequency.

Set the time between 10 and 10,000 ms in increments of 1 ms. The first digit of the setting is handled as zero. For example, if 144 is entered, the set value is handled as 140 ms.

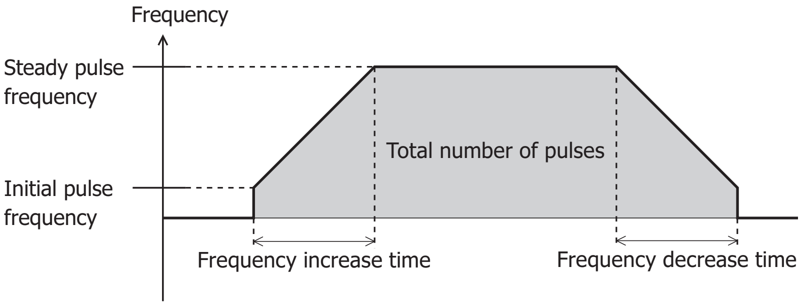
9. Frequency change curve

The frequency change curve can be selected from **Straight line** and **S-shaped curve**.

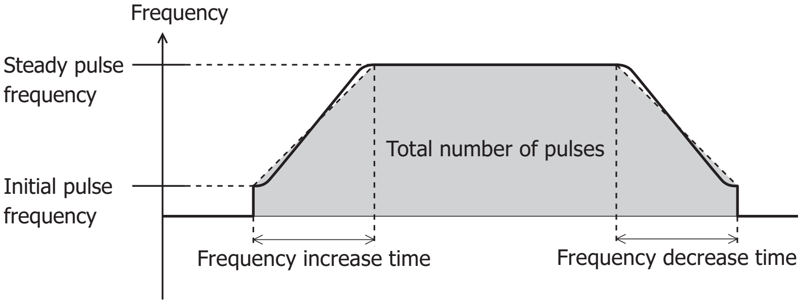
If **S-shaped curve** is selected, the initial frequency can be lower, which can suppress vibration and shocks more than **Straight line**.

This setting is supported by only the Plus CPU module transistor output type.

Straight line



S-shaped curve

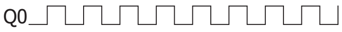
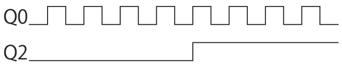
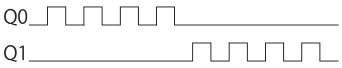


Notes:

- The S-shaped curve frequency change curve is approximated with a 3D function based on a setting value. The setting value of the approximation equation cannot be changed.
- The S-shaped curve frequency change curve changes to a linear curve in the following situations.
  - When the frequency change time is less than 100 ms
  - When the initial pulse frequency is less than 100 Hz

10. Reversible control enable

This setting enables or disables reversible control and selects the reversible control method from the following reversible control modes. There are two modes for the pulse output mode: single-pulse and dual-pulse. They can be combined with reversible control as follows. (This is an example when RAMP1 is used with the All-in-One CPU module.)

Reversible Control Enable	Operation	Pattern
Reversible control disabled	Select this option when using pulse output in a single direction. Pulse A and pulse B can be used independently.	
Reversible control **Single-pulse output**	Pulse A is used as pulse output. Pulse B on/off is used as reversible control.	
Reversible control **Dual-pulse output**	Pulse A is used as forward pulse (CW) output. Pulse B is used as reverse pulse (CCW) output.	

The outputs used on the FC6A Series MICROSmart vary based on the instruction used, the combination of the pulse output mode and reversible control, and the model used.

Instruction	Operating Condition	Output Used			
		All-in-One CPU Module		CAN J1939 All-in-One CPU Module/Plus CPU Module	
		Pulse Output	Reversible Control Output	Pulse Output	Reversible Control Output
RAMP1	Reversible control disabled	Q0	—	Q0	—
	Reversible control (single-pulse output)	Q0	Q2[*1]	Q0	Q1
	Reversible control (dual-pulse output)	Q0, Q1[*2]	—	Q0, Q1	—
RAMP2	Reversible control disabled	Q1	—	Q2	—
	Reversible control (single-pulse output)	Q1	Q3[*1]	Q2	Q3
	Reversible control (dual-pulse output)	—	—	Q2, Q3	—
RAMP3	Reversible control disabled	Q2	—	Q4	—
	Reversible control (single-pulse output)	—	—	Q4	Q5
	Reversible control (dual-pulse output)	—	—	Q4, Q5	—
RAMP4	Reversible control disabled	Q3	—	Q6	—
	Reversible control (single-pulse output)	—	—	Q6	Q7
	Reversible control (dual-pulse output)	—	—	Q6, Q7	—

[*1] When using single-pulse output mode with the All-in-One CPU module, Q2 or Q3 will be used, so an instruction that uses the same output cannot be used.

[*2] When using dual-pulse output mode with the All-in-One CPU module, Q1 will be used, so an instruction that uses the same output cannot be used.

11. Control direction

When reversible control is enabled, store 0 in this data register for forward operation and store 1 in this data register for reverse operation.

This is ignored when specify absolute position mode is **Enabled**. If the absolute position counter value subtracted from the target position is positive, **Forward** is automatically selected, and pulses are output. If negative, **Reverse** is automatically selected, and pulses are output.

12. Absolute Position Mode

When the target position is specified with preset count (13), pulses are output by automatically calculating the number of pulses and direction from the difference between the current position stored in the absolute position counter (D8240 to D8247) and the target position.

Specify Absolute Position	Overview
Disabled	The RAMP instruction will be executed by specifying **Control direction** and **Preset value**. The number of pulses specified by **Preset value** is output.
Enabled	The RAMP instruction will be executed by specifying the target absolute position (target position) in **Preset value**. The instruction will be executed by calculating the number of pulses to output and the direction from the absolute position managed by the absolute position counter and the target position specified by **Preset value**. The **Control direction** setting is ignored.

Notes:

- If Reversible control disabled is selected for reversible control enable, specify absolute position mode is disabled.
- If the corresponding absolute position counter initialized flag in D8239 (absolute position control status) is 0 (not initialized), a user program execution error will occur even if the instruction is executed with specify absolute position mode set to Enabled. After starting operation of the ladder program, execute the ABS instruction that corresponds to the applicable output one time.

13. Preset value

If specify absolute position mode is **Disabled**, set the total number of pulses to output between 1 and 100,000,000.

If specify absolute position mode is **Enabled**, specify the absolute position between -2,147,483,648 and 2,147,483,647. The number of pulses will be output in the amount of the absolute value of the value that is the result of subtracting the absolute position counter value from the absolute position.

14. Current value

Regardless of whether specify absolute position mode is enabled or disabled, the number of pulses output is stored in the data registers.

The current value is updated when the RAMP instruction is executed at each scan.

15. Error status

If a configuration error occurs when the RAMP instruction input changes from off to on, M8004 (user program execution error) is turned on and this register is set to the error code.

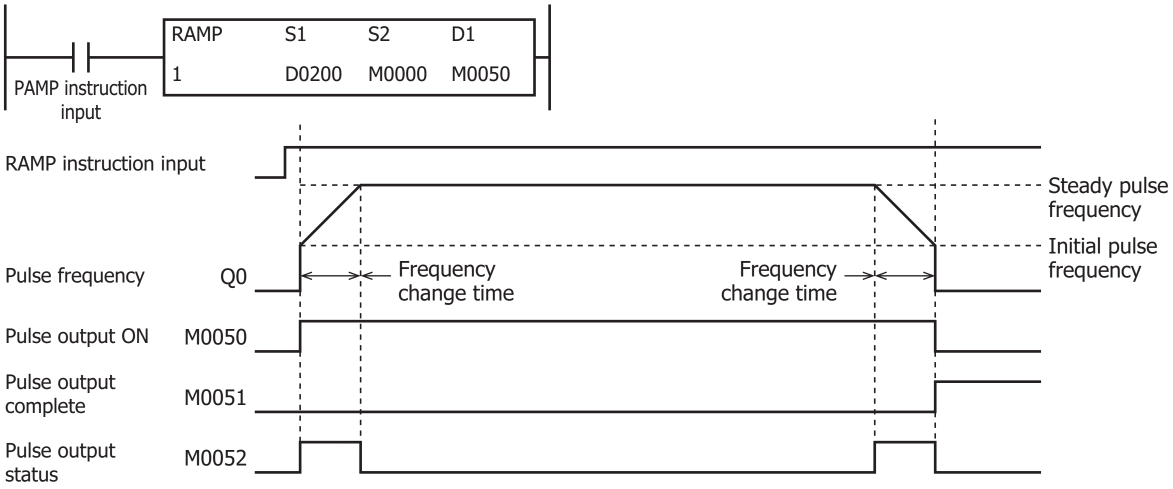
Error Code	Description	Details		
0	Normal	—		
2	Initial pulse frequency designation error	All-in-One CPU module	RAMP1, RAMP2	The pulse frequency was not set between 15 and 100,000.
			RAMP3, RAMP4	The pulse frequency was not set between 15 and 5,000.
		CAN J1939 All-in-One CPU module/Plus CPU module	RAMP1 to RAMP4	The pulse frequency was not set between 15 and 100,000.
3	Preset value designation error	When absolute position mode is disabled: The preset value was not set between 1 to 100,000,000. When absolute position mode is enabled: The preset value was not set between -2,147,483,648 to 2,147,483,647 or the difference between the configured preset value and absolute position counter was bigger than 100,000,000.		
4	Steady pulse frequency designation error	All-in-One CPU module	RAMP1, RAMP2	The pulse frequency was not set between 15 and 100,000.
			RAMP3, RAMP4	The pulse frequency was not set between 15 and 5,000.
		CAN J1939 All-in-One CPU module/Plus CPU module	RAMP1 to RAMP4	The pulse frequency was not set between 15 and 100,000.
5	Frequency change time designation error	The frequency change time was not set between 10 and 10,000.		
7	Control direction designation error	The control direction was not set to 0 or 1.		
8	Exceeded pulse count error	The number of frequency change pulses exceed the total number of output pulses.[*1]		
9	The initial pulse frequency was set to the same frequency as the steady pulse frequency or it was set to a value larger than the steady pulse frequency.[*2]			

[*1] The number of pulses in the frequency change region calculated by the initial pulse frequency, steady pulse frequency, and frequency change time has exceeded the total number of output pulses. Adjust the settings by decreasing the steady or initial pulse frequency or by shortening the frequency change time.

[*2] Set the initial pulse frequency so that it is lower than the steady pulse frequency.

RAMP1 instruction (reversible control disabled) timing chart

RAMP1 instruction, S1 is specified as D0200, D1 is specified as internal relay M0050

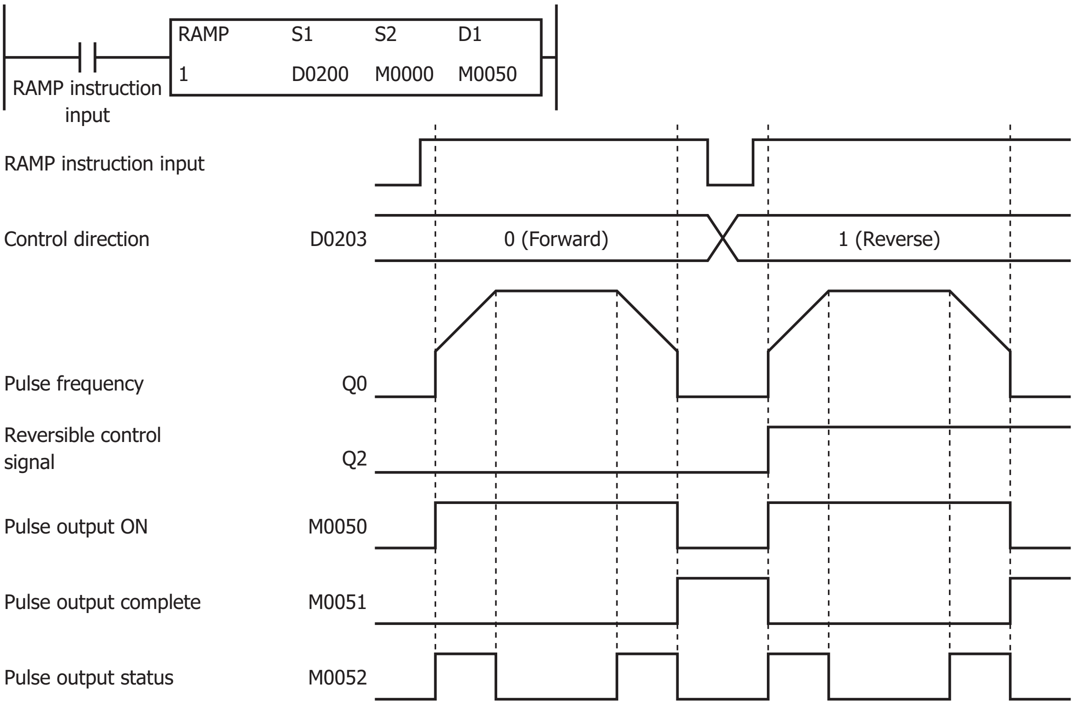


When the RAMP instruction input is on, pulses are output according to the settings configured by the control registers. M0050 turns on during pulse output. M0052 turns on or off while increasing or decreasing the pulse frequency. Pulses are output according to the frequency change time, until they reach the steady pulse frequency (from the initial pulse frequency). When the frequency change time is specified as 100, the steady pulse frequency is reached in 100 ms by increasing or decreasing the frequency every 10 ms. Pulse output stops when the pulses configured by the preset value are output. (The number of pulses is also counted while changing the frequency.) At this time, M0050 turns off and M0051 turns on.

If the RAMP instruction input turns off during pulse output, pulse output is canceled. If the RAMP instruction input turns on again, the pulse count is reset and pulse counting starts. Even if the contents of the control registers are changed during pulse output, the change is not reflected in the pulse output operation. The changed content is reflected the next time the RAMP instruction is executed.

RAMP1 instruction (reversible control enabled, single-pulse output mode) timing chart

RAMP1 instruction, S1 is specified as D0200, D1 is specified as internal relay M0050



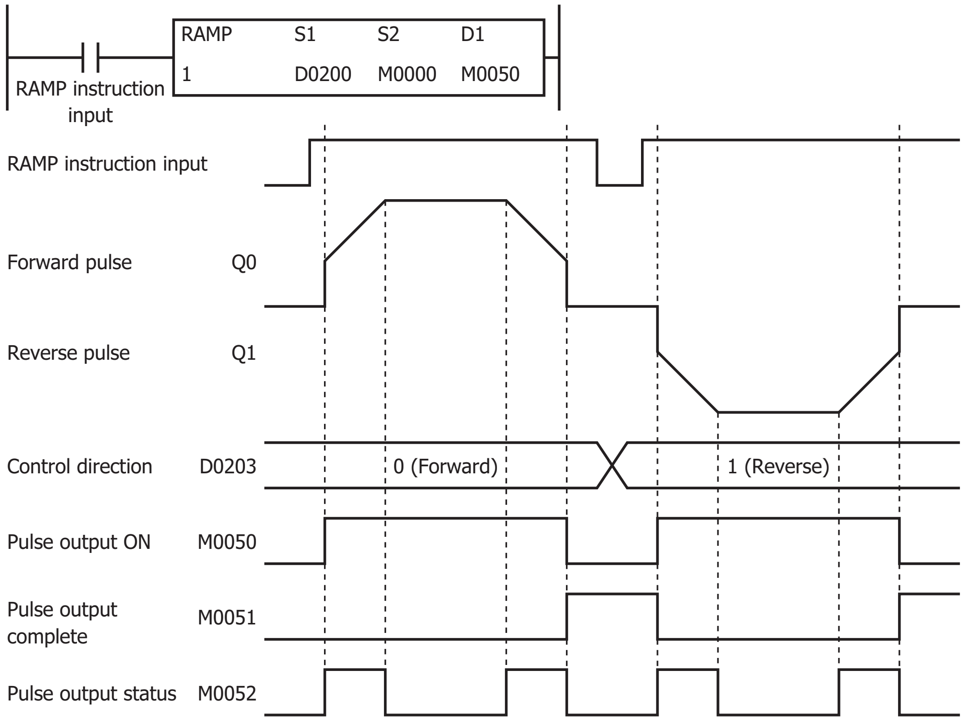
When the RAMP instruction input is on, pulses are output according to the settings configured by the control registers. The reversible control signal is output from Q2. When pulse output starts, M0050 turns on. M0052 turns on while the pulse frequency is increasing or decreasing. Pulses are output so that the frequency reaches the steady pulse frequency from the initial pulse frequency in the frequency change time. When the frequency change time is specified as 100, the steady pulse frequency is reached in 100 ms by increasing or decreasing the frequency every 10 ms.

Pulse output stops when the pulses configured by the preset value are output. (The number of pulses is also counted while changing the frequency.) At this time, M0050 turns off and M0051 turns on.

If the RAMP instruction input turns off during pulse output, pulse output is canceled. If this input turns on again, the operation starts from the beginning. Even if the contents of the data registers are changed during pulse output, the change is not reflected in the pulse output operation. The changed content is reflected the next time the RAMP instruction is started.

RAMP1 instruction (reversible control enabled, dual-pulse output mode) timing chart

RAMP1 instruction, S1 is specified as D0200, D1 is specified as internal relay M0050



When the RAMP instruction input changes from off to on, pulses are output from Q0 or Q1 according to the settings configured by the control registers.

When pulse output starts, M0050 turns on. M0052 turns on while the pulse frequency is increasing or decreasing.

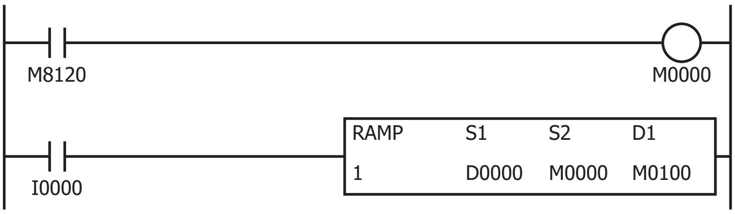
Pulses are output so that the frequency reaches the steady pulse frequency from the initial pulse frequency in the frequency change time. When the frequency change time is specified as 100, the steady pulse frequency is reached in 100 ms by increasing or decreasing the frequency every 10 ms.

Pulse output stops when the pulses configured by the preset value are output. (The number of pulses is also counted while changing the frequency.) At this time, M0050 turns off and M0051 turns on.

If the RAMP instruction input turns off during pulse output, pulse output is canceled. If this input turns on again, the operation starts from the beginning. Even if the contents of the data registers are changed during pulse output, the change is not reflected in the pulse output operation. The changed content is reflected the next time the RAMP instruction is started.

Example:

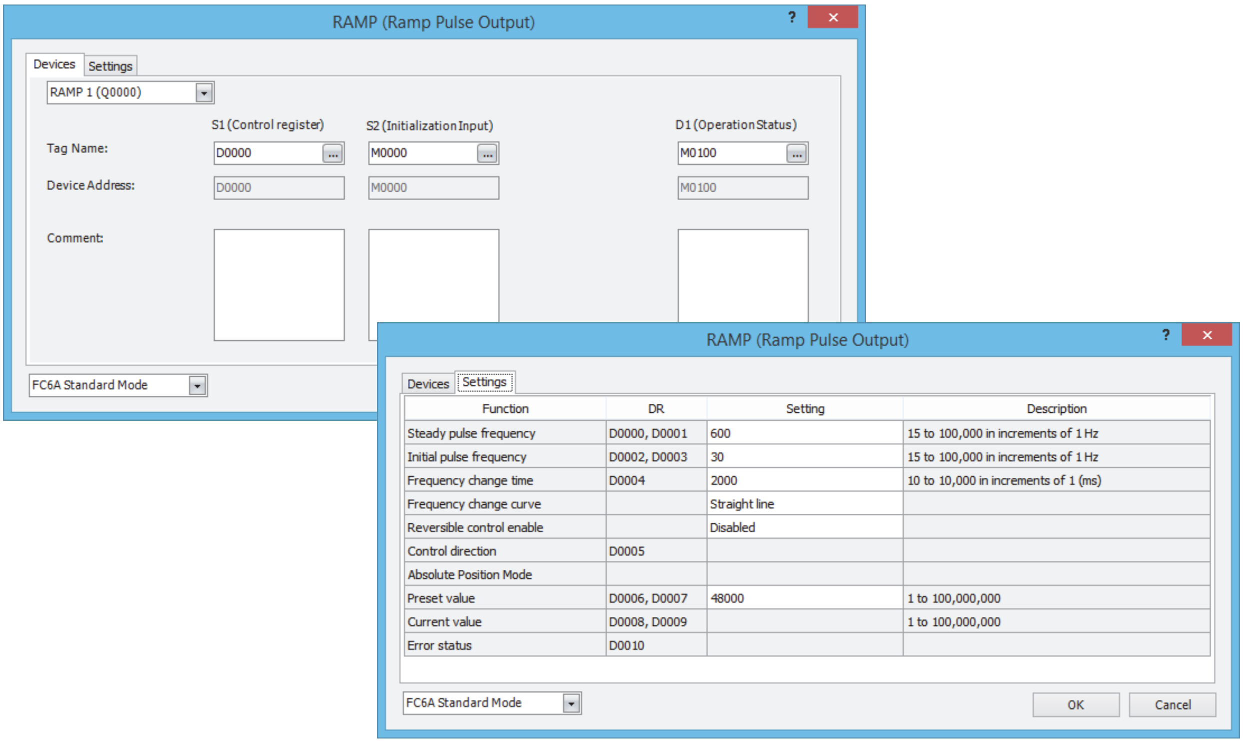
To output 48,000 pulses with the frequency change function (reversible control disabled) from Q0



Turn on initialization input (M0000)

When the RAMP instruction input (I0) turns on, pulse output starts

Setting



RAMP (Ramp Pulse Output)

Devices | Settings

RAMP 1 (Q0000)

	S1 (Control register)	S2 (Initialization Input)	D1 (Operation Status)
Tag Name:	D0000	M0000	M0100
Device Address:	D0000	M0000	M0100
Comment:			

FC6A Standard Mode

RAMP (Ramp Pulse Output)

Devices | Settings

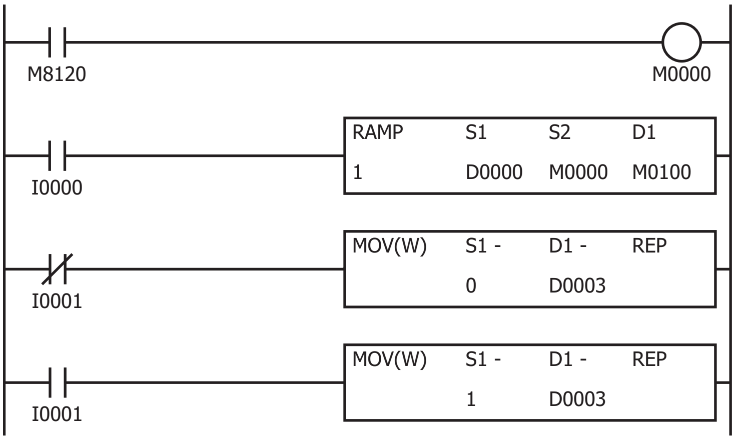
Function	DR	Setting	Description
Steady pulse frequency	D0000, D0001	600	15 to 100,000 in increments of 1 Hz
Initial pulse frequency	D0002, D0003	30	15 to 100,000 in increments of 1 Hz
Frequency change time	D0004	2000	10 to 10,000 in increments of 1 (ms)
Frequency change curve		Straight line	
Reversible control enable		Disabled	
Control direction	D0005		
Absolute Position Mode			
Preset value	D0006, D0007	48000	1 to 100,000,000
Current value	D0008, D0009		1 to 100,000,000
Error status	D0010		

FC6A Standard Mode

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0000, D0001	600	600 Hz
Initial pulse frequency	D0002, D0003	30	30 Hz
Frequency change time	D0004	2000	2,000 ms
Reversible control enable	—	Disable	—
Frequency change curve	—	Straight line	—
Control direction	D0005	—	—
Absolute Position Mode	—	—	—
Preset value	D0006, D0007	48000	Preset value=48,000

To output 100,000 pulses with the frequency change function (reversible control by single-pulse output) from Q0

When the RAMP instruction input (I0) changes from off to on, pulse output starts. When I1 is off, the reversible control signal (Q2) turns off (forward).
When I1 is on, the reversible control signal (Q2) turns on (reverse).



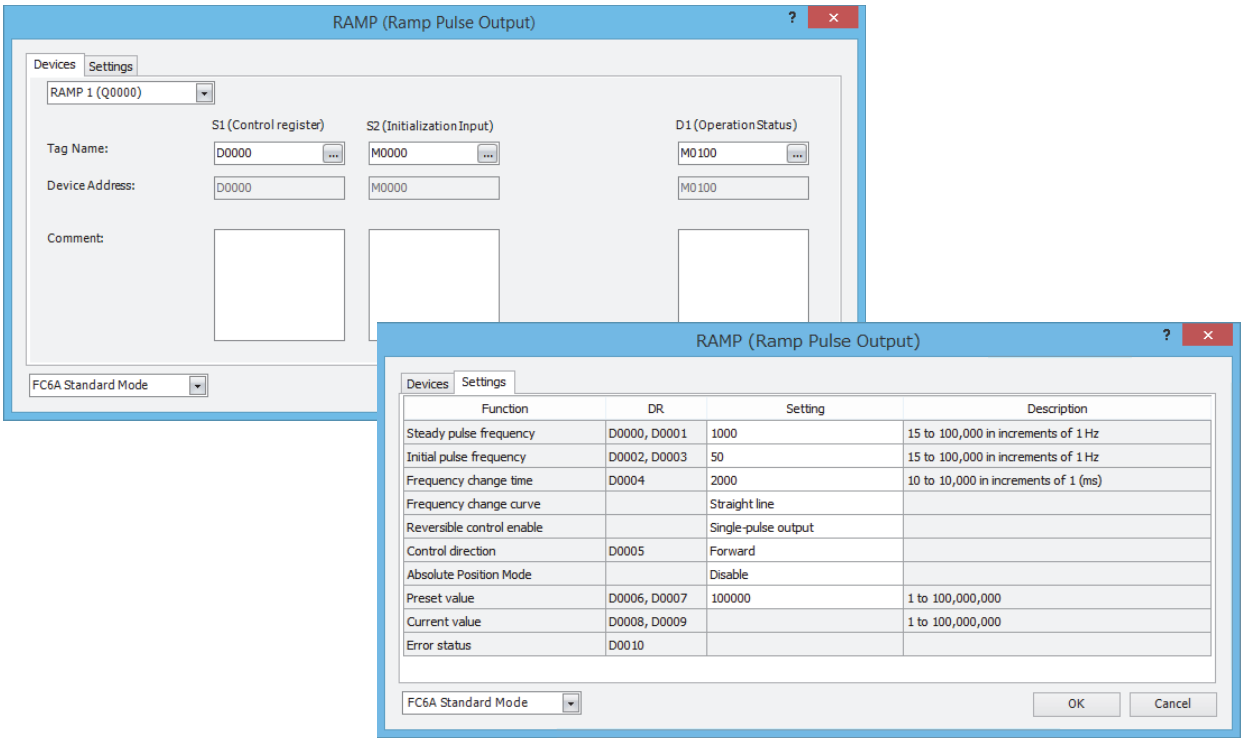
Turn on initialization input (M0000)

When the RAMP instruction input (I0) turns on, pulse output starts

When I1 is off, store 0 (forward) in control direction (D0003)

When I1 is on, store 1 (reverse) in control direction (D0003)

Setting



RAMP (Ramp Pulse Output)

Devices | Settings

RAMP 1 (Q0000)

	S1 (Control register)	S2 (Initialization Input)	D1 (Operation Status)
Tag Name:	D0000	M0000	M0100
Device Address:	D0000	M0000	M0100
Comment:			

FC6A Standard Mode

RAMP (Ramp Pulse Output)

Devices | Settings

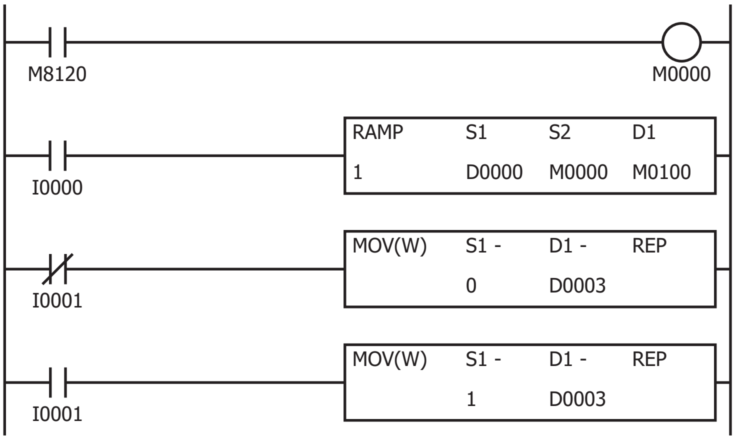
Function	DR	Setting	Description
Steady pulse frequency	D0000, D0001	1000	15 to 100,000 in increments of 1 Hz
Initial pulse frequency	D0002, D0003	50	15 to 100,000 in increments of 1 Hz
Frequency change time	D0004	2000	10 to 10,000 in increments of 1 (ms)
Frequency change curve		Straight line	
Reversible control enable		Single-pulse output	
Control direction	D0005	Forward	
Absolute Position Mode		Disable	
Preset value	D0006, D0007	100000	1 to 100,000,000
Current value	D0008, D0009		1 to 100,000,000
Error status	D0010		

FC6A Standard Mode

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0000, D0001	1000	1,000 Hz
Initial pulse frequency	D0002, D0003	50	50 Hz
Frequency change time	D0004	2000	2,000 ms
Frequency change curve	—	Straight line	—
Reversible control enable	—	Single-pulse output	—
Control direction	D0005	Forward	Forward=0
Absolute Position Mode	—	Disable	—
Preset value	D0006, D0007	100000	Preset value=100,000

To output 1,000,000 pulses with the frequency change function (reversible control by dual-pulse output)

When the RAMP instruction input (I0) changes from off to on, pulse output starts. For forward when I1 is off, pulses (CW) are output from Q0.
For reverse when I1 is on, pulses (CCW) are output from Q1.



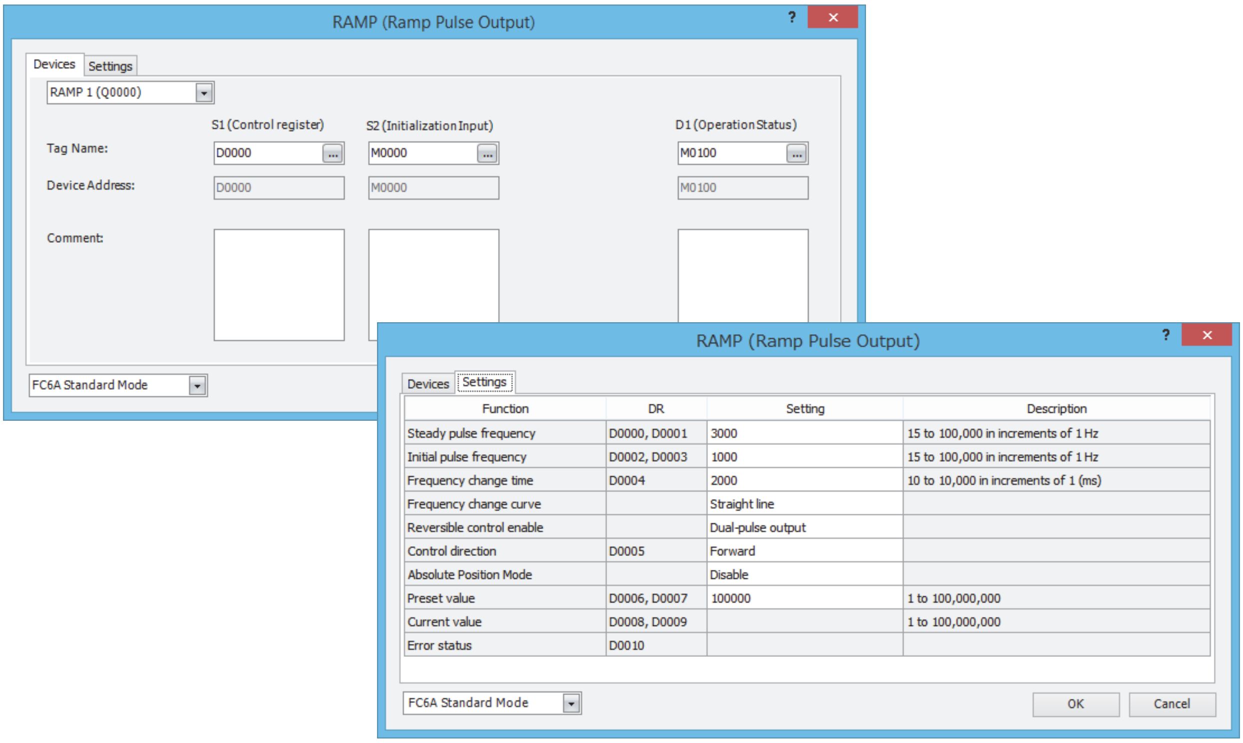
Turn on initialization input (M0000)

When the RAMP instruction input (I0) turns on, pulse output starts

When I1 is off, store 0 (forward) in control direction (D0003), and output pulses from Q0

When I1 is on, store 1 (reverse) in control direction (D0003), and output pulses from Q1

Setting



RAMP 1 (Q0000)

	S1 (Control register)	S2 (Initialization Input)	D1 (Operation Status)
Tag Name:	D0000	M0000	M0100
Device Address:	D0000	M0000	M0100
Comment:			

Function	DR	Setting	Description
Steady pulse frequency	D0000, D0001	3000	15 to 100,000 in increments of 1 Hz
Initial pulse frequency	D0002, D0003	1000	15 to 100,000 in increments of 1 Hz
Frequency change time	D0004	2000	10 to 10,000 in increments of 1 (ms)
Frequency change curve		Straight line	
Reversible control enable		Dual-pulse output	
Control direction	D0005	Forward	
Absolute Position Mode		Disable	
Preset value	D0006, D0007	100000	1 to 100,000,000
Current value	D0008, D0009		1 to 100,000,000
Error status	D0010		

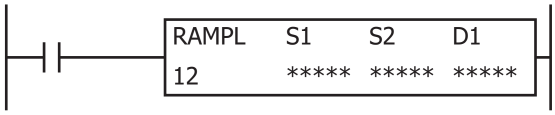
Function	Device Address	Setting Value	Details
Steady pulse frequency	D0000, D0001	3000	3,000 Hz
Initial pulse frequency	D0002, D0003	1000	1,000 Hz
Frequency change time	D0004	2000	2,000 ms
Frequency change curve	—	Straight line	—
Reversible control enable	—	Dual-pulse output	—
Control direction	D0005	Forward	Forward=0
Absolute Position Mode	—	Disable	—
Preset value	D0006, D0007	100000	Preset value=100,000

RAMPL (Linear Interpolation Control)

The RAMPL instruction outputs pulses with a frequency change function that operates simultaneously from two outputs, so that the trajectory of movement is linear.

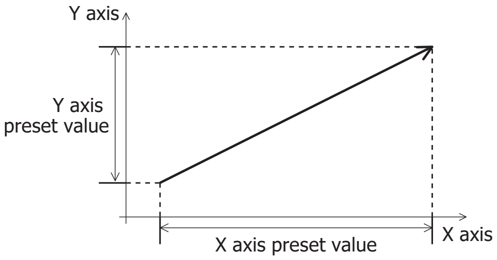
This instruction can be used with the Plus CPU module transistor output type and CAN J1939 All-in-One CPU module transistor output type.

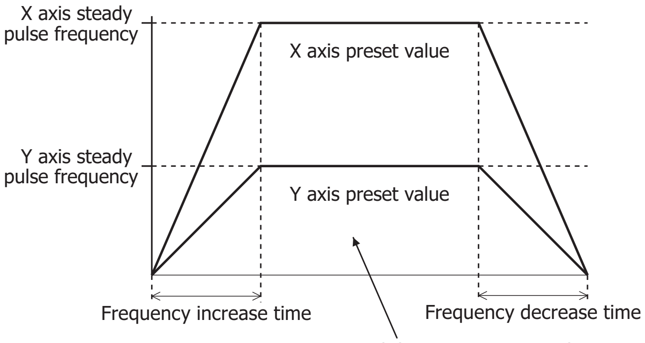
Ladder Diagram



Operation

When the input is on, the initial pulse frequency and steady pulse frequency are calculated from the preset value specified by S1 (target position), the combined initial pulse frequency, and the combined steady pulse frequency, so that the trajectory of movement is linear. Then the pulses are simultaneously output from the two specified outputs, and the pulse frequency increases at a constant rate until the frequency reaches the steady pulse frequency of the axes. After pulses of a constant speed are output at the steady pulse frequency, the pulses frequency decreases before reaching the preset value specified by S1, and then the pulse output stops when it reaches the preset value.





Frequency increase and decrease times are the same
Trapezoid area = Preset value

When the initialization input specified by S2 is turned on, the initial values configured in the WindLDR **RAMPL (Ramp Pulse Output with Liner Interpolation)** dialog box, on the **Common Settings** tab, are stored in the control registers.

The control status, including the pulse output status (output on/output direction/output complete), is stored in the operation status specified by D1.

Notes:

- If a pulse output instruction is simultaneously executed with the same output, a user program execution error will occur.
  Error code 48 will be stored in D8006 and instructions that are executed later will be canceled.
- The RAMPL instruction cannot be used in an interrupt program. If used in an interrupt program, a user program execution error will occur.
  Error code 18 will be stored in D8006 and instruction execution will be canceled.
- If a pulse output instruction is executed with the relay output type, a user program execution error will occur.
  Error code 19 will be stored in D8006 and instruction execution will be canceled.
- The RAMPL instruction operates only in absolute position mode with accompanying reversible control. After starting operation of the ladder program, execute the ABS instructions that correspond to the specified outputs to initialize the absolute position counters. If the corresponding absolute position counter initialized flags in D8239 (absolute position control status) are 0 (not initialized), a user program execution error will occur when the instruction is executed.
- For user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Devices

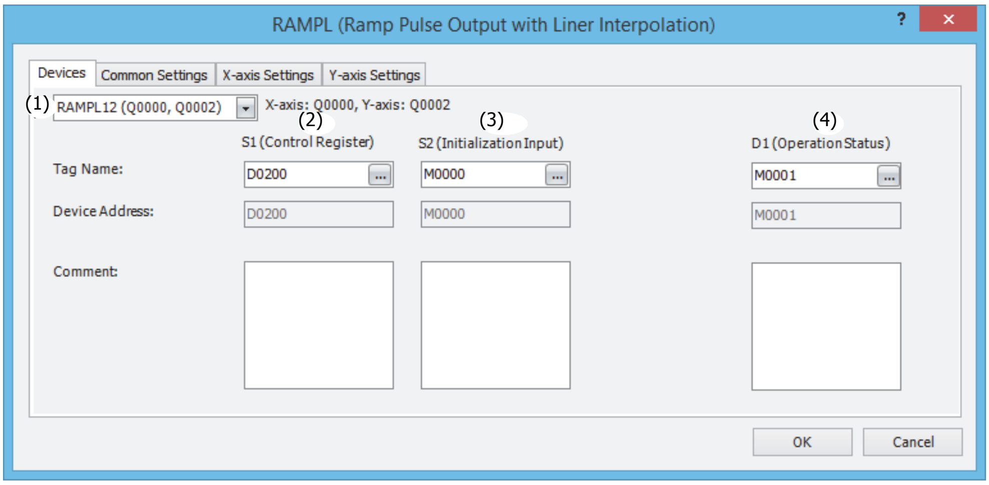
Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Control registers	—	—	—	—	—	—	X[*1]	—	—	—
S2 (Source 2)	Initialization input	X	—	X	—	—	—	—	—	—	—
D1 (Destination 1)	Operation status	—	—	X[*2]	—	—	—	—	—	—	—

*1 Special data registers cannot be used.

*2 Special internal relays cannot be used.

Settings

■ Device tab



RAMPL (Ramp Pulse Output with Liner Interpolation)

RAMPL12 (Q0000, Q0002) X-axis: Q0000, Y-axis: Q0002

	S1 (Control Register)	S2 (Initialization Input)	D1 (Operation Status)
Tag Name:	D0200	M0000	M0001
Device Address:	D0200	M0000	M0001
Comment:			

(1) Select instruction

Selects the RAMPL instruction to use from **RAMPL12**, **RAMPL13**, **RAMPL14**, **RAMPL23**, **RAMPL24**, and **RAMPL34**.

The combination of pulse outputs will depend on the instruction.

For details, see "(11) Reversible control" on page 18-32.

(2) S1 (source 1): Control register

S1 specifies the first data register of the data registers to use with the RAMPL12, RAMPL13, RAMPL14, RAMPL23, RAMPL24, or RAMPL34 instruction.
Starting from the first data register, 30 continuous words of data registers are used. Specify the first data register so that the device range is not exceeded.

Storage Destination	Function	Setting	Reference
Starting number+0	Combined steady pulse frequency (high word)*1	15 to 100,000 (increments of 1 Hz)	"(5) Combined steady pulse frequency" on page 18-30
Starting number+1	Combined steady pulse frequency (low word)*1		
Starting number+2	Combined initial pulse frequency (high word)*1	15 to 100,000 (increments of 1 Hz)	"(6) Combined initial pulse frequency" on page 18-30
Starting number+3	Combined initial pulse frequency (low word)*1		
Starting number+4	Frequency change time	10 to 10,000 (ms)	"(7) Frequency change time" on page 18-30
Starting number+5	— **Reserved** —		
Starting number+6	— **Reserved** —		
Starting number+7	— **Reserved** —		
Starting number+8	— **Reserved** —		
Starting number+9	Error status		"(8) Error status" on page 18-31

Storage Destination	Function		Setting	Reference
Starting number+10	X axis	Steady pulse frequency (high word)*1*2	15 to 100,000 (increments of 1 Hz)	"(9) Steady pulse frequency" on page 18-32
Starting number+11		Steady pulse frequency (low word)*1*2		
Starting number+12		Initial pulse frequency (high word)*1*2	15 to 100,000 (increments of 1 Hz)	"(10) Initial pulse frequency" on page 18-32
Starting number+13		Initial pulse frequency (low word)*1*2		
Starting number+14		— **Reserved** —		
Starting number+15		— **Reserved** —		
Starting number+16		Preset value (high word)*1	Specify absolute position mode -2,147,483,648 to 2,147,483,647 pulses	"(13) Preset value" on page 18-33
Starting number+17		Preset value (low word)*1		
Starting number+18		Current value (high word)*1	1 to 100,000,000 pulses*3	"(14) Current value" on page 18-33
Starting number+19		Current value (low word)*1		
Starting number+20	X axis	Steady pulse frequency (high word)*1*2	15 to 100,000 (increments of 1 Hz)	
Starting number+21		Steady pulse frequency (low word)*1*2		
Starting number+22		Initial pulse frequency (high word)*1*2	15 to 100,000 (increments of 1 Hz)	
Starting number+23		Initial pulse frequency (low word)*1*2		
Starting number+24		— **Reserved** —		
Starting number+25		— **Reserved** —		
Starting number+26		Preset value (high word)*1	Specify absolute position mode -2,147,483,648 to 2,147,483,647 pulses	"(13) Preset value" on page 18-33
Starting number+27		Preset value (low word)*1		
Starting number+28		Current value (high word)*1	1 to 100,000,000 pulses*3	"(14) Current value" on page 18-33
Starting number+29		Current value (low word)*1		

*1 The upper and lower data registers change according to the 32-bit data storage method specified. For details, see "32-bit Data Storage" on page 3-9.

*2 When the instruction input is turned on, the automatically calculated values are stored in the registers.

*3 The number of output pulses is stored in the data registers, regardless of the value of the absolute position counter.

(3) S2 (source 2): Initialization input

S2 specifies the initialization input.

When the initialization input is turned on, the initial values configured in the WindLDR **RAMPL (Ramp Pulse Output with Liner Interpolation)** dialog box, on the **Common Settings** tab, are stored in the control registers. An external input or an internal relay can be specified.

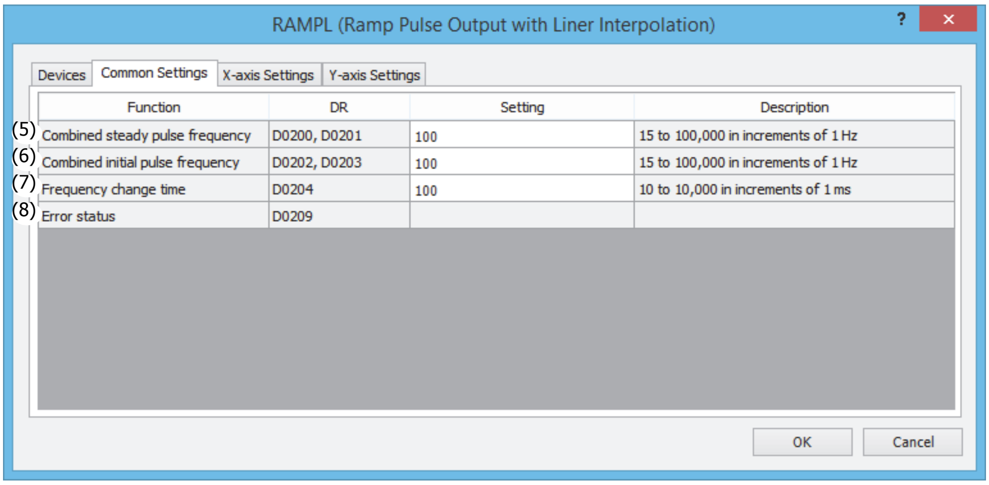
When the initialization input is on, the initial values are stored in the data registers with each scan. (Even when the RAMPL instruction is not executed (when not on), if the initialization input is turned on, the initial values are stored in the data registers.) To only initialize the values one time, use the initialization input in combination with the SOTU (single output up) instruction or the SOTD (single output down) instruction.

(4) D1 (destination 1): Operation status

D1 specifies the first internal relay of the internal relays to use with the RAMPL instruction. Starting from the specified internal relay, 4 sequential internal relays are used. Specify the first internal relay so that the device range is not exceeded.

Storage Destination	Function	Setting	
Starting number+0	Pulse output ON	0: Pulse output OFF 1: Pulse output ON	This relay turns on during pulse output. This relay turns off when pulse output stops. This relay turns off when the specified number of pulses are output and output ends.
Starting number+1	Pulse output complete	0: Pulse output not complete 1: Pulse output complete	This relay turns on when pulse output is complete. This relay turns off when pulse output starts.
Starting number+2	Pulse output status	0: Steady pulse output 1: Changing output pulse frequency	This relay turns off when the pulse output status is steady. This relay turns on when the pulse output is changing.
Starting number+3	Overflow	0: None 1: Overflow has occurred	This relay turns on when the pulses that were output have exceeded the preset value when enable pulse counting is set. Pulse output will continue even if an overflow occurs while changing the frequency or during steady operation. However, counting of the current value will stop when the overflow occurs.

■ **Common Settings tab**


RAMPL (Ramp Pulse Output with Liner Interpolation)

	Function	DR	Setting	Description
(5)	Combined steady pulse frequency	D0200, D0201	100	15 to 100,000 in increments of 1 Hz
(6)	Combined initial pulse frequency	D0202, D0203	100	15 to 100,000 in increments of 1 Hz
(7)	Frequency change time	D0204	100	10 to 10,000 in increments of 1 ms
(8)	Error status	D0209		

(5) Combined steady pulse frequency

This setting specifies the steady pulse frequency after the pulse frequency is increased. The output frequency error is within ±5%.

The steady pulse frequency for the X and Y axes that is calculated from the combined steady pulse frequency will be lower than the combined steady pulse frequency. Set this so that the steady pulse frequency of the X and Y axes does not fall below 15 Hz.

(6) Combined initial pulse frequency

Specifies the frequency when pulse output starts. The output frequency error is within ±5%.

The initial pulse frequency for the X and Y axes that is calculated from the combined initial pulse frequency will be lower than the combined initial pulse frequency. Set this so that the initial pulse frequency of the X and Y axes does not fall below 15 Hz.

(7) Frequency change time

Specify the time for increasing and decreasing the pulse frequency. Set the value in the range of 10 to 10,000 ms in increments of 1 ms. The first digit of the setting is handled as zero. For example, if 144 is entered, the set value is handled as 140 ms.

(8) Error status

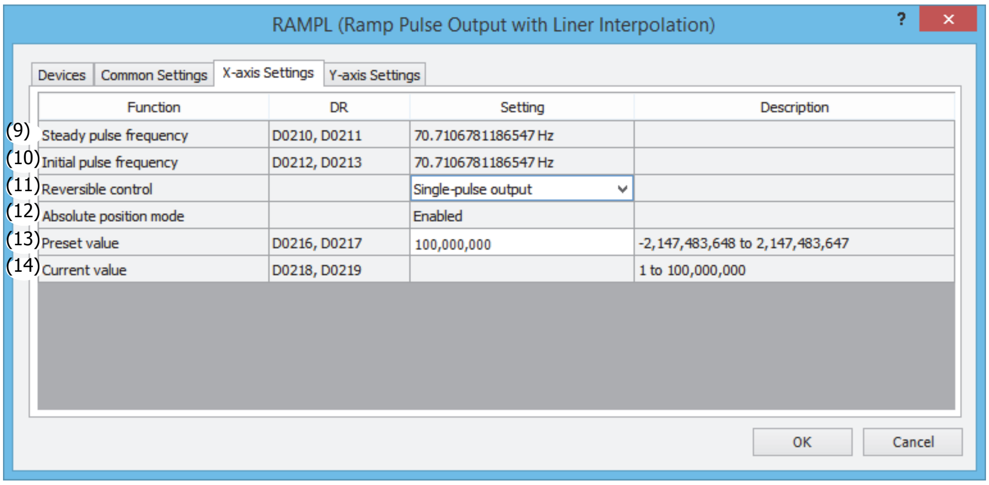
Outputs the error code that corresponds to the content of an error when there is an error in the settings. If a configuration error occurs when the RAMPL instruction input changes from off to on, M8004 (user program execution error) is turned on and this register is set to the error code.

Error Code	Status	Description
0	Normal	—
2	Combined initial pulse frequency designation error	The combined initial pulse frequency was not set between 15 and 100,000.
4	Combined steady pulse frequency designation error	The combined steady pulse frequency was not set between 15 and 100,000.
5	Frequency change time designation error	The frequency change time was not set between 10 and 10,000.
9	Frequency designation error	The combined initial pulse frequency was set to the same frequency as the combined steady pulse frequency or it was set to a value larger than the combined steady pulse frequency.*1
12	X axis initial pulse frequency designation error	The X axis initial pulse frequency calculated from the set values was not between 15 and 100,000.
13	X axis preset value designation error	The X axis preset value was not set between -2,147,483,648 and 2,147,483,647 or the difference between the configured preset value and absolute position counter was bigger than 100,000,000.
14	X axis steady pulse frequency designation error	The X axis steady pulse frequency calculated from the set values was not between 15 and 100,000.
18	X axis over preset value error	The number of X axis pulses when changing the pulse frequency has exceeded the total number of output pulses.*2
22	Y axis initial pulse frequency designation error	The Y axis initial pulse frequency calculated from the set values was not between 15 and 100,000.
23	Y axis preset value designation error	The Y axis preset value was not set between -2,147,483,648 and 2,147,483,647 or the difference between the configured preset value and absolute position counter was bigger than 100,000,000.
24	Y axis steady pulse frequency designation error	The Y axis steady pulse frequency calculated from the set values was not between 15 and 100,000.
28	Y axis over preset value error	The number of Y axis pulses when changing the pulse frequency has exceeded the total number of output pulses.*2

*1 Set the combined initial pulse frequency so that it is lower than the combined steady pulse frequency.

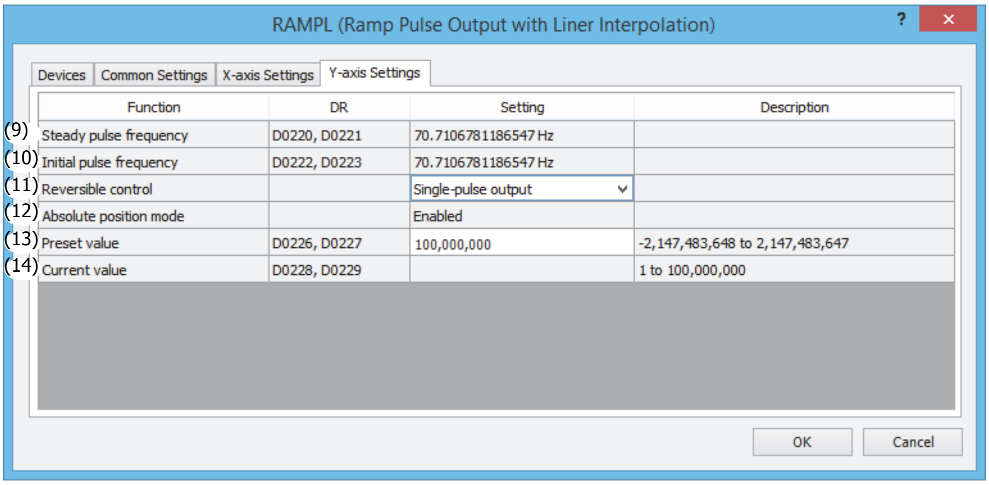
*2 The number of pulses in the frequency change area calculated by steady pulse frequency, initial pulse frequency, and frequency change time has exceed the total number of output pulses. Lower the steady pulse frequency, lower the initial pulse frequency, or shorten the frequency change time.

■ X-axis Settings tab, Y-axis Settings tab



RAMPL (Ramp Pulse Output with Liner Interpolation)

	Function	DR	Setting	Description
(9)	Steady pulse frequency	D0210, D0211	70.7106781186547 Hz	
(10)	Initial pulse frequency	D0212, D0213	70.7106781186547 Hz	
(11)	Reversible control		Single-pulse output	
(12)	Absolute position mode		Enabled	
(13)	Preset value	D0216, D0217	100,000,000	-2,147,483,648 to 2,147,483,647
(14)	Current value	D0218, D0219		1 to 100,000,000


RAMPL (Ramp Pulse Output with Liner Interpolation)

Devices | Common Settings | X-axis Settings | Y-axis Settings

	Function	DR	Setting	Description
(9)	Steady pulse frequency	D0220, D0221	70.7106781186547 Hz	
(10)	Initial pulse frequency	D0222, D0223	70.7106781186547 Hz	
(11)	Reversible control		Single-pulse output	
(12)	Absolute position mode		Enabled	
(13)	Preset value	D0226, D0227	100,000,000	-2,147,483,648 to 2,147,483,647
(14)	Current value	D0228, D0229		1 to 100,000,000

(9) Steady pulse frequency

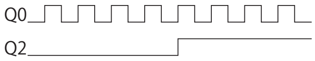
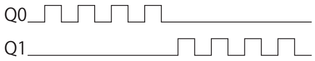
The steady pulse frequency of each axis is calculated from the amount of movement to the target position and the combined steady pulse frequency, and this value is stored in the data registers. When the instruction is turned on, the steady pulse frequency is calculated and updated. The combined steady pulse frequency on **Setting** is resolved for each axis, and that value is displayed here.

(10) Initial pulse frequency

The initial pulse frequency of each axis is calculated from the amount of movement to the target position and the combined initial pulse frequency, and this value is stored in the data registers. When the instruction is turned on, the initial pulse frequency is calculated and updated. The combined initial pulse frequency on **Setting** is resolved for each axis, and that value is displayed here.

(11) Reversible control

Select the reversible control method for the X and Y axes from the following reversible control modes. There are two modes for the pulse output mode: single-pulse output mode and dual-pulse output mode. Disabled cannot be selected.

Reversible Control Mode	Operation	Pattern
Reversible control: **Single-pulse output**	Pulse A is used as pulse output. Pulse B on/off is used as reversible control.	
Reversible control: **Dual-pulse output**	Pulse A is used as forward pulse (CW) output.Pulse B is used as reverse pulse (CCW) output.	

The outputs used on the Plus CPU module will depend on the instruction that is used.

Command	Operating Condition	Output Used			
		X Axis		Y Axis	
		Pulse Output	Reversible Control Output	Pulse Output	Reversible Control Output
RAMPL12	Reversible control: **Single-pulse output mode**	Q0	Q1	Q2	Q3
	Reversible control: **Dual-pulse output mode**	Q0, Q1	—	Q2, Q3	—
RAMPL13	Reversible control: **Single-pulse output mode**	Q0	Q1	Q4	Q5
	Reversible control: **Dual-pulse output mode**	Q0, Q1	—	Q4, Q5	—
RAMPL14	Reversible control: **Single-pulse output mode**	Q0	Q1	Q6	Q7
	Reversible control: **Dual-pulse output mode**	Q0, Q1	—	Q6, Q7	—

Command	Operating Condition	Output Used			
		X Axis		Y Axis	
		Pulse Output	Reversible Control Output	Pulse Output	Reversible Control Output
RAMPL23	Reversible control: **Single-pulse output mode**	Q2	Q3	Q4	Q5
	Reversible control: **Dual-pulse output mode**	Q2, Q3	—	Q4, Q5	—
RAMPL24	Reversible control: **Single-pulse output mode**	Q2	Q3	Q6	Q7
	Reversible control: **Dual-pulse output mode**	Q2, Q3	—	Q6, Q7	—
RAMPL34	Reversible control: **Single-pulse output mode**	Q4	Q5	Q6	Q7
	Reversible control: **Dual-pulse output mode**	Q4, Q5	—	Q6, Q7	—

Note: The outputs (pulse output and reversible control output) used by RAMPL instruction cannot be used by other instructions simultaneously.

(12) Absolute position mode

Absolute position mode is always **Enabled**. Specify the target position with the preset value (13) for the X and Y axes.

Note: If the absolute position counter initialized flags in D8239 (absolute position control status) for the specified pulse outputs are 0 (not initialized), a user program execution error will occur. After starting operation of the ladder program, execute the ABS instructions that correspond to the applicable outputs one time.

(13) Preset value

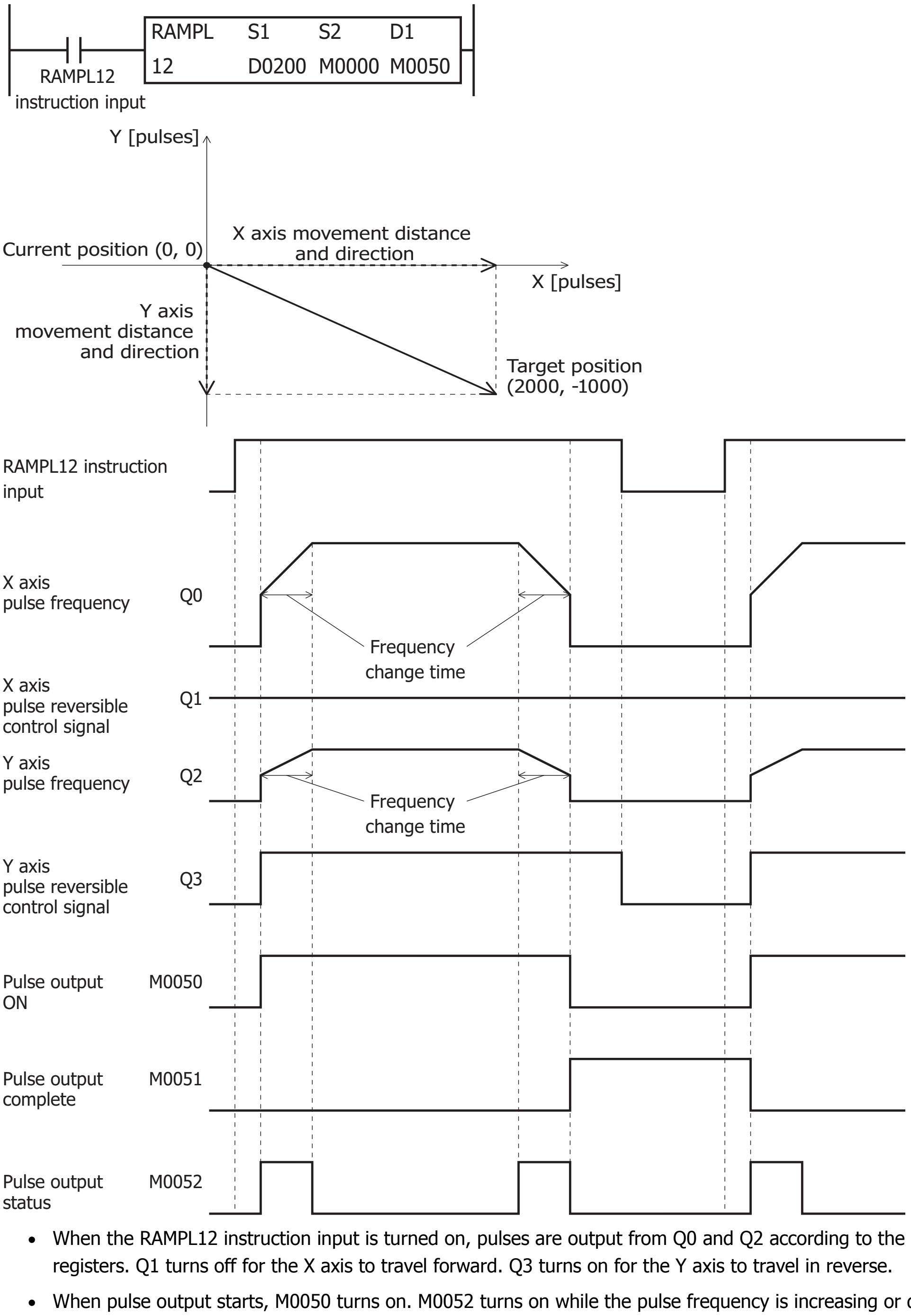
Specify the target position. The number of pulses will be output in the amount of the target value. This value is the result of subtracting the absolute position counter value from the target position. At that time, the pulses will be output by automatically switching between forward for positive and reverse for negative according to the sign of the subtracted value.

(14) Current value

The number of pulses output is stored in the data registers. The current value is updated at each scan when the RAMPL instruction is executed.

RAMPL12 Instruction (Reversible Control Enabled, Single-pulse Output) Timing Chart

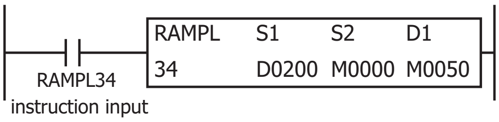
When data register D0200 is specified for S1 and internal relay M0050 is specified for D1 of the RAMPL12 instruction in order to move from the current position (0, 0) to the target position (2000, -1000)

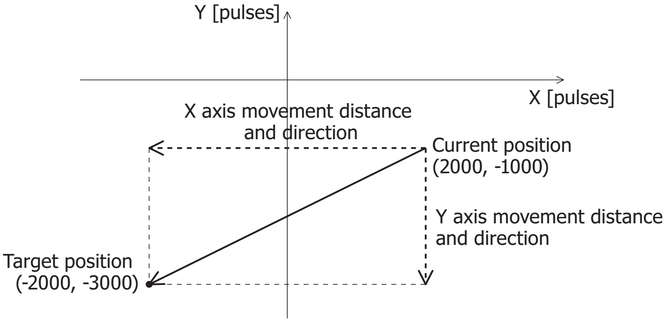


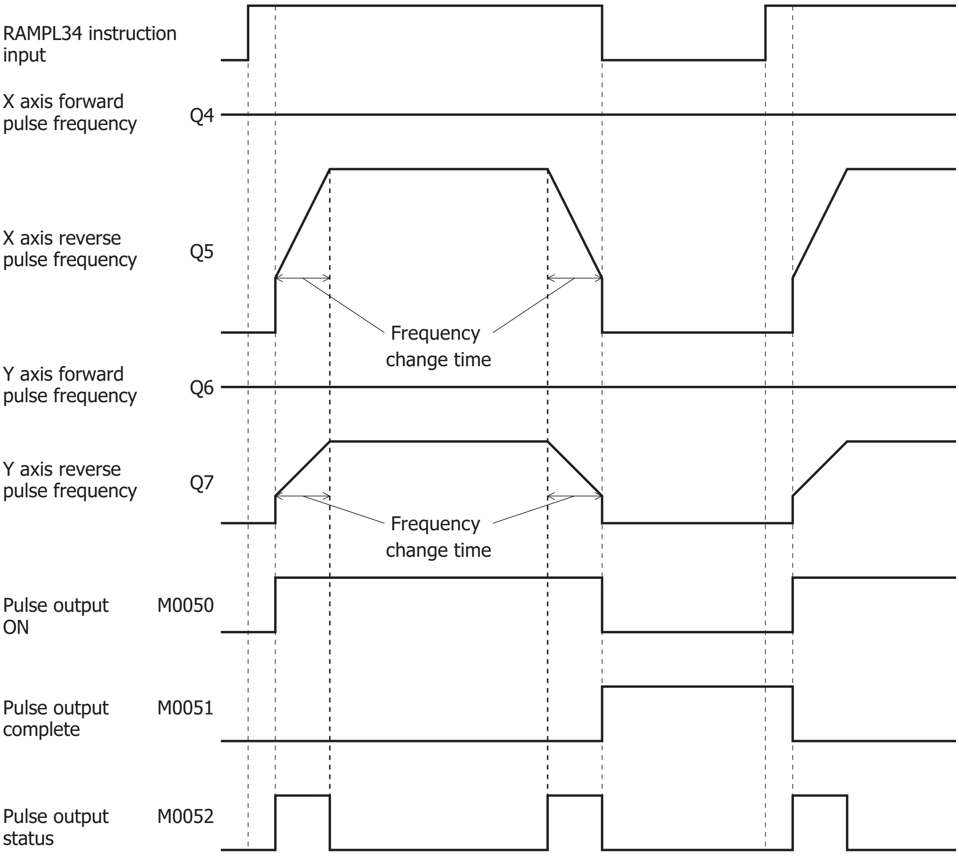
- When the RAMPL12 instruction input is turned on, pulses are output from Q0 and Q2 according to the settings configured by the control registers. Q1 turns off for the X axis to travel forward. Q3 turns on for the Y axis to travel in reverse.
- When pulse output starts, M0050 turns on. M0052 turns on while the pulse frequency is increasing or decreasing.
- The steady pulse frequency and initial pulse frequency of each axis are calculated from the preset value, combined steady pulse frequency, and combined initial pulse frequency, and then the pulses are output for each axis so that the frequency reaches the steady pulse frequency from the initial pulse frequency in the frequency change time. When the frequency change time is specified as 100, the steady pulse frequency is reached in 100 ms by increasing or decreasing the frequency every 10 ms.
- Pulse output stops when the pulses calculated from the difference between the current position and the target position are output. (The number of pulses is also counted while changing the frequency.) At this time, M0050 turns off and M0051 turns on.
- If the RAMPL12 instruction input turns off during pulse output, pulse output stops. If the input turns on again, the operation starts from the beginning.
- Even if the contents of the data registers are changed during pulse output, the change is not reflected in pulse output operation. The changed content is reflected the next time the RAMPL12 instruction is started.

RAMPL34 Instruction (Reversible Control Enabled, Dual-pulse Output) Timing Chart

When data register D0200 is specified for S1 and internal relay M0050 is specified for D1 of the RAMPL34 instruction in order to move from the current position (2000, -1000) to the target position (-2000, -3000)

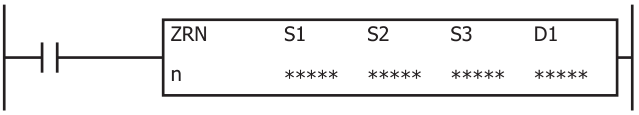






- When the RAMPL34 instruction input is turned on, pulses are output according to the settings configured by the control registers. X axis reverse pulses are output from Q5 and Y axis reverse pulses are output from Q7 in order to move both X and Y axes in reverse.
- When pulse output starts, M0050 turns on. M0052 turns on while the pulse frequency is increasing or decreasing.
- The steady pulse frequency and initial pulse frequency of each axis are calculated from the preset value, combined steady pulse frequency, and combined initial pulse frequency, and then the pulses are output for each axis so that the frequency reaches the steady pulse frequency from the initial pulse frequency in the frequency change time. When the frequency change time is specified as 100, the steady pulse frequency is reached in 100 ms by increasing or decreasing the frequency every 10 ms.
- Pulse output stops when the pulses calculated from the difference between the current position and the target position are output. (The number of pulses is also counted while changing the frequency.) At this time, M0050 turns off and M0051 turns on.
- If the RAMPL34 instruction input turns off during pulse output, pulse output stops. If the input turns on again, the operation starts from the beginning.
- Even if the contents of the data registers are changed during pulse output, the change is not reflected in pulse output operation. The changed content is reflected the next time the RAMPL34 instruction is started.

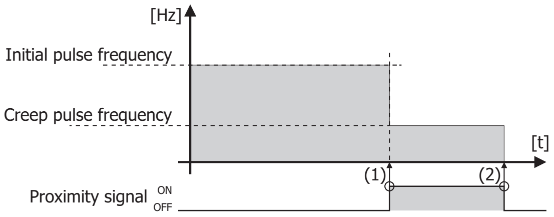
ZRN (Zero Return)



The ZRN instruction outputs pulses while monitoring multiple signals to perform a zero return.
When the input is on, pulses are output according to the frequency change settings stored in the control register specified by S1.
The pulse control information (output on/output complete/error) is stored in the internal relays specified by D2 as the operation status.
When the initialization input specified by S2 is turned on, the initial values configured in the WindLDR **ZRN (Zero Return)** dialog box are stored in the control registers.
The zero return method an be selected a **ZRN mode 0** or **ZRN mode 1**.

ZRN Mode 0

The zero return is performed by monitoring only the proximity signal specified by S3.



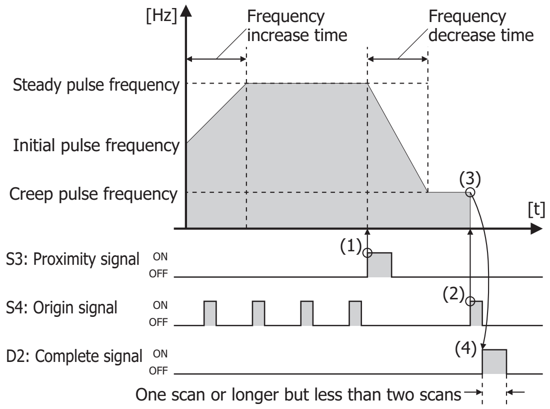
(1) When a rise in the proximity signal is detected, the frequency changes from the initial pulse frequency to the creep pulse frequency.

(2) When a fall in the proximity signal is detected, pulse output stops.

ZRN Mode 1

The zero return is performed by monitoring the proximity signal (decrease frequency trigger) specified by S3 and the origin signal (stop trigger) specified by S4.
After stopping pulse output with the origin signal, the complete signal specified by D2 can be output.
For ZRN mode 1, the timing to start monitoring the origin signal can be selected from the following two types.

Start monitoring the origin signal with the rise in the proximity signal



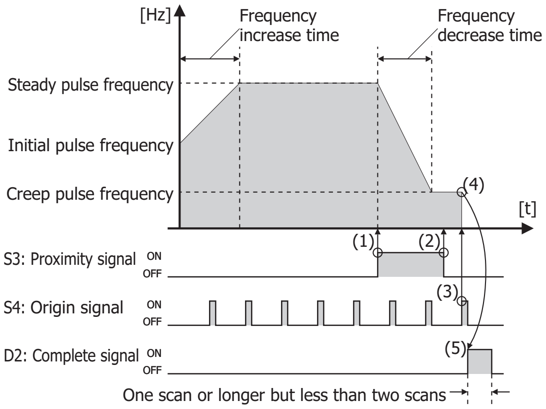
(1) The rise in the proximity signal is detected, the frequency starts being decreased, and at the same time the origin signal starts being monitored.

(2) The rise in the origin signal is detected and pulse output is stopped.

(3) When pulse output stops, the complete signal turns on at the same time.

(4) The on period for the complete signal is one scan or longer but less than two scans.

Start monitoring the origin signal with the fall in the proximity signal



(1) The rise in the proximity signal is detected and the frequency starts being decreased.

(2) The fall in the proximity signal is detected and the origin signal starts being monitored.

(3) The rise in the origin signal is detected and pulse output is stopped.

(4) When pulse output stops, the complete signal turns on at the same time.

(5) The on period for the complete signal is one scan or longer but less than two scans.

Notes:

- If a pulse output instruction is simultaneously executed with the same output, a user program execution error will occur. Error code 48 will be stored in D8006 and instructions that are executed later will be canceled.
- The ZRN instruction cannot be used in an interrupt program. If used in an interrupt program, a user program execution error will occur. Error code 18 will be stored in D8006 and instruction execution will be canceled.
- If a pulse output instruction is executed with the relay output type, a user program execution error will occur. Error code 19 will be stored in D8006 and instruction execution will be canceled.
- For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Devices

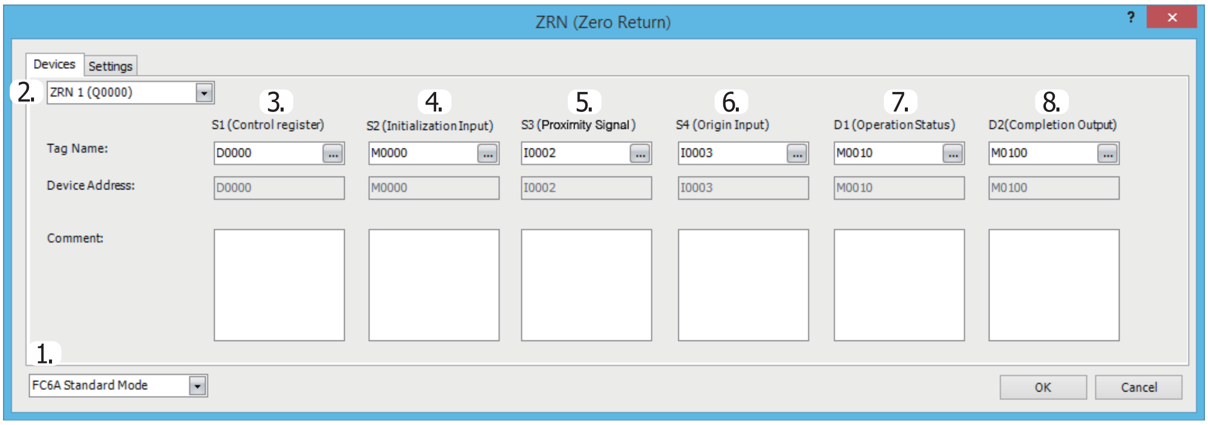
Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Control register	—	—	—	—	—	—	X*1	—	—	—
S2 (Source 2)	Initialization input	X	—	X	—	—	—	—	—	—	—
S3 (Source 3)	Proximity signal	X	—	X	—	—	—	—	—	—	—
S4 (Source 4)	Origin signal	X	—	X	—	—	—	—	—	—	—
D1 (Destination 1)	Operation status	—	—	X*2	—	—	—	—	—	—	—
D2 (Destination 2)	Complete signal	—	X	X*2	—	—	—	—	—	—	—

*1 Special data registers cannot be used.

*2 Special internal relays cannot be used. Only 0 can be specified as the first digit of the internal relay number. 1 to 7 cannot be specified.

Settings

■ Devices tab



ZRN 1 (Q0000)

	S1 (Control register)	S2 (Initialization Input)	S3 (Proximity Signal)	S4 (Origin Input)	D1 (Operation Status)	D2(Completion Output)
Tag Name:	D0000	M0000	I0002	I0003	M0010	M0100
Device Address:	D0000	M0000	I0002	I0003	M0010	M0100
Comment:						

FC6A Standard Mode

1. Select Mode

Selects the configuration mode. **FC6A Standard Mode** or **FC5A (except FC5A-D12X1E) Compatible Mode** can be selected.

Select **FC5A (except FC5A-D12X1E) Compatible Mode** to use the FC5A Series MICROSmart ZRN instruction specification.

When changing the PLC type from the FC5A/FC4A Series MICROSmart, **FC5A (except FC5A-D12X1E) Compatible Mode** is automatically selected.

The lowest frequency that can be used with **FC5A (except FC5A-D12X1E) Compatible Mode** is 20 Hz.

Note: The lowest frequency of pulses that can be output in FC5A (except FC5A-D12X1E) Compatible Mode is 20 Hz. A frequency lower than 20 Hz cannot be output. If such a value is specified, the pulse frequency error will occur.

The rest of this section is written under the assumption that **FC6A Standard Mode** has been selected.

Note: For details on the FC5A (except FC5A-D12X1E) Compatible Mode settings, refer to the ZRN instruction in Chapter 15 "Pulse Output Instructions" in the "FC5A Series MICROSmart Pentra User's Manual Advanced Volume".

2. Select instruction

This item selects which ZRN instruction to use ("ZRN1", "ZRN2", "ZRN3" or "ZRN4").

The output and the reversible control mode and frequency that can be set differ by the instruction and CPU module type.

For limitations due to the combination of instruction, reversible control mode, and the pulse output mode, see "17. Reversible control enable" on page 18-42.

3. S1 (source 1): Control register

S1 specifies the first data register of the data registers to use with ZRN1, ZRN2, ZRN3 or ZRN4 instructions.
Starting from the specified data register, 14 consecutive data registers are used.
Specify the first data register so that the device range is not exceeded.

Storage Destination	Function	Setting		Reference
		All-in-One CPU Module	CAN J1939 All-in-One CPU Module/ Plus CPU Module	
Starting number+0	Initial pulse frequency (high word)*1	ZRN1, ZRN2: 15 to 100,000 (increments of 1 Hz) ZRN3, ZRN4: 15 to 5,000 (increments of 1 Hz)	ZRN1 to ZRN4: 15 to 100,000 (increments of 1 Hz)	"10. Initial Pulse Frequency" on page 18-41
Starting number+1	Initial pulse frequency (low word)*1			
Starting number+2	Creep pulse frequency (high word)*1	ZRN1, ZRN2: 15 to 100,000 (increments of 1 Hz) ZRN3, ZRN4: 15 to 5,000 (increments of 1 Hz)	ZRN1 to ZRN4: 15 to 100,000 (increments of 1 Hz)	"11. Creep Pulse Frequency" on page 18-41
Starting number+3	Creep pulse frequency (low word)*1			
Starting number+4	Error status	10 to 10,000 ms		"12. Error status" on page 18-41
Starting number+5	**Reserved**			
Starting number+6	Steady pulse frequency (high word)*1 *2	ZRN1, ZRN2: 15 to 100,000 (increments of 1 Hz) ZRN3, ZRN4: 15 to 5,000 (increments of 1 Hz)	ZRN1 to ZRN4: 15 to 100,000 (increments of 1 Hz)	"13. Steady pulse frequency" on page 18-41
Starting number+7	Steady pulse frequency (low word)*1 *2			
Starting number+8	Acceleration time*2	10 to 10,000 ms		"14. Acceleration time" on page 18-41
Starting number+9	Deceleration time*2	10 to 10,000 ms		"15. Deceleration time" on page 18-42
Starting number+10	Control direction*2 *3	0: Forward 1: Reverse		"18. Control direction" on page 18-42
Starting number+11	**Reserved**			
Starting number+12	**Reserved**			
Starting number+13	**Reserved**			

*1 The upper and lower data registers change according to the 32-bit data storage method specified. For details, see "32-bit Data Storage" on page 3-9.

*2 When ZRN mode 0 is specified for the zero return method, starting number+6 to +10 are invalid.

*3 Valid only when **Reversible control (single-pulse output)** or **Reversible control (dual-pulse output)** is selected for reversible control enable.

4. S2 (source 2): Initialization Input

S2 specifies the initialization input. When the initialization input S2 is turned on, the initial values configured in the WindLDR **ZRN (Zero Return)** dialog box, on the **Settings** tab, are stored in the control registers. An external input or an internal relay can be specified. When the initialization input is on, the initial values are written to the data registers with each scan. (Even when the ZRN instruction is not executed (when not on), if the initialization input is turned on, the initial values are stored in the data registers.) To only initialize the values one time, use the initialization input in combination with the SOTU (single output up) instruction or the SOTD (single output down) instruction.

5. S3 (source 3): Proximity Signal

S3 specifies the proximity signal. The pulse frequency starts being decreased with the rise in the proximity signal.
An external input or an internal relay can be specified for the proximity signal.

Detection Speed	Input Device	Description
High-speed	I0, I1, I3, I4, I6, I7	An interrupt is used to read the proximity signal. The proximity signal can be read without being affected by the user program scan.
Normal	Inputs except I0, I1, I3, I4, I6, and I7 and Internal relays	The information updated in the END processing is read as the proximity signal. It is affected by the user program scan.

Notes:

- Do not use the same input or internal relay as the proximity signal for the ZRN1, ZRN2, ZRN3 and ZRN4 instructions. If they operate simultaneously, pulse output may not stop even if the proximity signal changes from on to off.
- To use the high-speed proximity signal, set the relevant input to **Normal Input** under **Special Input** on **Function Area Settings**. Do not use the input as interrupt input, catch input, high-speed counter, or frequency measurement.
- When using the high-speed proximity signal, ensure that no bounce occurs in the proximity signal.

6. S4 (source 4): Origin Input

S4 specifies the origin signal. The rise in the origin signal is detected and pulse output is stopped. An external input or an internal relay can be specified. When **ZRN mode 0** is selected for 9. Zero return method, the origin signal is ignored, even when specified.

Detection Speed	Input Device	Description
High-speed	I0, I1, I3, I4, I6, I7	An interrupt is used to read the origin signal. The origin signal is read without being affected by the user program scan.
Normal	Inputs except I0, I1, I3, I4, I6, and I7 and Internal relays	The information updated in the END processing is read as the origin signal. It is affected by the user program scan.

Notes:

- Do not use the same input or internal relay as the origin signal for the ZRN1, ZRN2, ZRN3, or ZRN4 instructions.
  If they operate simultaneously, pulse output may not stop even if the origin signal changes from on to off.
- To use a high-speed origin signal, set the relevant input to **Normal Input** under **Special Inputs** on **Function Area Settings**. Do not use the input as interrupt input, catch input, high-speed counter, or frequency measurement.
- When using a high-speed origin signal, ensure that no bounce occurs in the origin signal.

7. D1 (destination 1): Operation Status

D1 specifies the first internal relay of the internal relays that store the operation status.
Starting from the specified relay, 4 sequential relays are used.
Specify the first internal relay so that the device range is not exceeded.

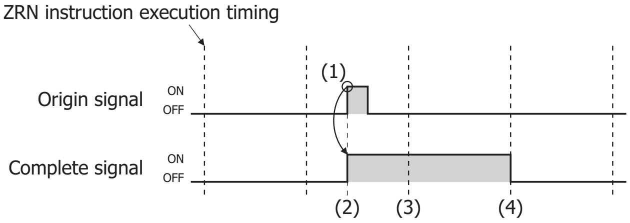
Storage Destination	Function	Setting	
Starting number+0	Pulse output ON	0: Pulse output OFF 1: Pulse output ON	This relay turns on during pulse output. This relay turns off when ZRN instruction input is turned off or when pulse output is complete.
Starting number+1	Pulse output complete	0: Pulse output not complete 1: Pulse output complete	This relay turns on when pulse output is complete. This relay turns off when pulse output starts.
Starting number+2	Pulse output status	0: Steady pulse output 1: Changing output pulse frequency	This relay turns on when the pulse output status is changing. This relay turns off when the pulse output status is steady.
Starting number+3	Zero return complete	0: Zero return not complete 1: Zero return complete	This relay turns on when the zero return operation is complete. This relay turns off when the zero return operation starts.

8. D2 (destination 2): Complete Output

D2 specifies the complete signal. When the rise in the complete signal is detected and pulse output is stopped, the complete signal can be output. An external output or an internal relay can be specified.

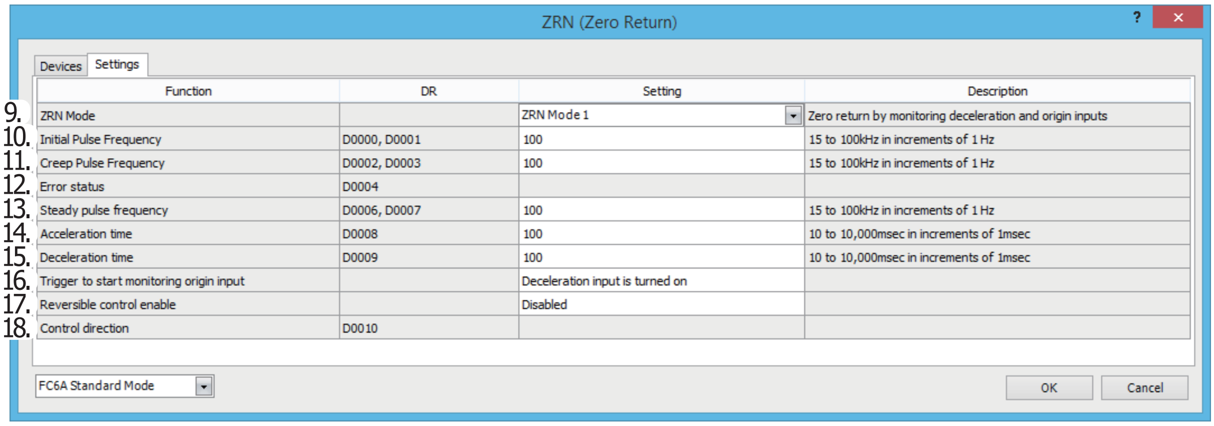
This can only be set when ZRN mode 1 is specified for the zero return method. This can be omitted if not using the complete signal. When ZRN mode 0 is specified, the complete signal is ignored, even when specified.

Note: When an external output (Q0 to Q17) is specified, the complete signal is turned on regardless of the ZRN instruction execution timing with the ladder and the I/O refresh timing. After the complete signal is turned on, it is reset to off when the ZRN instruction is executed a second time.



■ **Settings tab**

This tab configures the operation of the ZRN instruction functions.



ZRN (Zero Return)

Devices | Settings

	Function	DR	Setting	Description
9.	ZRN Mode		ZRN Mode 1	Zero return by monitoring deceleration and origin inputs
10.	Initial Pulse Frequency	D0000, D0001	100	15 to 100kHz in increments of 1 Hz
11.	Creep Pulse Frequency	D0002, D0003	100	15 to 100kHz in increments of 1 Hz
12.	Error status	D0004		
13.	Steady pulse frequency	D0006, D0007	100	15 to 100kHz in increments of 1 Hz
14.	Acceleration time	D0008	100	10 to 10,000msec in increments of 1msec
15.	Deceleration time	D0009	100	10 to 10,000msec in increments of 1msec
16.	Trigger to start monitoring origin input		Deceleration input is turned on	
17.	Reversible control enable		Disabled	
18.	Control direction	D0010		

FC6A Standard Mode

9. ZRN Mode

Select **ZRN mode 0** or **ZRN mode 1** for the zero return method according to the applicable system.

Setting	Description
ZRN mode 0	The zero return is performed by monitoring only the proximity signal.
ZRN mode 1	The zero return is performed by monitoring the proximity signal and the origin signal.

10. Initial Pulse Frequency

Specifies the frequency when pulse output starts.

Set between 15 Hz and 100,000 Hz in 1 Hz increments. When using the ZRN3 or ZRN4 instruction with the All-in-One CPU module, set between 15 Hz and 5,000 Hz in 1 Hz increments. The output frequency error is within ±5%.

11. Creep Pulse Frequency

After decreasing the pulse frequency has completed, this specifies the frequency when the pulses are steady.

Set between 15 Hz and 100,000 Hz in 1 Hz increments. When using the ZRN3 or ZRN4 instruction with the All-in-One CPU module, set between 15 Hz and 5,000 Hz in 1 Hz increments. The output frequency error is within ±5%.

12. Error status

Outputs the error code that corresponds to the content of an error when there is an error in the settings. If a configuration error occurs when the ZRN instruction is being executed, M8004 (user program execution error) is turned on and this relay is set to the error code.

Error Code	Status	Description
0	Normal	—
2	Pulse frequency designation error	The initial pulse frequency, creep pulse frequency, or steady pulse frequency was not set to a value between 15 to 100,000 Hz.
5	Frequency change time designation error	The frequency change time was not set between 10 and 10,000.
7	Control direction designation error	The control direction was not set to 0 or 1.
9	Initial pulse frequency designation error	The initial pulse frequency was set to the same frequency as the steady pulse frequency or it was set to a value larger than the steady pulse frequency.[*1]
10	Creep pulse frequency designation error	The creep pulse frequency was set to the same frequency as the steady pulse frequency or it was set to a value larger than the steady pulse frequency.[*2]

[*1] Set the initial pulse frequency so that it is lower than the steady pulse frequency.

[*2] Set the creep pulse frequency so that it is lower than the steady pulse frequency.

13. Steady pulse frequency

After increasing the pulse frequency has completed, this specifies the frequency when the pulses are steady.

Set between 15 Hz and 100,000 Hz in 1 Hz increments. When using the ZRN3 or ZRN4 instruction with the All-in-One CPU module, set between 15 Hz and 5,000 Hz in 1 Hz increments. The output frequency error is within ±5%.

14. Acceleration time

This setting specifies the time to increase the pulse frequency.

Set the time between 10 and 10,000 ms in increments of 1 ms. The first digit of the setting is handled as zero. For example, if 144 is entered, the set value is handled as 140 ms.

15. Deceleration time

This setting specifies the time to decrease the pulse frequency.

Set the time between 10 and 10,000 ms in increments of 1 ms. The first digit of the setting is handled as zero. For example, if 144 is entered, the set value is handled as 140 ms.

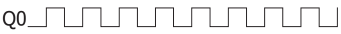
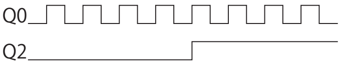
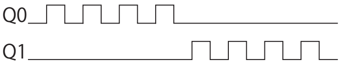
16. Trigger to start monitoring origin input

Specifies the timing to start the origin signal monitor.

Timing	Description
Proximity signal OFF→ON	After the proximity signal changes from off to on, the change from off to on in the origin signal is monitored.
Proximity signal ON→OFF	After the proximity signal changes from off to on and then next changes from on to off, the change from off to on in the origin signal is monitored.

17. Reversible control enable

Selects the reversible control method from the following reversible control modes when **ZRN mode 1** is selected for 9. Zero return method. (This is an example when ZRN1 is used with the All-in-One CPU module.)

Reversible Control Enable	Operation	Pattern
Reversible control disabled	Select this option when using pulse output in a single direction. Pulse A and pulse B can be used independently.	
Reversible control **Single-pulse output**	Pulse A is used as pulse output. Pulse B on/off is used as reversible control.	
Reversible control **Dual-pulse output**	Pulse A is used as forward pulse (CW) output.Pulse B is used as reverse pulse (CCW) output.	

The outputs used on the FC6A Series MICROSmart vary based on the instruction used, the combination of the pulse output mode and reversible control, and the model used.

Command	Operating Condition	Output Used			
		All-in-One CPU Module		CAN J1939 All-in-One CPU Module/Plus CPU Module	
		Pulse Output	Reversible Control Output	Pulse Output	Reversible Control Output
ZRN1	Reversible control disabled	Q0	—	Q0	—
	Reversible control Single-pulse output mode	Q0	Q2	Q0	Q1
	Reversible control Dual-pulse output mode	Q0, Q1	—	Q0, Q1	—
ZRN2	Reversible control disabled	Q1	—	Q2	—
	Reversible control Single-pulse output mode	Q1	Q3	Q2	Q3
	Reversible control Dual-pulse output mode	—	—	Q2, Q3	—
ZRN3	Reversible control disabled	Q2	—	Q4	—
	Reversible control Single-pulse output mode	—	—	Q4	Q5
	Reversible control Dual-pulse output mode	—	—	Q4, Q5	—
ZRN4	Reversible control disabled	Q3	—	Q6	—
	Reversible control Single-pulse output mode	—	—	Q6	Q7
	Reversible control Dual-pulse output mode	—	—	Q6, Q7	—

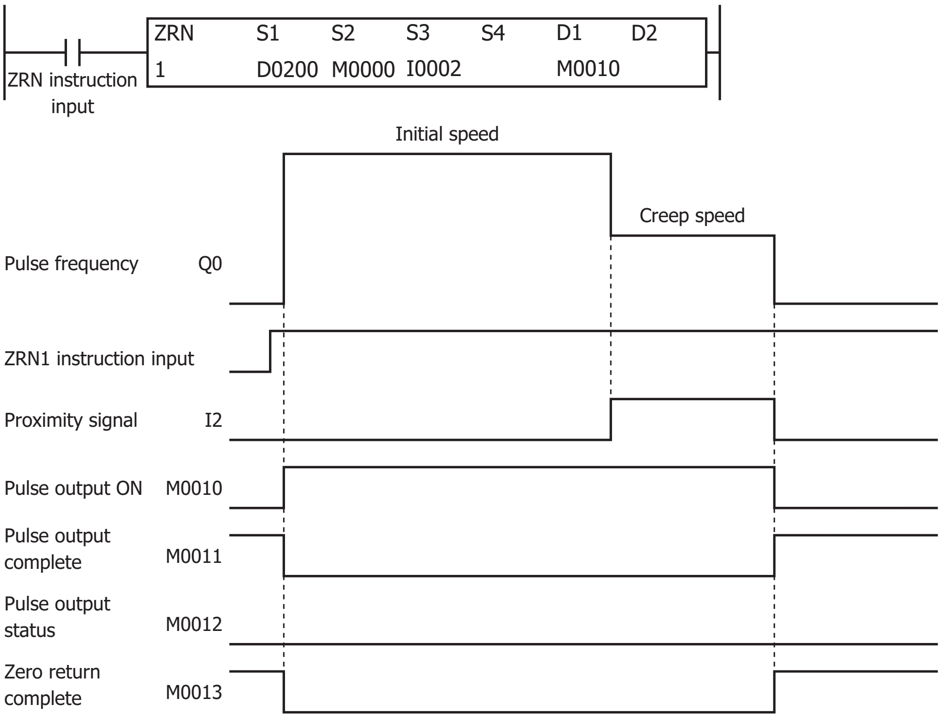
Note: The outputs (pulse output and reversible control output) used by ZRN instruction cannot be used by other instructions simultaneously.

18. Control direction

When reversible control is enabled, store 0 in this data register for forward operation and store 1 in this data register for reverse operation.

ZRM Mode 0 (When Using Only the Proximity Signal) Timing Chart

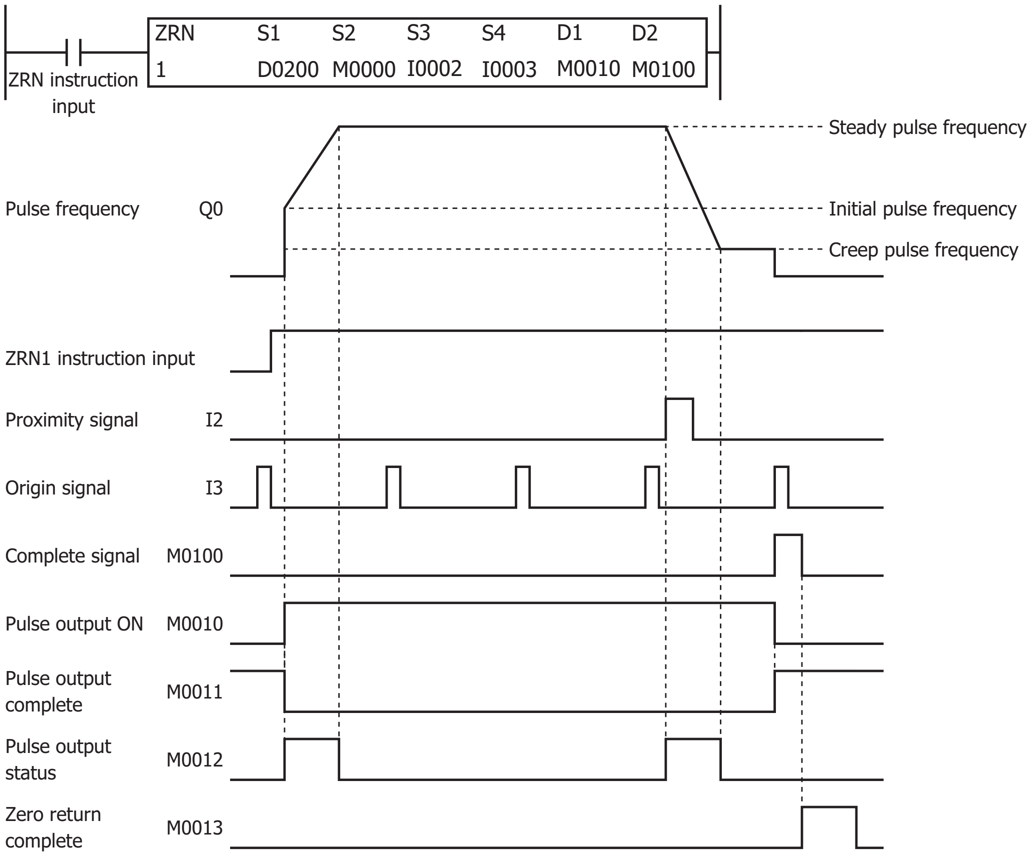
ZRN1 instruction, S1 is specified as data register D0200, S3 is specified as external input I2, D1 is specified as internal relay M0010



- When the ZRN1 instruction changes from off to on, pulses are output at the initial pulse frequency.
- When pulse output starts, M0010 turns on, and M0011 and M0013 turn off.
- The pulses are steady so M0012 turns off.
- When I2 changes from off to on, pulses are output at the creep pulse frequency.
- When I2 changes from on to off, pulse output stops.
- When pulse output stops, M0010 turns off, and M0011 and M0013 turn on.
- If the ZRN1 instruction input turns off during pulse output, pulse output stops. If the input turns on again, the operation starts from the beginning.
- Even if the contents of the data registers are changed during pulse output, the change is not reflected in pulse output operation. The changed content is reflected the next time the ZRN1 instruction is started.

ZRM Mode 1 (When Using the Proximity Signal and Origin Signal) Timing Chart

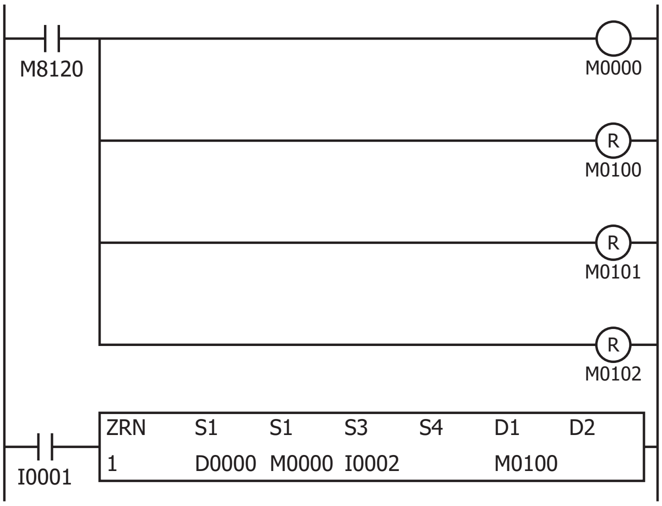
ZRN1 instruction, S1 is specified as data register D0200, S3 is specified as external input I2, D1 is specified as internal relay M0010



- When the ZRN1 instruction changes from off to on, pulses are output at the initial pulse frequency, and then pulses are output so that the frequency reaches the steady pulse frequency from the initial pulse frequency in the frequency change time. When the frequency change time is specified as 100, the steady pulse frequency is reached in 100 ms by increasing or decreasing the frequency every 10 ms.
- When pulse output starts, M0010 turns on and M0011 turns off.
- While the pulse speed is increasing or decreasing, M0012 turns on.
- When I2 changes from off to on, the pulse speed starts to decrease and reaches the creep pulse frequency.
- When I3 is detected as changing from off to on, pulse output stops.
- When pulse output stops, M0100 turns on. M0100 turns on for one scan or longer but less than two scans.
- When M0100 changes from on to off and the zero return operation has completed, M0013 turns on.
- Then when pulse output stops, M0010 turns off and M0011 turns on.
- If the ZRN1 instruction input turns off during pulse output, pulse output stops. If the input turns on again, the operation starts from the beginning.
- Even if the contents of the data registers are changed during pulse output, the change is not reflected in pulse output operation. The changed content is reflected the next time the ZRN1 instruction is started.

Example:

To perform a zero return operation with proximity signal I2, initial pulse frequency 3 kHz, and creep pulse frequency 800 Hz



Turn on initialization input (M0000)

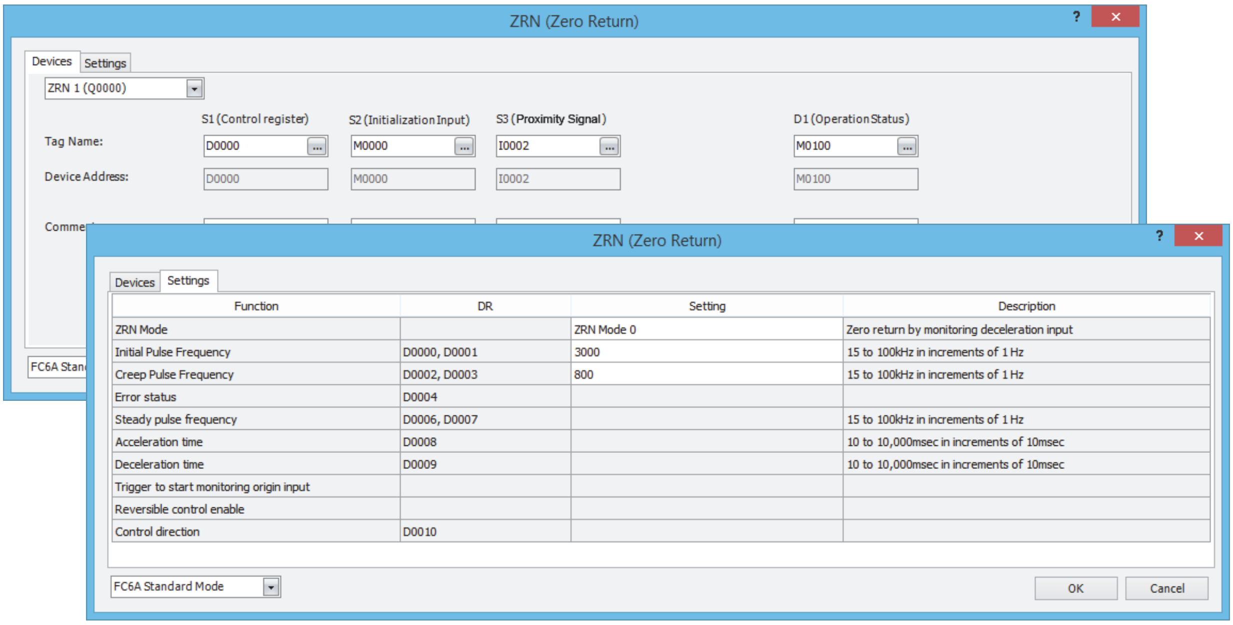
Pulse output ON (M0100) off

Pulse output complete (M0101) off

Pulse output status (M0102) off

When the ZRN1 instruction input turns on, pulse output starts

Setting



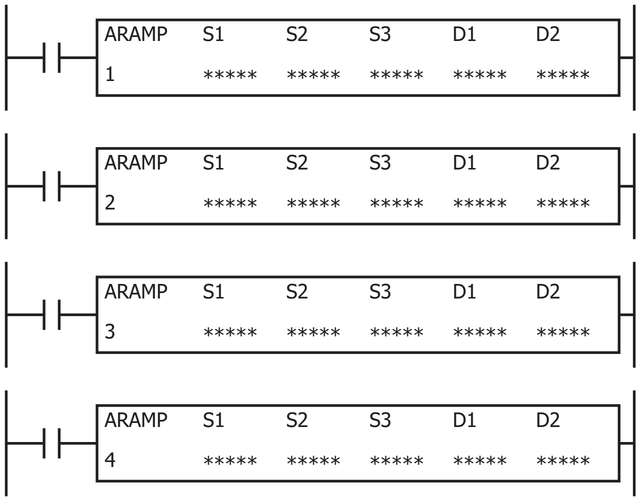
	S1 (Control register)	S2 (Initialization Input)	S3 (Proximity Signal)	D1 (Operation Status)
Tag Name:	D0000	M0000	I0002	M0100
Device Address:	D0000	M0000	I0002	M0100
Comme				

Function	DR	Setting	Description
ZRN Mode		ZRN Mode 0	Zero return by monitoring deceleration input
Initial Pulse Frequency	D0000, D0001	3000	15 to 100kHz in increments of 1 Hz
Creep Pulse Frequency	D0002, D0003	800	15 to 100kHz in increments of 1 Hz
Error status	D0004		
Steady pulse frequency	D0006, D0007		15 to 100kHz in increments of 1 Hz
Acceleration time	D0008		10 to 10,000msec in increments of 10msec
Deceleration time	D0009		10 to 10,000msec in increments of 10msec
Trigger to start monitoring origin input			
Reversible control enable			
Control direction	D0010		

FC6A Standard Mode

Function	Device Address	Preset Value	Details
Initial pulse frequency	D0000, D0001	3000	3,000 Hz
Creep pulse frequency	D0002, D0003	800	800 Hz

ARAMP (RAMP with Table)


ARAMP 1 S1 ***** S2 ***** S3 ***** D1 ***** D2 *****
ARAMP 2 S1 ***** S2 ***** S3 ***** D1 ***** D2 *****
ARAMP 3 S1 ***** S2 ***** S3 ***** D1 ***** D2 *****
ARAMP 4 S1 ***** S2 ***** S3 ***** D1 ***** D2 *****

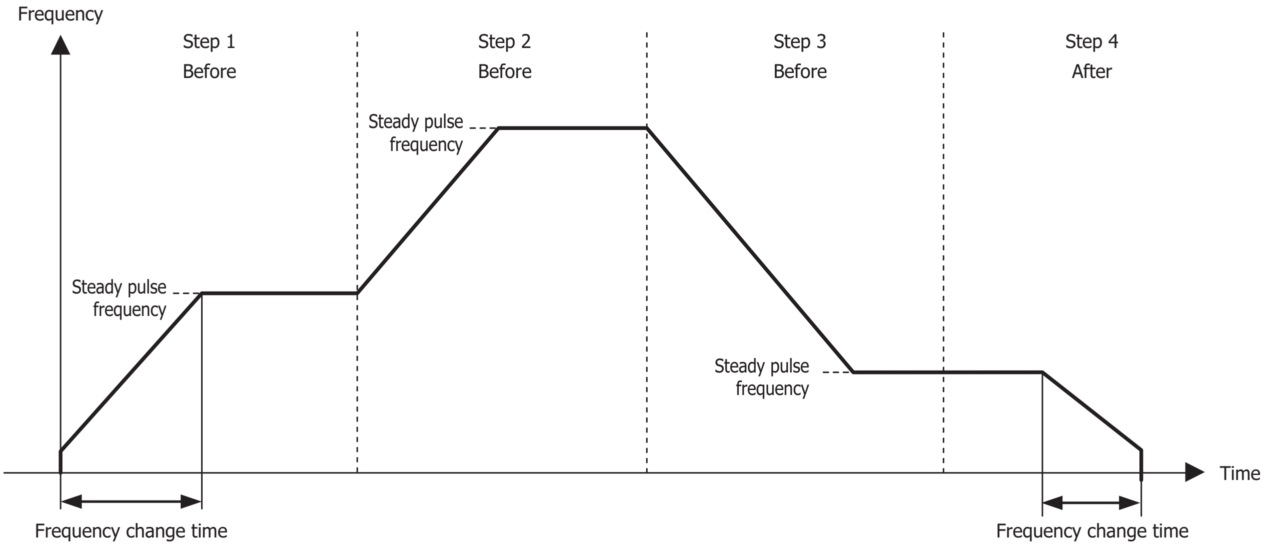
The ARAMP instructions output pulses with the frequency change function according to the information in the frequency table.

A frequency change and target frequency are set for each step, and the pulse frequency is controlled through the combination of these steps.

When the number of pulses that were output reaches the preset value, the next step that is specified for each step is executed. You can configure a maximum of 18 steps.

The pulse output operation can be selected from the following two operations with the step option settings.

- The pulse frequency is changed at a constant rate until it reaches the steady pulse frequency, and then a steady frequency of pulses is output at the steady pulse frequency.
  (Step 1 to 3 operations in the diagram below)
- After outputting pulses that maintain the frequency in the previous step, the frequency is changed at a constant rate until it reaches the steady pulse frequency.
  (Step 4 operation in the diagram below)



When the initialization input specified by S2 is turned on, the initial values configured in the WindLDR **ARAMP (Ramp Pulse Output with Table)** are stored in the control registers.
When the interrupt input specified by S3 is turned on, the step being executed is aborted and the interrupt step is executed.
The preset value and the steady pulse frequency for the running step are stored in the monitor registers specified by D1.
The control status including the pulse output status (output on/output direction/output complete) is stored in the operation status specified by D2.

Notes:

- If a pulse output instruction is simultaneously executed with the same output, a user program execution error will occur.
  Error code 48 will be stored in D8006 and instructions that are executed later will be canceled.
- The ARAMP instruction cannot be used in an interrupt program. If used in an interrupt program, a user program execution error will occur.
  Error code 18 will be stored in D8006 and instruction execution will be canceled.
- If a pulse output instruction is executed with the relay output type, a user program execution error will occur.
  Error code 19 is stored in D8006 and instruction execution is canceled.
- For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Devices

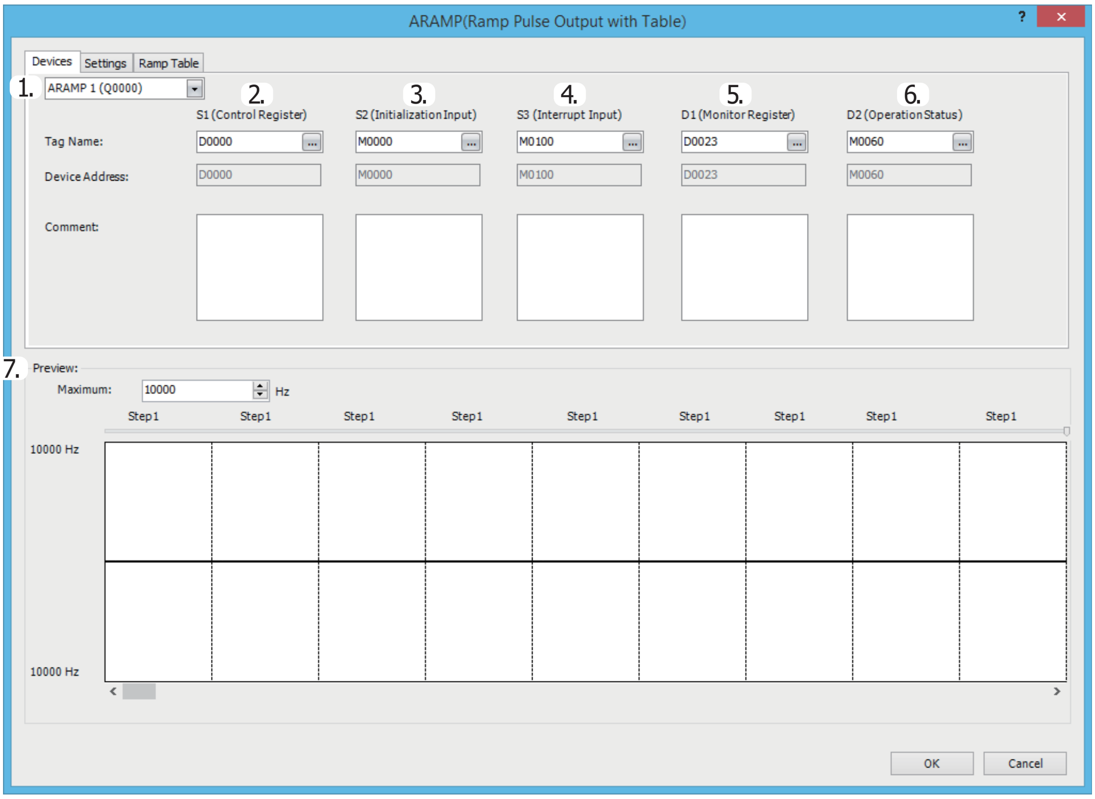
Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Control register	—	—	—	—	—	—	X*1	—	—	—
S2 (Source 2)	Initialization input	X	—	X	—	—	—	—	—	—	—
S3 (Source 3)	Interrupt input	X	—	X	—	—	—	—	—	—	—
D1 (Destination 1)	Monitor register	—	—	—	—	—	—	X*1	—	—	—
D2 (Destination 2)	Operation status	—	—	X*2	—	—	—	—	—	—	—

*1 Special data registers cannot be used.

*2 Special internal relays cannot be used. Only 0 can be specified as the first digit of the internal relay number. 1 to 7 cannot be specified.

Settings

■ Devices tab



ARAMP(Ramp Pulse Output with Table)

Devices | Settings | Ramp Table

1. ARAMP 1 (Q0000)

	2. S1 (Control Register)	3. S2 (Initialization Input)	4. S3 (Interrupt Input)	5. D1 (Monitor Register)	6. D2 (Operation Status)
Tag Name:	D0000	M0000	M0100	D0023	M0060
Device Address:	D0000	M0000	M0100	D0023	M0060
Comment:					

7. Preview:

Maximum: 10000 Hz

1. Select instruction

This item selects which ARAMP instruction to use ("ARAMP1", "ARAMP2", "ARAMP3" or "ARAMP4").

The output, reversible control mode, and operation mode that can be selected differ by the instruction and CPU module type.

For limitations based on the combination of instruction, reversible control mode, and the pulse output mode, see "8. Reversible control enable" on page 18-52.

2. S1 (source 1): Control Register

S1 specifies the first data register of the data registers to use with the ARAMP1, ARAMP2, ARAMP3 or ARAMP4 instruction.
Starting from the specified data register, "2+8 x N (N: number of steps)" consecutive data registers are used.
The range of data registers that can be specified depends on the number of steps.
If the number of steps is 1, 10 words are required.
If the number of steps is 18, 146 words are required.
Specify the first data register so that the device range is not exceeded.
The steps operate with the settings that are configured when the steps start. If the settings for a step are changed after it starts running, those changes are not reflected while the step is running.

Storage Destination	Function	Setting		Reference
		All-in-One CPU Module	CAN J1939 All-in-One CPU Module/ Plus CPU Module	
Starting number+0	Interrupt number	1 to 18		"11. Interrupt step number" on page 18-52
Starting number+1	**Reserved**			
Step 1 (8 words)				
Starting number+2, Starting number+3	Steady pulse frequency	ARAMP1, ARAMP2: 15 to 100,000 (increments of 1 Hz) ARAMP3, ARAMP4: 15 to 5,000 (increments of 1 Hz)	ARAMP1 to ARAMP4: 15 to 100,000 (increments of 1 Hz)	"13. Steady pulse frequency" on page 18-53
Starting number+4	**Reserved**			
Starting number+5	Frequency change time	10 to 10,000 ms		"14. Frequency change time" on page 18-53
Starting number+6	Preset value (high word)*1	1 to 100,000,000 pulses		"15. Preset value" on page 18-53
Starting number+7	Preset value (low word)*1			
Starting number+8	Step options	0 to 3		"16. Step options" on page 18-53
Starting number+9	Next step number	1 to 18		"17. Next step number" on page 18-54
Step 2 (8 words)				
Starting number+10, Starting number+11	Steady pulse frequency	ARAMP1, ARAMP2: 15 to 100,000 (increments of 1 Hz) ARAMP3, ARAMP4: 15 to 5,000 (increments of 1 Hz)	ARAMP1 to ARAMP4: 15 to 100,000 (increments of 1 Hz)	"13. Steady pulse frequency" on page 18-53
:	:	:		:
Starting number+17	Next step number	1 to 18		"17. Next step number" on page 18-54
: Step N (8 words)				
Starting number+2+N x 8-8 Starting number+3+N x 8-8	Steady pulse frequency	ARAMP1, ARAMP2: 15 to 100,000 (increments of 1 Hz) ARAMP3, ARAMP4: 15 to 5,000 (increments of 1 Hz)	ARAMP1 to ARAMP4: 15 to 100,000 (increments of 1 Hz)	"13. Steady pulse frequency" on page 18-53
:	:	:		:
Starting number+9+N x 8-8	Next step number	1 to 18		"17. Next step number" on page 18-54

*1 The upper and lower data registers change according to the 32-bit data storage method specified.
For details, see "32-bit Data Storage" on page 3-9.

3. S2 (source 2): Initialization Input

S2 specifies the initialization input.

When the initialization input S2 is turned on, the initial values configured in the WindLDR **ARAMP (Advanced Ramp)** dialog box, on the **Settings** tab, are stored in the control registers. An external input or an internal relay can be specified.

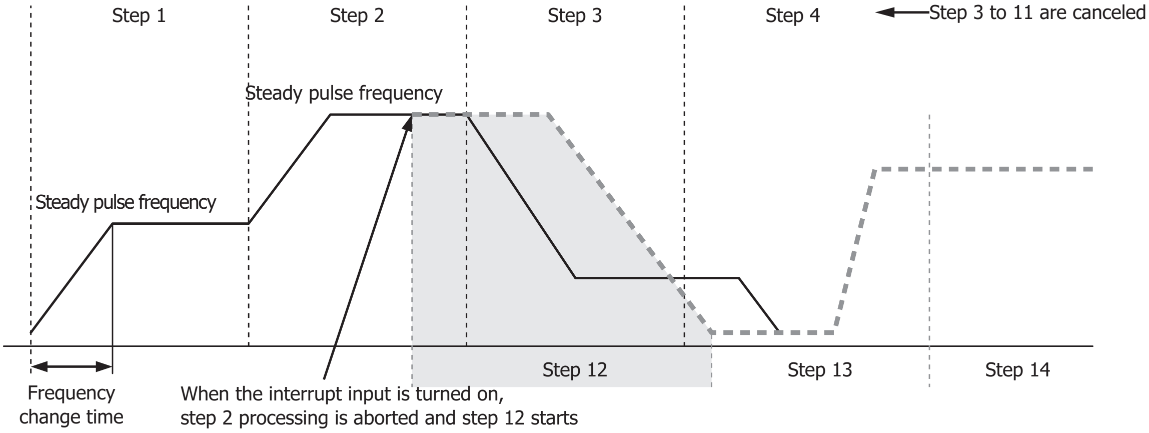
When the initialization input is on, the initial values are stored in the data registers with each scan. (Even when the ARAMP instruction is not executed (when not on), if the initialization input is turned on, the initial values are stored in the data registers.) To only initialize the values one time, use the initialization input in combination with the SOTU (single output up) instruction or the SOTD (single output down) instruction.

4. S3 (source 3): Interrupt Input

S3 specifies the interrupt input.

When the interrupt input changes from off to on, the pulse output process for the running step is aborted, and pulse output starts with the settings for the step configured by the interrupt step number (10). When pulse output for the interrupt step completes, pulse output starts for the next step according to the next step number (16).

The following example shows the interrupt input turning on during step 2 when configured to output pulses in order from step 1 to step 18.



When the interrupt input turns on, step 2 which is currently outputting pulses is aborted, and pulse output starts from the step number set as the interrupt step number (step 12). When step 12 completes, pulses are output for the next step according to the set order of steps.

An external input or internal relay can be specified as the interrupt input.
The detection speed differs by the used device.

Detection Speed	Device Number	Description
High-speed	I0, I1, I3, I4, I6, I7	An interrupt is used to read the interrupt input. The interrupt input can be read without being affected by the user program scan.
Normal	Inputs except I0, I1, I3, I4, I6, and I7 and Internal relays	The information updated in the END processing is read as the interrupt input. It is affected by the user program scan.

S3 is omitted if not using an interrupt input.

Notes:

- Do not use the same input or internal relay as the interrupt input signal for the ARAMP1, ARAMP2, ARAMP3, or ARAMP4 instructions.
- To use a high-speed interrupt input signal, set the relevant input to **Normal Input** under **Special Inputs** on **Function Area Settings**. Do not use the input as interrupt input, catch input, high-speed counter, or frequency measurement.
- When using the high-speed interrupt input, ensure that no bounce occurs in the interrupt input.
- When the interrupt input is turned on, the control direction (forward or reverse) is unchanged regardless of the control direction of the interrupted step.

5. D1 (destination 1): Monitor Register

D1 specifies the first data register of the data registers to use with ARAMP1, ARAMP2, ARAMP3 or ARAMP4. Starting from the specified data register, 11 consecutive data registers are used. Specify the first data register so that the device range is not exceeded. The contents of the monitor registers are read-only.

Address	Function	Setting		Reference
		All-in-One CPU Module	CAN J1939 All-in-One CPU Module/ Plus CPU Module	
Starting number+0	Next step number	0 to 18		"Next step number" on page 18-50
Starting number+1	Running step number	1 to 18		"Running step number" on page 18-50
Starting number+2	Steady pulse frequency monitor (high word)*1	ARAMP1, ARAMP2: 15 to 100,000 (increments of 1 Hz) ARAMP3, ARAMP4: 15 to 5,000 (increments of 1 Hz)	ARAMP1 to ARAMP4: 15 to 100,000 (increments of 1 Hz)	"Steady pulse frequency monitor" on page 18-50
Starting number+3	Frequency change time monitor (low word)*1			
Starting number+4	**Reserved**			
Starting number+5	Frequency change time monitor	10 to 10,000 (increments of 1 ms) The first digit of the setting is handled as zero.		"Frequency change time monitor" on page 18-50
Starting number+6	Preset value monitor (high word)*1	1 to 100,000,000 pulses		"Preset value monitor" on page 18-50
Starting number+7	Preset value monitor (low word)*1			
Starting number+8	Current value (high word)*1	1 to 100,000,000 pulses		"Current value" on page 18-50
Starting number+9	Current value (low word)*1			
Starting number+10	Error status	0 to 9		"Error status" on page 18-51

*1 The upper and lower data registers change according to the 32-bit data storage method specified.
For details, see "32-bit Data Storage" on page 3-9.

Next step number

This register stores the number of the step to execute next.

If the next step is 0, pulse output ends after the currently executed step is complete.

Running step number

This register stores the number of the step that is currently being executed.

Steady pulse frequency monitor

This register stores the steady pulse frequency for the step that is currently being executed.

Frequency change time monitor

This register stores the frequency change time for the step that is currently being executed.

Set the time between 10 and 10,000 ms in increments of 1 ms. The first digit of the setting is handled as zero. For example, if 144 is entered, the set value is handled as 140 ms.

Preset value monitor

This register stores the number of pulses to output for the running step.

Current value

This register stores the number of pulses that have been output for the step that is currently being executed. The current value is updated when the ARAMP instruction is executed at each scan.

Error status

Outputs the error code that corresponds to the content of an error when there is an error in the settings. If a configuration error occurs when a step starts executing, a user program execution error will occur, and the error code 20 is stored in D8006.

Error Code	Status	Description		
0	Normal	—		
3	Preset value designation error	The preset value was not set between 1 to 100,000,000.		
4	Steady pulse frequency designation error	All-in-One CPU module	ARAMP1, ARAMP2	The steady pulse frequency was not set between 15 and 100,000.
			ARAMP3, ARAMP4	The steady pulse frequency was not set between 15 and 5,000.
		CAN J1939 All-in-One CPU module/Plus CPU module	ARAMP1 to ARAMP4	The steady pulse frequency was not set between 15 and 100,000.
5	Frequency change time designation error	The frequency change time was not set between 10 and 10,000.		
7	Step options designation error	The step options were not set to a valid value.		
8	Next step number destination error	The next step number was not set between 0 and 18.		
9	Interrupt number destination error	The interrupt number was not set between 1 and 18.		

6. D2 (destination 2): Operation Status

D2 specifies the starting number of the internal relays to use with ARAMP1, ARAMP2, ARAMP3 and ARAMP4 instructions.

Starting from the specified internal relay, 5 sequential internal relays are used.

Specify the starting number so that the device range is not exceeded.

Address	Description		
Starting number+0	Pulse output ON	0: Pulse output OFF 1: Pulse output ON	This relay turns on during pulse output. This relay turns off when the ARAMP instruction output stops. This relay turns off when the specified number of pulses are output and output ends
Starting number+1	Pulse output complete	0: Pulse output not complete 1: Pulse output complete	This relay turns on when pulse output completes. This relay turns on when the step number currently being executed is 0. This relay turns off when the ARAMP instruction output starts.
Starting number+2	Pulse output status	0: Steady pulse output 1: Changing output pulse frequency	This relay turns off when the pulse output status is steady. This relay turns on when the pulse output is changing.
Starting number+3	Overflow	0: None 1: An overflow has occurred	This relay turns on when the pulses output exceed the configured preset value. Pulse output continues even if an the overflow occurs during steady output or while the pulse frequency is changing. However, pulse counting (current value data register) is suspended at the point when the overflow occurred.
Starting number+4	Pulse output direction	0: Forward 1: Reverse	This relay turns off when the output direction of the pulses being output is forward. This relay turns on when the output direction of the pulses being output is reverse.

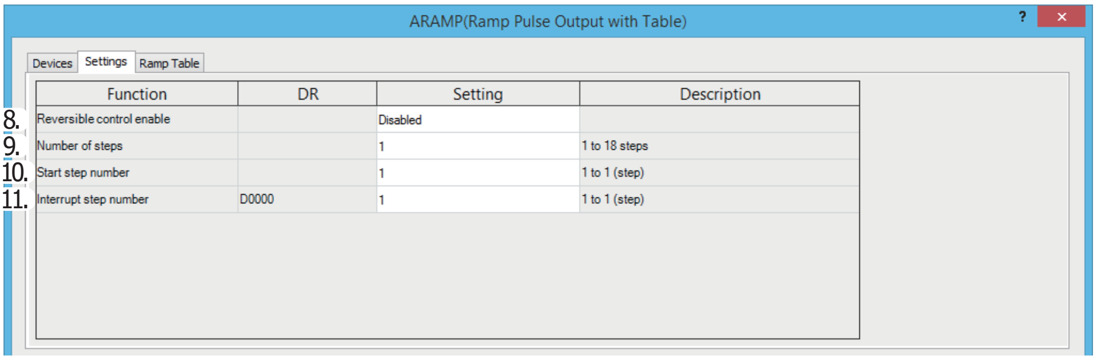
7. Preview

Displays a preview of the configured ARAMP instruction operation. The change in pulse output frequency, forward/reverse operation, and the execution order of the steps can be checked.

The vertical axis indicates the pulse frequency and the horizontal axis indicates time.

The width of each step is locked, so the actual proportion of the horizontal axis is not accurate.

■ Settings tab



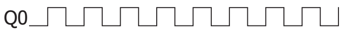
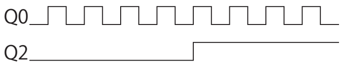
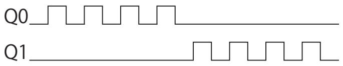
ARAMP(Ramp Pulse Output with Table)

Devices | Settings | Ramp Table

	Function	DR	Setting	Description
8.	Reversible control enable		Disabled	
9.	Number of steps		1	1 to 18 steps
10.	Start step number		1	1 to 1 (step)
11.	Interrupt step number	D0000	1	1 to 1 (step)

8. Reversible control enable

This setting enables or disables reversible control and selects the reversible control method from the following modes. There are two modes for the pulse output mode: single-pulse and dual-pulse. They can be combined with reversible control as follows. (This is an example when ARAMP is used with the All-in-One CPU module.)

Reversible Control Enable	Operation	Pattern
Reversible control disabled	Select this option when using pulse output in a single direction. Pulse A and pulse B can be used independently.	
Reversible control **Single-pulse output**	Pulse A is used as pulse output. Pulse B on/off is used as reversible control.	
Reversible control **Dual-pulse output**	Pulse A is used as forward pulse (CW) output. Pulse B is used as reverse pulse (CCW) output.	

The outputs used on the FC6A Series MICROSmart varies based on the instruction used, the combination of the pulse output mode and reversible control, and the model used.

Instruction	Operating Condition	Output Used			
		All-in-One CPU Module		CAN J1939 All-in-One CPU Module/Plus CPU Module	
		Pulse Output	Reversible Control Output	Pulse Output	Reversible Control Output
ARAMP1	Reversible control disabled	Q0	—	Q0	—
	Reversible control (single-pulse output)	Q0	Q2[*1]	Q0	Q1
	Reversible control (dual-pulse output)	Q0, Q1[*2]	—	Q0, Q1	—
ARAMP2	Reversible control disabled	Q1	—	Q2	—
	Reversible control (single-pulse output)	Q1	Q3[*1]	Q2	Q3
	Reversible control (dual-pulse output)	—	—	Q2, Q3	—
ARAMP3	Reversible control disabled	Q2	—	Q4	—
	Reversible control (single-pulse output)	—	—	Q4	Q5
	Reversible control (dual-pulse output)	—	—	Q4, Q5	—
ARAMP4	Reversible control disabled	Q3	—	Q6	—
	Reversible control (single-pulse output)	—	—	Q6	Q7
	Reversible control (dual-pulse output)	—	—	Q6, Q7	—

[*1] When using single-pulse output, Q2 or Q3 will be used, so a Q2 or Q3 pulse output instruction cannot be used.

[*2] When using dual-pulse output, Q1 will be used, so a Q1 pulse output instruction cannot be used.

9. Number of steps

This setting specifies the number of steps. The maximum is 18.

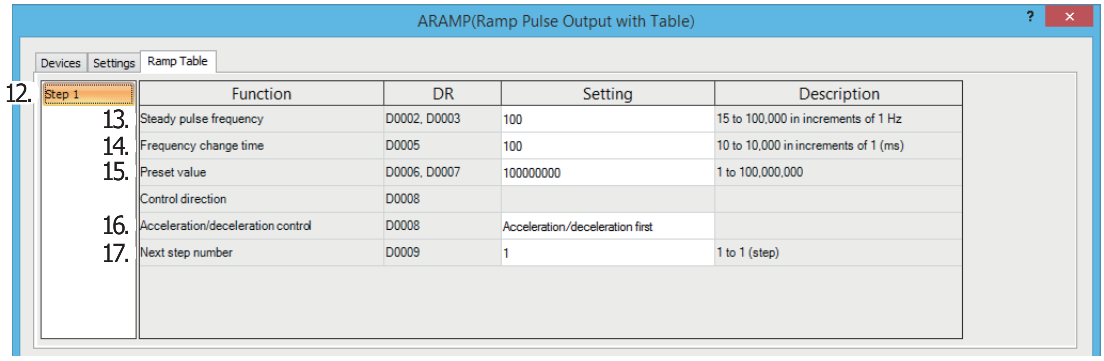
10. Start step number

When the input changes from off to on, pulse output starts with the settings for the step configured as the start step number.

11. Interrupt step number

When the interrupt input changes from off to on, the pulse output process for the running step is aborted, and pulse output restarts with the settings for the step configured by the interrupt step number.

- **Ramp Table tab**



ARAMP(Ramp Pulse Output with Table)

Devices | Settings | Ramp Table

12. Step 1

	Function	DR	Setting	Description
13.	Steady pulse frequency	D0002, D0003	100	15 to 100,000 in increments of 1 Hz
14.	Frequency change time	D0005	100	10 to 10,000 in increments of 1 (ms)
15.	Preset value	D0006, D0007	100000000	1 to 100,000,000
	Control direction	D0008		
16.	Acceleration/deceleration control	D0008	Acceleration/deceleration first	
17.	Next step number	D0009	1	1 to 1 (step)

12. Step numbers

This option selects the step number to configure. The output frequency error is ±5%.

13. Steady pulse frequency

This setting specifies the frequency at the steady output state before or after changing the frequency.

CPU Module Type	Instruction	Steady Pulse Frequency
All-in-One CPU module	ARAMP1, ARAMP2	Set this value in the range of 15 to 100,000 (increments of 1 Hz).
	ARAMP3, ARAMP4	Set this value in the range of 15 to 5,000 (increments of 1 Hz).
CAN J1939 All-in-One CPU module/Plus CPU module	ARAMP1 to ARAMP4	Set this value in the range of 15 to 100,000 (increments of 1 Hz).

14. Frequency change time

This setting specifies the time to change the pulses.

Set this value in the range of 10 to 10,000 (increments of 1 ms).

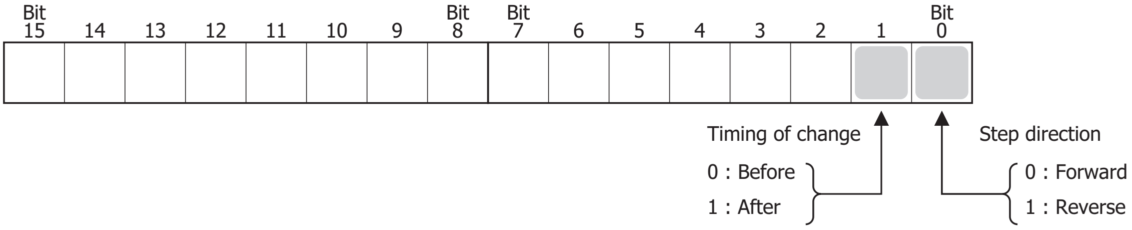
The first digit of the setting is handled as zero. For example, if 144 is entered, the set value is handled as 140 ms.

15. Preset value

This setting configures the number of pulses to output from 1 to 100,000,000.

16. Step options

This setting configures the step direction and the execution timing of the change in the pulses.

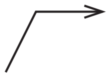
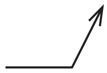
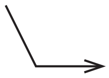
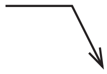


Control direction

When reversible control is enabled, store 0 in this data register for forward operation and store 1 in this data register for reverse operation.

Acceleration/deceleration control

The frequency changes as shown in the following diagram according to the setting for the execution timing of the change. In the Before column, the frequency changes and then becomes steady. When the number of pulses in the preset value is output, the instruction transitions to the next step. In the After column, the steady pulse frequency is maintained, then the frequency changes, and the instruction transitions to the next step.

	Before	After
Increase		
Decrease		

Note: Depending on the combination of steps and the interrupt input timing, forward and reverse may switch at high frequencies. Program the user program so that forward and reverse can be switched according to the application specifications that will be used.

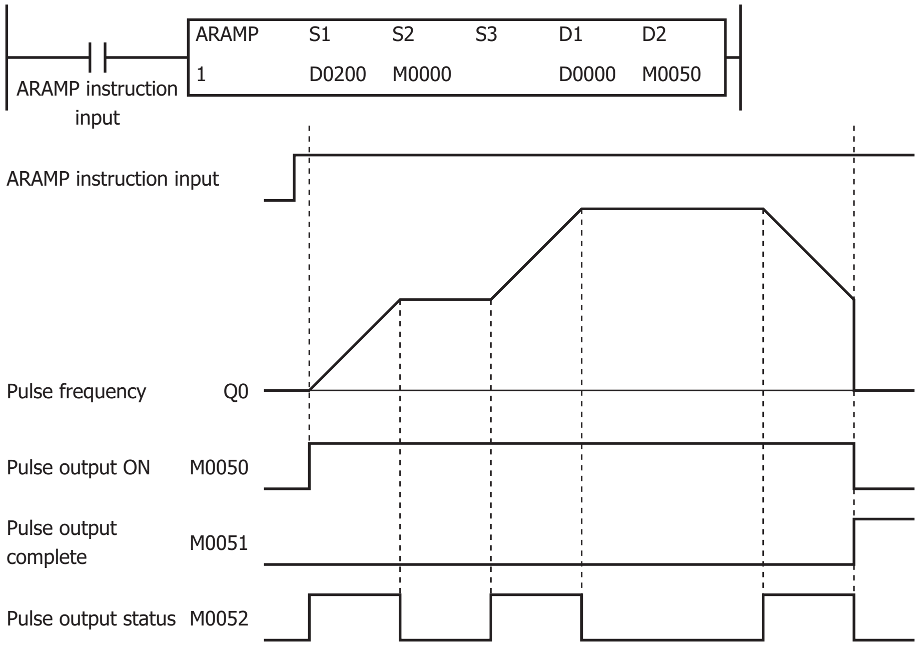
17. Next step number

This setting specifies the number of the next step to be executed after the output for the currently executed step is completed.

When the next step number is set to 0, that step becomes the last step, and when pulse output at that step is complete, pulse output will end.

ARAMP1 instruction (reversible control disabled) timing chart

ARAMP1 instruction, S1 is specified as data register D0200, S2 is specified as internal relay M0000, S3 is disabled, D1 is specified as data register D0000, D2 is specified as internal relay M0050

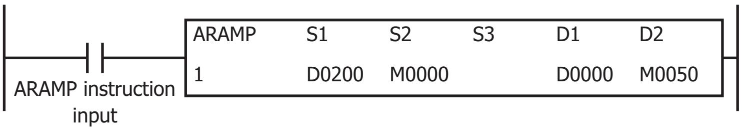


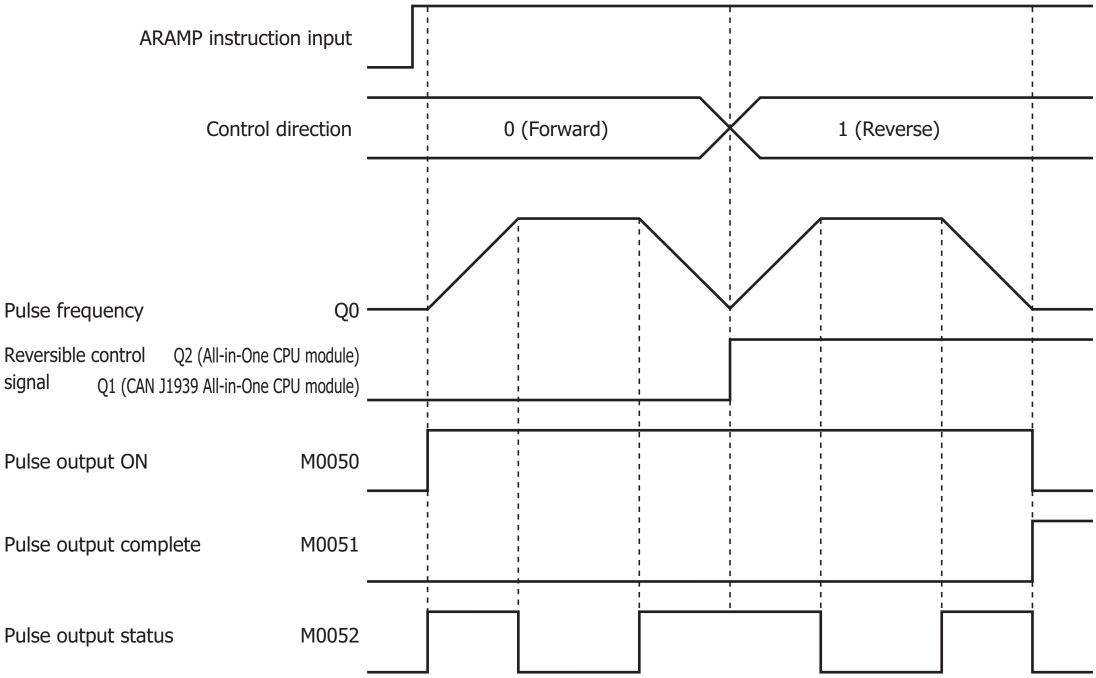
When the ARAMP instruction input changes from off to on, pulses are output according to the settings configured by the data registers. When pulse output starts, M0050 turns on. M0052 turns on while the pulse frequency is increasing or decreasing. When the number of pulses configured for each step is output, the next step is executed. When pulse output is complete for the step where the next step number is set to 0, then the pulses will stop. In this situation, M0050 turns off and M0051 turns on.

If the ARAMP instruction input turns off during pulse output, pulse output ends. If the ARAMP instruction input turns on again, the operation starts from the beginning. Even if the contents of the data registers are changed during pulse output, the change is not reflected in the pulse output operation. The changed content is reflected the next time the ARAMP instruction is started.

ARAMP1 instruction (single-pulse output reversible control enabled) timing chart

ARAMP1 instruction, S1 is specified as data register D0200, S2 is specified as internal relay M0000, S3 is disabled, D1 is specified as data register D0000, D2 is specified as internal relay M0050





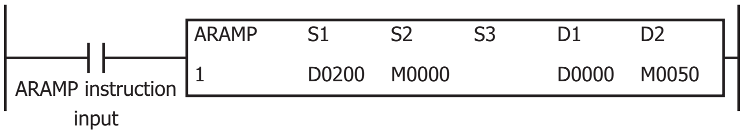
When the ARAMP instruction input changes from off to on, pulses are output from Q0 according to the settings configured by the data registers. The reversible control signal is output from Q2 or Q1. When pulse output starts, M0050 turns on. M0052 turns on while the pulse frequency is increasing or decreasing. When the number of pulses configured for each step is output, the next step is executed. When pulse output is complete for the step where the next step number is set to 0, then the pulses will stop. In this situation, M0050 turns off and M0051 turns on.

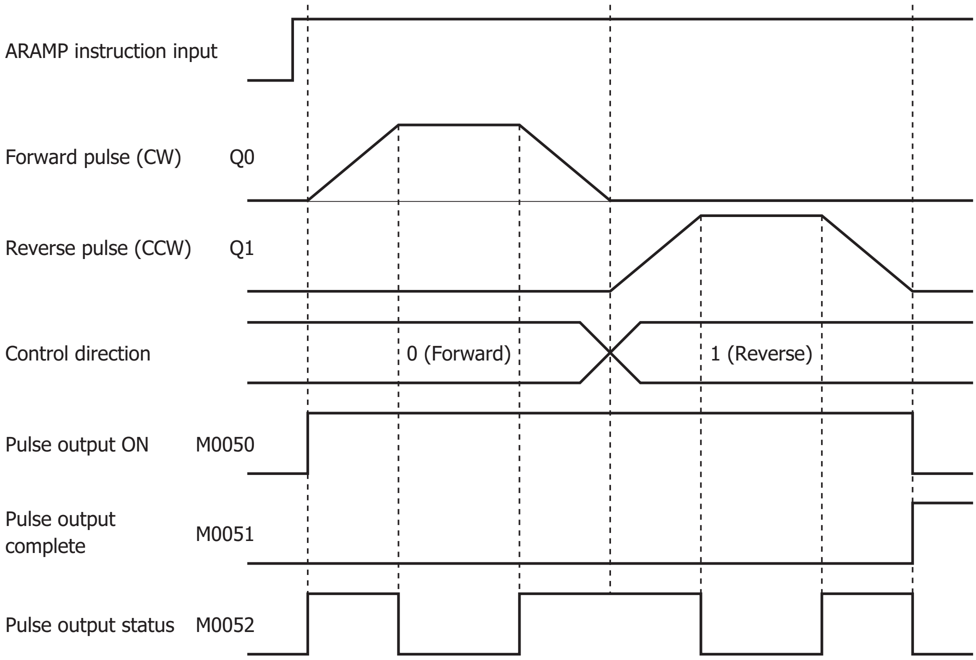
If the ARAMP instruction input turns off during pulse output, pulse output ends. If this input turns on again, the operation starts from the beginning. Even if the contents of the data registers are changed during pulse output, the change is not reflected in the pulse output operation. The changed content is reflected the next time the ARAMP instruction is started.

In this application, switching between forward and reverse must be executed when the pulse frequency is decreased to the minimum value, so a step is inserted to decrease the frequency before reversing.

ARAMP1 instruction (dual-pulse output reversible control enabled) timing chart

All-in-One CPU module, ARAMP1 instruction, S1 is specified as data register D0200, S2 is specified as internal relay M0000, S3 is disabled, D1 is specified as data register D0000, D2 is specified as internal relay M0050





When the ARAMP instruction input changes from off to on, pulses are output from Q0 according to the settings configured by the data registers. The reversible control signal is output from Q0 or Q1. When pulse output starts, M0050 turns on. M0052 turns on while the pulse frequency is increasing or decreasing. The pulses increase and decrease according to the frequency change time until they reach the steady pulse frequency from the current pulse frequency. When the configured number of pulses are output, the pulses stop. In this situation, M0050 turns off and M0051 turns on.

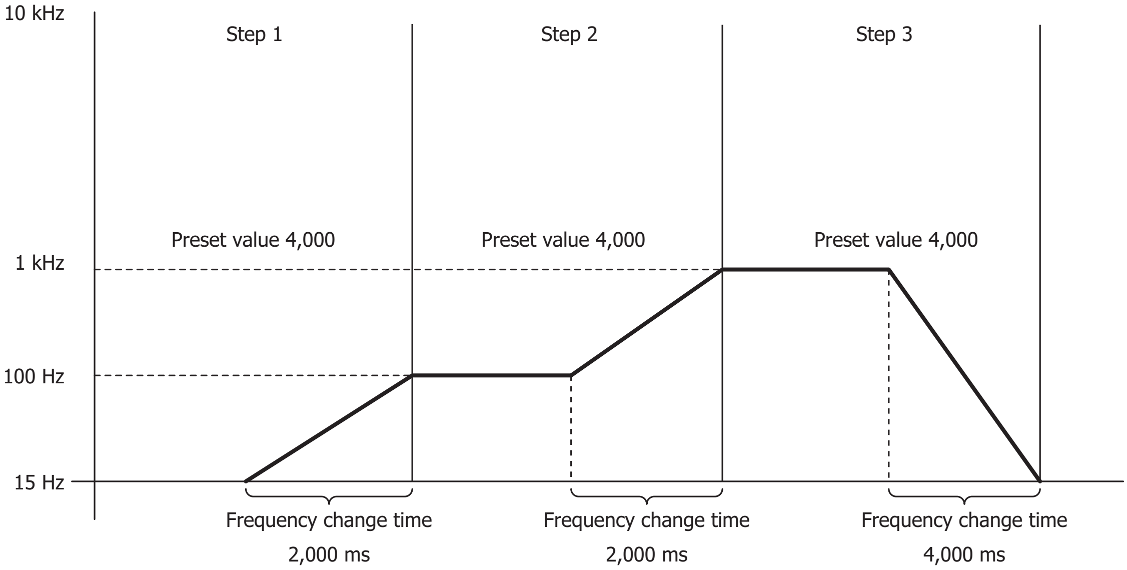
If the ARAMP instruction input turns off during pulse output, pulse output ends. If the ARAMP instruction input turns on again, the operation starts from the beginning. Even if the contents of the data registers are changed during pulse output, the change is not reflected in the pulse output operation. The changed content is reflected the next time the ARAMP instruction is started.

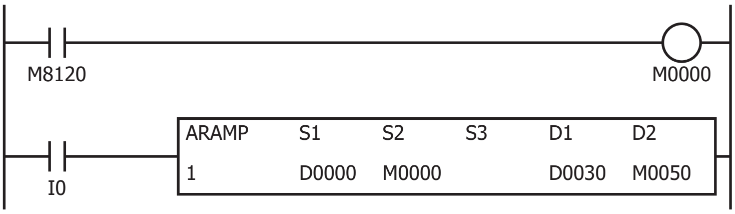
In this application, switching between forward and reverse must be executed when the pulse frequency is decreased to the minimum value, so a step is inserted to decrease the frequency before reversing.

Sample program

When outputting pulses as shown in the diagram below with the frequency change function (reversible control disabled) using the following settings

The pulses are output from Q0.

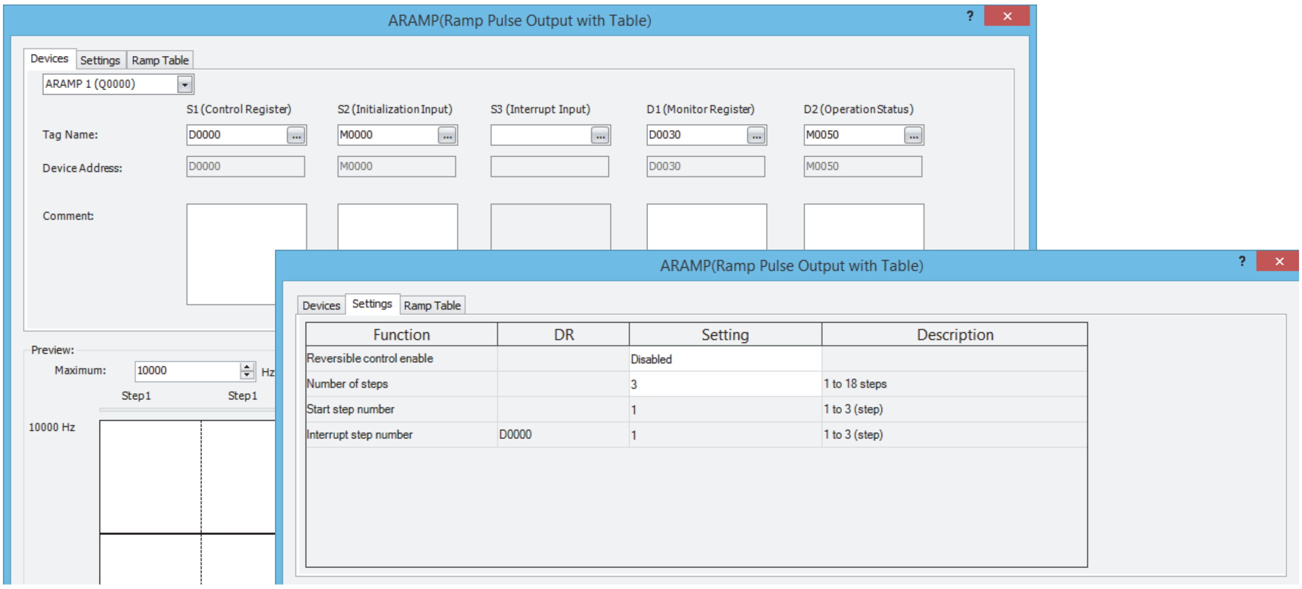




Initialize the data registers when the PLC starts

When the ARAMP instruction input (I0) turns on, pulse output starts

Basic settings



ARAMP(Ramp Pulse Output with Table)

Devices | Settings | Ramp Table

ARAMP 1 (Q0000)

	S1 (Control Register)	S2 (Initialization Input)	S3 (Interrupt Input)	D1 (Monitor Register)	D2 (Operation Status)
Tag Name:	D0000	M0000		D0030	M0050
Device Address:	D0000	M0000		D0030	M0050
Comment:					

Preview:
Maximum: 10000 Hz
Step1 Step1
10000 Hz

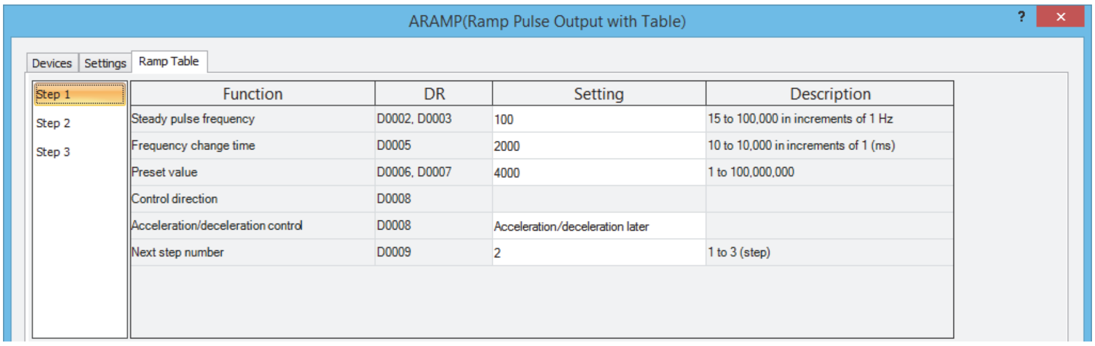
ARAMP(Ramp Pulse Output with Table)

Devices | Settings | Ramp Table

Function	DR	Setting	Description
Reversible control enable		Disabled	
Number of steps		3	1 to 18 steps
Start step number		1	1 to 3 (step)
Interrupt step number	D0000	1	1 to 3 (step)

Function	Device Address	Setting Value	Details
Reversible control enable	—	Disable	—
Number of steps	—	3	—
Start step number	—	1	Step 1
Interrupt step number	D0000	1	—

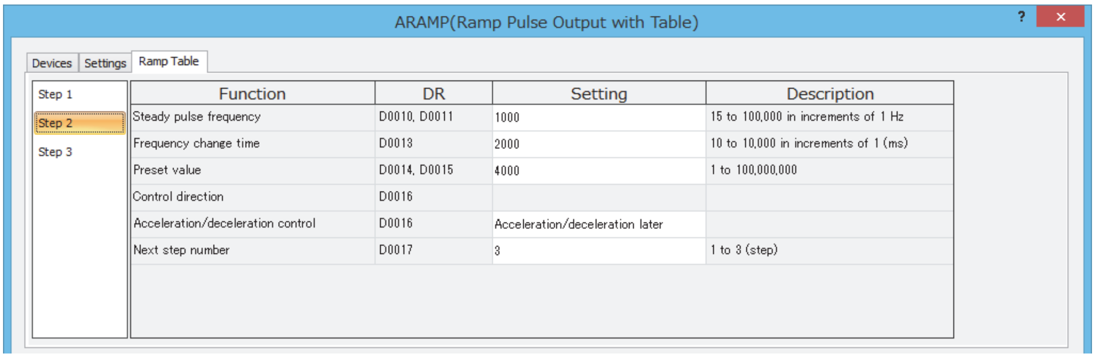
Step 1 settings



Function	DR	Setting	Description
Steady pulse frequency	D0002, D0003	100	15 to 100,000 in increments of 1 Hz
Frequency change time	D0005	2000	10 to 10,000 in increments of 1 (ms)
Preset value	D0006, D0007	4000	1 to 100,000,000
Control direction	D0008		
Acceleration/deceleration control	D0008	Acceleration/deceleration later	
Next step number	D0009	2	1 to 3 (step)

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0002, D0003	100	100 Hz
Frequency change time	D0005	2000	2,000 ms
Preset value	D0006, D0007	4000	Preset value=4,000
Control direction	D0008	—	—
Acceleration/deceleration control	D0008	Acceleration/deceleration later	After=2
Next step number	D0009	2	Step 2

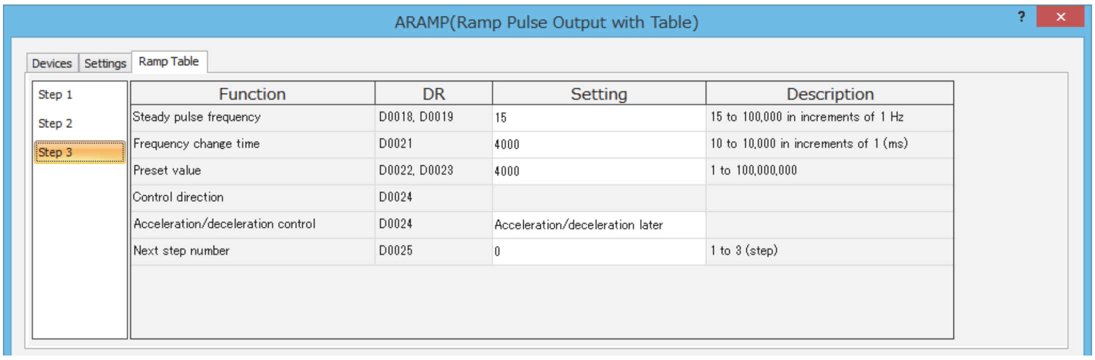
Step 2 settings



Function	DR	Setting	Description
Steady pulse frequency	D0010, D0011	1000	15 to 100,000 in increments of 1 Hz
Frequency change time	D0013	2000	10 to 10,000 in increments of 1 (ms)
Preset value	D0014, D0015	4000	1 to 100,000,000
Control direction	D0016		
Acceleration/deceleration control	D0016	Acceleration/deceleration later	
Next step number	D0017	3	1 to 3 (step)

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0010, D0011	1000	1,000 Hz
Frequency change time	D0013	2000	2,000 ms
Preset value	D0014, D0015	4000	Preset value=4,000
Control direction	D0016	—	—
Acceleration/deceleration control	D0016	Acceleration/deceleration later	After=2
Next step number	D0017	3	Step 3

Step 3 settings

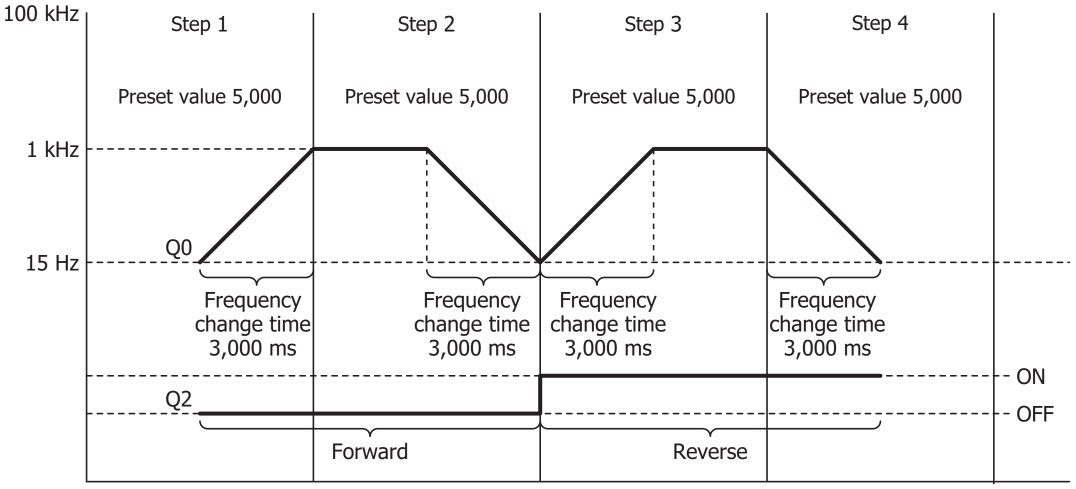


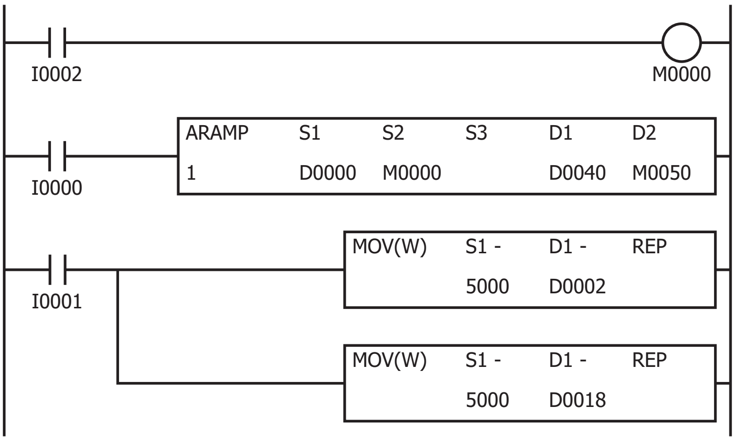
Function	DR	Setting	Description
Steady pulse frequency	D0018, D0019	15	15 to 100,000 in increments of 1 Hz
Frequency change time	D0021	4000	10 to 10,000 in increments of 1 (ms)
Preset value	D0022, D0023	4000	1 to 100,000,000
Control direction	D0024		
Acceleration/deceleration control	D0024	Acceleration/deceleration later	
Next step number	D0025	0	1 to 3 (step)

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0018, D0019	15	15 Hz
Frequency change time	D0021	4000	4,000 ms
Preset value	D0022, D0023	4000	Preset value=4,000
Control direction	D0024	—	—
Acceleration/deceleration control	D0024	Acceleration/deceleration later	After=2
Next step number	D0025	0	0=End output

When outputting pulses as shown in the diagram below with the frequency change function (single-pulse output reversible control enabled) using the following settings

Pulses are output from Q0 and the reversible control signal is output from Q2.



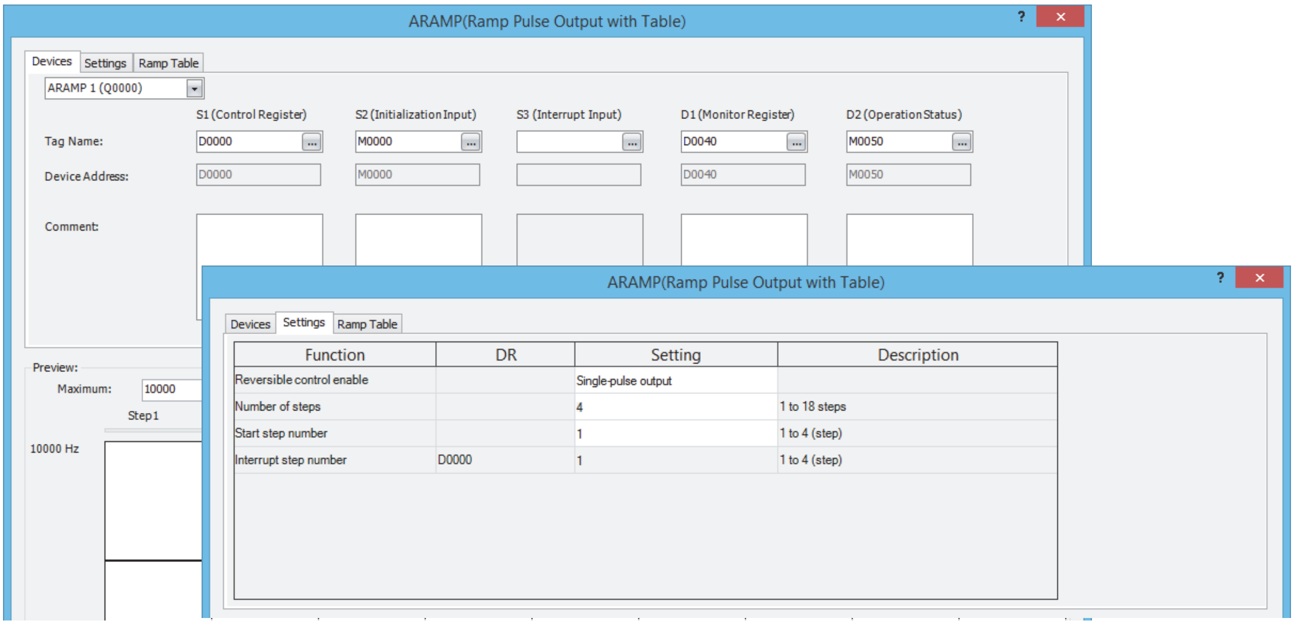


When I2 turns on, the initialization input (M0000) is turned on and the ARAMP instruction control register values are initialized

When the ARAMP instruction input (I0) turns on, pulse output starts

When the MOV instruction input (I1) turns on, the steady pulse frequencies for step 2 and step 4 (D0002, D0003 and D0018, D0019) are changed to 5 kHz.

Basic settings



ARAMP(Ramp Pulse Output with Table)

Devices | Settings | Ramp Table

ARAMP 1 (Q0000)

	S1 (Control Register)	S2 (Initialization Input)	S3 (Interrupt Input)	D1 (Monitor Register)	D2 (Operation Status)
Tag Name:	D0000	M0000		D0040	M0050
Device Address:	D0000	M0000		D0040	M0050
Comment:					

Preview:
Maximum: 10000
Step1
10000 Hz

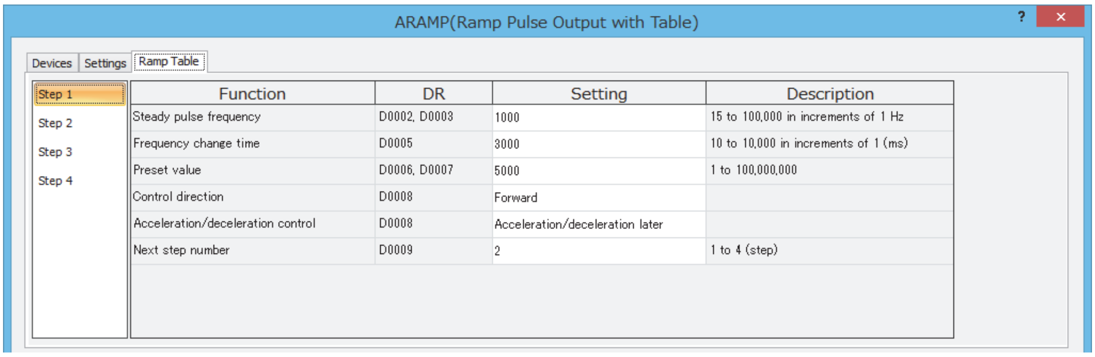
ARAMP(Ramp Pulse Output with Table)

Devices | Settings | Ramp Table

Function	DR	Setting	Description
Reversible control enable		Single-pulse output	
Number of steps		4	1 to 18 steps
Start step number		1	1 to 4 (step)
Interrupt step number	D0000	1	1 to 4 (step)

Function	Device Address	Setting Value	Details
Reversible control enable	—	Single-pulse output	—
Number of steps	—	4	—
Start step number	—	1	Step 1
Interrupt step number	D0000	1	—

Step 1 settings



ARAMP(Ramp Pulse Output with Table)

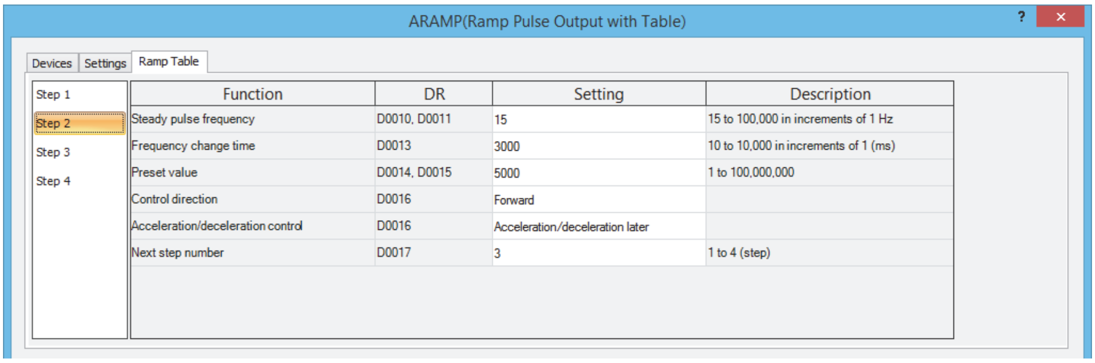
Devices | Settings | Ramp Table

Step 1
Step 2
Step 3
Step 4

Function	DR	Setting	Description
Steady pulse frequency	D0002, D0003	1000	15 to 100,000 in increments of 1 Hz
Frequency change time	D0005	3000	10 to 10,000 in increments of 1 (ms)
Preset value	D0006, D0007	5000	1 to 100,000,000
Control direction	D0008	Forward	
Acceleration/deceleration control	D0008	Acceleration/deceleration later	
Next step number	D0009	2	1 to 4 (step)

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0002, D0003	1000	1,000 Hz
Frequency change time	D0005	3000	3,000 ms
Preset value	D0006, D0007	5000	Preset value=5,000
Control direction	D0008	Forward	Forward
Acceleration/deceleration control	D0008	Acceleration/deceleration later	After=2
Next step number	D0009	2	Step 2

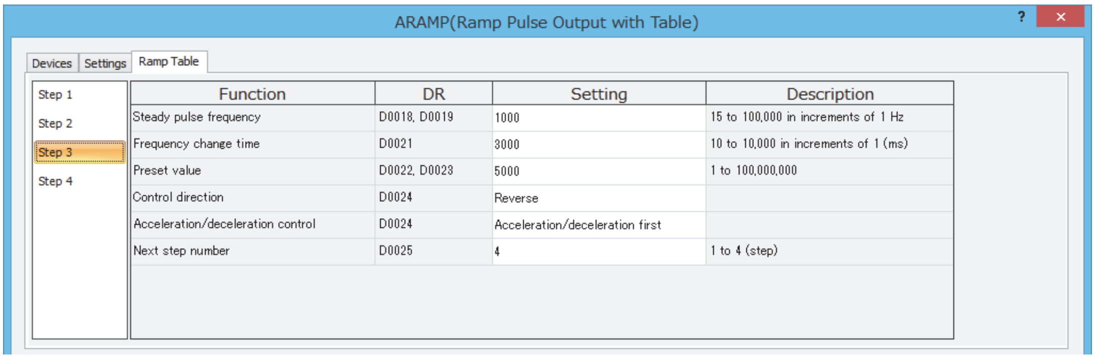
Step 2 settings



Function	DR	Setting	Description
Steady pulse frequency	D0010, D0011	15	15 to 100,000 in increments of 1 Hz
Frequency change time	D0013	3000	10 to 10,000 in increments of 1 (ms)
Preset value	D0014, D0015	5000	1 to 100,000,000
Control direction	D0016	Forward	
Acceleration/deceleration control	D0016	Acceleration/deceleration later	
Next step number	D0017	3	1 to 4 (step)

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0010, D0011	15	15 Hz
Frequency change time	D0013	3000	3,000 ms
Preset value	D0014, D0015	5000	Preset value=5,000
Control direction	D0016	Forward	Forward
Acceleration/deceleration control	D0016	Acceleration/deceleration later	After=2
Next step number	D0017	3	Step 3

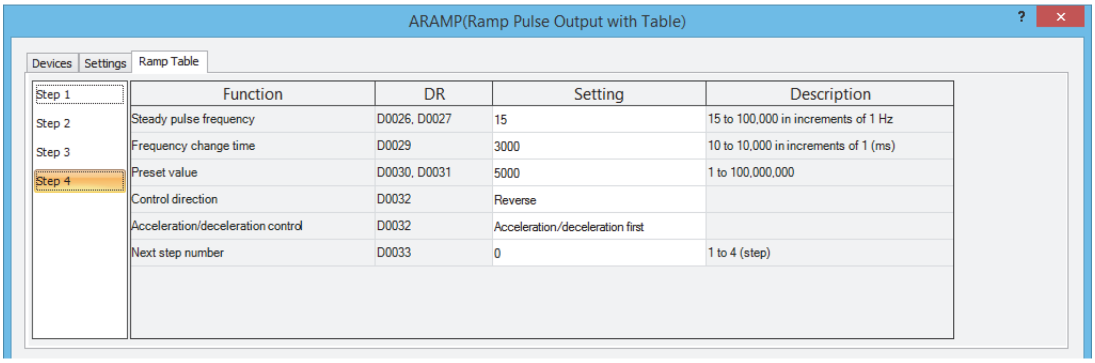
Step 3 settings



Function	DR	Setting	Description
Steady pulse frequency	D0018, D0019	1000	15 to 100,000 in increments of 1 Hz
Frequency change time	D0021	3000	10 to 10,000 in increments of 1 (ms)
Preset value	D0022, D0023	5000	1 to 100,000,000
Control direction	D0024	Reverse	
Acceleration/deceleration control	D0024	Acceleration/deceleration first	
Next step number	D0025	4	1 to 4 (step)

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0018, D0019	1000	1,000 Hz
Frequency change time	D0021	3000	3,000 ms
Preset value	D0022, D0023	5000	Preset value=5,000
Control direction	D0024	Reverse	Reverse
Acceleration/deceleration control	D0024	Acceleration/deceleration first	Before=0
Next step number	D0025	4	Step 4

Step 4 settings

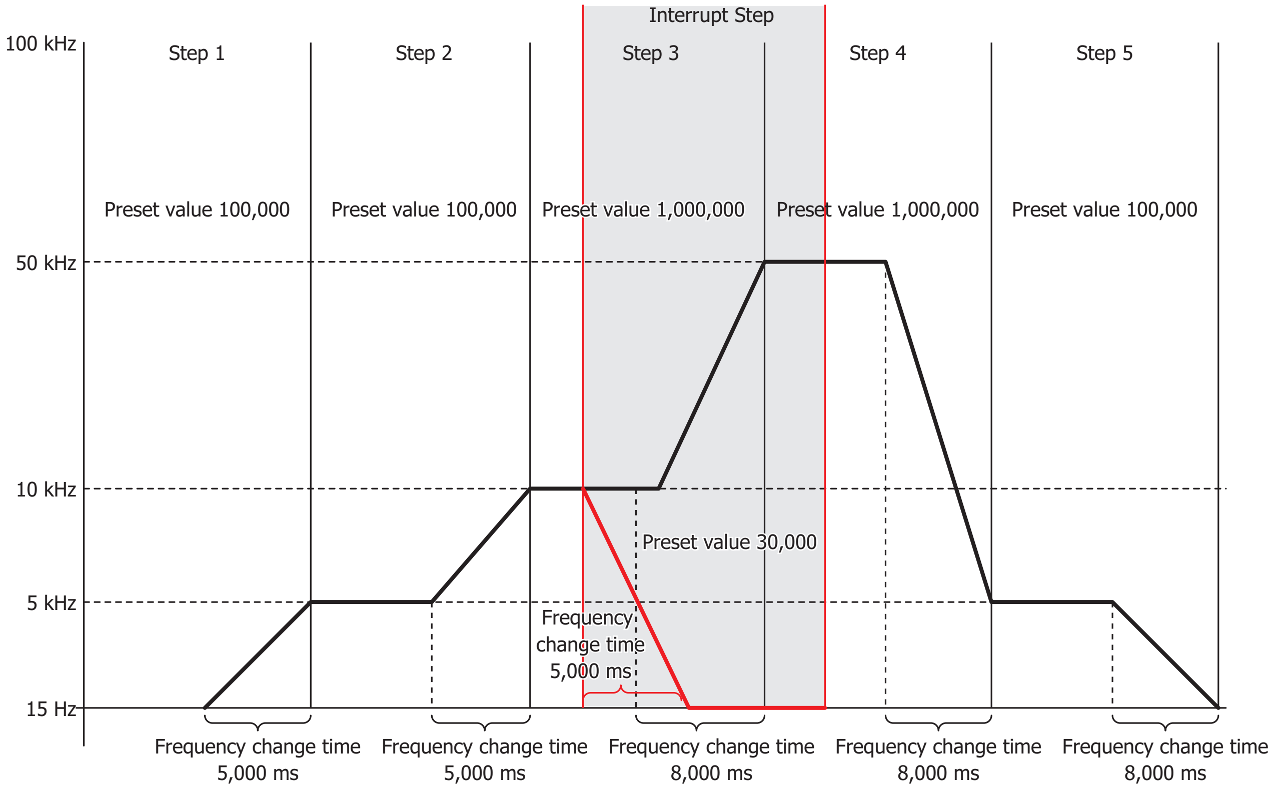


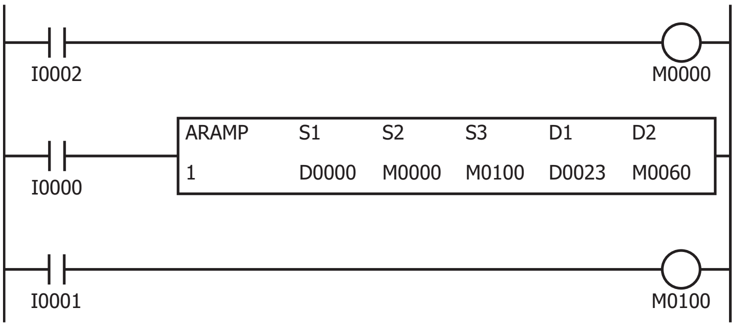
Function	DR	Setting	Description
Steady pulse frequency	D0026, D0027	15	15 to 100,000 in increments of 1 Hz
Frequency change time	D0029	3000	10 to 10,000 in increments of 1 (ms)
Preset value	D0030, D0031	5000	1 to 100,000,000
Control direction	D0032	Reverse	
Acceleration/deceleration control	D0032	Acceleration/deceleration first	
Next step number	D0033	0	1 to 4 (step)

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0026, D0027	15	15 Hz
Frequency change time	D0029	3000	3,000 ms
Preset value	D0030, D0031	5000	Preset value=5,000
Control direction	D0032	Reverse	Reverse
Acceleration/deceleration control	D0032	Acceleration/deceleration first	Before=0
Next step number	D0033	0	Step 0 (end)

When outputting pulses as shown in the diagram below with the frequency change function (single-pulse output reversible control disabled) using the following settings

The pulses are output from Q0.



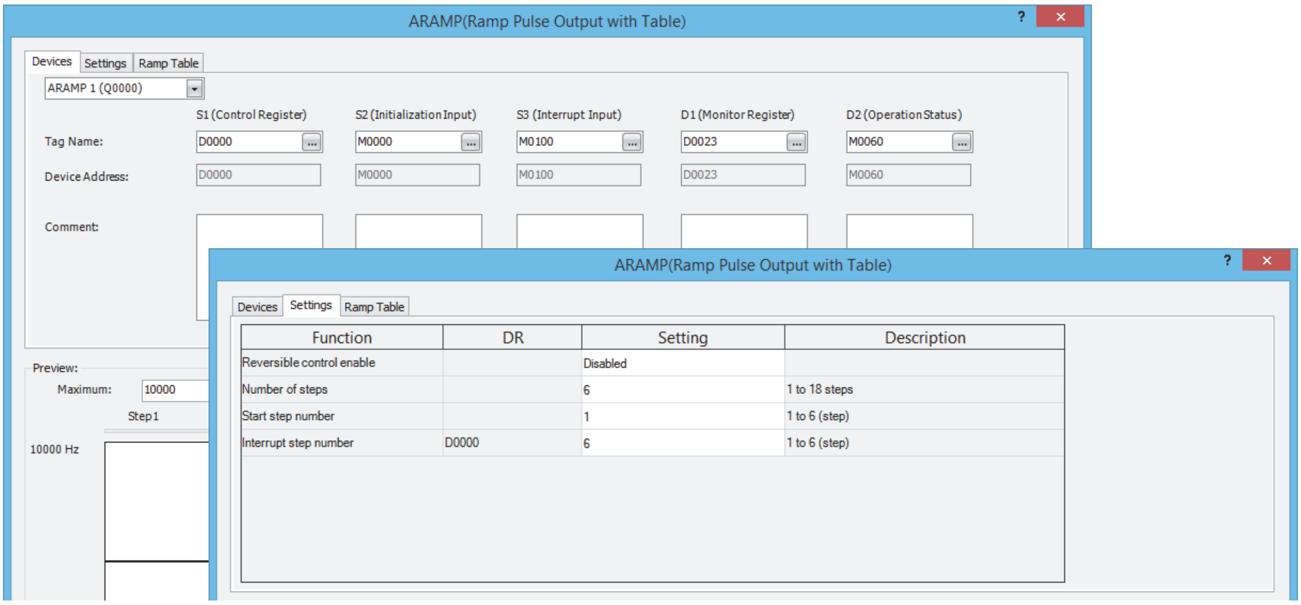


When I2 turns on, the initialization input (M0000) is turned on and the ARAMP instruction control register values are initialized

When the ARAMP instruction input (I0) turns on, pulse output starts

When I1 turns on, the interrupt input (M0100) is turned on and the interrupt step is executed

Basic settings



ARAMP 1 (Q0000)

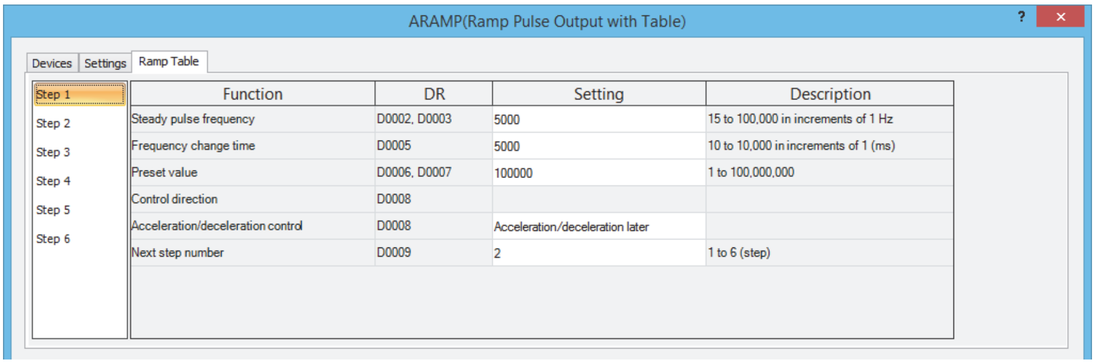
	S1 (Control Register)	S2 (Initialization Input)	S3 (Interrupt Input)	D1 (Monitor Register)	D2 (Operation Status)
Tag Name:	D0000	M0000	M0100	D0023	M0060
Device Address:	D0000	M0000	M0100	D0023	M0060
Comment:					

Maximum: 10000

Function	DR	Setting	Description
Reversible control enable		Disabled	
Number of steps		6	1 to 18 steps
Start step number		1	1 to 6 (step)
Interrupt step number	D0000	6	1 to 6 (step)

Function	Device Address	Setting Value	Details
Reversible control enable	—	Disable	—
Number of steps	—	6	—
Start step number	—	1	Step 1
Interrupt step number	D0000	6	Step 6

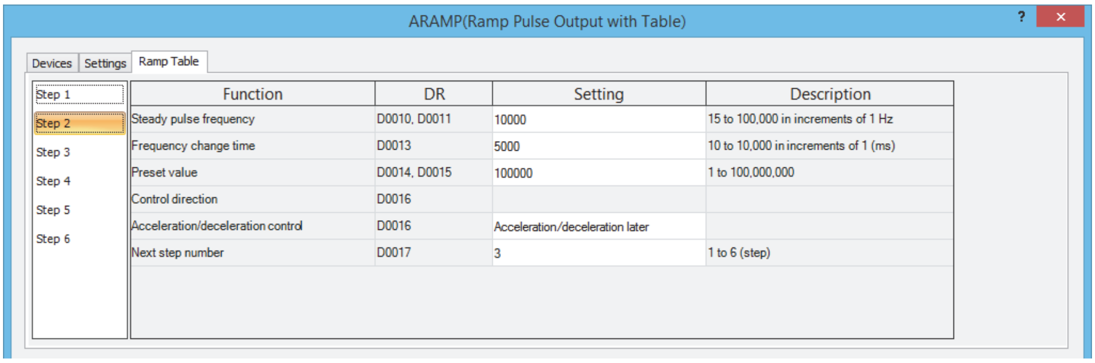
Step 1 settings



Function	DR	Setting	Description
Steady pulse frequency	D0002, D0003	5000	15 to 100,000 in increments of 1 Hz
Frequency change time	D0005	5000	10 to 10,000 in increments of 1 (ms)
Preset value	D0006, D0007	100000	1 to 100,000,000
Control direction	D0008		
Acceleration/deceleration control	D0008	Acceleration/deceleration later	
Next step number	D0009	2	1 to 6 (step)

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0002, D0003	5000	5,000 Hz
Frequency change time	D0005	5000	5,000 ms
Preset value	D0006, D0007	100000	Preset value=100,000
Control direction	D0008	—	—
Acceleration/deceleration control	D0008	Acceleration/deceleration later	After=2
Next step number	D0009	2	Step 2

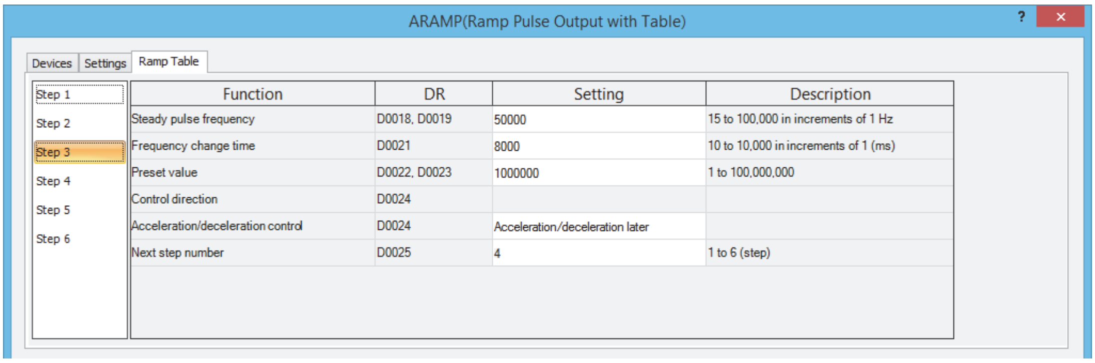
Step 2 settings



Function	DR	Setting	Description
Steady pulse frequency	D0010, D0011	10000	15 to 100,000 in increments of 1 Hz
Frequency change time	D0013	5000	10 to 10,000 in increments of 1 (ms)
Preset value	D0014, D0015	100000	1 to 100,000,000
Control direction	D0016		
Acceleration/deceleration control	D0016	Acceleration/deceleration later	
Next step number	D0017	3	1 to 6 (step)

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0010, D0011	10000	10,000 Hz
Frequency change time	D0013	5000	5,000 ms
Preset value	D0014, D0015	100000	Preset value=100,000
Control direction	D0016	—	—
Acceleration/deceleration control	D0016	Acceleration/deceleration later	After=2
Next step number	D0017	3	Step 3

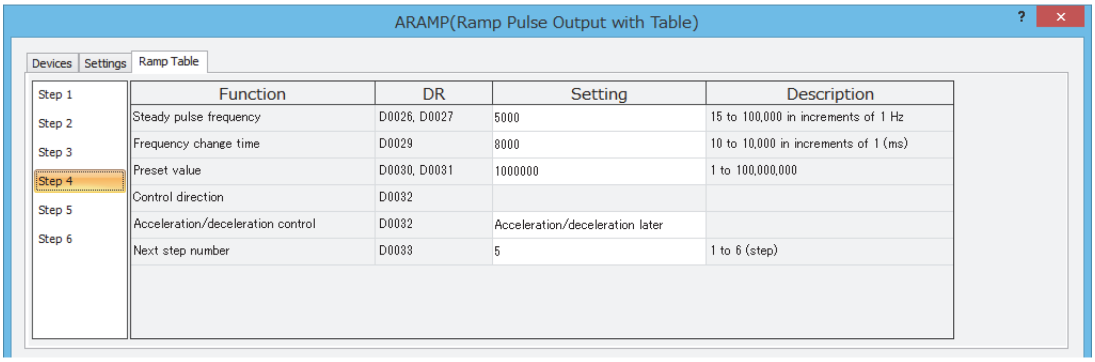
Step 3 settings



Function	DR	Setting	Description
Steady pulse frequency	D0018, D0019	50000	15 to 100,000 in increments of 1 Hz
Frequency change time	D0021	8000	10 to 10,000 in increments of 1 (ms)
Preset value	D0022, D0023	1000000	1 to 100,000,000
Control direction	D0024		
Acceleration/deceleration control	D0024	Acceleration/deceleration later	
Next step number	D0025	4	1 to 6 (step)

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0018, D0019	50000	50,000 Hz
Frequency change time	D0021	8000	8,000 ms
Preset value	D0022, D0023	1000000	Preset value=1,000,000
Control direction	D0024	—	—
Acceleration/deceleration control	D0024	Acceleration/deceleration later	After=2
Next step number	D0025	4	Step 4

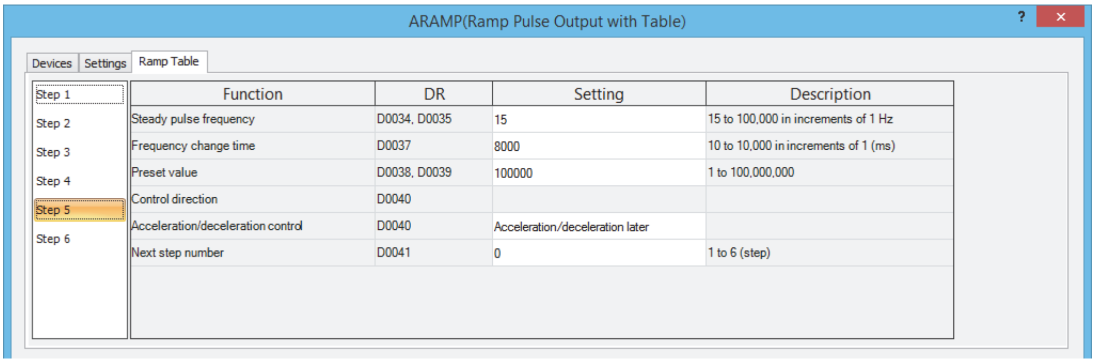
Step 4 settings



Function	DR	Setting	Description
Steady pulse frequency	D0026, D0027	5000	15 to 100,000 in increments of 1 Hz
Frequency change time	D0029	8000	10 to 10,000 in increments of 1 (ms)
Preset value	D0030, D0031	1000000	1 to 100,000,000
Control direction	D0032		
Acceleration/deceleration control	D0032	Acceleration/deceleration later	
Next step number	D0033	5	1 to 6 (step)

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0026, D0027	5000	5,000 Hz
Frequency change time	D0029	8000	8,000 ms
Preset value	D0030, D0031	1000000	Preset value=1,000,000
Control direction	D0032	—	—
Acceleration/deceleration control	D0032	Acceleration/deceleration later	After=2
Next step number	D0033	5	Step 5

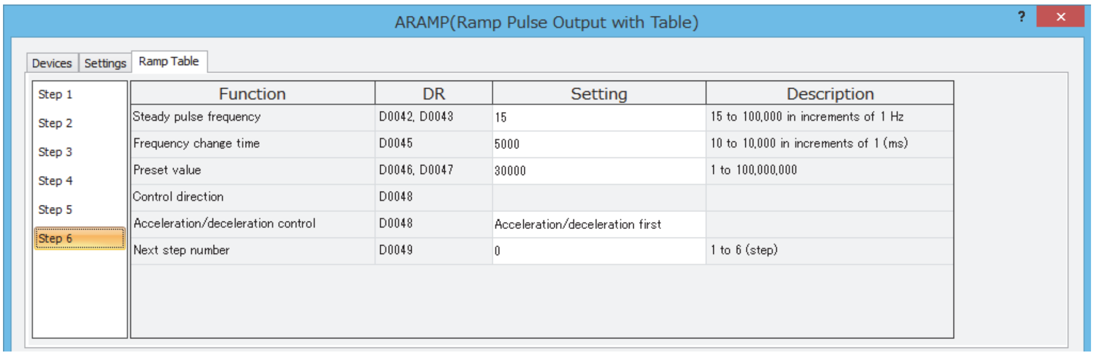
Step 5 settings



Function	DR	Setting	Description
Steady pulse frequency	D0034, D0035	15	15 to 100,000 in increments of 1 Hz
Frequency change time	D0037	8000	10 to 10,000 in increments of 1 (ms)
Preset value	D0038, D0039	100000	1 to 100,000,000
Control direction	D0040		
Acceleration/deceleration control	D0040	Acceleration/deceleration later	
Next step number	D0041	0	1 to 6 (step)

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0034, D00035	15	15 Hz
Frequency change time	D0037	8000	8,000 ms
Preset value	D0038, D0039	100000	Preset value=100,000
Control direction	D0040	—	—
Acceleration/deceleration control	D0040	Acceleration/deceleration later	After=2
Next step number	D0041	0	0=End output

Step 6 settings

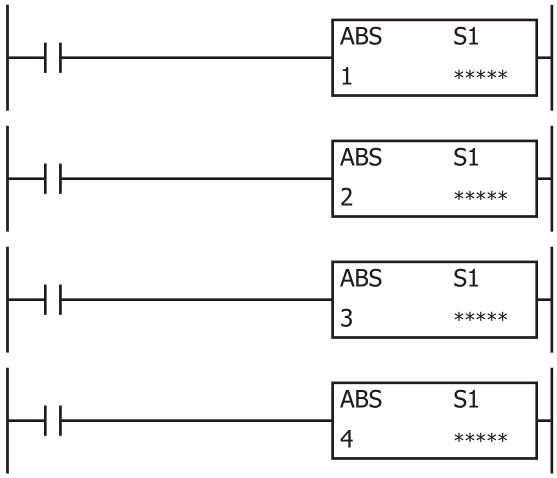


Function	DR	Setting	Description
Steady pulse frequency	D0042, D0043	15	15 to 100,000 in increments of 1 Hz
Frequency change time	D0045	5000	10 to 10,000 in increments of 1 (ms)
Preset value	D0046, D0047	30000	1 to 100,000,000
Control direction	D0048		
Acceleration/deceleration control	D0048	Acceleration/deceleration first	
Next step number	D0049	0	1 to 6 (step)

Function	Device Address	Setting Value	Details
Steady pulse frequency	D0042, D0043	15	15 Hz
Frequency change time	D0045	5000	5000 ms
Preset value	D0046, D0047	30000	Preset value=30,000
Control direction	D0048	—	—
Acceleration/deceleration control	D0048	Acceleration/deceleration first	Before=0
Next step number	D0049	0	0=End output

ABS (Set Absolute Position)

The ABS instruction initializes the absolute position counter for pulse output.



Operation

When the input is on, the values of special data registers D8240 to D8247 (absolute position counters) are updated to the initial values specified by S1.

When updated, 1 is stored in the absolute position counter initialized flag in special data register D8239 (absolute position control status).

Notes:

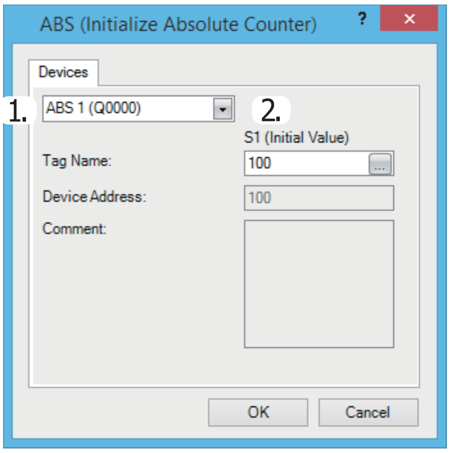
- Do not execute the ABS instruction while the RAMP, ARAMP, ZRN, or JOG instructions are being executed. The absolute position counters may be initialized with an unintended value.
- Program the user program so that the ABS instruction will be executed after user program execution starts and before specifying the absolute position and executing the RAMP (Trapezoidal Control) instruction.
- When multiple ABS instructions are executed in the same scan, the content of the last executed ABS instruction is reflected in the absolute value counters.
- The ABS instruction cannot be used in an interrupt program. If used in an interrupt program, a user program execution error will occur. Error code 18 will be stored in D8006 and instruction execution will be canceled.
- If a pulse output instruction is executed with the relay output type, a user program execution error will occur.
  Error code 19 will be stored in D8006 and instruction execution will be canceled.
- For user program execution errors, see "User Program Execution Errors" on page 3-10.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Initial value	—	—	—	—	—	—	X	—	X	—

Settings

■ Devices tab



ABS (Initialize Absolute Counter)

Devices

1. ABS 1 (Q0000)

2. S1 (Initial Value)

Tag Name: 100

Device Address: 100

Comment:

1. Select instruction

This item selects which ABS instruction to use ("ABS1", "ABS2", "ABS3", or "ABS4").

The absolute position counter to be initialized differs with the instruction.

Command	Absolute Position Counter	Special Data Register*1*2*3	Description	Instructions that Update the Absolute Position Counter*4
ABS1	Absolute Position Counter 1	High word D8240	-2147483648 to 2147483647	ABS1/RAMP1*5/ARAMP1/ZRN1/JOG1
		Low word D8241		
ABS2	Absolute Position Counter 2	High word D8242		ABS2/RAMP2*5/ARAMP2/ZRN2/JOG2
		Low word D8243		
ABS3	Absolute Position Counter 3	High word D8244		ABS3/RAMP3*5/ARAMP3/ZRN3/JOG3
		Low word D8245		
ABS4	Absolute Position Counter 4	High word D8246		ABS4/RAMP4*5/ARAMP4/ZRN4/JOG4
		Low word D8247		

*1 The upper and lower data registers change according to the 32-bit data storage method specified. For details, see "32-bit Data Storage" on page 3-9.

*2 The values of the absolute position counters are retained by the backup battery even when the power is turned off. When a keep data error occurs, the values are initialized to 0.

*3 The special data registers are read only. To change the values of D8240 to D8247, use the ABS instruction.

*4 When the RAMP, ARAMP, ZRN, and JOG instructions are used with Reversible control disabled, the absolute position counters are not updated.

*5 When specify absolute position mode is enabled in the RAMP instruction, the target position is specified as the absolute position. When using specify absolute position mode in the RAMP instruction, initialize the corresponding absolute position counter in advance with the ABS instruction. For details, see "RAMP (Trapezoidal Control)" on page 18-15.

The initialized status of D8240 to D8247 can be checked with the absolute position counter initialized flags defined in D8239 (absolute position control status).
When the FC6A Series MICROSmart switches from stop to run, 0 (not initialized) is stored in the absolute position counter initialized flags. Then when the absolute position counters are initialized using the ABS instruction, 1 (initialized) is stored.

Device Number	Parameter Name	Bit Position	Description
D8239	Absolute Position Control Status	bit0: Absolute Position Counter 1 Initialized Flag	0: Not initialized 1: Initialized
		bit1: Absolute Position Counter 2 Initialized Flag	
		bit2: Absolute Position Counter 3 Initialized Flag	
		bit3: Absolute Position Counter 4 Initialized Flag	
		bit4 to bit15	Reserved

2. S1 (source 1): Initial Value

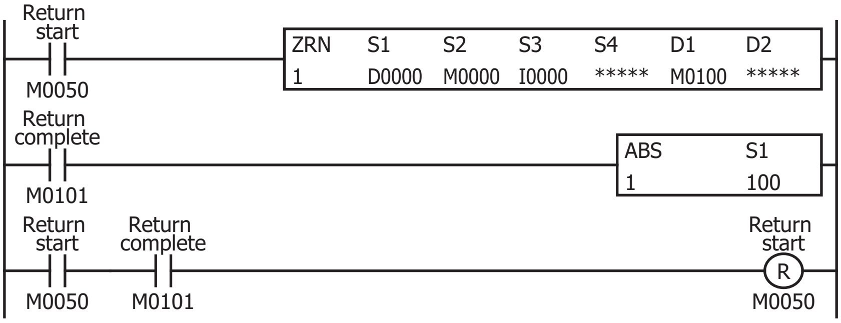
Specifies the first data register of the devices that store the value for initialization or a constant.
Starting from the specified data register, 2 continuous words of data registers are used.

Specify the first data register so that the device range is not exceeded. The data type is L (long).

Storage Destination	Setting	
Starting number+0	Initial value (high word)[*1]	-2147483648 to 2147483647
Starting number+1	Initial value (low word)[*1]	

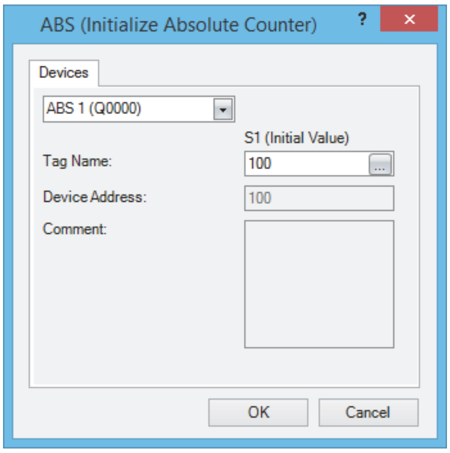
[*1] The upper and lower data registers change according to the 32-bit data storage method specified. For details, see "32-bit Data Storage" on page 3-9.

Example:



After the power is turned on, the ZRN1 instruction is started and a zero return is performed.
After the zero return operation completes, absolute position counter 1 is initialized to 100 with the ABS1 instruction.

Setting


ABS (Initialize Absolute Counter)

Devices

ABS 1 (Q0000)

S1 (Initial Value)

Tag Name: 100

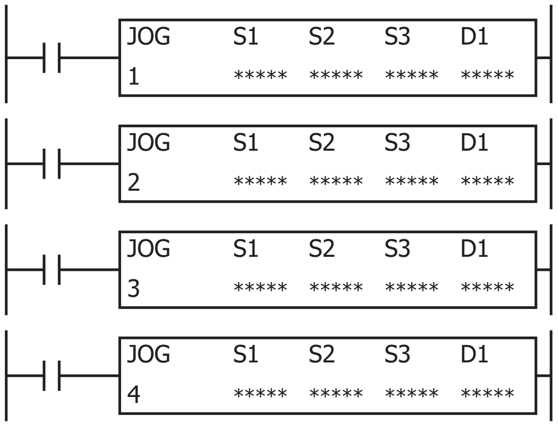
Device Address: 100

Comment:

Function	Preset Value
Select instruction	ABS1
Tag name	100 (constant)

JOG (JOG Operation)

The JOG instruction output pulses with a frequency change.



Operation

When the input is on and the JOG input specified by S3 is turned on, pulses of the initial pulse frequency specified by S1 are output, and then the pulse frequency is increased at a constant rate until it reaches the steady pulse frequency. When the steady pulse frequency is reached, steady pulses are output.

When the JOG input specified by S3 is turned off, the pulses slow at a constant rate until the initial pulse frequency is reached, and when the initial pulse frequency is reached, pulse output stops.

The pulse control information (output on/output complete/error) is stored in the internal relays specified by D1 as the operation status.

When the initialization input specified by S2 is turned on, the initial values configured in the WindLDR **JOG (JOG Control)** dialog box, on the **Settings** tab, are stored in the control registers.

Notes:

- If a pulse output instruction is simultaneously executed with the same output, a user program execution error will occur.
  Error code 48 will be stored in D8006 and instructions that are executed later will be canceled.
- The JOG instruction cannot be used in an interrupt program. If used in an interrupt program, a user program execution error will occur. Error code 18 will be stored in D8006 and instruction execution will be canceled.
- If a pulse output instruction is executed with the relay output type, a user program execution error will occur.
  Error code 19 will be stored in D8006 and instruction execution will be canceled.
- For user program execution errors, see "User Program Execution Errors" on page 3-10.

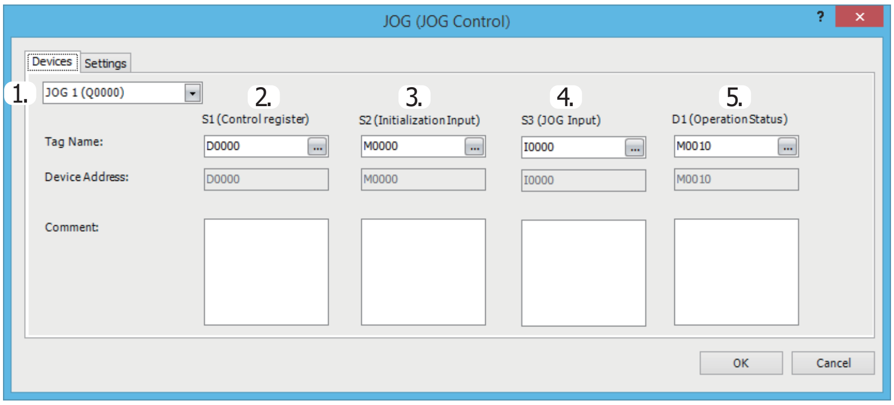
Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Control register	—	—	—	—	—	—	▲	—	—	—
S2 (Source 2)	Initialization input	X	—	X	—	—	—	—	—	—	—
S3 (Source 3)	JOG input	X	—	X	—	—	—	—	—	—	—
D1 (Destination 1)	Status code	—	—	▲	—	—	—	—	—	—	—

▲ Special data registers cannot be designated as S1. Special internal relays cannot be designated as D1.

Settings

■ Devices tab



JOG (JOG Control)

Devices | Settings

1. JOG 1 (Q0000)

	2. S1 (Control register)	3. S2 (Initialization Input)	4. S3 (JOG Input)	5. D1 (Operation Status)
Tag Name:	D0000	M0000	I0000	M0010
Device Address:	D0000	M0000	I0000	M0010
Comment:				

1. Select instruction

This item selects which JOG instruction to use ("JOG1", "JOG2", "JOG3", or "JOG4").
The output and reversible control mode that can be selected differ by the instruction and CPU module type.

For limitations due to the combination of instruction, reversible control mode, and the pulse output mode, see "10. Reversible control enable" on page 18-75.

2. S1 (source 1): Control register

S1 specifies the first data register of the data registers to use with the JOG1, JOG2, JOG3, or JOG4 instruction.
Starting from the specified data register, 12 continuous words of data registers are used.
Specify the first data register so that the device range is not exceeded.

Storage Destination	Function	Setting		Reference
		All-in-One CPU Module	CAN J1939 All-in-One CPU Module/ Plus CPU Module	
Starting number+0	Steady pulse frequency (high word)[*1]	JOG1, JOG2: 15 to 100,000 (increments of 1 Hz)	JOG1 to JOG4: 15 to 100,000 (increments of 1 Hz)	"6. Steady pulse frequency" on page 18-75
Starting number+1	Steady pulse frequency (low word)[*1]	JOG3, JOG4: 15 to 5,000 (increments of 1 Hz)		
Starting number+2	Initial pulse frequency (high word)[*1]	JOG1, JOG2: 15 to 100,000 (increments of 1 Hz)	JOG1 to JOG4: 15 to 100,000 (increments of 1 Hz)	"7. Initial pulse frequency" on page 18-75
Starting number+3	Initial pulse frequency (low word)[*1]	JOG3, JOG4: 15 to 5000 (increments of 1 Hz)		
Starting number+4	Acceleration time	10 to 10,000 (ms)		"8. Acceleration time" on page 18-75
Starting number+5	Deceleration time	10 to 10,000 (ms)		"9. Deceleration time" on page 18-75
Starting number+6	Control direction	0: Forward 1: Reverse		"11. Control direction" on page 18-76
Starting number+7	**Reserved**			
Starting number+8	Current value (high word)[*1]	1 to 100,000,000 pulses[*2]		"12. Current value" on page 18-76
Starting number+9	Current value (low word)[*1]			
Starting number+10	Error status	0 to 3		"13. Error status" on page 18-76
Starting number+11	**Reserved**			

[*1] The upper and lower data registers change according to the 32-bit data storage method specified. For details, see "32-bit Data Storage" on page 3-9.

[*2] When starting pulse output, the current value is reset.

3. S2 (source 2): Initialization Input

S2 specifies the initialization input.

When the initialization input is turned on, the initial values configured in the WindLDR **JOG (JOG)** dialog box, on the **Settings** tab, are stored in the control registers. An external input or an internal relay can be specified.

When the initialization input is on, the initial values are stored in the data registers with each scan. (Even when the JOG instruction is not executed (when not on), if the initialization input is turned on, the initial values are stored in the data registers.) To only initialize the values one time, use the initialization input in combination with the SOTU (single output up) instruction or the SOTD (single output down) instruction.

4. S3 (source 3): JOG Input

S3 specifies the jog input.

When the JOG input is turned on, pulses of the initial pulse frequency are output, and then the pulse frequency is increased until it reaches the steady pulse frequency.

When the JOG input is turned off, pulse output stops after the pulse frequency is decreased until the initial pulse frequency is reached.

An external input or an internal relay can be specified.

When the JOG input is turned off while the pulse frequency is increasing, pulse output stops after the pulse frequency is decreased until the initial pulse frequency is reached.

When the JOG input is turned on while the pulse frequency is decreasing, the pulse frequency is increased until the steady pulse frequency is reached.

5. D1 (destination 1): Operation Status

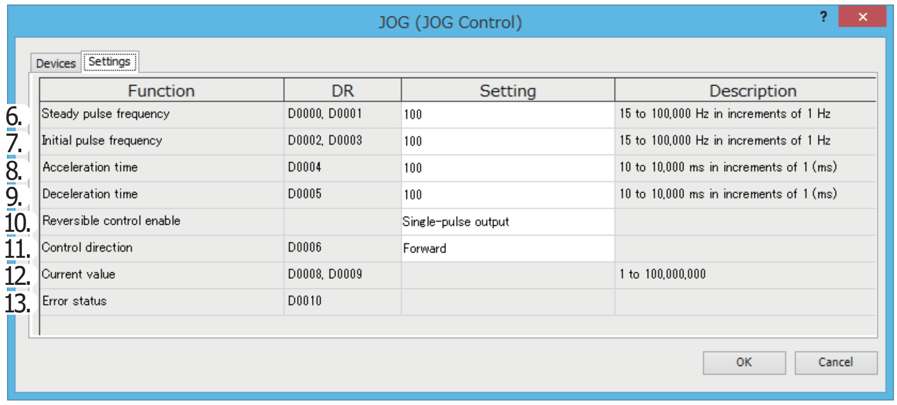
D1 specifies the first internal relay of the internal relays to use with the JOG instruction.

Starting from the specified relay, 4 sequential relays are used.

Specify the first internal relay so that the device range is not exceeded.

Storage Destination	Function	Setting	
Starting number+0	Pulse output ON	0: Pulse output OFF 1: Pulse output ON	This relay turns on during pulse output. This relay turns off when pulse output stops.
Starting number+1	Pulse output complete	0: Pulse output not complete 1: Pulse output complete	This relay turns on when pulse output completes. This relay turns off when pulse output starts.
Starting number+2	Pulse output status	0: Steady pulse output 1: Changing output pulse frequency	This relay turns on when the pulse output status is steady. This relay turns off when the pulse output status is changing frequency.
Starting number+3	**Reserved**		

■ Settings tab



JOG (JOG Control)

	Function	DR	Setting	Description
6.	Steady pulse frequency	D0000, D0001	100	15 to 100,000 Hz in increments of 1 Hz
7.	Initial pulse frequency	D0002, D0003	100	15 to 100,000 Hz in increments of 1 Hz
8.	Acceleration time	D0004	100	10 to 10,000 ms in increments of 1 (ms)
9.	Deceleration time	D0005	100	10 to 10,000 ms in increments of 1 (ms)
10.	Reversible control enable		Single-pulse output	
11.	Control direction	D0006	Forward	
12.	Current value	D0008, D0009		1 to 100,000,000
13.	Error status	D0010		

6. Steady pulse frequency

This setting specifies the steady pulse frequency after the pulse frequency is increased. Set between 15 and 100 kHz in 1 Hz increments. The output frequency error is within ±5%.
For JOG3 and JOG4 with the All-in-One CPU module, set between 15 and 5 kHz in 1 Hz increments.

7. Initial pulse frequency

Specifies the frequency when pulse output starts. Set between 15 and 100 kHz in 1 Hz increments. The output frequency error is within ±5%.
For JOG3 and JOG4 with the All-in-One CPU module, set between 15 and 5 kHz in 1 Hz increments.

8. Acceleration time

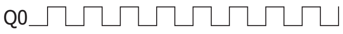
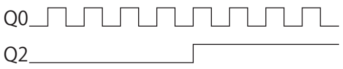
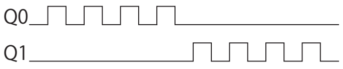
Specifies the time to increase the pulse frequency. Set the value in the range of 10 to 10,000 ms in increments of 1 ms. The first digit of the setting is handled as zero. For example, if 144 is entered, the set value is handled as 140 ms.

9. Deceleration time

This setting specifies the time to decrease the pulse frequency. Set the value in the range of 10 to 10,000 ms in increments of 1 ms. The first digit of the setting is handled as zero. For example, if 144 is entered, the set value is handled as 140 ms.

10. Reversible control enable

This setting enables or disables reversible control and selects the reversible control method from the following reversible control modes. There are two modes for the pulse output mode: single-pulse output mode and dual-pulse output mode. They can be combined with reversible control as follows. (This is an example when JOG1 is used with the All-in-One CPU module.)

Reversible Control Enable	Operation	Pattern
Reversible control disabled	Select this option when using pulse output in a single direction. Pulse A and pulse B can be used independently.	
Reversible control **Single-pulse output**	Pulse A is used as pulse output. Pulse B on/off is used as reversible control.	
Reversible control **Dual-pulse output**	Pulse A is used as forward pulse (CW) output.Pulse B is used as reverse pulse (CCW) output.	

The used outputs differ by the instruction used, the combination of the pulse output mode and reversible control, and the model used.

Command	Operating Condition	Output Used			
		All-in-One CPU Module		CAN J1939 All-in-One CPU Module/Plus CPU Module	
		Pulse Output	Reversible Control Output	Pulse Output	Reversible Control Output
JOG1	Reversible control disabled	Q0	—	Q0	—
	Reversible control **Single-pulse output**	Q0	Q2[*1]	Q0	Q1
	Reversible control **Dual-pulse output**	Q0, Q1[*2]	—	Q0, Q1	—
JOG2	Reversible control disabled	Q1	—	Q2	—
	Reversible control **Single-pulse output**	Q1	Q3[*1]	Q2	Q3
	Reversible control **Dual-pulse output**	—	—	Q2, Q3	—

Command	Operating Condition	Output Used			
		All-in-One CPU Module		CAN J1939 All-in-One CPU Module/Plus CPU Module	
		Pulse Output	Reversible Control Output	Pulse Output	Reversible Control Output
JOG3	Reversible control disabled	Q2	—	Q4	—
	Reversible control **Single-pulse output**	—	—	Q4	Q5
	Reversible control **Dual-pulse output**	—	—	Q4, Q5	—
JOG4	Reversible control disabled	Q3	—	Q6	—
	Reversible control **Single-pulse output**	—	—	Q6	Q7
	Reversible control **Dual-pulse output**	—	—	Q6, Q7	—

*1 When using single-pulse output mode with the All-in-One CPU module, Q2 or Q3 will be used, so an instruction that uses the same output cannot be used.

*2 When using dual-pulse output mode with the All-in-One CPU module, Q1 will be used, so an instruction that uses the same output cannot be used.

11. Control direction

When reversible control is enabled, store 0 in this data register for forward operation and store 1 in this data register for reverse operation.

12. Current value

The number of pulses outputs is stored in the data registers.

The current value is updated when the JOG instruction is executed at each scan.

When starting pulse output, the current value is reset.

13. Error status

Outputs the error code that corresponds to the content of an error when there is an error in the settings. If a configuration error occurs when a step starts running, a user program execution error will occur, and error code 20 is stored in D8006.

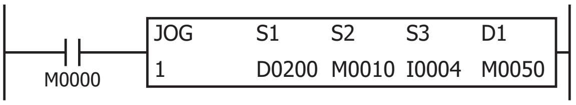
Error Code List

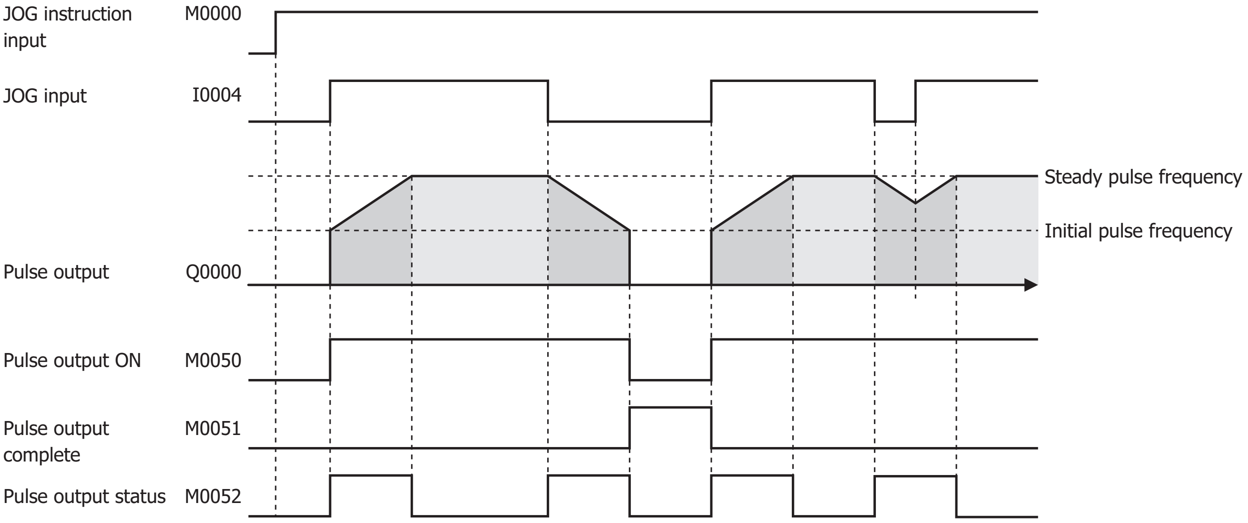
Error Code	Status	Description		
0	Normal	—		
2	Initial pulse frequency designation error	All-in-One CPU module	JOG1, JOG2	The initial pulse frequency was not set between 15 and 100,000.
			JOG3, JOG4	The initial pulse frequency was not set between 15 and 5,000.
		CAN J1939 All-in-One CPU module/Plus CPU module	JOG1 to JOG4	The initial pulse frequency was not set between 15 and 100,000.
4	Steady pulse frequency designation error	All-in-One CPU module	JOG1, JOG2	The steady pulse frequency was not set between 15 and 100,000.
			JOG3, JOG4	The steady pulse frequency was not set between 15 and 5,000.
		CAN J1939 All-in-One CPU module/Plus CPU module	JOG1 to JOG4	The steady pulse frequency was not set between 15 and 100,000.
5	Frequency change time designation error	The acceleration time was not set between 10 and 10,000.		
7	Control direction designation error	The control direction was not set to 0 or 1.		
9	Pulse frequency designation error	The initial pulse frequency was set to the same frequency as the steady pulse frequency or it was set to a value larger than the steady pulse frequency.*1		

*1 Set the initial pulse frequency so that it is lower than the steady pulse frequency.

JOG1 Instruction (Reversible Control Disabled) Timing Chart

With the All-in-One CPU module, JOG1 instruction, S1 is specified as D0200, D1 is specified as internal relay M0050

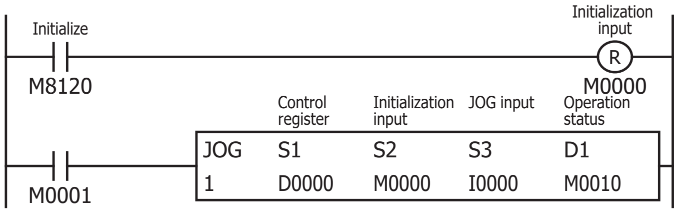




- When the JOG1 input is on, the I4 monitor starts. When I4 is turned on, pulses are output according to the settings configured by the control registers.
- M0050 turns on during pulse output. M0052 turns on while the pulse frequency is increasing, and M0052 turns off while decreasing.
- Pulses are output so that the frequency reaches the steady pulse frequency from the initial pulse frequency in the acceleration time. The frequency increases or decreases every 10 ms.
- When I4 turns off, the frequency starts to decrease, and then when the frequency reaches the initial pulse frequency, pulse output stops. At this time, M0050 turns off and M0051 turns on.
- If I4 turns off while the frequency is increasing, the frequency starts to decrease, and then when the frequency reaches the initial pulse frequency, pulse output stops.
- If I4 turns on while the frequency is decreasing, it once again starts to increase, and pulses are output so the frequency reaches the steady pulse frequency.
- Even if the contents of the control registers are changed during pulse output, the change is not reflected in the operation of pulse output. The changed content will be reflected when pulses are next output after pulse output is first stopped and M0050 is turned off.

Example:

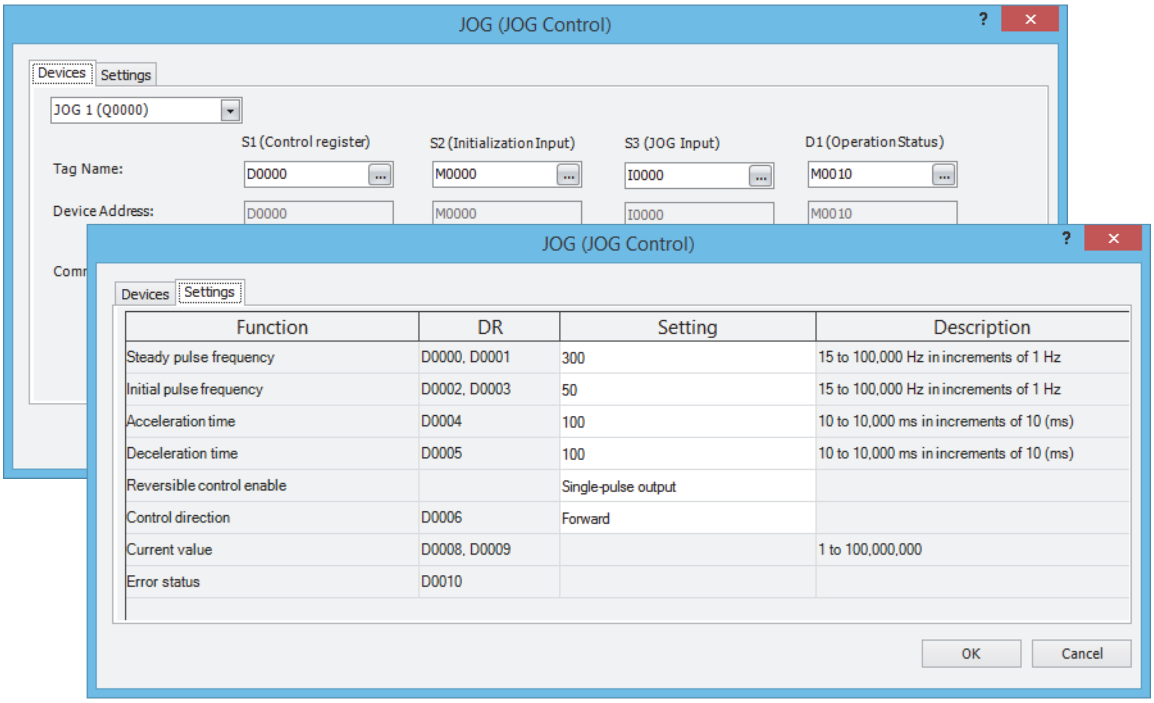
When performing a JOG operation by outputting pulses with a frequency change (reversible control disabled) from Q0



Turn on initialization input (M0000)

When the JOG instruction input (M0001) is on and the JOG input (I0) is turned on, pulse output starts.

Setting



JOG 1 (Q0000)

	S1 (Control register)	S2 (Initialization Input)	S3 (JOG Input)	D1 (Operation Status)
Tag Name:	D0000	M0000	I0000	M0010
Device Address:	D0000	M0000	I0000	M0010

Function	DR	Setting	Description
Steady pulse frequency	D0000, D0001	300	15 to 100,000 Hz in increments of 1 Hz
Initial pulse frequency	D0002, D0003	50	15 to 100,000 Hz in increments of 1 Hz
Acceleration time	D0004	100	10 to 10,000 ms in increments of 10 (ms)
Deceleration time	D0005	100	10 to 10,000 ms in increments of 10 (ms)
Reversible control enable		Single-pulse output	
Control direction	D0006	Forward	
Current value	D0008, D0009		1 to 100,000,000
Error status	D0010		

Pulse Monitor

Function Description

This function monitors the pulses that the FC6A Series MICROSmart outputs from the output ports with the pulse output instructions and graphs the trajectory and waveform of each axis.

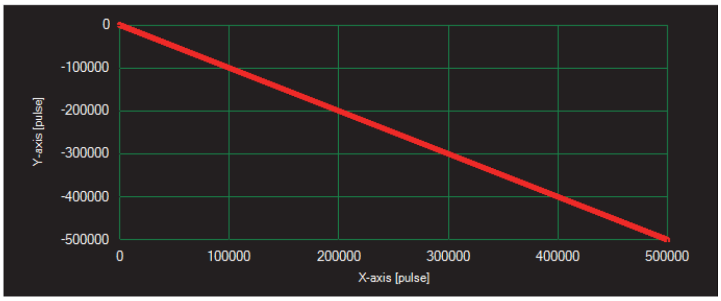
The Pulse Monitor can be used only with the Plus CPU module.

Notes:

- Data is accumulated in the PLC in order for WindLDR to acquire data at a regular interval. If the speed at which the PLC accumulates data is faster than the speed at which WindLDR reads that data and the data in internal memory is overwritten, the Pulse Monitor will be suspended.
- Depending on the sampling cycle of the pulse monitor or the communications environment, the graph may be drawn with a delay in regard to PLC operations.

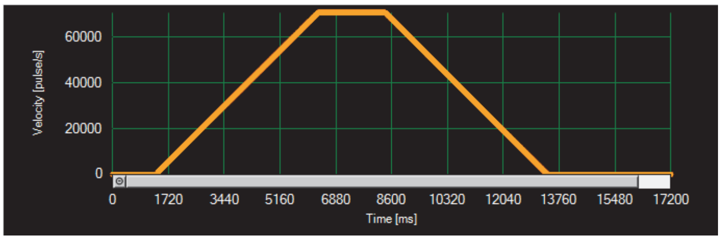
Trajectory

The trajectory on a plane is displayed based on the number of pulses that corresponds to the specified output port.
The trajectory graph is displayed only when the PLC is operating in absolute position mode.



Waveform

This graph displays the pulse output speed of the specified output port.
The pulse output time (the relative time from the start of the pulse monitor) is plotted on the horizontal axis, and the number of pulses that corresponds to the output time is plotted on the vertical axis.

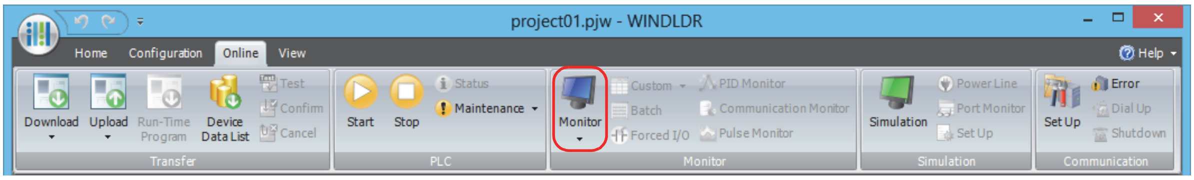


Starting the Pulse Monitor

Operation Procedure

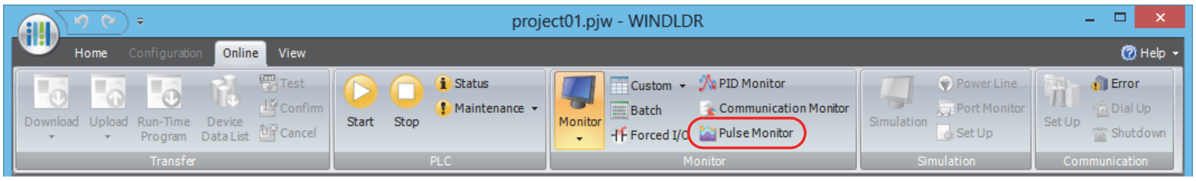
1. On the **Online** tab, in the **Monitor** group, click **Monitor**.

The monitor will start.

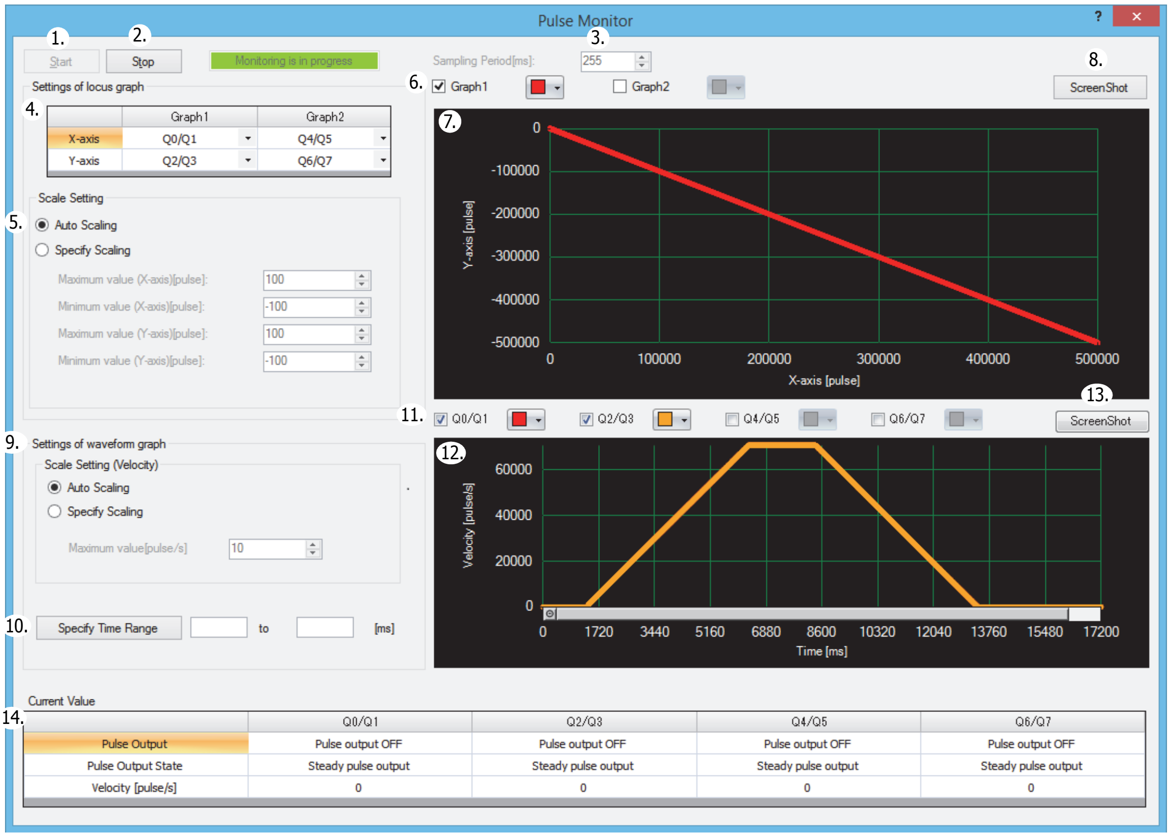


2. On the **Online** tab, in the **Monitor** group, click **Pulse Monitor**.

The **Pulse Monitor** dialog box is displayed.



Pulse Monitor Dialog Box



Sampling Period[ms]: 255

☑ Graph1 🟥　□ Graph2

Settings of locus graph

	Graph1	Graph2
X-axis	Q0/Q1	Q4/Q5
Y-axis	Q2/Q3	Q6/Q7

Scale Setting

● Auto Scaling

○ Specify Scaling

Maximum value (X-axis)[pulse]: 100

Minimum value (X-axis)[pulse]: -100

Maximum value (Y-axis)[pulse]: 100

Minimum value (Y-axis)[pulse]: -100

Settings of waveform graph

Scale Setting (Velocity)

● Auto Scaling

○ Specify Scaling

Maximum value[pulse/s]: 10

☑ Q0/Q1 🟥　☑ Q2/Q3 🟧　□ Q4/Q5　□ Q6/Q7

Current Value

	Q0/Q1	Q2/Q3	Q4/Q5	Q6/Q7
Pulse Output	Pulse output OFF	Pulse output OFF	Pulse output OFF	Pulse output OFF
Pulse Output State	Steady pulse output	Steady pulse output	Steady pulse output	Steady pulse output
Velocity [pulse/s]	0	0	0	0

1. Start

Click this button to start the Pulse Monitor. WindLDR will acquire data from the PLC at a regular interval.

2. Stop

Click this button to stop the Pulse Monitor.

3. Sampling Period

Specifies the sampling frequency. The sampling frequency is specified between 5 and 255 ms in increments of 1 ms.

4. Output Port Setting

Select the output port for the X-axis and Y-axis to display on the trajectory graph from the following. The selected ports can be changed after the Pulse Monitor is started.

X-axis: **Q0/Q1**, **Q2/Q3**, **Q4/Q5**, **Q6/Q7**

Y-axis: **Q0/Q1**, **Q2/Q3**, **Q4/Q5**, **Q6/Q7**

5. Scale Setting

Select the scale of the X-axis and Y-axis on the trajectory graph from **Auto Scaling** or **Specify Scaling**.

Auto Scaling: The maximum value of the graph is automatically updated according to the maximum value of the data.

Specify Scaling: Manually enter the maximum and minimum values of each axis. The graph will not be displayed if the value is larger than the maximum value or smaller than the minimum value.

6. Graph Setting

Select the check boxes for the graphs to display on the trajectory graph, and select the colors for the graphs.

7. Trajectory Graph

The trajectory graph is drawn in this area. Cursors are displayed on the graph, and the graph can be displayed in conjunction with the waveform graph.

8. ScreenShot

This button saves the trajectory graph as an image.

When you click **ScreenShot**, the **Select File** dialog box will be displayed, and you can save the graph as a bitmap file.

9. Scale Setting (Velocity)

Select the scale of speed (Y-axis) on the waveform graph from **Auto Scaling** or **Specify Scaling**.

Auto Scaling: The maximum value of the graph is automatically updated according to the maximum value of the data.

Specify Scaling: Manually enter the maximum value of speed (Y-axis). The graph will not be displayed if the value is larger than the maximum value.

10. Specify Time Range

Specifies the range of the graph to display as relative times from the start of monitoring.
The graph is enlarged or shrunk according to this setting.

11. Graph Setting

Select the check boxes for the output ports to display on the waveform graph, and select the color for the graph.

12. Waveform Graph

The waveform graph is drawn in this area. A scrollbar is displayed, and the data for a fixed interval is displayed. Move the scrollbar and you can view data in the past.

13. ScreenShot

This button saves the waveform graph as an image.

When you click **ScreenShot**, the **Select File** dialog box will be displayed, and you can save the graph as a bitmap file.

14. Current Value

This area displays the current values of the waveform graph.

Pulse Output: Displays **Pulse output ON** or **Pulse output OFF**.

Pulse Output State: Displays **Steady pulse output** or **Frequency change**.

Velocity: Displays the pulse speed being output.

Positioning Control

This section provides an overview of positioning control as well as WindLDR settings, wiring diagrams, and practical examples.

Positioning Control Overview

Positioning control can be performed with the pulse output instructions.

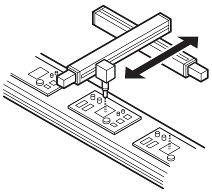
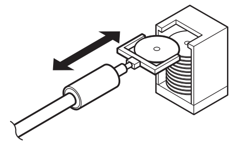
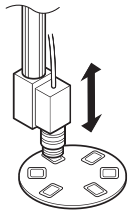
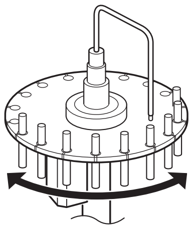
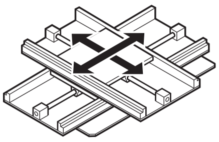
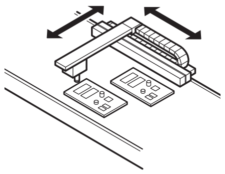
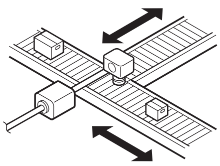
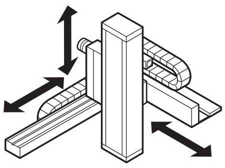
The All-in-One CPU module can simultaneously control up to two axes.

The Plus CPU module and CAN J1939 All-in-One CPU module can simultaneously control up to four axes.

Pulse Output Instruction Examples

Instruction	Operation Description
RAMP	This positioning instruction performs trapezoidal control at a constant frequency change rate.
RAMPL	This positioning instruction simultaneously controls two axes towards a target position.
ZRN	This instruction returns the axis to the origin in order to move from the origin at the next operation.
ARAMP	This instruction enables multi-step speed control, and the speed can be changed using an interrupt input as a trigger.
ABS	This instruction initializes the absolute position counter.

Application Examples

Overview	Description			
Number of axes	One axis			
Application examples	Pick and release 	Storing components 	Focus position adjustment 	Rotary table 
Optimal instructions	RAMP, ARAMP, ZRN			
Overview	**Description**			
Number of axes	Two axes			Three axes
Application examples	XY axis stage 	Imaging inspection 	Sorting conveyed goods 	XYZ axis stage 
Optimal instructions	RAMPL+ABS ZRN		RAMP+ABS ZRN	RAMPL+RAMP+ABS RAMP+ABS ZRN

Pulse Output Instruction Setting Items

Reversible Control Enable

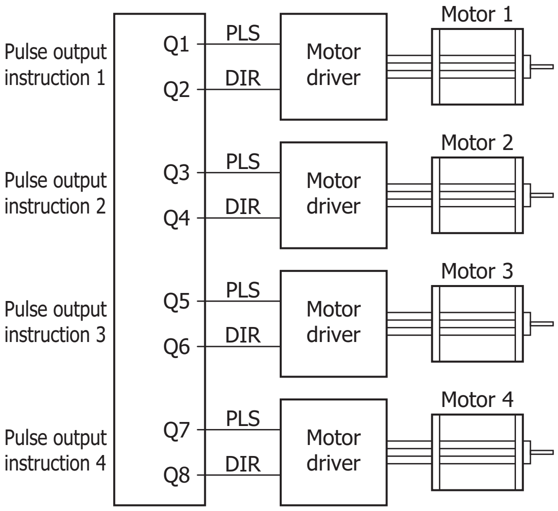
Reversible control enable is the mode that controls the rotation direction of the motor. It can be used with the RAMP/RAMPL/ZRN/ARAMP pulse output instructions. There are three modes for reversible control enable: disabled, single-pulse output mode, and dual-pulse output mode.

See the description of each instruction for more information about operations.

Note: If the setting is different between the PLC and the motor driver, the motor cannot be correctly controlled (for example, the motor will only work in one direction).

Plus CPU Module and CAN J1939 All-in-One CPU Module Example

Single-pulse output

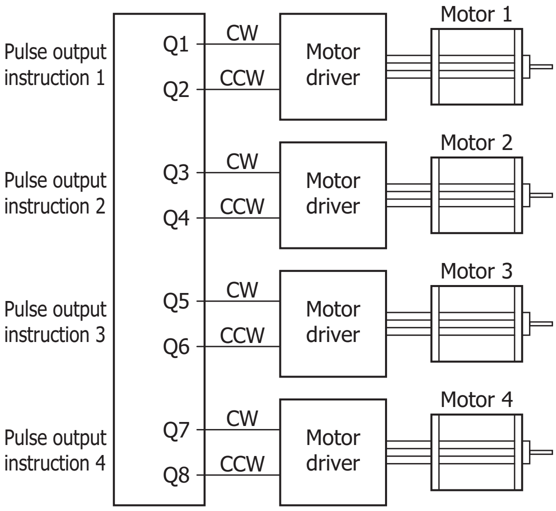


Pulse output instruction 1
Pulse output instruction 2
Pulse output instruction 3
Pulse output instruction 4
Q1
Q2
Q3
Q4
Q5
Q6
Q7
Q8
PLS
DIR
Motor driver
Motor 1
Motor 2
Motor 3
Motor 4

PLS: Pulse output

DIR: Reversible control

Dual-pulse output



Pulse output instruction 1
Pulse output instruction 2
Pulse output instruction 3
Pulse output instruction 4
Q1
Q2
Q3
Q4
Q5
Q6
Q7
Q8
CW
CCW
Motor driver
Motor 1
Motor 2
Motor 3
Motor 4

CW: Forward pulse

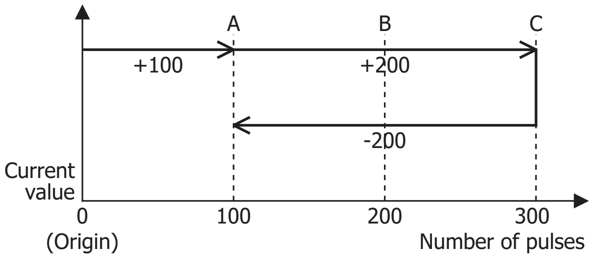
CCW: Reverse pulse

Specify Absolute Position Mode

Specify absolute position mode specifies the pulse position as an absolute position. It can be used with the RAMP/RAMPL pulse output instructions. When specify absolute position mode is enabled, the target position set for the pulse output instruction can be specified as an absolute position. See the description of each instruction for more information about operations.

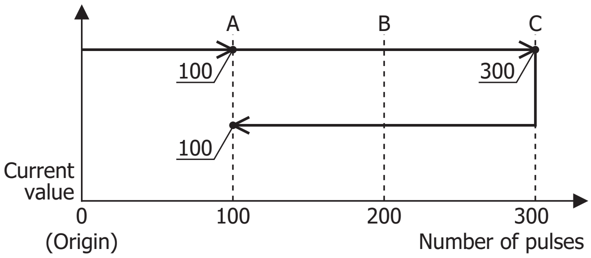
Disabled

Destination	No. of Pulses	Control Direction
Point A → Point B	100	Forward
Point B → Point C	200	Forward
Point C → Point B	200	Reverse



Enabled

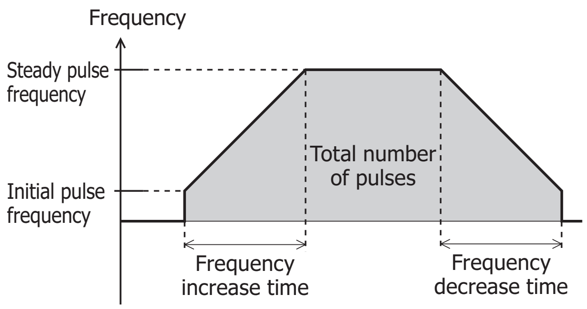
Destination	No. of Pulses (Target Position)
Point A → Point B	100
Point B → Point C	300
Point C → Point B	100



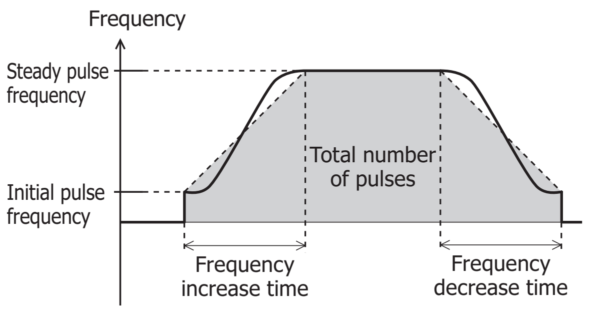
Frequency Change Curve

The frequency change curve is a function that controls the acceleration/deceleration frequency. It can be used with the RAMP pulse output instruction. There are two settings: straight line and S-shaped curve. The straight line setting accelerates and decelerates at a constant rate when starting and stopping, and this absorbs starting and stopping shocks. The S-shaped curve setting lessens shocks and reduces vibration by reducing acceleration when starting and stopping. See the description of each instruction for more information about operations.

Straight line

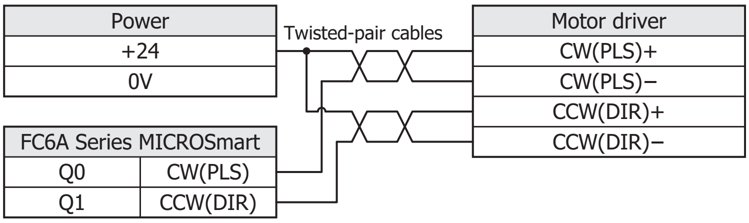


S-shaped curve



Wiring Diagram (Between FC6A Series MICROSmart and Motor Driver)

FC6A Series MICROSmart: Transistor sink output type

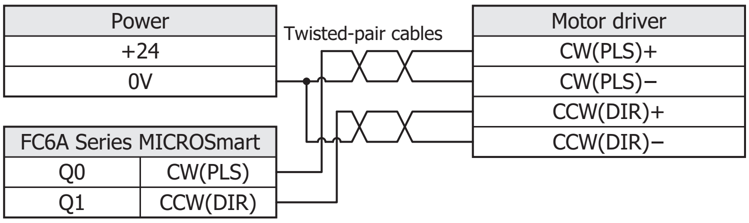


Power
+24
0V

FC6A Series MICROSmart	
Q0	CW(PLS)
Q1	CCW(DIR)

Motor driver
CW(PLS)+
CW(PLS)−
CCW(DIR)+
CCW(DIR)−

FC6A Series MICROSmart: Transistor protection source output type



Power
+24
0V

FC6A Series MICROSmart	
Q0	CW(PLS)
Q1	CCW(DIR)

Motor driver
CW(PLS)+
CW(PLS)−
CCW(DIR)+
CCW(DIR)−

⚠ **Caution** Do not reverse the power supply connection under any circumstances. Doing so may damage the components.
When wiring the components, carefully read and understand the manual for the motor driver first.

Practical Examples

Example of Single Axis Control Using the RAMP Instruction

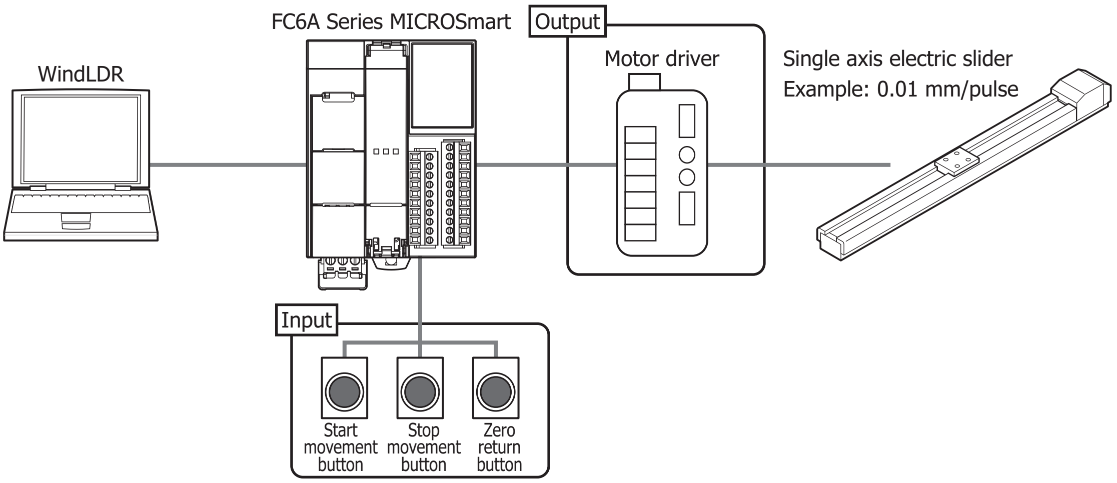
■ **Application**

Single axis electric slider

■ **System configuration diagram**

Electric slider specification is treated as 0.01 mm per pulse in this example.

To learn the actual travel distance of the electric slider that will be used, check the electric slider specifications (travel distance per pulse).



■ **Operation**

When the stop movement switch is pressed for 3 seconds, ABS instruction will be executed to establish the origin.

When the start movement button is pressed, the electric slider moves as follows: Point A → Point C → Point B → Point A. The electric slider stops when it reaches Point A the second time.

When the stop movement button is pressed, the moving electric slider stops.

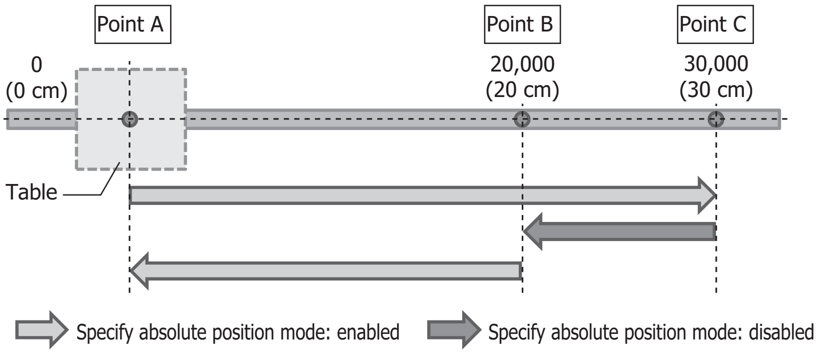
When the zero return button is pressed, the electric slider returns to the origin (Point A) from the stopped position.

Movement from Point A to Point C and from Point B to Point A is executed with specify absolute position mode enabled.

Movement from Point C to Point B is executed with specify absolute position mode disabled.

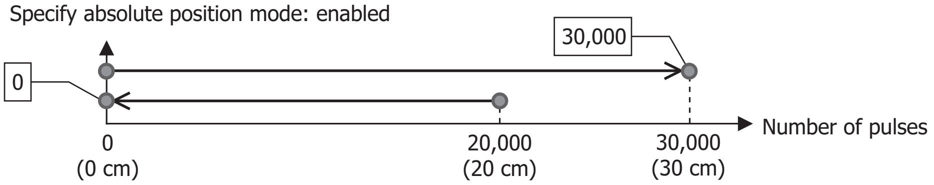
If the frequency change curve is set to **S-shaped curve**, shocks and vibration are suppressed because the initial acceleration can be reduced.

Position	X (cm)
Point A	0
Point B	20
Point C	30

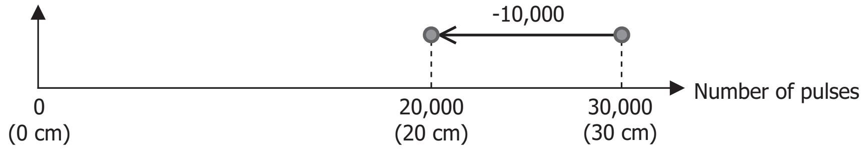

Point A
Point B
Point C
0
(0 cm)
20,000
(20 cm)
30,000
(30 cm)
Table
Specify absolute position mode: enabled
Specify absolute position mode: disabled

Distances in parentheses have been calculated at 0.01 mm per pulse.

Specify absolute position mode: enabled



Specify absolute position mode: disabled

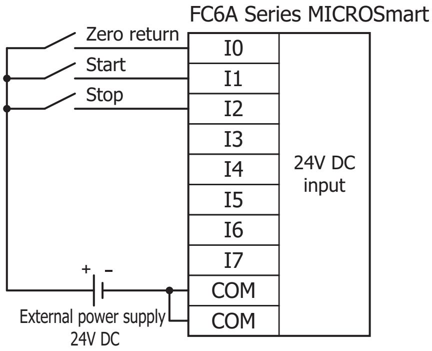


Distances in parentheses have been calculated at 0.01 mm per pulse.

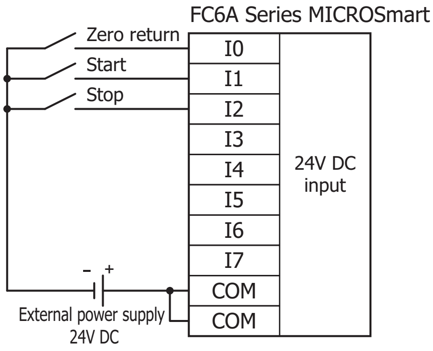
■ **Wiring diagram**

Input terminals

DC sink input wiring

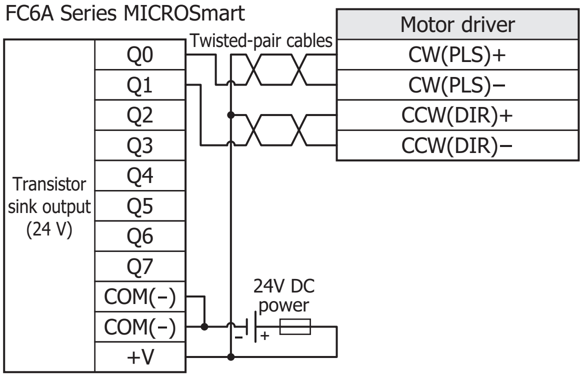


DC source input wiring

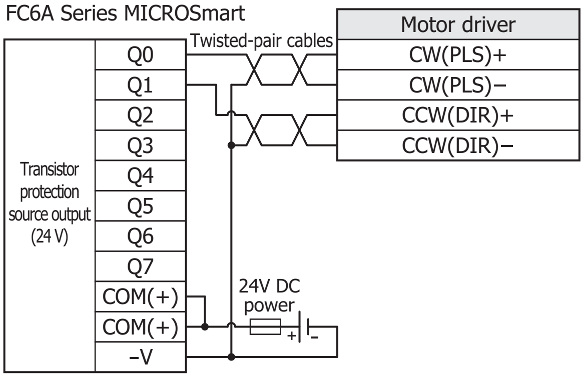


Output terminals

Transistor sink output type



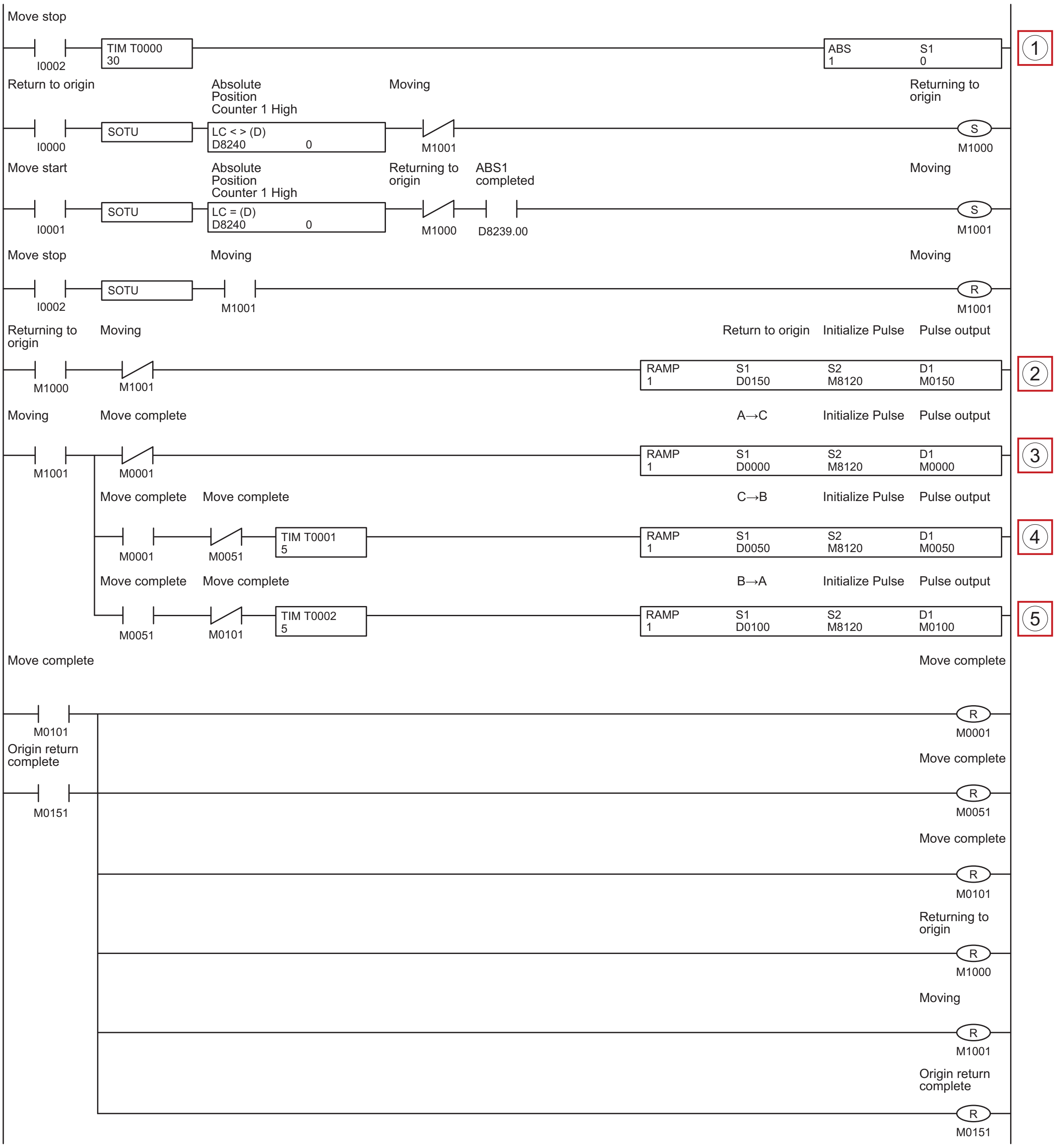
Transistor protection source output type



■ Assigned addresses

Device	Unit	Range	Description
Input	Bit	I0	Zero return button
		I1	Start movement button
		I2	Stop movement button
Internal relay	Bit	M10-M13	RAMP instruction operation status
		M50-M53	
		M100-M103	
		M150-M153	
		M1000	Zero return bit
		M1001	Start movement bit
		M1002	Stop movement bit
Special internal relay		M8120	Initialize pulse
Data register	Bit Word	D10-D20	RAMP instruction control registers
		D50-D60	
		D100-D110	
		D150-D160	
Timer		T0	Timer settings
		T1	
Counter		C0	Counter settings

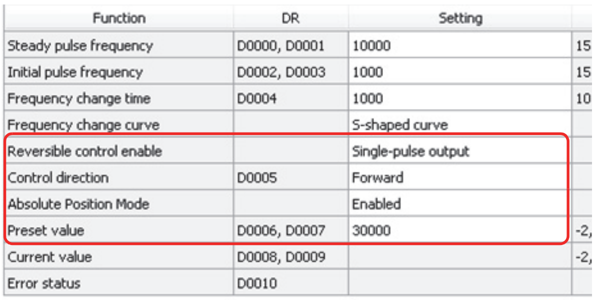
■ Ladder diagram



No.	Description
(1)	Origin positioning
(2)	Zero return
(3)	Move Point A → Point C
(4)	Move Point C → Point B
(5)	Move Point B → Point A

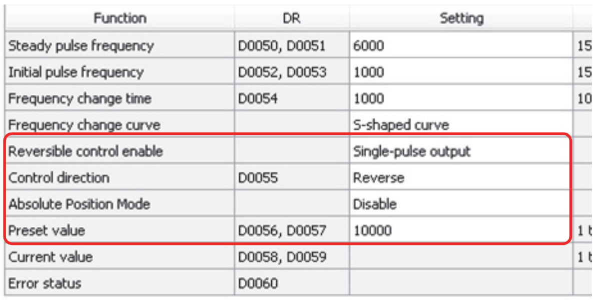
■ Settings tabs

Point A → Point C



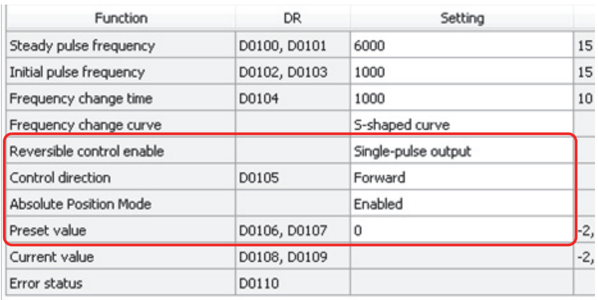
Function	DR	Setting	
Steady pulse frequency	D0000, D0001	10000	15
Initial pulse frequency	D0002, D0003	1000	15
Frequency change time	D0004	1000	10
Frequency change curve		S-shaped curve	
Reversible control enable		Single-pulse output	
Control direction	D0005	Forward	
Absolute Position Mode		Enabled	
Preset value	D0006, D0007	30000	-2,
Current value	D0008, D0009		-2,
Error status	D0010		

Point C → Point B



Function	DR	Setting	
Steady pulse frequency	D0050, D0051	6000	15
Initial pulse frequency	D0052, D0053	1000	15
Frequency change time	D0054	1000	10
Frequency change curve		S-shaped curve	
Reversible control enable		Single-pulse output	
Control direction	D0055	Reverse	
Absolute Position Mode		Disable	
Preset value	D0056, D0057	10000	1 t
Current value	D0058, D0059		1 t
Error status	D0060		

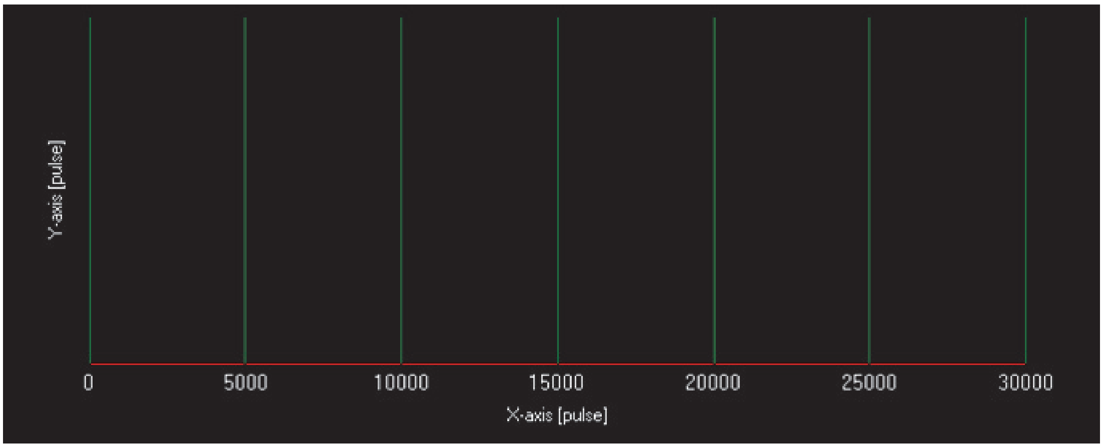
Point B → Point A



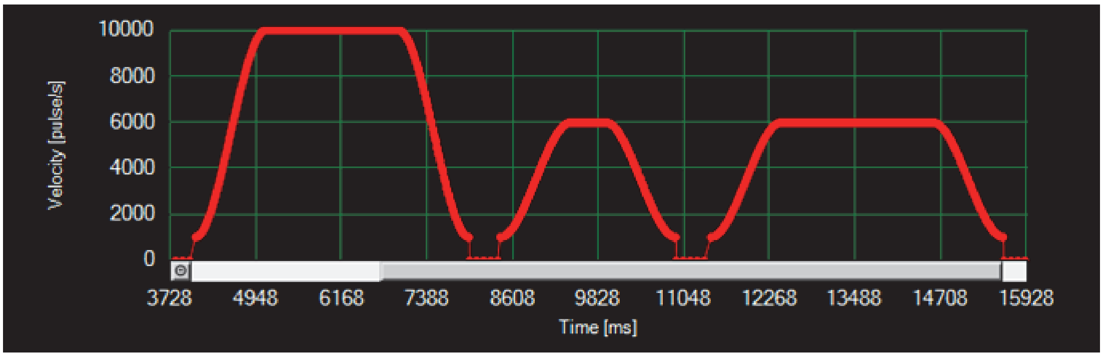
Function	DR	Setting	
Steady pulse frequency	D0100, D0101	6000	15
Initial pulse frequency	D0102, D0103	1000	15
Frequency change time	D0104	1000	10
Frequency change curve		S-shaped curve	
Reversible control enable		Single-pulse output	
Control direction	D0105	Forward	
Absolute Position Mode		Enabled	
Preset value	D0106, D0107	0	-2,
Current value	D0108, D0109		-2,
Error status	D0110		

■ Pulse monitor window

Trajectory



Waveform



Example of Simultaneous Dual Axis Control Using the RAMPL Instruction

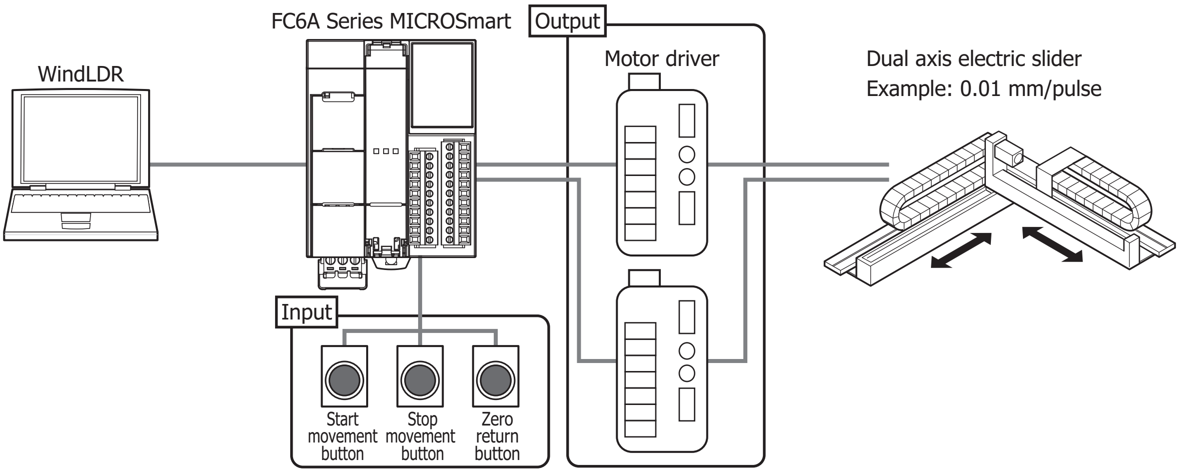
■ Application

Dual axis electric slider

■ System configuration diagram

Electric slider specification is treated as 0.01 mm per pulse in this example.

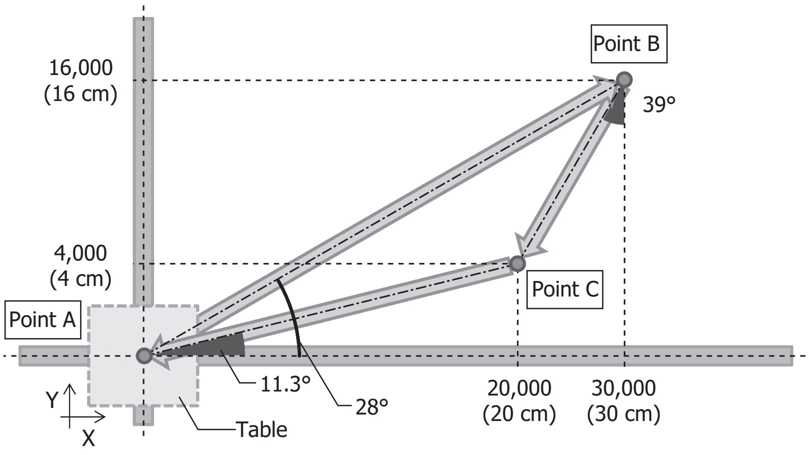
To learn the actual travel distance of the electric slider that will be used, check the electric slider specifications (travel distance per pulse).



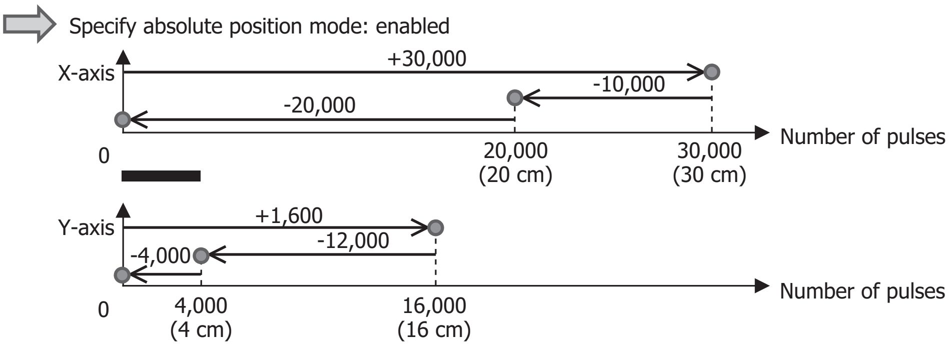
■ Operation

When the stop movement switch is pressed for 3 seconds, ABS instruction will be executed to establish the origin.
When the start movement button is pressed, the electric slider moves as follows: Point A → Point B → Point C → Point A. The electric slider stops when it reaches Point A the second time.
When the stop movement button is pressed, the moving electric slider stops.
When the zero return button is pressed, the electric slider returns to the origin (Point A) from the stopped position.

Position	X (cm)	Y (cm)
Point A	0	0
Point B	30	16
Point C	20	4



Distances in parentheses have been calculated at 0.01 mm per pulse.

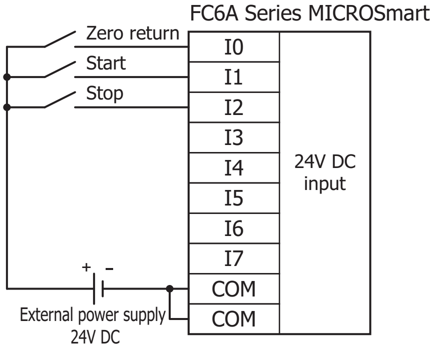


Distances in parentheses have been calculated at 0.01 mm per pulse.

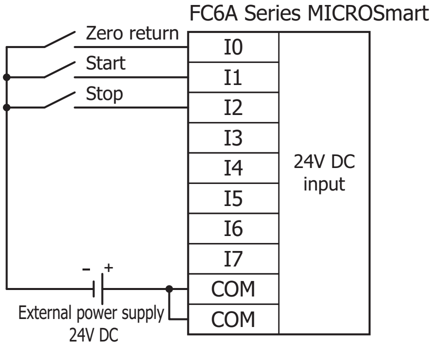
■ Wiring diagram

Input terminals

DC sink input wiring

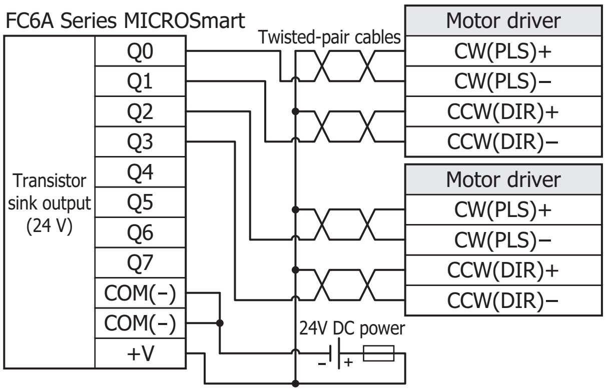


DC source input wiring

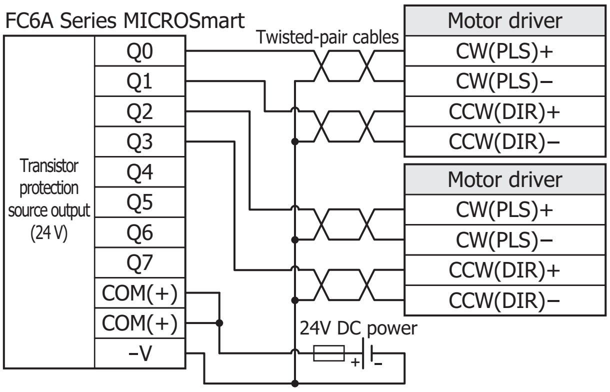


Output terminals

Transistor sink output type



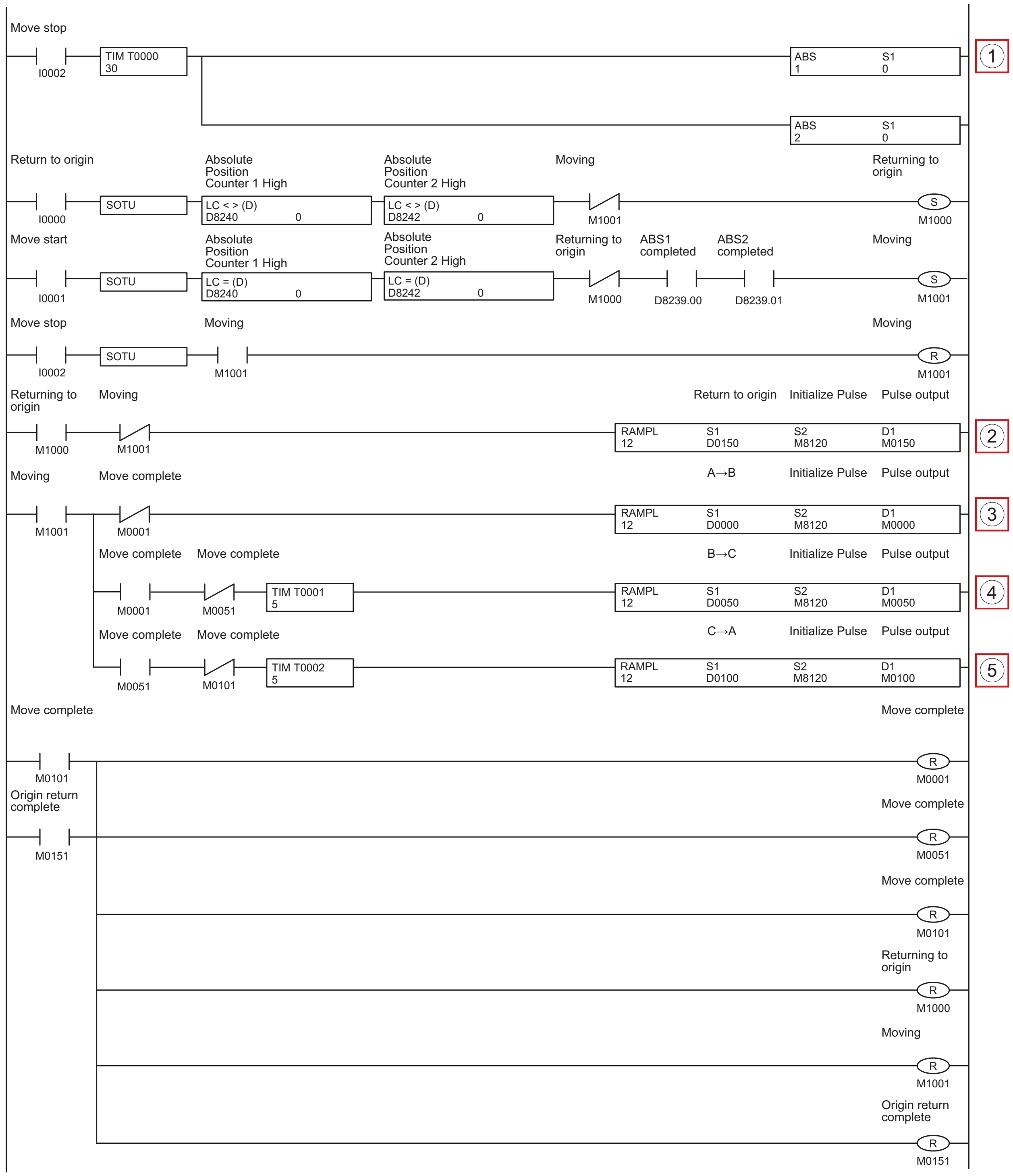
Transistor protection source output type



■ Assigned addresses

Device	Unit	Range	Description
Input	Bit	I0	Zero return button
		I1	Start movement button
		I2	Stop movement button
Internal relay	Bit	M10-M13	RAMPL instruction operation status
		M50-M53	
		M100-M103	
		M150-M153	
		M1000	Zero return bit
		M1001	Start movement bit
		M1002	Stop movement bit
Special internal relay		M8120	Initialize pulse
Data register	Bit Word	D10-D39	RAMPL instruction control registers
		D50-D79	
		D100-D129	
		D150-D179	
Timer		T0	Timer settings
		T1	
Counter		C0	Counter settings

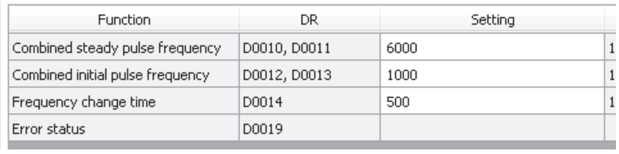
■ Ladder diagram



No.	Description
(1)	Origin positioning
(2)	Zero return
(3)	Move Point A → Point B
(4)	Move Point B → Point C
(5)	Move Point C → Point A

■ Settings tabs

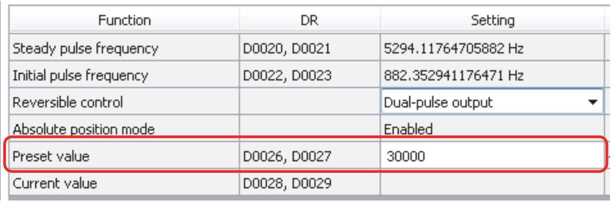
Common settings



Function	DR	Setting	
Combined steady pulse frequency	D0010, D0011	6000	1
Combined initial pulse frequency	D0012, D0013	1000	1
Frequency change time	D0014	500	1
Error status	D0019		

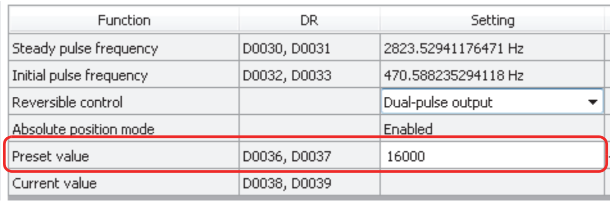
Point A → Point B

X axis



Function	DR	Setting
Steady pulse frequency	D0020, D0021	5294.11764705882 Hz
Initial pulse frequency	D0022, D0023	882.352941176471 Hz
Reversible control		Dual-pulse output
Absolute position mode		Enabled
Preset value	D0026, D0027	30000
Current value	D0028, D0029	

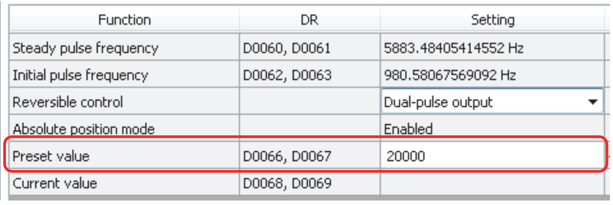
Y axis



Function	DR	Setting
Steady pulse frequency	D0030, D0031	2823.52941176471 Hz
Initial pulse frequency	D0032, D0033	470.588235294118 Hz
Reversible control		Dual-pulse output
Absolute position mode		Enabled
Preset value	D0036, D0037	16000
Current value	D0038, D0039	

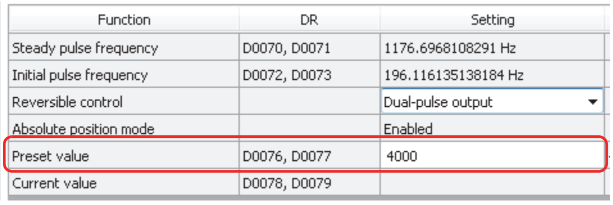
Point B → Point C

X axis



Function	DR	Setting
Steady pulse frequency	D0060, D0061	5883.48405414552 Hz
Initial pulse frequency	D0062, D0063	980.58067569092 Hz
Reversible control		Dual-pulse output
Absolute position mode		Enabled
Preset value	D0066, D0067	20000
Current value	D0068, D0069	

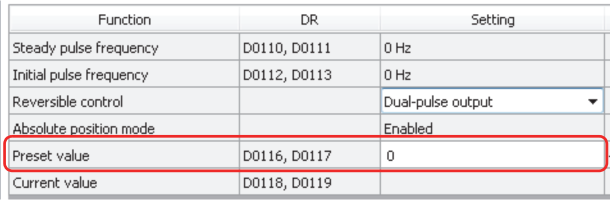
Y axis



Function	DR	Setting
Steady pulse frequency	D0070, D0071	1176.6968108291 Hz
Initial pulse frequency	D0072, D0073	196.116135138184 Hz
Reversible control		Dual-pulse output
Absolute position mode		Enabled
Preset value	D0076, D0077	4000
Current value	D0078, D0079	

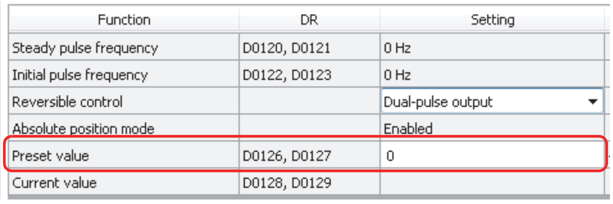
Point C → Point A

X axis



Function	DR	Setting
Steady pulse frequency	D0110, D0111	0 Hz
Initial pulse frequency	D0112, D0113	0 Hz
Reversible control		Dual-pulse output
Absolute position mode		Enabled
Preset value	D0116, D0117	0
Current value	D0118, D0119	

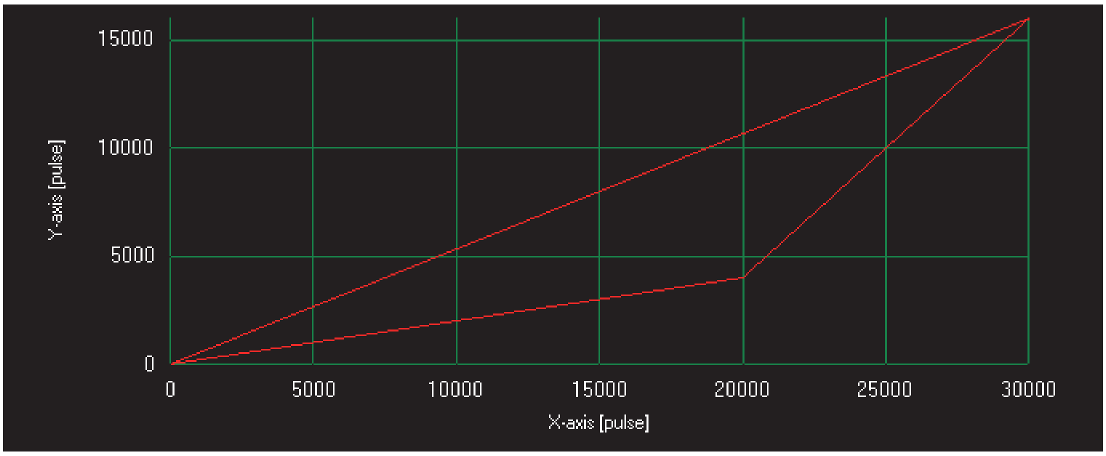
Y axis



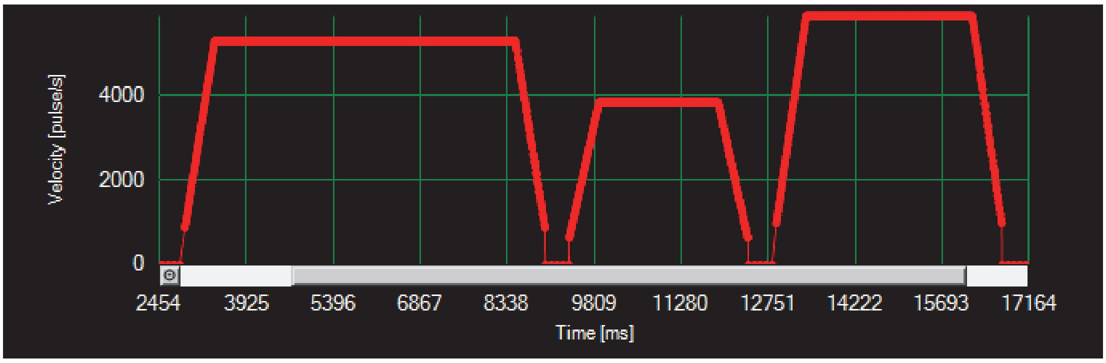
Function	DR	Setting
Steady pulse frequency	D0120, D0121	0 Hz
Initial pulse frequency	D0122, D0123	0 Hz
Reversible control		Dual-pulse output
Absolute position mode		Enabled
Preset value	D0126, D0127	0
Current value	D0128, D0129	

■ Pulse monitor window

Trajectory



Waveform



19: PID Control Instruction

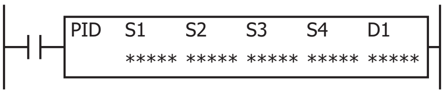
Introduction

This chapter describes the PID control instructions that perform auto tuning and PID control for temperature control and other purposes.

PID (PID Control)

Executes PID control and outputs that result. When auto tuning is performed, the optimal PID parameters (proportional gain, integral time, and derivative time) and control action are automatically calculated.

This instruction is compatible with the FC5A Series MICROSmart PID instruction. When temperature control is performed using the FC5A Series MICROSmart PID instruction, the same temperature control can be achieved by replacing it with the FC6A Series MICROSmart and the FC6A Series MICROSmart PID instruction.



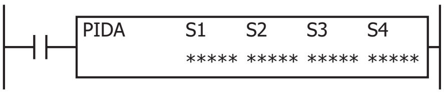
Notes:

- The PID instruction and the PIDA instruction can be mixed in a single program, and a maximum of 32 instructions can be used.
- For details on the PID instruction, see Chapter 14 "PID Instruction" in the "FC5A Series MICROSmart Pentra User's Manual Advanced Volume".

PIDA (PID Control)

Executes PID control and outputs that result. When auto tuning is performed, the optimal PID constants are calculated. When auto/manual mode is switched, the balance-less and bumpless function automatically operates to prevent rapid changes in the output manipulated variable.

A maximum of four alarm types, out of nine, can be set to detect process variable errors for multiple conditions.

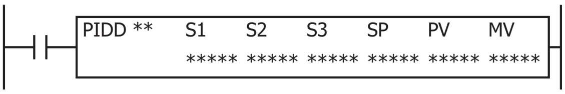


Notes:

- The PID instruction and the PIDA instruction can be mixed in a single program, and a maximum of 32 instructions can be used.
- For details on the PIDA instruction, see "PIDA (PID Control)" on page 19-3.

PIDD (PID with Derivative Decay)

Executes PID control and outputs that result. The PID control parameters stored in the control registers are handled as data type F (float), and fine adjustments can be made to the control parameters such as proportional gain, integral gain, and derivative gain. Multiple PIDD instructions can also be combined to execute the cascade control.



Notes:

- A maximum of 32 PIDD instructions can be used in a user program. PID and PIDA instructions are counted separately.
- For details on the PIDD instruction, see "PIDD (PID with Derivative Decay)" on page 19-26.

Differences between PID, PIDA, PIDD Instructions and PID Module

There are three types of PID control instructions, the PID instruction, the PIDA instruction, and the PIDD instruction. PID control can also be performed using the PID module. The differences between the various PID instructions and the PID module are as follows.

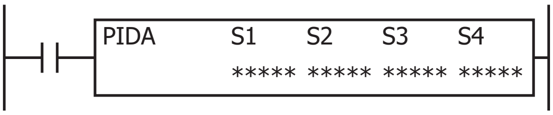
Item		PID Instruction	PIDA Instruction	PIDD Instruction	PID Module
Limit on number of instructions		Both PID and PIDA instructions: 32 total		32	—
Data type of parameters		W (Word)/I (Integer)		F (Float)	W (Word)/ I (Integer) (To tenths place)
Cascade control function		No	No	Yes	Yes
Heating/cooling control		No	No	No	Yes
Auto/manual switching		Yes	Yes	Yes	Yes
Auto tuning	Method	Step response method	Limit cycle method	No	Ultimate gain method
	Execution timing	Execute when starting the PID instruction	Execute at any time		Can be executed at any time
ARW (anti-reset windup)		Yes	Yes	Yes	Yes
Balance-less and bumpless function		No	Yes	Yes	Yes
Alarm types	Upper limit alarm Lower limit alarm	Yes	Yes	No	Yes
	Upper/lower limits alarm Upper/lower limits range alarm Process high alarm Process low alarm Upper limit alarm with standby Lower limit alarm with standby Upper/lower limits alarm with standby	No	Yes	No	Yes
Parameters		PIDST instruction, PID instruction	PIDA instruction	PIDD instruction	PID module settings
Monitor in WindLDR	Check and modify parameters	Use Device Monitor	Use a dedicated monitor screen		
	Trend graph display	No	Yes		
IDEC instruction compatibility		FC5A Series MICROSmart PID instruction	FT1A Touch PID instruction	No	No

PIDA (PID Control)

The PID instruction executes PID control and outputs the result. When auto tuning is performed, the optimal PID parameters are calculated.

When auto/manual mode is switched, the balance-less and bumpless function automatically operates to prevent rapid changes in the output manipulated variable.

Four alarm types, out of nine, can be set to detect process variable errors for multiple conditions.



Operation

When the input is on, auto tuning and PID control are performed.

Warning

- Special technical knowledge about the PID control is required to use the PIDA function of the FC6A Series MICROSmart. Use of the PIDA function without understanding the PID control may cause the FC6A Series MICROSmart to perform unexpected operation, resulting in disorder of the control system, damage, or accidents.
- When using the PIDA instruction for feedback control, emergency stop and interlocking circuits must be configured outside the FC6A Series MICROSmart. If these are configured in the internal circuit and the process variable is not input correctly, there is a risk of no longer being able to perform proper feedback control, which may damage connected devices or cause accidents.
  For PID control, see "PID Control" on page 19-23.

Caution

The PIDA instruction cannot be used in an interrupt program. If used in an interrupt program, a user program execution error occurs, the execution of the instruction is canceled, and the next instruction is executed.
For user program execution errors, see "User Program Execution Errors" on page 3-10.

Note: The PIDA instruction and the PID instruction can be mixed in a single program, and a maximum of 32 instructions can be used.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Control registers	—	—	—	—	—	—	X*1	—	—	—
S2 (Source 2)	Initialization input	X	—	X	—	—	—	—	—	—	—
S3 (Source 3)	Control relays	—	—	X*2	—	—	—	—	—	—	—
S4 (Source 4)	Set point	—	—	—	—	—	—	X*1	—	—	—

*1 Special data registers cannot be used.

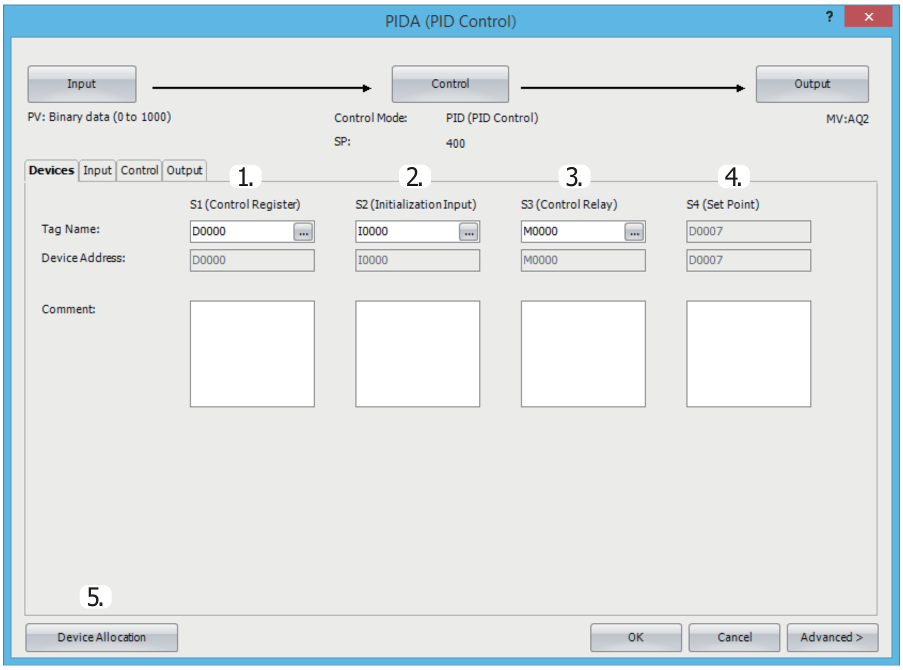
*2 Special internal relays cannot be used. Only 0 can be specified as the first digit of the internal relay number. 1 to 7 cannot be specified.

Settings

The **PIDA (PID Control)** dialog box contains the **Devices** tab, **Input** tab, **Control** tab, and **Output** tab.

The **Devices** tab configures the devices used with the PIDA instruction. The **Input**, **Control**, and **Output** tabs configure the initial values of the parameters for the PIDA instruction.

■Devices tab


PIDA (PID Control)

PV: Binary data (0 to 1000)

Control Mode: PID (PID Control)

SP: 400

MV:AQ2

	S1 (Control Register)	S2 (Initialization Input)	S3 (Control Relay)	S4 (Set Point)
Tag Name:	D0000	I0000	M0000	D0007
Device Address:	D0000	I0000	M0000	D0007
Comment:				

1. S1 (source 1): Control Register

Specify the first device to store the PIDA instruction parameters.

The devices that can be specified are data registers D0000 to D7960 and D10000 to D61960.

40 words are used starting from the specified data register.

When the initialization input is turned on, the control registers are initialized with the values set on the **Input** tab, the **Control** tab, and the **Output** tab.

For initialization, see "2. S2 (source 2): Initialization Input" on page 19-4.

For the control registers, see "S1: Control Registers" on page 19-16.

2. S2 (source 2): Initialization Input

Specify the device to initialize the control registers.

The values configured on the **Input** tab, **Control** tab, and **Output** tab are stored in the control registers and control relays when the initialization input turns on*1. An external input or an internal relay can be specified.

*1 If you want the initialization to be performed just one time, please use a SOTU (single output up) or a SOTD (single output down) instruction.

3. S3 (source 3): Control Relay

Specify the device to output control of PID control and the control results of the PIDA instruction.

16 bits are used starting from the specified device.

The devices that can be specified are internal relays M0000 to M7980 and M10000 to M21230. Special internal relays cannot be specified.

The role of each bit is different. You can switch between auto and manual mode for the PIDA instruction and execute auto tuning by turning the bits on and off. The PIDA instruction control result and process variable error are also output as alarms.

For tag names, device addresses, and comments, see "S3: Control Relay" on page 19-21.

4. S4 (source 4): Set Point

Shows the device that stores the PID control set point. When a data register is specified in S1 (source 1), S1+7 is automatically allocated.

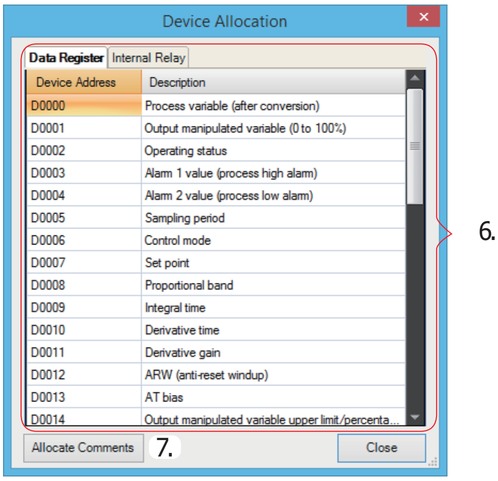
The set point (S1+7) is one data register out of the 40 words of data registers used by the control registers (1). This is automatically displayed when a first device is specified in the control registers (1).

5. Device Allocation

Click this button to display the **Device Allocation** dialog box. The table containing the data registers and internal relays and corresponding PIDA instruction settings is displayed in the dialog box (6).

Click **Allocate Comments** (7) to set the content in the table as the comments of the devices.

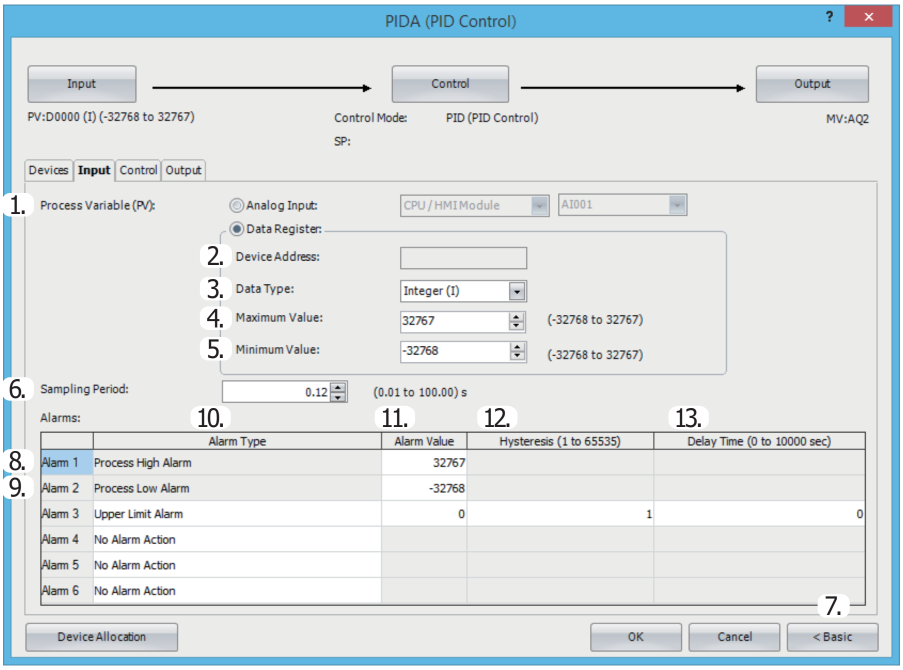
Device Allocation dialog box



Device Address	Description
D0000	Process variable (after conversion)
D0001	Output manipulated variable (0 to 100%)
D0002	Operating status
D0003	Alarm 1 value (process high alarm)
D0004	Alarm 2 value (process low alarm)
D0005	Sampling period
D0006	Control mode
D0007	Set point
D0008	Proportional band
D0009	Integral time
D0010	Derivative time
D0011	Derivative gain
D0012	ARW (anti-reset windup)
D0013	AT bias
D0014	Output manipulated variable upper limit/percenta...

■Input tab

This tab configures the input parameters for the PIDA instruction.



PV:D0000 (I) (-32768 to 32767)

Control Mode: PID (PID Control)

SP:

MV:AQ2

Devices | **Input** | Control | Output

1. Process Variable (PV): ○ Analog Input: CPU / HMI Module AI001
   ● Data Register:
2. Device Address:
3. Data Type: Integer (I)
4. Maximum Value: 32767 (-32768 to 32767)
5. Minimum Value: -32768 (-32768 to 32767)
6. Sampling Period: 0.12 (0.01 to 100.00) s

Alarms:

	10. Alarm Type	11. Alarm Value	12. Hysteresis (1 to 65535)	13. Delay Time (0 to 10000 sec)
8. Alarm 1	Process High Alarm	32767		
9. Alarm 2	Process Low Alarm	-32768		
Alarm 3	Upper Limit Alarm	0	1	0
Alarm 4	No Alarm Action			
Alarm 5	No Alarm Action			
Alarm 6	No Alarm Action			

1. Process Variable (S1+0)

Configure the input for the PID control. The process variable is linearly converted and given to the PID control for the input value. The value can be selected as **Analog Input** or **Data Register**.

When Analog Input is specified

Specify the module with the analog input you want to specify as the process variable and its analog input number to set as PID control input.

Analog Module Configure:	Select the analog input module. Select the module as the CPU/HMI module or expansion module 1 to 15.
Analog Input No.:	Select the input number on the analog input module. Select as AI0 to AI7. The device range that can be used depends on the model.

The analog input value is also stored in a control register (S1+0) as a read-only value.
For control registers, see "1. S1 (source 1): Control Register" on page 19-4.

Notes:

- To use an analog input, the analog input must be configured in advance. For details on analog input settings, see Chapter 9 "Analog I/O Modules" in the "FC6A Series MICROSmart User's Manual".
- When using an analog input, the analog volume value cannot be used as the process variable. However, the analog volume value can be used as the process variable by specifying **Data Register** as the process variable and specifying a special data register that stores the analog volume value.
- When **Analog Input** is specified, the minimum value and the maximum value of the analog input settings are used as the minimum value and the maximum value for the input.

When Data Register is specified

The value stored in S1+0 of the control registers is used as the PID control input.
The range of stored values is minimum value (5) ≤ process variable ≤ maximum value (4). For control registers, see "1. S1 (source 1): Control Register" on page 19-4.
Configure parameters 2. Device Address through 5. Minimum Value when **Data Register** is selected as the process variable.

2. Device Address

Shows the first data register (S1+0) of the control registers configured on the **Devices** tab.

3. Data Type

Select the process variable (1) data type as W (word) or I (integer).

4. Maximum Value, 5. Minimum Value

Set the process variable (1) range as a maximum value and a minimum value. Set the input settings in PID control (process variable, Alarm 1 (process high alarm) value, and Alarm 2 (process low alarm) value) and the set point within the maximum and minimum value range. The maximum value and minimum value can be configured within the range of the data type (3).

- When the data type is I (integer): -32,768 to +32,767.
- When the data type is W (word): 0 to 65,535

Note: If the process variable (1) is larger than the maximum value (4), or if the process variable (1) is smaller than the minimum value (5), status code 109 is stored in the operating status (S1+2) and PID control stops.
For status code details, see "Operation status (S1+2)" on page 19-19.

6. Sampling Period (S1+5)

Set the cycle to execute the PID control.

PID control will only be executed at the end of the scan.

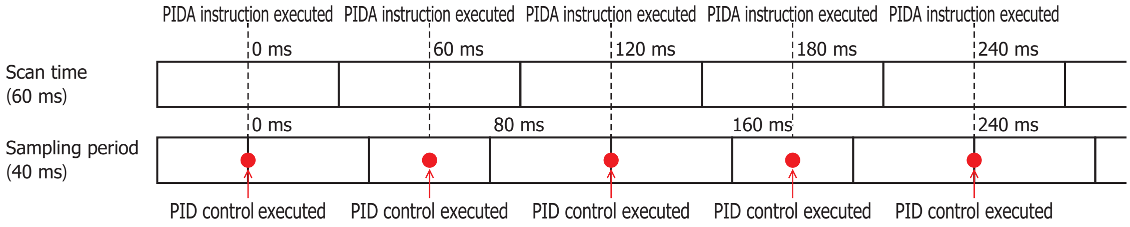
When the sampling period is set lower than the scan time, PID control is not executed with the set sampling period, it will be executed with the same period as the scan time.

When the sampling period is set longer than the scan time, if the scan has not finished when the sampling period has elapsed, PID control will wait until the scan ends and then be executed.

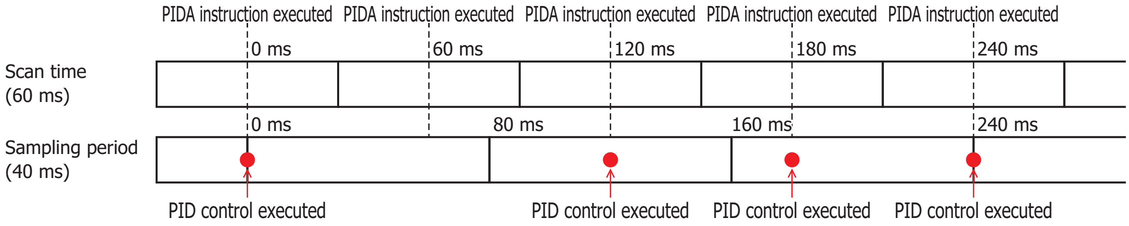
Due to the above factors, an error will occur between the sampling period and the actual PID control execution timing.

The sampling period can be set from 0.01 to 100.00 seconds in 0.01 second increments.

Sampling period ≤ Scan time



Sampling period > Scan time



7. Advanced/Basic

Click this button to show or hide the advanced settings.

On the **Input** tab, Sampling Period (6) can be set as an advanced setting.

8. Alarm 1 (Process High Alarm) (S1+3)

Set the upper limit value of the process variable (1).

The alarm 1 value can be set in the range between the alarm 2 value (process low alarm) (9) and the maximum value (4).

If the process variable is higher than or equal to the alarm 1 value, the alarm 1 output (S3+3) turns on.

If the process variable is lower than the alarm 1 value, the alarm 1 output (S3+3) turns off.

9. Alarm 2 (Process Low Alarm) (S1+4)

Set the lower limit value of the process variable (1).

The alarm 2 value can be set in the range between the minimum value (5) and the alarm 1 value (process high alarm) (8).

If the process variable is lower than or equal to the alarm 2 value, the alarm 2 output (S3+4) turns on.

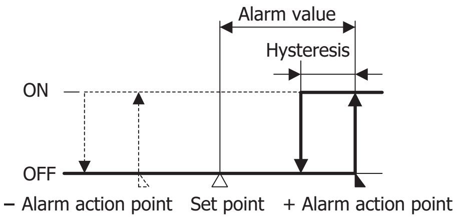
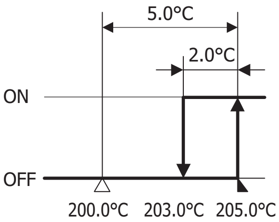
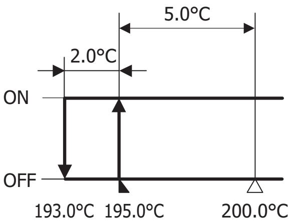
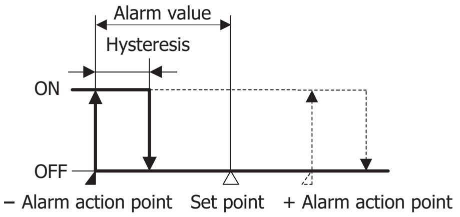
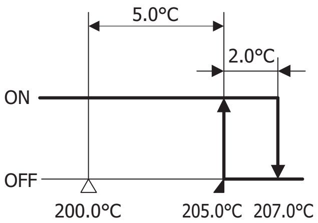
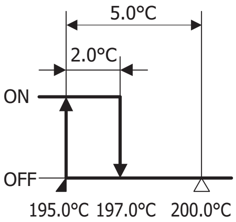
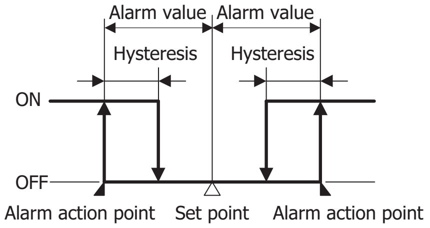
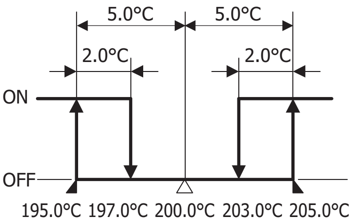
If the process variable is higher than the alarm 2 value, the alarm 2 output (S3+4) turns off.

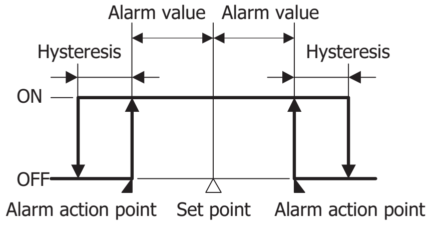
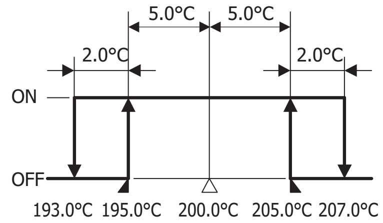
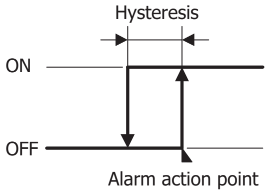
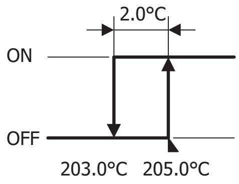
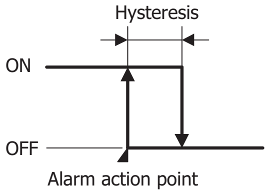
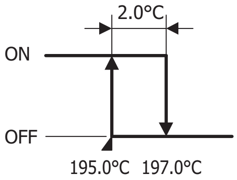
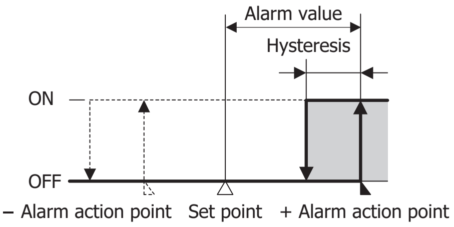
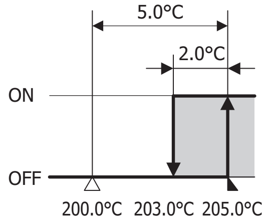
A maximum of four alarms (Alarm 3 to Alarm 6) can be configured for the process variable in PID control.

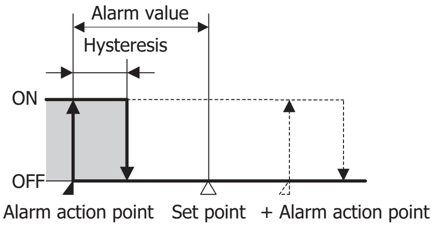
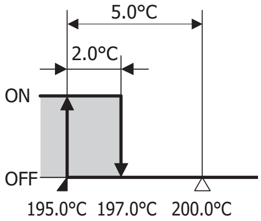
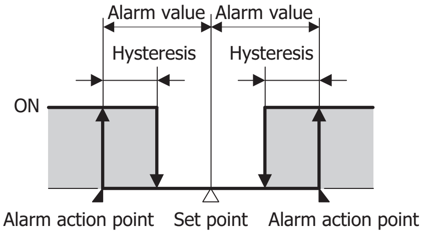
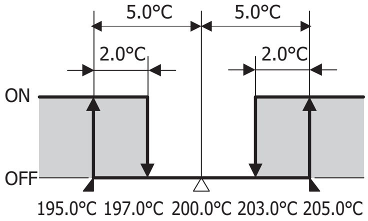
Individually configure Alarm Type (10), Alarm Value (11), Hysteresis (12), and Delay Time (13).

10. Alarm Type

Select the type of alarm action from the following.

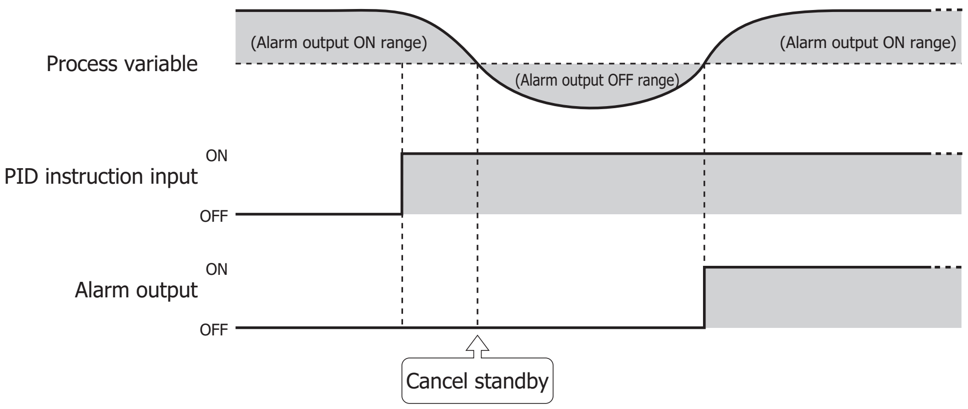
Type	Action	Example
No Alarm Action	Do not output an alarm.	-
Upper Limit Alarm	Turns ON the alarm output when process variable ≥ (set point + alarm value). Turns OFF the alarm output when process variable ≤ (set point + alarm value - hysteresis). When (set point + alarm value - hysteresis) < process variable < (set point + alarm value), the alarm output maintains the state of the previous scan.  Alarm value Hysteresis ON OFF − Alarm action point Set point + Alarm action point	Set point: 200.0°C Alarm value: 5.0°C Hysteresis: 2.0°C  5.0°C 2.0°C ON OFF 200.0°C 203.0°C 205.0°C When process variable ≥ 205.0°C, the alarm output is turned ON. When process variable ≤ 203.0°C, the alarm output is turned OFF.
		Set point: 200.0°C Alarm value: -5.0°C Hysteresis: 2.0°C  5.0°C 2.0°C ON OFF 193.0°C 195.0°C 200.0°C When process variable ≥ 195.0°C, the alarm output is turned ON. When process variable ≤ 193.0°C, the alarm output is turned OFF.
Lower Limit Alarm	Turns ON the alarm output when process variable ≤ (set point + alarm value). Turns OFF the alarm output when process variable ≥ (set point + alarm value + hysteresis). When (set point + alarm value) < process variable < (set point + alarm value + hysteresis), the alarm output maintains the state of the previous scan.  Alarm value Hysteresis ON OFF − Alarm action point Set point + Alarm action point	Set point: 200.0°C Alarm value: 5.0°C Hysteresis: 2.0°C  5.0°C 2.0°C ON OFF 200.0°C 205.0°C 207.0°C When process variable ≤ 205.0°C, the alarm output is turned ON. When process variable ≥ 207.0°C, the alarm output is turned OFF.
		Set point: 200.0°C Alarm value: -5.0°C Hysteresis: 2.0°C  5.0°C 2.0°C ON OFF 195.0°C 197.0°C 200.0°C When process variable ≤ 195.0°C, the alarm output is turned ON. When process variable ≥ 197.0°C, the alarm output is turned OFF.
Upper/Lower Limit Alarm	Turns ON the alarm output when process variable ≥ (set point + alarm value). Turns ON the alarm output when process variable ≤ (set point - alarm value). Turns OFF the alarm output when (set point - alarm value + hysteresis) ≤ process variable ≤ (set point + alarm value - hysteresis). When (set point + alarm value - hysteresis) < process variable < (set point + alarm value), the alarm output maintains the state of the previous scan. When (set point - alarm value) < process variable < (set point - alarm value + hysteresis), the alarm output maintains the state of the previous scan.  Alarm value Alarm value Hysteresis Hysteresis ON OFF Alarm action point Set point Alarm action point	Set point: 200.0°C Alarm value: 5.0°C Hysteresis: 2.0°C  5.0°C 5.0°C 2.0°C 2.0°C ON OFF 195.0°C 197.0°C 200.0°C 203.0°C 205.0°C When process variable ≥ 205.0°C, the alarm output is turned ON. When process variable ≤ 195.0°C, the alarm output is turned ON. When 197.0°C ≤ process variable ≤ 203.0°C, the alarm output is turned OFF.

Type	Action	Example
Upper/Lower Limit Range Alarm	Turns ON the alarm output when (set point - alarm value) ≤ process variable ≤ (set point + alarm value). Turns OFF the alarm output when process variable ≥ (set point + alarm value + hysteresis). Turns OFF the alarm output when process variable ≤ (set point - alarm value - hysteresis). When (set point + alarm value) < process variable < (set point + alarm value + hysteresis), the alarm output maintains the state of the previous scan. When (set point - alarm value - hysteresis) < process variable < (set point - alarm value), the alarm output maintains the state of the previous scan.  Alarm value Alarm value Hysteresis Hysteresis ON OFF Alarm action point Set point Alarm action point	Set point: 200.0°C Alarm value: 5.0°C Hysteresis: 2.0°C  5.0°C 5.0°C 2.0°C 2.0°C ON OFF 193.0°C 195.0°C 200.0°C 205.0°C 207.0°C When 195.0°C ≤ process variable ≤ 205.0°C, the alarm output is turned ON. When process variable ≥ 207.0°C, the alarm output is turned OFF. When process variable ≤ 193.0°C, the alarm output is turned OFF.
Process High Alarm	Turns ON the alarm output when process variable ≥ alarm value. Turns OFF the alarm output when process variable ≤ (alarm value - hysteresis). When (alarm value - hysteresis) < process variable < alarm value, the alarm output maintains the state of the previous scan.  	Alarm value: 205.0°C Hysteresis: 2.0°C  When process variable ≥ 205.0°C, the alarm output is turned ON. When process variable ≤ 203.0°C, the alarm output is turned OFF.
Process Low Alarm	Turns ON the alarm output when process variable ≤ alarm value. Turns OFF the alarm output when process variable ≥ (alarm value + hysteresis). When alarm value < process variable < (alarm value + hysteresis), the alarm output maintains the state of the previous scan.  	Alarm value: 195.0°C Hysteresis: 2.0°C  When process variable ≤ 195.0°C, the alarm output is turned ON. When process variable ≥ 197.0°C, the alarm output is turned OFF.
Upper Limit Alarm with Standby	Turns ON the alarm output when process variable ≥ (set point + alarm value). Turns OFF the alarm output when process variable ≤ (set point + alarm value - hysteresis). When (set point + alarm value - hysteresis) < process variable < (set point + alarm value), the alarm output maintains the state of the previous scan. The gray-colored portion is where the standby function operates.  Alarm value Hysteresis ON OFF − Alarm action point Set point + Alarm action point	Set point: 200.0°C Alarm value: 5.0°C Hysteresis: 2.0°C  5.0°C 2.0°C ON OFF 200.0°C 203.0°C 205.0°C When process variable ≥ 205.0°C, the alarm output is turned ON. When process variable ≤ 203.0°C, the alarm output is turned OFF.

Type	Action	Example
Lower Limit Alarm with Standby	Turns ON the alarm output when process variable ≤ (set point + alarm value). Turns OFF the alarm output when process variable ≥ (set point + alarm value + hysteresis). When (set point + alarm value) < process variable < (set point + alarm value + hysteresis), the alarm output maintains the state of the previous scan. The gray-colored portion is where the standby function operates.  Alarm value Hysteresis ON OFF − Alarm action point Set point + Alarm action point	Set point: 200.0°C Alarm value: -5.0°C Hysteresis: 2.0°C  5.0°C 2.0°C ON OFF 195.0°C 197.0°C 200.0°C When process variable ≤ 195.0°C, the alarm output is turned ON. When process variable ≥ 197.0°C, the alarm output is turned OFF.
Upper/Lower Limit Alarm with Standby	Turns ON the alarm output when process variable ≥ (set point + alarm value). Turns ON the alarm output when process variable ≤ (set point - alarm value). Turns OFF the alarm output when (set point - alarm value + hysteresis) ≤ process variable ≤ (set point + alarm value - hysteresis). When (set point + alarm value - hysteresis) < process variable < (set point + alarm value), the alarm output maintains the state of the previous scan. When (set point - alarm value) < process variable < (set point - alarm value + hysteresis), the alarm output maintains the state of the previous scan. The gray-colored portion is where the standby function operates.  Alarm value Alarm value Hysteresis Hysteresis ON Alarm action point Set point Alarm action point	Set point: 200.0°C Alarm value: 5.0°C Hysteresis: 2.0°C  5.0°C 5.0°C 2.0°C 2.0°C ON OFF 195.0°C 197.0°C 200.0°C 203.0°C 205.0°C When process variable ≥ 205.0°C, the alarm output is turned ON. When process variable ≤ 195.0°C, the alarm output is turned ON. When 197.0°C ≤ process variable ≤ 203.0°C, the alarm output is turned OFF.

Notes:

- Alarm action point
  This indicates the point where the alarm output changes from OFF to ON.
- Standby function
  This function does not immediately turn ON the alarm output when starting execution of the PIDA instruction, even when the process variable is in the alarm output ON range.
  The standby function is canceled when the process variable enters the alarm output OFF range, and when the process variable enters the alarm output ON range, the alarm output will turn ON.
  If the set point is changed, the standby function will be enabled again.



11. Alarm Value

Set the value that will be the trigger condition for Alarm Type (10). The value to set differs by the alarm type.
The content for the alarm value is as follows.

Alarm Type	Alarm Value	Range
Upper Limit Alarm	Set the deviation from the set point.	When the process variable is current/voltage or a data register: -32,768 to 32,767*1 When the process variable is a thermocouple or resistance thermometer and the data type is set to Celsius or Fahrenheit: -3,276.8 to 3,276.7ºC/ºF*1
Lower Limit Alarm		
Upper/Lower Limit Alarm		When the process variable is current/voltage or a data register: 0 to 65,535*1 When the process variable is a thermocouple or resistance thermometer and the data type is set to Celsius or Fahrenheit: 0.0 to 6,553.5ºC/ºF*1
Upper/Lower Limit Range Alarm		
Process High Alarm	Set the value where the alarm will operate as an absolute value.	Minimum to maximum*1
Process Low Alarm		
Upper Limit Alarm with Standby	Set the deviation from the set point.	When the process variable is current/voltage or a data register: -32,768 to 32,767*1 When the process variable is a thermocouple or resistance thermometer and the data type is set to Celsius or Fahrenheit: -3,276.8 to 3,276.7ºC/ºF*1
Lower Limit Alarm with Standby		
Upper/Lower Limit Alarm with Standby	Set the deviation from the set point.	When the process variable is current/voltage or a data register: 0 to 65,535*1 When the process variable is a thermocouple or resistance thermometer and the data type is set to Celsius or Fahrenheit: 0.0 to 6,553.5ºC/ºF*1

*1 No alarm action when the alarm value is 0.

12. Hysteresis

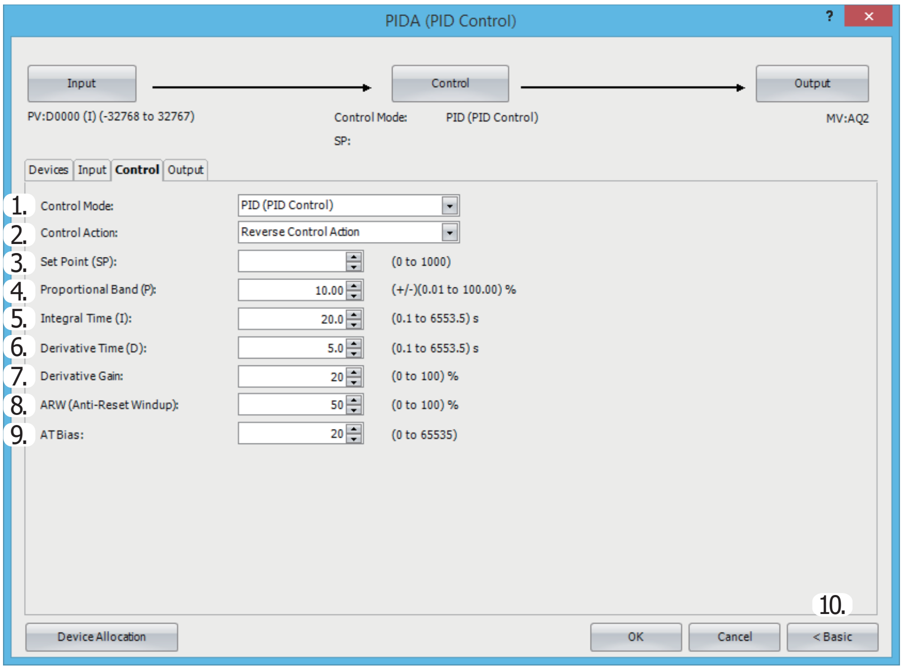
Set the range from the alarm action point to when the alarm output changes from ON to OFF.
When hysteresis is set to a large value, the alarm output will not change by slight variations. When hysteresis is set to a small value, the alarm output will change by slight variations near the alarm action point, which may negatively impact connected equipment.
The range is 0.1 to 6,553.5 (°C/°F) or 1 to 65,535 (voltage/current input).
However, when the alarm type is **Upper/Lower Limit Alarm** or **Upper/Lower Limit Alarm with Standby**, you must set hysteresis so that 1 ≤ hysteresis < Alarm Value (11) is satisfied.

13. Delay Time

Set the time until the alarm output is turned ON after the condition to turn the alarm output ON is satisfied.
The delay time prevents the alarm output from turning ON unintentionally when there are fluctuations in the process variable due to the effects of noise.
After the condition to turn the alarm output ON has been satisfied, if it is no longer satisfied by when the delay time elapses, the alarm output does not turn ON.
The range is 0 to 10,000 s. When 0, the delay time is disabled.

■Control tab

This tab configures the control parameters for the PIDA instruction.



PIDA (PID Control)

PV:D0000 (I) (-32768 to 32767)

Control Mode: PID (PID Control)

SP:

MV:AQ2

1. Control Mode: PID (PID Control)
2. Control Action: Reverse Control Action
3. Set Point (SP): (0 to 1000)
4. Proportional Band (P): 10.00 (+/-)(0.01 to 100.00) %
5. Integral Time (I): 20.0 (0.1 to 6553.5) s
6. Derivative Time (D): 5.0 (0.1 to 6553.5) s
7. Derivative Gain: 20 (0 to 100) %
8. ARW (Anti-Reset Windup): 50 (0 to 100) %
9. AT Bias: 20 (0 to 65535)

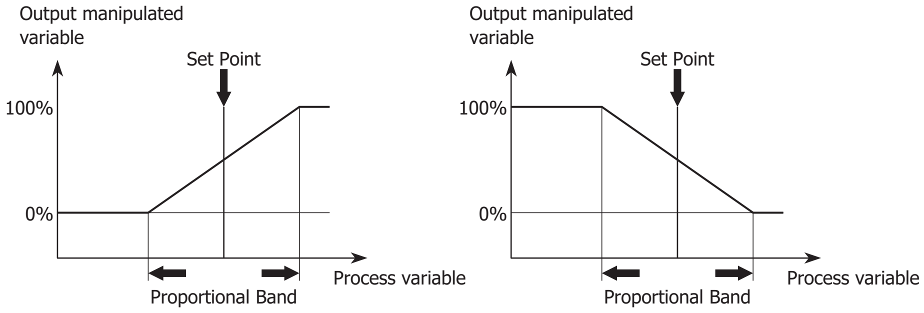
1. Control Mode (S1+6)

Select **PID (PID Control), P (Proportional Control), PI (PI Control)** or **PD (PD Control)** according to the characteristics of the target application.

For details about each control, see "PID Control" on page 19-23.

2. Control Action (S3+0)

- Select the control action from **Direct Control Action** or **Reverse Control Action**.
- Direct control action operates to increase the output manipulated variable when the process variable is higher than the set point. Cooling equipment uses this operation. Reverse control action operates to increase the output manipulated variable when the process variable is lower than the set point. Heating furnaces uses this operation. (S3+0) is read-only.



Direct control action

Reverse control action

3. Set Point (S1+7)

The set point can be set between the minimum value and the maximum value of the process variable. The setting unit for the set point is as follows.

When **Analog Input** thermocouple input or resistance thermometer input is set for the process variable, it can be set in units of 0.1°C/°F.

When **Data Register** or **Analog Input** voltage input, current input, thermistor, or resistance measurement is set for the process variable, it can be set in units of 1.

4. Proportional Band (S1+8)

The proportional action changes the output proportional to the deviation between the set point and process variable.
If the process variable is in the range of the proportional band, the control output (S3+6) turns on or in proportional to the deviation. If the process variable is out of the proportional band, the control output (S3+6) is always on or off.
When the proportional band is made larger, the control output (S3+6) turns on and off from a state where the deviation is large, so overshooting (process variable goes above the set point), undershooting (process variable falls below the set point), and hunting (process variable is in an unstable undulating state) decrease, but it takes time until the process variable reaches the set point and the offset between the set point and the process variable becomes larger.
When the proportional band is made smaller, the control output starts turning on and off at near the set point, so the time until the process variable reaches the set point decreases and the offset also becomes smaller, but hunting increases. If the proportional band is set extremely small, the control is the same as on/off operation.
When auto tuning is used, the appropriate proportional band can be automatically set for the target application. For details, see "Auto Tuning (AT)" on page 19-24.
The proportional band can be set from ±0.01 to ±100.00% in 0.01% increments.

5. Integral Time (S1+9)

With only the proportional action, an offset occurs between the set point and the process variable even when the target application is in a stable state. In order to bring this offset close to 0, the integral action is required. Integral time is a factor that determines the manipulated variable by the integral action. If the integral time is too short, the integral action becomes too large, which causes hunting of a long cycle. Conversely, if the integral time is too long, the processing until the set point is reached will take time. The integral execution range is - proportional band to + proportional band. If the process variable goes out of the proportional band due to a change in the set point or a disturbance, the integration calculation stops. As a result, the response of the output manipulated variable against the change in set point improves and undershoot and overshoot can be reduced.
When auto tuning is used, the appropriate integral time can be automatically set for the target application. For details, see "Auto Tuning (AT)" on page 19-24.
The integral time can be set from 0.1 to 6,553.5 seconds in 0.1 second increments.

6. Derivative Time (S1+10)

When the set point changes or when the difference between the set point and the process variable increases due to a disturbance, the derivative action quickly brings the process variable close to the set point by increasing the manipulated variable. When the derivative time is decreased, the derivative action becomes weaker and the response to rapid temperature changes becomes slower. The action to control rapid temperature increases also becomes weaker, so the time to reach the set point will become shorter, but overshoot will occur more easily. When the derivative time is increased, the derivative action becomes large and the response to rapid temperature changes becomes faster. The action to control rapid temperature increases also becomes more powerful, so the time to reach the set point will become longer, but overshoot will occur less easily.
When auto tuning is used, the appropriate derivative time can be automatically set for the target application. For details, see "Auto Tuning (AT)" on page 19-24.
The derivative time can be set from 0.1 to 6,553.5 seconds in 0.1 second increments.

7. Derivative Gain (S1+11)

When the derivative gain is set to a small value, the output manipulated variable is more easily affected by noise and changes in the set point. When the derivative gain is set to a large value, the output manipulated variable is less easily affected by noise and changes in the set point, but stability at normal times decreases. When noise or changes in the process variable occur, normally set this to 20 to 30%.
The derivative gain can be set from 0 to 100% in 1% increments.

8. ARW (Anti-Reset Windup) (S1+12)

Set the point to start the integral action. If the integral term is enabled from the start of execution of the PIDA instruction, overshoot may be caused. Overshoot can be controlled by delaying the integral action with ARW linked to the proportional band. ARW is normally 100%, which results in appropriate control with little overshoot. If ARW is set too small, overshoot can be eliminated, but an offset may occur.
The ARW can be set from 0 to 100%.
For example, when ARW=50% and proportional band=20.00%, the integral action starts from when the deviation becomes smaller than 10%.

9. AT (Auto Tuning) Bias (S1+13)

Set the bias value for the auto tuning.
The auto tuning start point is determined by the AT bias value in 1 increments.

Process Variable Value	AT Start Point
When process variable ≤ (set point - AT bias value)	Set point - AT bias value
When process variable ≥ (set point + AT bias value)	Set point + AT bias value
When (set point - AT bias value) < process variable < (set point + AT bias value)	Set point

For details, see "Auto Tuning (AT)" on page 19-24.

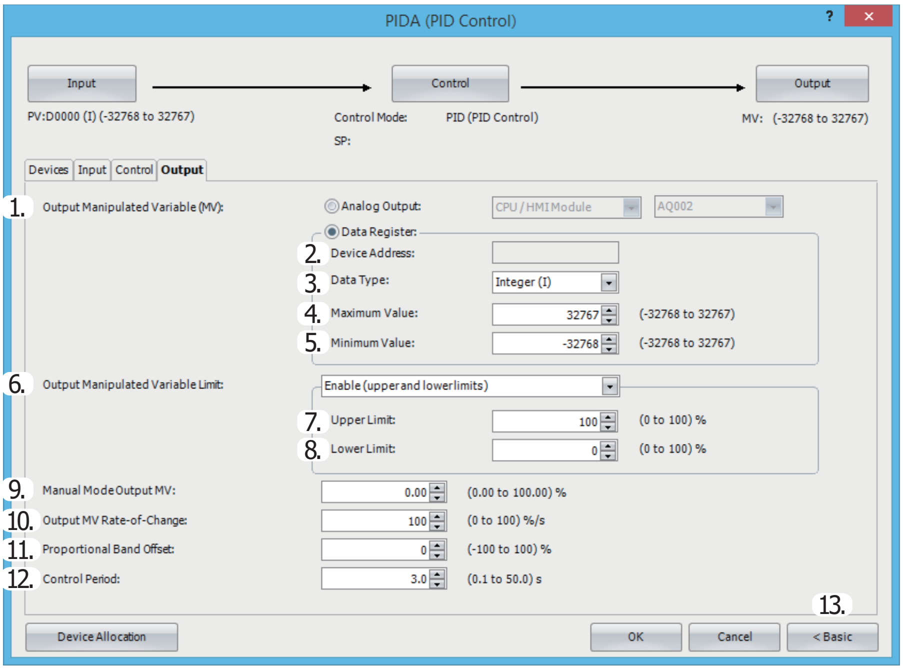
The AT bias can be set in the range of process variable minimum value < (set point - AT bias setting value) or process variable maximum value > (set point + AT bias setting value) in units of 1.

10. Advanced/Basic

Click this button to show or hide the advanced settings.
On the **Control** tab, Derivative Gain (7) can be set as an advanced setting.

■Output tab

This tab configures the output parameters for the PIDA instruction.



PV:D0000 (I) (-32768 to 32767)

Control Mode: PID (PID Control)

SP:

MV: (-32768 to 32767)

1. Output Manipulated Variable (MV): ○ Analog Output: CPU / HMI Module AQ002

● Data Register:

2. Device Address:

3. Data Type: Integer (I)

4. Maximum Value: 32767 (-32768 to 32767)

5. Minimum Value: -32768 (-32768 to 32767)

6. Output Manipulated Variable Limit: Enable (upper and lower limits)

7. Upper Limit: 100 (0 to 100) %

8. Lower Limit: 0 (0 to 100) %

9. Manual Mode Output MV: 0.00 (0.00 to 100.00) %

10. Output MV Rate-of-Change: 100 (0 to 100) %/s

11. Proportional Band Offset: 0 (-100 to 100) %

12. Control Period: 3.0 (0.1 to 50.0) s

1. Output Manipulated Variable (analog value) (S1+19)

Set the output for the PID control. The value can be selected as **Analog Output** or **Data Register**. The manipulated variable calculated with the PIDA instruction according to the output manipulated variable limit (6) setting is stored.

When Analog Output is specified

Configure the module with the analog output you want to configure as the output manipulated variable (analog value) destination and its analog number.

Analog Module Configure:	Select the analog output module. Select the module as the CPU module, HMI module, or expansion module 1 to 15.
Analog Output No.:	Select the output number on the analog output module. Select as AQ0 to AQ7.

- For auto mode, the analog value stored in the output manipulated variable (S1+19) undergoes linear conversion in the range of the maximum value and the minimum value of the analog output and is output.
- For manual mode, the value stored in the manual mode output manipulated variable (S1+17) undergoes linear conversion in the range of the maximum value and the minimum value of the analog output and is output.

For switching between auto/manual mode, see "Auto/manual mode (S3+1)" on page 19-21.
The value of the output manipulated variable is automatically stored in the special data register that corresponds to the analog output.
For allocating special data registers, see "Special Internal Relay Device Addresses" on page 2-4.

Note: To use an analog output, the analog output must be configured in advance.
For details on analog output settings, see Chapter 9 "Analog I/O Modules" in the "FC6A Series MICROSmart User's Manual".

When Data Register is specified

Specify the device to store the output manipulated variable.

- For auto mode, the output manipulated variable for PID control is stored in output manipulated variable (analog value) (S1+19) for the control register.
- It is stored in the range of minimum value (5) ≤ output manipulated variable ≤ maximum value (4).
- For manual mode, the stored value is that the one, stored in the manual
  mode output manipulated variable (S1+17), is converted linearly in the range of minimum value (5) and maximum value (4).

Configure parameters 2. Device Address through 5. Minimum Value when data register is selected as the process variable.

2. Device Address

Shows the control register (S1+19) configured on the devices tab when **Data Register** is specified as the output manipulated variable (1).

3. Data Type

Select the output manipulated variable (1) data type as W (word) or I (integer).

4. Maximum Value, 5. Minimum Value

When Analog Output is specified for Output Manipulated Variable

This setting is disabled.

When Data Register is specified for Output Manipulated Variable

A value that is the PIDA instruction output manipulated variable having undergone linear conversion in the range of maximum value (4) and minimum value (5) is stored in the set data register.

6. Output Manipulated Variable Limit

Set the output manipulated variable by restricting the manipulated variable calculated by the PIDA instruction.

When Disabled

The manipulated variable calculated by the PIDA instruction is the output manipulated variable.

When Enable (upper and lower limits)

When the manipulated variable calculated by the PIDA instruction exceeds the output manipulated variable upper limit or lower limit, the upper limit value or the lower limit value is set as the output manipulated variable.

When Enable (percentage)

The result of the manipulated variable calculated by the PIDA instruction multiplied by 1 to 99% is set as the output manipulated variable. The output manipulated variable lower limit (8) is disabled at this time.

7. Output Manipulated Variable Upper Limit/percentage (S1+14)

When Output manipulated variable restriction is Output manipulated variable (upper limit, lower limit)

Set the upper limit value of the output manipulated variable for the control output. When the manipulated variable calculated by the PIDA instruction is higher than or equal to the output manipulated variable upper limit value, the set upper limit value is set as the output manipulated variable for the control output. When setting the output manipulated variable upper limit value, you must set the output manipulated variable upper limit value so that it satisfies output manipulated variable lower limit value < output manipulated variable upper limit value.
The output manipulated variable upper limit value can be set from 0 to 100% in 1% increments.

When Enable (percentage)

Set the percentage of the output manipulated variable. The result of the manipulated variable calculated by the PIDA instruction multiplied by 1 to 99% is set as the output manipulated variable. The output manipulated variable (percentage) can be set from 1 to 99% in 1% increments.

Output manipulated variable = manipulated variable calculated by the PIDA instruction x output manipulated variable (scaling factor)

8. Output Manipulated Variable Lower Limit (S1+15)

When Enable (upper and lower limits)

Set the lower limit value of the output manipulated variable for the control output. When the manipulated variable calculated by the PIDA instruction is lower than or equal to the output manipulated variable lower limit value, the set lower limit value is set as the output manipulated variable for the control output. When setting the output manipulated variable lower limit value, you must set the output manipulated variable lower limit value so that it satisfies output manipulated variable lower limit value < output manipulated variable upper limit value. The output manipulated variable lower limit value can be set from 0 to 100% in 1% increments.

When Enable (percentage)

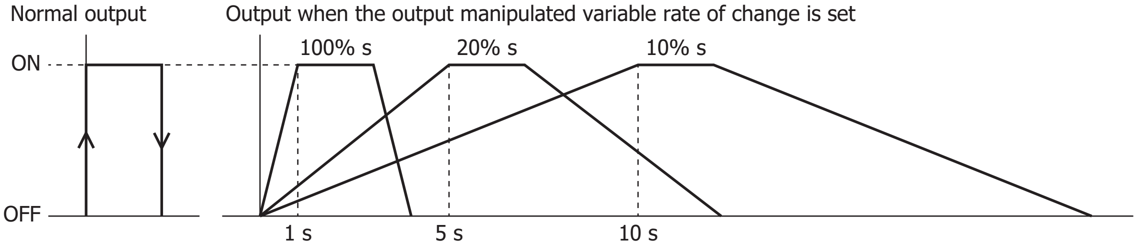
The output manipulated variable lower limit is disabled.

9. Manual Mode Output MV (S1+17)

Set the output manipulated variable for manual mode. To enable this function, Auto/manual mode (S3+1) must be turned on.
The manual mode output manipulated variable is enabled for the output manipulated variable, the output manipulated variable (digital value), and the control output.
For switching between auto/manual mode, see "Auto/manual mode (S3+1)" on page 19-21.
When the output manipulated variable limit is **Output manipulated variable (upper limit, lower limit)**, the manual mode output manipulated variable must be set between output manipulated variable lower limit value and output manipulated variable upper limit value. The manual mode output manipulated variable can be set from 0.00 to 100.00% in 0.01% increments.

10. Output MV Rate-of-Change (S1+16)

Set the output manipulated variable to change in a second. When the output manipulated variable rate of change is 0%/second, this function is disabled.
When the difference between the set point and process variable is large, the normal output switches from off to on as shown in the diagram on the next page, but by setting the output manipulated variable rate of change, the rate of change for the output manipulated variable can be changed as shown in the diagram on the next page.
The output manipulated variable rate of change can be set from 0 to 100%/second in 1% increments.


Normal output
Output when the output manipulated variable rate of change is set
ON
OFF
100% s
20% s
10% s
1 s
5 s
10 s

This function is suitable for the control of high-temperature heaters (elements composed of molybdenum, tungsten, platinum, used at approximately 1,500 to 1,800°C) that burn out when rapidly energized.

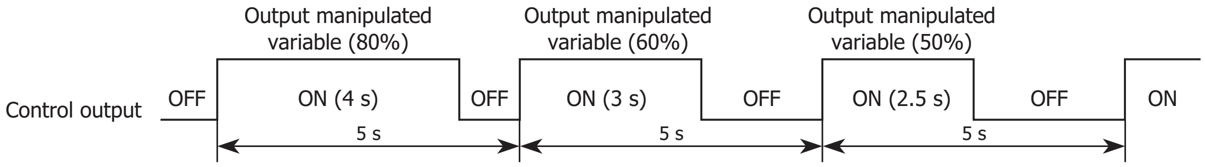
11. Proportional Band Offset (S1+20)

Sets the offset for the proportional band. The output manipulated variable (S1+1) can be increased or decreased by the value set as the proportional band offset.
For example, if the proportional band offset is set to 20%, the output manipulated variable (S1+1) is increased by 20%.
The proportional band offset can be set from -100 to 100%.

12. Control Period (S1+21)

Set the cycle to perform on/off control of the control (S3+5) output according to the output manipulated variable. The on pulse width for the control period varies according to the output manipulated variable.
The control period can be set from 0.1 to 50.0 seconds in 0.1 second increments.

When control period: 5 seconds (set value 50)


Output manipulated variable (80%)
Output manipulated variable (60%)
Output manipulated variable (50%)
Control output
OFF
ON (4 s)
OFF
ON (3 s)
OFF
ON (2.5 s)
OFF
ON
5 s
5 s
5 s

13. Advanced/Basic

Click this button and you can show or hide the advanced settings. On the **Output** tab, Output Manipulated Variable Limit (6), Output Manipulated Variable Upper Limit/Percentage (7), Output Manipulated Variable Lower Limit (8), Manual Mode Output MV (9), Output MV Rate-of-Change (10), Proportional Band Offset (11), and Control Period (12) can be set as advanced settings.

S1: Control Registers

Storage Destination	Function	Setting Details	R/W
S1+0	Process variable	• When **Analog Input** is specified AI0 to AI5 can be selected. S1+0 is read-only. • When **Data Register** is specified Store a value that satisfies process variable minimum value ≤ process variable ≤ process variable maximum value. S1+0 is readable/writable.	R/W
S1+1	Output manipulated variable	Store a value between 0 and 100 (0% to 100%).	R
S1+2	Operation status	Store the PIDA instruction execution status or error status. For details, see "Operation status (S1+2)" on page 19-19.	R
S1+3	Alarm 1 (process high alarm)	Set a value that satisfies alarm 2 value (process low alarm) < alarm 1 value ≤ process variable maximum. Operates as process variable maximum value when alarm 1 value ≤ alarm 2 value (process low alarm) or alarm 1 value > process variable maximum value.	R/W
S1+4	Alarm 2 (process low alarm)	Set a value that satisfies process variable minimum value ≤ alarm 2 value < alarm 1 value (process high alarm) Operates as process variable minimum value when alarm 2 value < process value minimum value or alarm 1 value (process high alarm) ≤ alarm 2 value.	R/W
S1+5	Sampling period	Set a value between 1 and 10,000 (0.01 to 100.00 seconds). Sampling period of 0 operates as 0.01 seconds and 10,001 or higher operates as 100.00 seconds.	R/W
S1+6	Control mode	Select the control mode. 0: PID control 1: Proportional control 2: PI control 3: PD control Operates as 0 (PID Control) when a value other than those listed above.	R/W
S1+7	Set point	Set a value that satisfies process variable minimum value ≤ set point ≤ process variable maximum value. When set point < process variable minimum value or process variable maximum value < set point, a set point configuration error occurs and the previous set point is retained.	R/W
S1+8	Proportional band	Set a value between 1 and 10,000 (±0.01 to ±100.00%) Proportional band of 0 operates as ±0.01% and 10,001 or higher operates as ±100.00%.	R/W

Storage Destination	Function	Setting Details	R/W
S1+9	Integral time	Set a value between 1 and 65,535 (0.1 to 6553.5 seconds). When integral time is 0, integral time is disabled.	R/W
S1+10	Derivative time	Set a value between 1 and 65,535 (0.1 to 6553.5 seconds). When derivative time is 0, derivative time is disabled.	R/W
S1+11	Derivative gain	Set a value between 0 and 100 (0 to 100%) Derivative gain of 0 operates as 0% and 101 or higher operates as 100%.	R/W
S1+12	ARW	Set a value between 0 and 100 (0 to 100%) ARW of 101 or higher operates as 100%.	R/W
S1+13	AT bias	Set a value that satisfies process variable minimum value < (set point - AT bias setting value) and process variable maximum value > (set point + AT bias setting value). When a value is set that does not satisfy both conditions listed above, 133 is stored in the operation status (S1+2).	R/W
S1+14	Output manipulated variable upper limit/percentage	• When **Enable (upper and lower limits)** Set a value between 0 and 100 (0 to 100%) Operates as 100 for a value other than 0 to 100. • When **Enable (percentage)** Set a value between 10001 and 10099 (1 to 99%) Operates as the "Output manipulated variable upper limit" for a value other than 10001 to 10099. Output manipulated variable = manipulated variable calculated by the PIDA instruction x scaling factor	R/W
S1+15	Output manipulated variable lower limit	Set a value between 0 and 100 (0 to 100%) Operates as 0 for a value other than 0 to 100. Set a value that satisfies Output manipulated variable lower limit value < Output manipulated variable upper limit value.	R/W
S1+16	Output manipulated variable rate of change	Set a value between 0 and 100 (0 to 100%/second) If not 0 to 100, output manipulated variable rate of change is disabled.	R/W
S1+17	Manual mode output manipulated variable	Set a value between 0 and 10000 (0.00 to 100.00%). Set a value that satisfies output manipulated variable lower limit value ≤ manual mode output manipulated variable ≤ output manipulated variable upper limit value. When the output manipulated variable limit enable (S3+2) is ON, this operates as the output manipulated variable lower limit value for a value other than one between the output manipulated variable lower limit value and the output manipulated variable upper limit value.	R/W
S1+18	Output manipulated variable (% value)	Output manipulated variable is stored as a value between -32,768 and +32,767 (-327.68 to +327.67%). The output manipulated variable (% value) is unknown while auto tuning is executing. For manual mode, the manual mode output manipulated variable value is not stored.	R
S1+19	Output manipulated variable (Analog value)	• When **Analog Output** is specified for the output manipulated variable The S1+1 output manipulated variable (%) undergoes linear conversion in the range of the minimum and maximum values of the analog output, and that value is stored. • When **Data Register** is specified for the output manipulated variable The output manipulated variable (%) stored in S1+1 is linearly converted with the minimum and maximum output values, and the converted value is stored in S1+19.	R
S1+20	Proportional band offset	Set a value between -100 and 100 (-100 to 100%) Proportional band offset of -101 or lower operates as -100% and 101 or higher operates as 100%.	R/W
S1+21	Control period	Set a value between 1 and 500 (0.1 to 50.0 seconds) Control period of 0 operates as 0.1 seconds and 501 or higher operates as 50.0 seconds.	R/W

Storage Destination	Function	Setting Details	R/W
S1+22	Alarm 3 action	Set the Alarm 3 action type. 0: No Alarm Action 1: Upper Limit Alarm 2: Lower Limit Alarm 3: Upper/Lower Limit Alarm 4: Upper/Lower Limit Range Alarm 5: Process High Alarm 6: Process Low Alarm 7: Upper Limit Alarm with Standby 8: Lower Limit Alarm with Standby 9: Upper/Lower Limit Alarm with Standby Operates as 0 (No Alarm Action) for value other than those listed above.	R/W
S1+23	Alarm 3 value	Set the value that will be the trigger condition for the Alarm 3 action. For details, see "Alarm 3 value (S1+23)" on page 19-20.	R/W
S1+24	Alarm 3 hysteresis	Set the range from the alarm action point to when the alarm output changes from ON to OFF. Set a value that satisfies the following conditions in the range of 1 to 65,535 (0.1 to 6,553.5 (°C/°F), 1 to 65,535 (voltage/current input)). • When Alarm 3 action is set to **Upper/Lower Limit Alarm** or **Upper/Lower Limit Alarm with Standby** 1 ≤ Alarm 3 hysteresis < Alarm 3 value • When Alarm 3 action is not set to **Upper/Lower Limit Alarm** or **Upper/Lower Limit Alarm with Standby** 1 ≤ Alarm 3 hysteresis ≤ 65,535 When 0, hysteresis operates as 1 (0.1 (°C/°F) or 1 (voltage/current input)).	R/W
S1+25	Alarm 3 delay time	Set the time until the alarm output is turned ON after the condition to turn the alarm output ON is satisfied. Set as 0 (delay time disabled) or in the range of 1 to 10,000 (1 to 10,000 seconds). When 10,001 or higher, delay time operates as 10,000 seconds.	R/W
S1+26	Alarm 4 action	Set the Alarm 4 action type. The setting details are the same as Alarm 3 action (S1+22).	R/W
S1+27	Alarm 4 value	Set the value that will be the trigger condition for the Alarm 4 action. The setting details are the same as Alarm 3 value (S1+23).	R/W
S1+28	Alarm 4 hysteresis	Set the range from the alarm action point to when the alarm output changes from ON to OFF. The setting details are the same as Alarm 3 hysteresis (S1+24).	R/W
S1+29	Alarm 4 delay time	Set the time until the alarm output is turned ON after the condition to turn the alarm output ON is satisfied. The setting details are the same as Alarm 3 delay time (S1+25).	R/W
S1+30	Alarm 5 action	Set the Alarm 5 action type. The setting details are the same as Alarm 3 action (S1+22).	R/W
S1+31	Alarm 5 value	Set the value that will be the trigger condition for the Alarm 5 action. The setting details are the same as Alarm 3 value (S1+23).	R/W
S1+32	Alarm 5 hysteresis	Set the range from the alarm action point to when the alarm output changes from ON to OFF. The setting details are the same as Alarm 3 hysteresis (S1+24).	R/W
S1+33	Alarm 5 delay time	Set the time until the alarm output is turned ON after the condition to turn the alarm output ON is satisfied. The setting details are the same as Alarm 3 delay time (S1+25).	R/W
S1+34	Alarm 6 action	Set the Alarm 6 action type. The setting details are the same as Alarm 3 action (S1+22).	R/W
S1+35	Alarm 6 value	Set the value that will be the trigger condition for the Alarm 6 action. The setting details are the same as Alarm 3 value (S1+23).	R/W
S1+36	Alarm 6 hysteresis	Set the range from the alarm action point to when the alarm output changes from ON to OFF. The setting details are the same as Alarm 3 hysteresis (S1+24).	R/W
S1+37	Alarm 6 delay time	Set the time until the alarm output is turned ON after the condition to turn the alarm output ON is satisfied. The setting details are the same as Alarm 3 delay time (S1+25).	R/W
S1+38 **S1+39**	— Reserved —	—	—

Operation status (S1+2)

Indicates the PIDA instruction execution status or error status. It stores the status code.

- X in the table indicates the elapsed time from the start of AT until the set point is reached. X is incremented by 1 every 10 minutes. 9 is stored if the elapsed time is equal to or longer than 90 minutes.
- x in the table indicates the elapsed time from the start of PID instruction until the set point is reached. x is incremented by 1 every 10 minutes. 9 is stored if the elapsed time is equal to or longer than 90 minutes.
- If an error code (status code value is equal to or greater than 100 and less than 120) is set for the operation status, execution for PID control is stopped. Set the correct parameters and then turn on the PIDA instruction input once again.

Status Code	Status Description	Status Classification
1x	Auto tuning in progress	AT is normally executed
2x	Auto tuning ended	
5x	P/PI/PID control in progress	PID control is normally executed
6x	Set point is reached (Changes from 5x to 6x onace the set point is reached.)	
103	Output manipulated variable restriction upper limit value < output manipulated variable restriction lower limit value was set.	PID control execution stop
104	— Reserved —	—
106	A value was set for the set point that does not satisfy process variable minimum value ≤ set point ≤ process variable maximum value.	PID control execution stop
109	Process variable > process variable maximum value or process variable < process variable minimum value was set.	PID control execution stop
120	The control mode was set to a value other than 0 to 3.	PID control execution continues
121	The sampling period was set to 0 or 10,001 or greater.	PID control execution continues
122	The proportional band was set to 0 or a value of 10,001 or greater.	PID control execution continues
123	The derivative gain was set to 0 or a value of 101 or greater.	PID control execution continues
124	The integral start factor (ARW) was set to a value of 101 or greater.	PID control execution continues
125	A value was set for the alarm 1 value (process high alarm) that does not satisfy alarm 2 value (process low alarm) < alarm 1 value (process high alarm) ≤ process variable maximum value.	PID control execution continues
126	A value was set for the alarm 2 value (process low alarm) that does not satisfy process variable minimum value ≤ alarm 2 value (process low alarm) < alarm 1 value (process high alarm).	PID control execution continues
127	The output manipulated variable restriction upper limit value was set to a value between 101 to 10,000 or 10,100 to 65,535.	PID control execution continues
128	The output manipulated variable restriction lower limit value was set to a value of 101 or higher. PID execution continues	PID control execution continues
129	The output manipulated variable rate of change was set to a value of 101 or higher.	PID control execution continues
130	The manual mode output manipulated variable was set to a value that does not satisfy output manipulated variable restriction lower limit value ≤ manual mode output manipulated variable ≤ output manipulated variable restriction upper limit value.	PID control execution continues
131	The proportional band offset value was set to a value that is not between -100 to 100.	PID control execution continues
132	The control cycle was set to 0 or a value of 501 or greater.	PID control execution continues
133	A value was set that does not satisfy process variable minimum value < (set point - AT bias setting value) and process variable maximum value > (set point + AT bias setting value).	PID control execution continues
134	The PIDA instruction is executed, and from the second and subsequent scans, a value is set for the set point that does not satisfy minimum value ≤ set point ≤ maximum value.	PID control execution continues (Maintains the set point in the normal range up to the previous scan.)
135 to 199	— Reserved —	—
200	The alarm type for Alarm 3 was set to a value of 10 or higher.	PID control execution continues
201	The alarm type for Alarm 4 was set to a value of 10 or higher.	PID control execution continues
202	The alarm type for Alarm 5 was set to a value of 10 or higher.	PID control execution continues
203	The alarm type for Alarm 6 was set to a value of 10 or higher.	PID control execution continues
204 to 209	— Reserved —	—
210	The action point for Alarm 3 with the set alarm value exceeded the process variable minimum value or maximum value.	PID control execution continues
211	The action point for Alarm 4 with the set alarm value exceeded the process variable minimum value or maximum value.	PID control execution continues

Status Code	Status Description	Status Classification
212	The action point for Alarm 5 with the set alarm value exceeded the process variable minimum value or maximum value.	PID control control execution continues
213	The action point for Alarm 6 with the set alarm value exceeded the process variable minimum value or maximum value.	PID control execution continues
214 to 219	— Reserved —	—
220	The action point for Alarm 3 with the set hysteresis exceeded the process variable minimum value or maximum value.	PID control execution continues
221	The action point for Alarm 4 with the set hysteresis exceeded the process variable minimum value or maximum value.	PID control execution continues
222	The action point for Alarm 5 with the set hysteresis exceeded the process variable minimum value or maximum value.	PID control execution continues
223	The action point for Alarm 6 with the set hysteresis exceeded the process variable minimum value or maximum value.	PID control execution continues
224 to 229	— Reserved —	—
230	The Alarm 3 delay time was set to a value of 10,001 or higher.	PID control execution continues
231	The Alarm 4 delay time was set to a value of 10,001 or higher.	PID control execution continues
232	The Alarm 5 delay time was set to a value of 10,001 or higher.	PID control execution continues
233	The Alarm 6 delay time was set to a value of 10,001 or higher.	PID control execution continues
234 to 239	— Reserved —	—
240	When the Alarm 3 alarm type was set to the upper/lower limit alarm or the upper/lower limit alarm with standby, the Alarm 3 hysteresis was set to a value that does not satisfy hysteresis < alarm value.	PID control execution continues
241	When the Alarm 4 alarm type was set to the upper/lower limit alarm or the upper/lower limit alarm with standby, the Alarm 4 hysteresis was set to a value that does not satisfy hysteresis < alarm value.	PID control execution continues
242	When the Alarm 5 alarm type was set to the upper/lower limit alarm or the upper/lower limit alarm with standby, the Alarm 5 hysteresis was set to a value that does not satisfy hysteresis < alarm value.	PID control execution continues
243	When the Alarm 6 alarm type was set to the upper/lower limit alarm or the upper/lower limit alarm with standby, the Alarm 6 hysteresis was set to a value that does not satisfy hysteresis < alarm value.	PID control execution continues
244 to 65535	— Reserved —	—

Alarm 3 value (S1+23)

Set the value that will be the trigger condition for the alarm. The value to set differs by the alarm type.
The content for the alarm value is as follows.

Alarm Type	Alarm Value	Range
Upper Limit Alarm	Set the deviation value from the set point.	-32,768 to 32,767[*1]
Lower Limit Alarm		
Upper/Lower Limit Alarm		0 to 65,535[*1]
Upper/Lower Limit Range Alarm		
Process High Alarm	Set the value where the alarm will operate as an absolute value.	Minimum to maximum
Process Low Alarm		
Upper Limit Alarm with Standby	Set the deviation value from the set point.	-32,768 to 32,767[*1]
Lower Limit Alarm with Standby		
Upper/Lower Limit Alarm with Standby		0 to 65,535[*1]

[*1] No alarm action when the alarm value is 0.

S2: Initialization Input

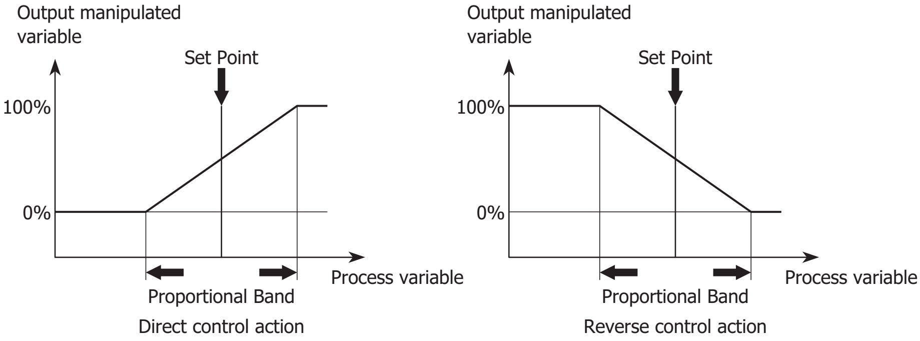
When the initialize input is on, the control registers (data registers) are overwritten with the values configured on the WindLDR input tab, control tab, and output tab. If you want the initialization to be performed just one time, please use a SOTU (shot up) or a SOTD (shot down) instruction.

S3: Control Relay

Device	Function	Setting Details	R/W
S3+0	Control action	0 (OFF): Reverse control action 1 (ON): Direct control action	R
S3+1	Auto/manual mode	0 (OFF): Auto 1 (ON): Manual	R/W
S3+2	Output manipulated variable limit enable	0 (OFF): Disabled 1 (ON): Enabled (operates with S1+14, 15 settings)	R/W
S3+3	Alarm 1 output	Turns on when process variable (S1+0) ≥ alarm 1 value (process high alarm) (S1+3).	R
S3+4	Alarm 2 output	Turns on when process variable (S1+0) ≤ alarm 2 value (process low alarm) (S1+4).	R
S3+5	Control output	Turns on and off according to the control period and output manipulated variable.	R
S3+6	AT execution	Starts auto tuning when execute auto tuning is turned on. Stops auto tuning if execute auto tuning is turned off while auto tuning is executed.	R/W
S3+7	AT complete output	Turns on after auto tuning ends	R
S3+8	Alarm 3 output	Turns ON when the process variable (S1+0) is outside the Alarm 3 action (S1+22) range. Turns OFF when the process variable (S1+0) is inside the Alarm 3 action (S1+22) range.	R
S3+9	Alarm 4 output	Turns ON when the process variable (S1+0) is outside the Alarm 4 action (S1+26) range. Turns OFF when the process variable (S1+0) is inside the Alarm 4 action (S1+26) range.	R
S3+10	Alarm 5 output	Turns ON when the process variable (S1+0) is outside the Alarm 5 action (S1+30) range. Turns OFF when the process variable (S1+0) is inside the Alarm 5 action (S1+30) range.	R
S3+11	Alarm 6 output	Turns ON when the process variable (S1+0) is outside the Alarm 6 action (S1+34) range. Turns OFF when the process variable (S1+0) is inside the Alarm 6 action (S1+34) range.	R
S3+12 to S3+15	— Reserved —	—	—

Operation action (S3+0)

In direct control action, PID instruction operates to increase the output manipulated variable when the process variable is higher than the set point. Cooling equipment uses this operation. In reverse control action, PID instruction operates to increase the output manipulated variable when the process variable is lower than the set point. Heating furnaces uses this operation.



Auto/manual mode (S3+1)

In auto mode, PIDA instruction controls the output according to the manipulated variable calculated by the PID instruction. In manual mode, PIDA instruction controls the output with the manual mode output manipulated variable (S1+17). To use manual mode, you must set the manual mode output manipulated variable in advance.

Output manipulated variable limit enable (S3+2)

You can enable or disable the output manipulated variable restriction using the output manipulated variable upper limit value (S1+14) and the output manipulated variable lower limit value (S1+15).

Alarm 1 output (S3+3)

Turns on if the process variable (S1+0) is higher than or equal to the alarm 1 value (process high alarm) (S1+3). This is read-only data.

Alarm 2 output (S3+4)

Turns on if the process variable (S1+0) is lower than or equal to the alarm 2 value (process low alarm) (S1+4). This is read-only data.

Control output (S3+5)

When PIDA instruction is in auto mode, the control output turns on and off according to the manipulated variable calculated by the PIDA instruction and the control period (S1+21). When PIDA instruction is in manual mode, the control output turns on and off according to the manual mode output manipulated variable (S1+17) and the control period.

AT execution (S3+6)

When the execute auto tuning is turned on, auto tuning is initiated. Once auto tuning is finished, it is turned off automatically. The auto tuning is terminated if it is turned off while auto tuning is executed.

AT complete output (S3+7)

The auto tuning completion output is turned on when the auto tuning is successfully completed.

Alarm 3 output (S3+8)

Turns ON when the process variable (S1+0) is outside the Alarm 3 action (S1+22) range.
Turns OFF when the process variable (S1+0) is inside the Alarm 3 action (S1+22) range. This is read-only data.

Alarm 4 output (S3+9)

Turns ON when the process variable (S1+0) is outside the Alarm 4 action (S1+26) range.
Turns OFF when the process variable (S1+0) is inside the Alarm 4 action (S1+26) range. This is read-only data.

Alarm 5 output (S3+10)

Turns ON when the process variable (S1+0) is outside the Alarm 5 action (S1+30) range.
Turns OFF when the process variable (S1+0) is inside the Alarm 5 action (S1+30) range. This is read-only data.

Alarm 6 output (S3+11)

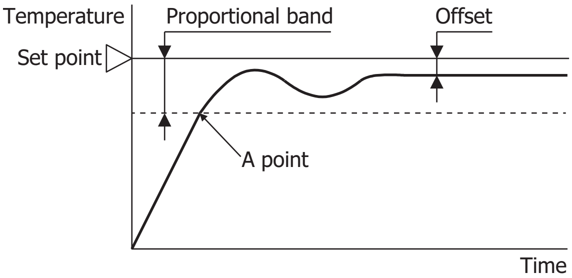
Turns ON when the process variable (S1+0) is outside the Alarm 6 action (S1+34) range.
Turns OFF when the process variable (S1+0) is inside the Alarm 6 action (S1+34) range. This is read-only data.

PID Control

PID control performs adjustment operations to cancel the deviation between a single set point and the process variable that is present in normal temperature control. The types of PID control that can be used with the FC6A Series MICROSmart are as follows.

■Proportional control

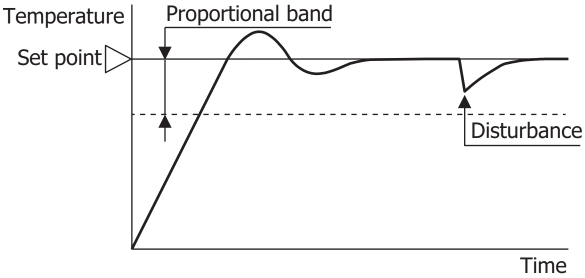
Proportional control outputs the manipulated variable in the proportional band that is proportional to the deviation between the set point and the process variable. The output is turned on until the process variable reaches the A point (the proportional band), and when it exceeds this (it enters the proportional band), the control output starts to be turned on and off at the control cycle, and then when the set point is exceeded, the control output is turned completely off. Along with the rise in temperature from the A point to the set point, the control output on time becomes shorter and the off time becomes longer. An offset always occurs with Proportional control. The P action is suitable for processes with no wasted time such as gas pressure control and level control. When set to Proportional control, integral time and derivative time are automatically ignored.



- When the proportional band is made smaller, the control output turns on or off from near the set point, so the time until the process variable rises in temperature to the set point decreases and the offset also becomes smaller, but hunting increases.
- When the proportional band is made larger, the control output turns on or off from a temperature considerably lower than the target temperature, so auto tuning and hunting eliminate, but it takes time until the process variable rises in temperature to the set point and the offset between the set point and the process variable becomes larger.

■PI control

The I action automatically corrects the offset generated by Proportional control. However, for rapid changes in temperature due to disturbances, it will take time until the temperature stabilizes. The PI action is suitable for temperature control with a slow rate of change. When set to PI control, derivative time is automatically 0.

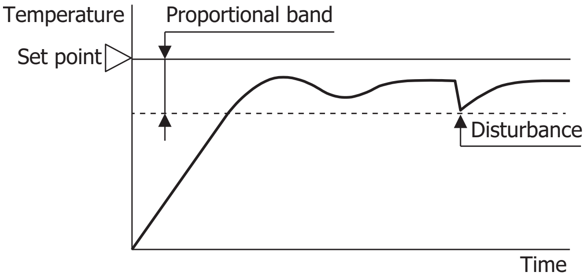


- When the integral time is decreased, I (integral) control becomes more powerful and the offset can be corrected in a short amount of time, but this may cause long cycles of hunting.
- When the integral time is increased, I (integral) control becomes weaker and it takes time to correct the offset.

■PD control

Compared to Proportional control, PD control increases the response to rapid temperature changes due to disturbances, stabilizes control in a short amount of time, and attempts to improve the transient response characteristics. PD control is suitable to temperature control with a fast rate of change.

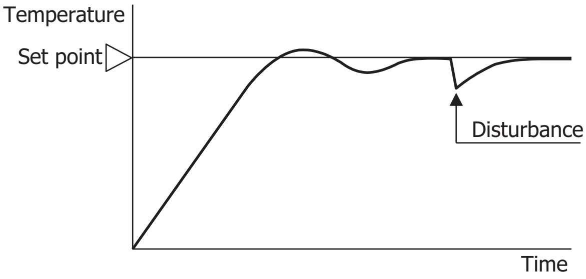
When set to PD control, integral time is automatically 0.



- When the derivative time is decreased, the D (derivative) control becomes weaker and the response to rapid temperature changes becomes slower. The action to control rapid temperature increases also becomes weaker, so the rise in temperature time to the set point will become faster, but overshooting the set point will occur more easily.
- When the derivative time is increased, the D (derivative) control becomes more powerful and the response to rapid temperature changes becomes faster. The action to control rapid temperature increases also becomes more powerful, so the rise in temperature time to the set point will become slower, but overshooting the set point will occur less easily.

■PID control

PID controls overshooting and hunting in Proportional control, corrects the offset in I (integral) control, and makes rapid temperature changes due to disturbances converge on the set point in a short amount of time in D (derivative) control. Ideal temperature control can be performed by using PID control. The parameters for the proportional band, integral time, and derivative time in PID control can be automatically set with auto tuning.



Auto Tuning (AT)

Optimal temperature control parameters will differ according to the characteristics of the subject to control. When performing PID control, the proportional band (P), integral time (I), and derivative time (D) can be automatically set by executing auto tuning.

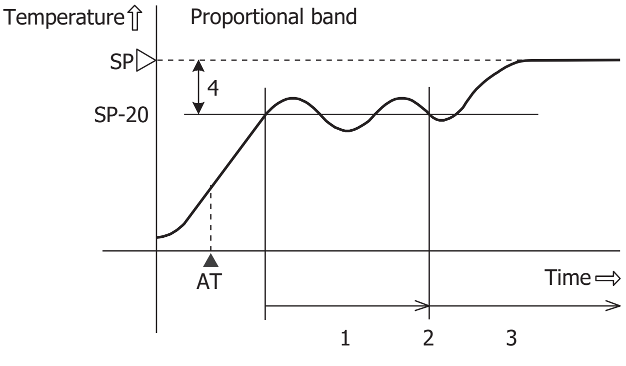
- Perform auto tuning during test operation.
- When auto tuning is executed near normal temperatures, temperature fluctuations cannot be imposed, so auto tuning may not end normally. In such cases, set the values for P, I, and D manually.
- When auto turning is executed once, so long as the process is not changed, it is not necessary to execute auto tuning again.

■Auto tuning (AT)

In order to automatically set the proportional band (P), integral time (I), and derivative time (D), fluctuations must be forcibly imposed on the subject to control to set each value to the optimal value. In order to perform optimal auto tuning, fluctuations must be imposed when the process variable has reached a value near the set point. Fluctuations can be imposed when the process variable is near the set point by setting the AT bias. The relationship between the set point, AT bias, auto tuning start point, and fluctuation start point is as follows.

Process variable (PV) ≤ set point (SP) - AT bias setting value

When the AT bias setting is set to 20°C, fluctuations will start when the process variable (PV) reaches a temperature 20°C lower than the set point (SP).

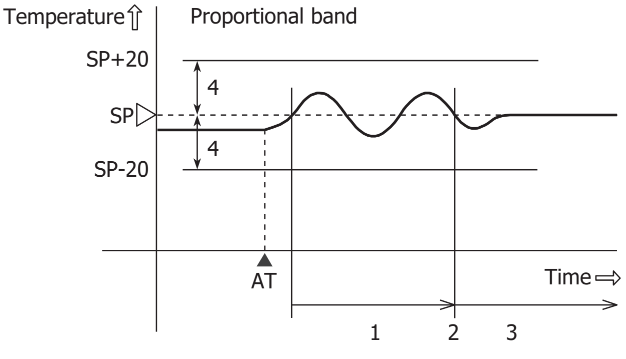


1. Measuring PID constants
2. Calculate PID constants
3. Control with the PID constants set with AT
4. AT bias setting value (20)

▲ AT: When the AT execution bit is turned on

Set point (SP) - AT bias setting value < process variable (PV) < set point (SP) + AT bias setting value

Fluctuations will start when the process variable (PV) reaches the set point (SP).

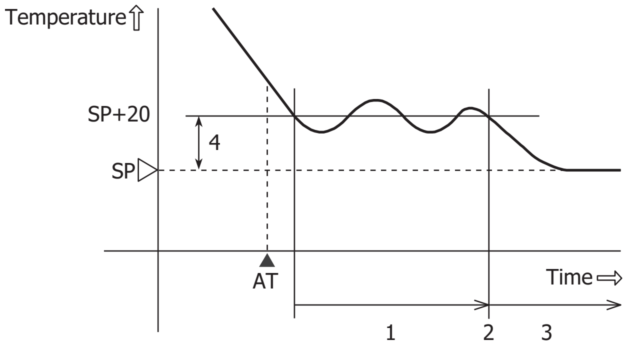


1. Measuring PID constants
2. Calculate PID constants
3. Control with the PID constants set with AT
4. AT bias setting value (20)

▲ AT: When the AT execution bit is turned on

Process variable (PV) ≥ set point (SP) + AT bias setting value

When the AT bias setting is set to 20°C, fluctuations will start when the process variable (PV) reaches a temperature 20°C higher than the set point (SP).



1. Measuring PID constants
2. Calculate PID constants
3. Control with the PID constants set with AT
4. AT bias setting value (20)

▲ AT: When the AT execution bit is turned on

■To execute auto tuning (AT)

To execute auto tuning, turn on the auto tuning execution flag (S3+6) when the PIDA instruction input is on. The values for P, I, and D will be automatically calculated. While auto tuning is executing, 1 is stored in the operation status (S1+2).

When auto tuning ends, the auto tuning execution flag (S3+6) turns off automatically, 2 is stored in the operation status (S1+2), and the auto tuning completed output (S3+7) turns on.

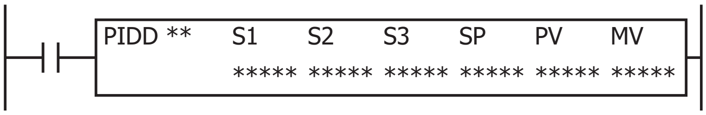
■To cancel auto tuning (AT) execution

To cancel auto tuning while it is executing, turn off the auto tuning execution flag (S3+6). When the auto tuning execution flag (S3+6) is turned off, auto tuning stops, 2 is stored in the operation status (S1+2), and the auto tuning completed output (S3+7) turns on.

If auto tuning is canceled without completing, the values for P, I, and D are restored to the values before auto tuning was executed.

PIDD (PID with Derivative Decay)

Executes PID control and outputs that result. The PID control parameters stored in the control registers are handled as data type F (float), and fine adjustments can be made to the control parameters such as proportional gain, integral gain, and derivative gain. Multiple PIDD instructions can also be combined to execute the cascade control.



Operation

When the input is on, PID control is performed.

Warning Knowledge of PID control is required to use the PIDD instruction. If this instruction is used without an understanding of PID control, there is a possibility of control that was not intended by the user. Please use the PIDD instruction with a full understanding of PID control and the PIDD instruction.
When performing feedback control using the PIDD instruction, emergency stop and interlocking circuits must be configured outside the FC6A Series MICROSmart according to the user application. If these are configured in the internal circuit and the process variable is not input correctly, there is a risk of no longer being able to perform proper feedback control, which may damage connected devices or cause accidents.

Caution The PIDD instruction cannot be used in an interrupt program. If the instruction is used in an interrupt program, a user program execution error occurs, the execution of the instruction is canceled, and the next instruction is executed.
For user program execution errors, see "User Program Execution Errors" on page 3-10.

Note: A maximum of 32 PIDD instructions can be used in a user program.

Valid Devices

The control register uses 100 words, and the control relay uses 32 internal relays.

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Control registers	—	—	—	—	—	—	X*1	—	—	—
S2 (Source 2)	Initialization input	X	—	X	—	—	—	—	—	—	—
S3 (Source 3)	Control relays	—	—	X*2	—	—	—	—	—	—	—

*1 Special data registers cannot be used.
*2 Special internal relays cannot be used. Only 0 can be specified as the first digit of the internal relay number. 1 to 7 cannot be specified.

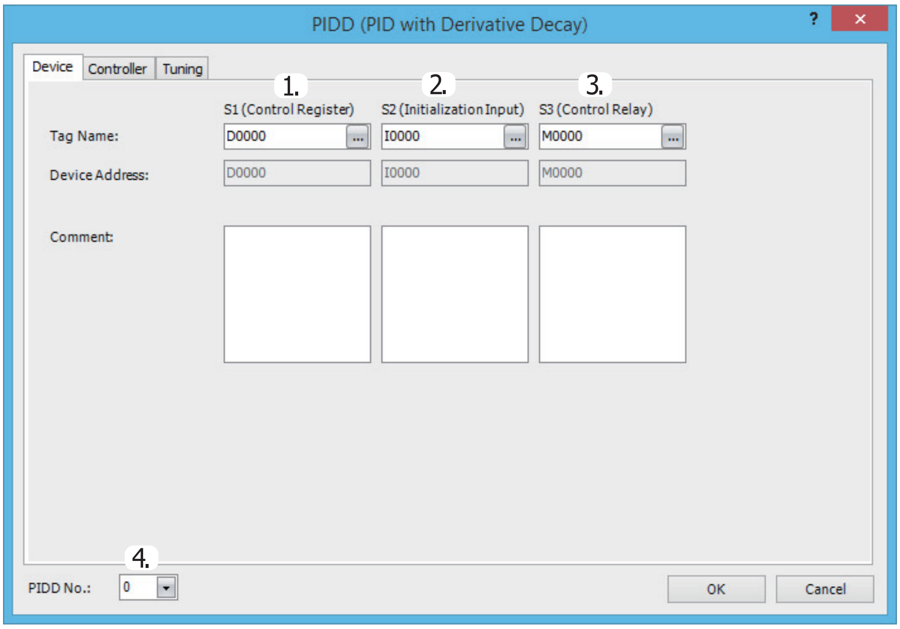
Settings

The **PIDD (PID with Derivative Decay)** dialog box contains the **Device** tab, **Controller** tab, and **Tuning** tab.

The devices used with the PIDD instruction are configured in the **Device** tab.

The initial values of the PIDD instruction parameters are configured in **Controller** and **Tuning** tabs.

■Device tab



PIDD (PID with Derivative Decay)

Device | Controller | Tuning

	1. S1 (Control Register)	2. S2 (Initialization Input)	3. S3 (Control Relay)
Tag Name:	D0000	I0000	M0000
Device Address:	D0000	I0000	M0000
Comment:			

4. PIDD No.: 0

1. S1 (source 1): Control Register

Specify the first device of the data register range that stores the PIDD instruction control parameters.

Starting from the specified data register, 100 words of data registers are used. The devices that can be specified as data registers are data registers D0000 to D7900 and D10000 to D61900.

When the PIDD instruction initialization input is turned on, the control registers are initialized with the values set on the **Controller** and **Tuning** tabs.

For the control registers, see "S1: Control Registers" on page 19-34.

2. S2 (source 2): Initialization Input

Specify the device that initializes the PIDD instruction control registers and control relays.

The device that can be specified as the initialization input is an external input or internal relay.

The parameters set on the **Controller** and **Tuning** tabs in the PIDD instruction dialog box are the initial values of the PIDD instruction control registers and control relays. These initial values are downloaded to the PLC as the user program and stored in the PLC's ROM. When the initialization input is turned on, the initial values of the PIDD instruction in ROM are stored in the control registers and control relays.

When the initialization input is on, the initial values are stored with each scan. To only initialize the values one time, use the initialization input in combination with the SOUT (single output up) instruction or the SOTD (single output down) instruction.

3. S3 (source 3): Control Relay

Specify the bit devices that will control PID control and store the status of PIDD instructions.

Starting from the specified internal relay, 32 internal relays are used. The devices that can be specified are internal relays M0 to M7960 and M10000 to M21210. Special internal relays cannot be specified.

The role of each control relay bit is different. The operation of the PIDD instruction, such as direct control action/reverse control action, derivative action enabled/disabled, self tuning enabled/disabled, and auto mode/manual mode/cascade control mode, can be toggled by turning the corresponding bit on and off.

For the control relays, see "S3: Control Relay" on page 19-21.

4. PIDD No.

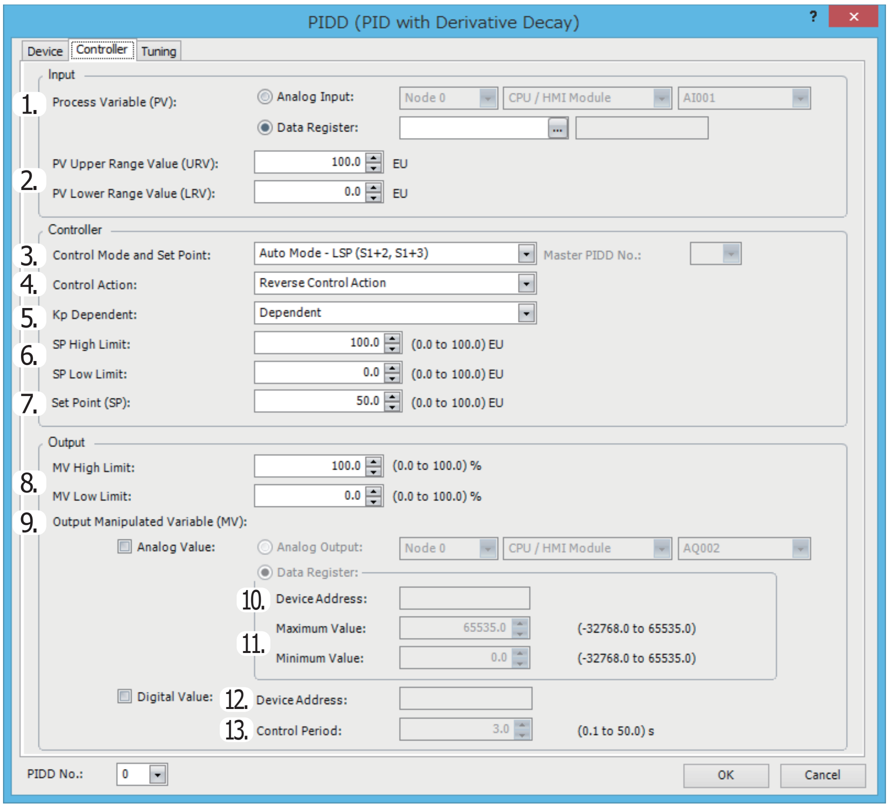
Assign a unique number from 0 to 31 to identify each PIDD instruction.

When performing the cascade control, specify the master PIDD instruction by its PIDD number.

■Controller tab

This tab is used to configure the control parameters of the PIDD instruction.

To store the initial settings of the PIDD instruction that are set on the **Controller** tab in the control registers and control relays, turn on the initialization input for the corresponding PIDD instruction after the user program is downloaded to the FC6A Series MICROSmart.



PIDD (PID with Derivative Decay)

Device | Controller | Tuning

Input

1. Process Variable (PV): ○ Analog Input: Node 0 / CPU / HMI Module / AI001; ● Data Register:

2. PV Upper Range Value (URV): 100.0 EU
PV Lower Range Value (LRV): 0.0 EU

Controller

3. Control Mode and Set Point: Auto Mode - LSP (S1+2, S1+3) Master PIDD No.:

4. Control Action: Reverse Control Action

5. Kp Dependent: Dependent

6. SP High Limit: 100.0 (0.0 to 100.0) EU
SP Low Limit: 0.0 (0.0 to 100.0) EU

7. Set Point (SP): 50.0 (0.0 to 100.0) EU

Output

8. MV High Limit: 100.0 (0.0 to 100.0) %
MV Low Limit: 0.0 (0.0 to 100.0) %

9. Output Manipulated Variable (MV):
□ Analog Value: ○ Analog Output: Node 0 / CPU / HMI Module / AQ002; ● Data Register:

10. Device Address:

11. Maximum Value: 65535.0 (-32768.0 to 65535.0)
Minimum Value: 0.0 (-32768.0 to 65535.0)

□ Digital Value: 12. Device Address:

13. Control Period: 3.0 (0.1 to 50.0) s

PIDD No.: 0

OK | Cancel

1. Process Variable (PV) (S1+0, S1+1)

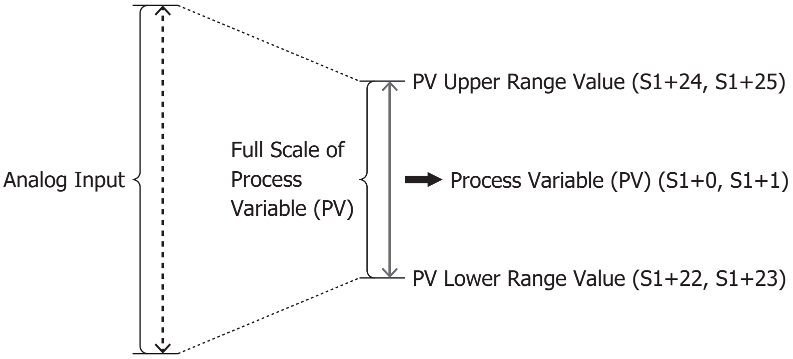
Select **Analog Input** or **Data Register** as the process variable (PV) for PID control.

When Analog Input is specified

A built-in analog input in the CPU or an analog input of the analog I/O cartridge or analog I/O module can be selected as the process variable (PV) of the PIDD instruction. Specify the analog input with the module and analog input number.

Node:	Select the node of the module having the analog input to specify.	
	Node 0:	Plus CPU module/Expansion interface remote master module
	Node 1 to 10:	Expansion interface remote slave module
Module:	Select the module having the analog input to specify.	
	CPU/HMI Module:	Built-in analog input, analog I/O cartridge
	Expansion Module 1 to 15:	Analog I/O module
Analog Input No.:	Select the analog input number from AI0 to AI7.	

Process Variable (PV) (S1+0, S1+1) is the value of the input value of the specified analog input converted to the full scale in the range of PV Lower Range Value (S1+22, S1+23) to PV Upper Range Value (S1+24, S1+25).


Analog Input
Full Scale of Process Variable (PV)
PV Upper Range Value (S1+24, S1+25)
Process Variable (PV) (S1+0, S1+1)
PV Lower Range Value (S1+22, S1+23)

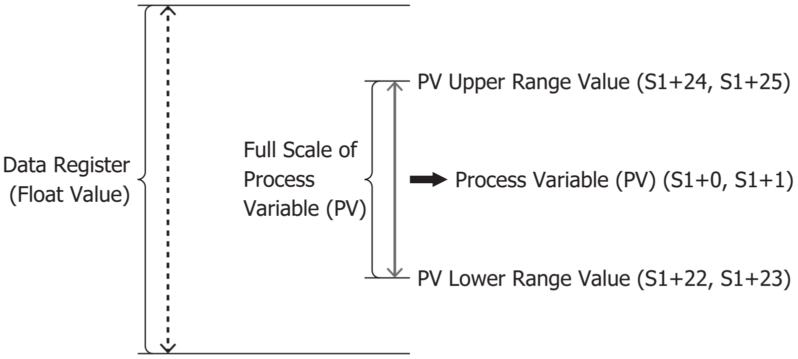
Notes:

- When **Analog Input** is specified, the analog I/O cartridge and analog I/O module must be configured in the Module Configuration Editor. For details on how to configure analog inputs, see Chapter 9 "Analog I/O Modules" in the "FC6A Series MICROSmart User's Manual".
- The analog volume on the CPU module cannot be specified as an analog input.

When Data Register is specified

The PIDD instruction loads the value (data type F (float)) stored in the two data registers starting from the specified data register as the process variable.

If the data register value is higher than PV Upper Range Value, the process variable will be PV Upper Range Value. If the data register value is lower than PV Lower Range Value, the process variable will be PV Lower Range Value. The process variable is stored in Process Variable (PV) (S1+0, S1+1).


Data Register (Float Value)
Full Scale of Process Variable (PV)
PV Upper Range Value (S1+24, S1+25)
Process Variable (PV) (S1+0, S1+1)
PV Lower Range Value (S1+22, S1+23)

2. PV Upper Range Value (URV) (S1+24, S1+25), PV Lower Range Value (LRV) (S1+22, S1+23)

Sets the full scale of Process Variable (PV) (S1+0, S1+1) with the minimum and maximum values.

If the process variable is higher than PV Upper Range Value (S1+24, S1+25), Process Variable (PV) (S1+0, S1+1) will be PV Upper Range Value. If the process variable is lower than PV Lower Range Value (S1+22, S1+23), Process Variable (PV) (S1+0, S1+1) will be PV Lower Range Value. Error code 109 will be stored in Error Status (S1+34, S1+35) in those cases.

PV Upper Range Value (S1+24, S1+25) and PV Lower Range Value (S1+22, S1+23) are set as data type F (float) between -32,768.0 to 65,535.0.

3. Control Mode and Set Point (S3+1 to S3+3)

Specifies the PID control mode and the referenced set point.

Select **Auto Mode - LSP (S1+2, S1+3)** or **Cascade Control - RSP (S1+4, S1+5)**.

When Auto Mode - LSP (S1+2, S1+3) is selected

The PIDD instruction becomes Auto Mode. Auto Mode Select (S3+2) will be on, Manual Mode Select (S3+1) will be off, and Cascade Control Mode Select (S3+3) will be off.

Store the set point in Set Point (SP) (S1+2, S1+3) in the range of SP Low Limit and SP High Limit.

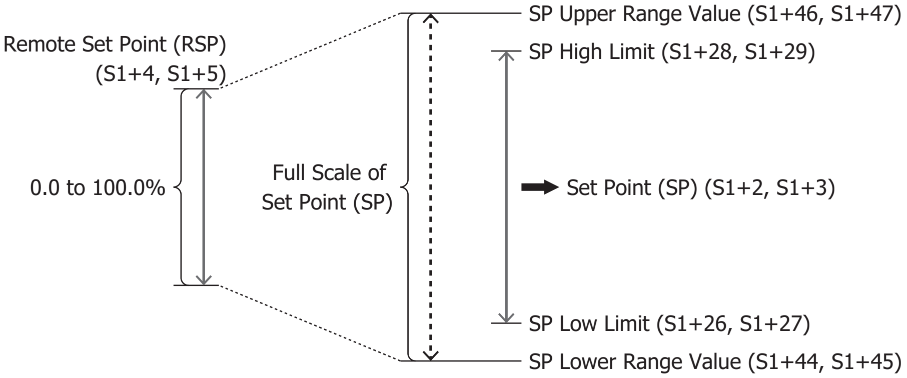
If the value of Set Point (SP) (S1+2, S1+3) is higher than SP High Limit, the PID control set point will be SP High Limit. If the value of Set Point (SP) (S1+2, S1+3) is lower than SP Low Limit, the PID control set point will be SP Low Limit. If the set point is outside the range of the high limit and low limit, error code 106 will be stored in Error Status (S1+34, S1+35).

When Cascade Control - RSP (S1+4, S1+5) is selected

The PIDD instruction becomes Cascade Control Mode. Cascade Control Mode Select (S3+3) will be on, Manual Mode Select (S3+1) will be off, and Auto Mode Select (S3+2) will be off.

Store the set point in Remote Set Point (RSP) (S1+4, S1+5) in the range of 0.0 and 100.0%. When another PIDD instruction is specified as Master PIDD No., Output Manipulated Variable (MV) (S1+16, S1+17) of the specified PIDD instruction is automatically loaded into Remote Set Point (RSP) (S1+4, S1+5).

Remote Set Point (RSP) (S1+4, S1+5) will be converted to the full scale of SP Lower Range Value (S1+44, S1+45) to SP Upper Range Value (S1+46, S1+47). If the converted set point is higher than SP High Limit (S1+28, S1+29), the set point will be SP High Limit. If the converted set point is lower than SP Low Limit (S1+26, S1+27), the set point will be SP Low Limit. The converted set point is stored in Set Point (SP) (S1+2, S1+3).


SP Upper Range Value (S1+46, S1+47)
Remote Set Point (RSP)
(S1+4, S1+5)
SP High Limit (S1+28, S1+29)
0.0 to 100.0%
Full Scale of
Set Point (SP)
Set Point (SP) (S1+2, S1+3)
SP Low Limit (S1+26, S1+27)
SP Lower Range Value (S1+44, S1+45)

4. Control Action (S3+0)

Select **Reverse Control Action** or **Direct Control Action**. When **Reverse Control Action** is selected, Control Action (S3+0) will be off. When **Direct Control Action** is selected, Control Action (S3+0) will be on.

For Reverse Control Action, when PIDD instruction Process Variable (PV) becomes larger than Set Point (SP), Output Manipulated Variable (MV) becomes smaller. Reverse Control Action is used for applications such as heating control.

For Direct Control Action, when PIDD instruction Process Variable (PV) becomes larger than Set Point (SP), Output Manipulated Variable (MV) also becomes larger. Direct Control Action is used for applications such as cooling control.

5. Kp Dependent (S3+8)

Select **Dependent** or **Independent**. When **Dependent** is selected, Kp Dependent (S3+8) will be on. When **Independent** is selected, Kp Dependent (S3+8) will be off.

When Kp Dependent is selected, the integral and derivative actions grow larger in proportion to Kp (Gain) (S1+6, S1+7). When Kp Independent is selected, Kp (Gain) (S1+6, S1+7) has no effect on the integral and derivative actions.

6. SP High Limit (S1+28, S1+29), SP Low Limit (S1+26, S1+27)

Sets the upper limit and lower limit of Set Point (SP) (S1+2, S1+3).

If the value of Set Point (SP) (S1+2, S1+3) is higher than SP High Limit (S1+28, S1+29), the PIDD control set point will be SP High Limit. If the value of Set Point (SP) (S1+2, S1+3) is lower than SP Low Limit (S1+26, S1+27), the PIDD control set point will be SP Low Limit.

SP High Limit (S1+28, S1+29) and SP Low Limit (S1+26, S1+27) are set as data type F (float) between SP Lower Range Value (S1+44, S1+45) and SP Upper Range Value (S1+46, S1+47).

7. Set Point (SP) (S1+2, S1+3)

Sets the set point (SP) for PID control.

Set Point (SP) (S1+2, S1+3) is set as data type F (float) between SP Low Limit (S1+26, S1+27) and SP High Limit (S1+28, S1+29).

8. MV High Limit (S1+32, S1+33), MV Low Limit (S1+30, S1+31)

Sets the upper limit and lower limit of Output Manipulated Variable (MV) (S1+16, S1+17).

If the output manipulated variable calculated with PID control (0 to 100%) is greater than MV High Limit (S1+32, S1+33), Output Manipulated Variable (MV) (S1+16, S1+17) will be MV High Limit. If the output manipulated variable calculated with PID control (0 to 100%) is less than MV Low Limit (S1+30, S1+31), Output Manipulated Variable (MV) (S1+16, S1+17) will be MV Low Limit.

MV High Limit (S1+32, S1+33) and MV Low Limit (S1+30, S1+31) are set as data type F (float) between 0.0 and 100.0 (0.0 and 100.0%).

9. Output Manipulated Variable (MV) (S1+16, S1+17)

If the **Analog Value** check box is selected, select **Analog Output** or **Data Register** to output the analog value.
If the **Analog Value** check box is cleared, the analog value is not output.
If the **Digital Value** check box is selected, Control Output (S3+14) turns on and off.
If the **Digital Value** check box is cleared, Control Output (S3+14) does not turn on and off.

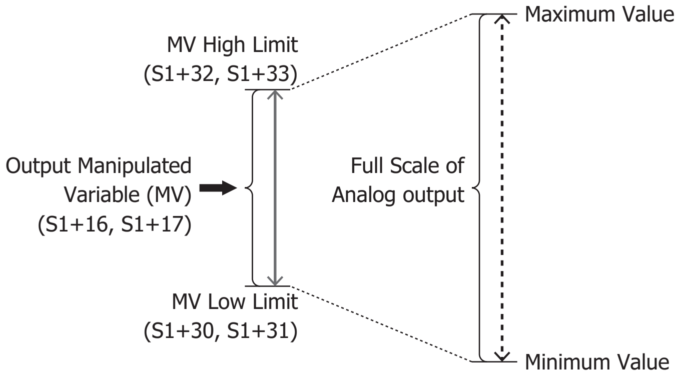
When Analog Output is specified

An analog output of an analog I/O cartridge or analog I/O module can be selected. Specify the analog output with the node, module, and analog output number.

Node:	Select the node of the module having the analog output to specify.	
	Node 0:	Plus CPU module/Expansion interface remote master module
	Node 1 to 10:	Expansion interface remote slave module
Module:	Select the module having the analog output to specify.	
	CPU/HMI Module:	Analog I/O cartridge
	Expansion Module 1 to 15:	Analog I/O module
Analog Input No.:	Select the analog output number from AO0 to AO7.	

- For auto mode, Output Manipulated Variable (S1+16, S1+17) is converted to the full scale in the range of the minimum value and maximum value of the analog output and output from the specified analog output number.
- For manual mode, Manual Mode Output MV (S1+14, S1+15) is converted to the full scale in the range of the minimum value and maximum value of the analog output and output from the specified analog output number.

The analog output value converted to the full scale can also be checked with Output Manipulated Variable (Analog Value) (S1+66, S1+67).
For switching between auto/manual mode, see "S3: Control Relay" on page 19-37.

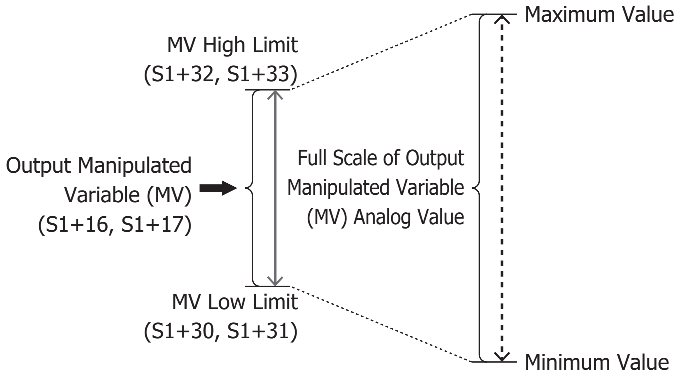


Note: To use an analog output, the analog output must be configured in advance.
For details on analog output settings, see Chapter 9 "Analog I/O Modules" in the "FC6A Series MICROSmart User's Manual".

When Data Register is specified

- For auto mode, the value of Output Manipulated Variable (MV) (S1+16, S1+17) converted to the full scale in the range of the minimum value and maximum value is stored in Output Manipulated Variable (MV) Analog Value (S1+66, S1+67).
- For manual mode, the value of Manual Mode Output MV (S1+14, S1+15) converted to the full scale in the range of the minimum value and maximum value is stored in Output Manipulated Variable (MV) Analog Value (S1+66, S1+67).

For switching between auto/manual mode, see "S3: Control Relay" on page 19-37.



When Digital Value is specified

- For auto mode, Control Output (S3+14) turns on and off from Output Manipulated Variable (MV) (S1+16, S1+17) and Control Period (S1+64, S1+65). The on pulse width for the control period varies according to the output manipulated variable.
- For manual mode, Control Output (S3+14) turns on and off from Manual Mode Output MV (S1+14, S1+15) and Control Period (S1+64, S1+65). The on pulse width for the control period varies according to Manual Mode Output MV (S1+14, S1+15).

When Data Register is selected for Output Manipulated Variable (MV), Device Address (10) and Maximum Value/Minimum Value (11) can be set.

10. Device Address

Shows the device address to store the Output Manipulated Variable (MV) Analog Value (S1+66, S1+67).

11. Maximum Value, Minimum Value

Maximum Value and Minimum Value are set to perform the full scale conversion from Output Manipulated Variable (MV) (S1+16, S1+17) to Output Manipulated Variable (MV) Analog Value (S1+66, S1+67). Set these values between -32,768.0 and 65,535.0.

When Digital Value is selected for Output Manipulated Variable (MV), Device Address (12) and Control Period (S1+64, S1+65) (13) can be set.

12. Device Address

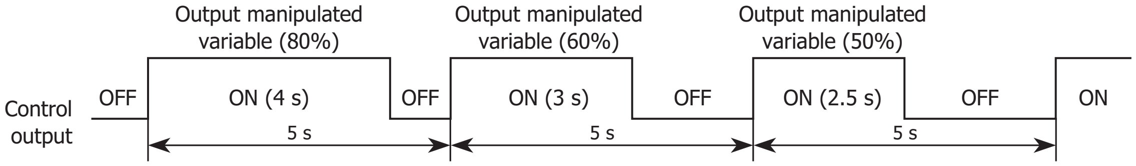
Shows the Control Output (S3+14) device address.

13. Control Period (S1+64, S1+65)

Set the cycle to perform on/off control of Control Output (S3+14) from Output Manipulated Variable (MV) (S1+16, S1+17). The on pulse width for the control period varies according to the output manipulated variable.

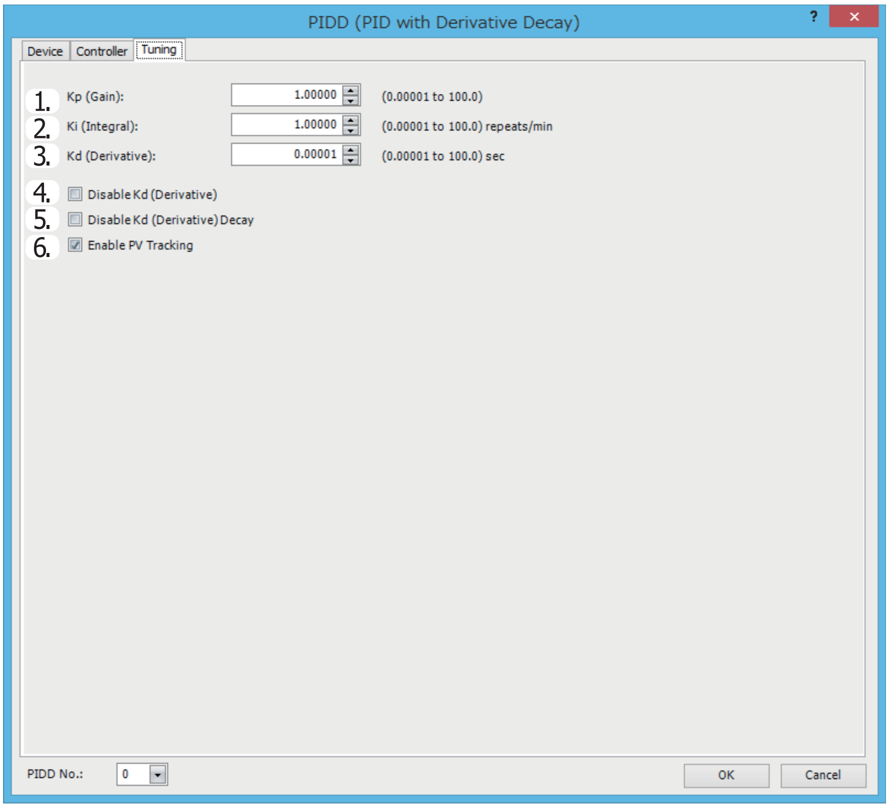
Control Period is set as data type F (float) between 0.1 and 50.0 seconds in 0.1 second increments.

When control period: 5.0 seconds (set value 5.0)


Output manipulated variable (80%)
Output manipulated variable (60%)
Output manipulated variable (50%)
Control output
OFF
ON (4 s)
OFF
ON (3 s)
OFF
ON (2.5 s)
OFF
ON
5 s
5 s
5 s

■Tuning tab

This tab is used to configure the tuning parameters of the PIDD instruction.
To store the initial values of the PIDD instruction that are set on the **Tuning** tab in the control registers and control relays, turn on the initialization input for the corresponding PIDD instruction after the user program is downloaded to the FC6A Series MICROSmart.



1. Kp (Gain): 1.00000 (0.00001 to 100.0)
2. Ki (Integral): 1.00000 (0.00001 to 100.0) repeats/min
3. Kd (Derivative): 0.00001 (0.00001 to 100.0) sec
4. □ Disable Kd (Derivative)
5. □ Disable Kd (Derivative) Decay
6. ☑ Enable PV Tracking

PIDD No.: 0

1. Kp (Gain) (S1+6, S1+7)

Sets the proportional gain for PID control. The proportional action changes the output in proportion to the difference (called the offset) between the set point (SP) and the process variable (PV). If the process variable is within the range of the proportional band, the output manipulated variable proportional to the deviation is output.
When the proportional gain is made smaller, overshooting (process variable goes above the set point), undershooting (process variable falls below the set point), and hunting (process variable is in an unstable undulating state) decrease, but it takes time until the process variable reaches the set point and the offset between the set point and the process variable becomes larger. When proportional gain is made larger, the time until the process variable reaches the set point decreases and the offset becomes smaller, but hunting increases.
Kp (Gain) (S1+6, S1+7) is set as data type F (float) between 0.00001 and 100.0 (0.00001 and 100.0%).

2. Ki (Integral) (S1+8, S1+9)

Sets the integral gain for PID control. Integral gain is the factor that determines the output manipulated variable by the integral action. With only the proportional action, an offset occurs between the set point and the process variable even when the subject to control has reached a stable state. In order to bring this offset close to 0, the integral action is required.

When integral gain is increased, the integral action becomes stronger and the set point is quickly reached, but this causes hunting with a long cycle. When integral gain is decreased, it will take time to reach the set point.

Ki (Integral) (S1+8, S1+9) is set as data type F (float) between 0.00001 and 100.0 (0.00001 and 100.0 times/minute).

3. Kd (Derivative) (S1+10, S1+11)

Sets the derivative gain for PID control. Derivative gain is the factor that determines the output manipulated variable by the derivative action. The derivative action changes the output manipulated variable and rapidly stabilizes the process variable when there is a change in the process variable due to a factor such as an external disturbance.

When derivative gain is decreased, the derivative action becomes weaker and the response to rapid process variable changes becomes slower. When derivative gain is increased, the derivative action becomes stronger and the response to rapid changes in the process variable becomes faster, but the control becomes sensitive to changes in the process variable and changes in the output manipulated variable will become more severe.

Kd (Derivative) (S1+10, S1+11) is set as data type F (float) between 0.00001 and 100.0 (0.00001 and 100.0 seconds).

4. Disable Kd (Derivative) (S3+9)

Select the **Disable Kd (Derivative)** check box to disable the derivative action. Derivative Action (S3+9) will be on.

Clear the **Disable Kd (Derivative)** check box to enable the derivative action. Derivative Action (S3+9) will be off.

5. Disable Kd (Derivative) Decay (S3+10)

Select the **Disable Kd (Derivative) Decay** check box to disable the derivative decay. Derivative Decay (S3+10) will be on.

Clear the **Disable Kd (Derivative) Decay** check box to enable the derivative decay. The derivative decay is applied to the derivative action, and it prevents drastic changes in the output manipulated variable. Derivative Decay (S3+10) will be off.

6. Enable PV Tracking (S3+7)

Select the **Enable PV Tracking** check box to enable PV tracking. PV Tracking (S3+7) will be on. When the PIDD instruction is in Manual Mode and PV tracking is enabled, the value of Process Variable (PV) (S1+0, S1+1) is copied to Set Point (SP) (S1+2, S1+3). When switching from Manual Mode to Auto Mode, the output manipulated variable (MV) will be maintained because no offset will occur between the process variable (PV) and set point (SP).

Clear the **Enable PV Tracking** check box to disable PV tracking. PV Tracking (S3+7) will be off.

S1: Control Registers

Allocation	Function	Setting Details	R/W	Modifiable during Execution
S1+0, S1+1	Process Variable (PV)	Stores the process variable for PID control. Read-only.	R	—
S1+2, S1+3	Set Point (SP)	Sets the set point for PID control. Set the set point between SP Low Limit and SP High Limit. If the set point is higher than SP High Limit, the instruction operates with SP High Limit. If the set point is lower than SP Low Limit, the instruction operates with SP Low Limit. If the set point is out of range, error code 106 will be stored in Error Status (S1+34, S1+35).	R/W	Yes
S1+4, S1+5	Remote Set Point (RSP)	Set the master PIDD instruction output manipulated variable between 0.0 and 100.0%. For Cascade Control Mode, the value of Remote Set Point (RSP) will be converted to the full scale of SP Lower Range Value (S1+44, S1+45) to SP Upper Range Value (S1+46, S1+47). If the converted set point is higher than SP High Limit (S1+28, S1+29), the set point will be SP High Limit. If the converted set point is lower than SP Low Limit (S1+26, S1+27), the set point will be SP Low Limit. The converted set point is stored in Set Point (SP) (S1+2, S1+3).	R/W	Yes
S1+6, S1+7	Kp (Gain)	Set proportional gain between 0.00001 and 100.0 (0.00001 and 100.0%). If the proportional gain value is less than 0.00001, the instruction operates with 0.00001%. If the proportional gain value is greater than 100.0, the instruction operates with 100.0%. If the value is out of range, error code 122 will be stored in Error Status (S1+34, S1+35).	R/W	Yes
S1+8, S1+9	Ki (Integral)	Set integral gain between 0.00001 and 100.0 (0.00001 and 100.0 times/minute). If the integral gain value is less than 0.00001, the instruction operates with 0.00001 times/minute. If the integral gain value is greater than 100.0, the instruction operates with 100.0 times/minute.	R/W	Yes
S1+10, S1+11	Kd (Derivative)	Set derivative gain between 0.00001 and 100.0 (0.00001 and 100.0 seconds). If the derivative gain value is less than 0.00001, the instruction operates with 0.00001 seconds. If the derivative gain value is greater than 100.0, the instruction operates with 100.0 seconds.	R/W	Yes
S1+12, S1+13	— **Reserved** —	—	—	—

Allocation	Function	Setting Details	R/W	Modifiable during Execution
S1+14, S1+15	Manual Output Manipulated Variable	When the PIDD instruction is in Manual Mode, set the output manipulated variable between 0.0 and 100.0%. If the value of Manual Output Manipulated Variable is less than 0.0%, the instruction operates with 0.0. If the value of Manual Output Manipulated Variable is greater than 100.0, the instruction operates with 100.0. If the value is out of range, error code 130 will be stored in Error Status (S1+34, S1+35).	R/W	Yes
S1+16, S1+17	Output Manipulated Variable (MV)	Stores the output manipulated variable of the PIDD instruction. The value is stored between MV Low Limit (S1+30, S1+31) and MV High Limit (S1+32, S1+33).	R	—
S1+18 to S1+21	— **Reserved** —	—	—	—
S1+22, S1+23	PV Lower Range Value (LRV)	Set PV Lower Range Value between -32,768.0 and 65,535.0. If the PV Lower Range Value is outside the range of -32,768.0 and 65,535.0, or if PV Upper Range Value ≤ PV Lower Range Value, PV Lower Range Value will be reverted to the original value, and the execution of PID control will continue. If the PV Lower Range Value is out of the range, error code 102 will be stored in Error Status (S1+34, S1+35). However, if PV Lower Range Value is out of the range or if PV Upper Range Value ≤ PV Lower Range Value at the first scan of PIDD instruction execution, the execution of PID control will stop. The execution of PID control will start once the value within the range is set and the error is eliminated.	R/W	Yes
S1+24, S1+25	PV Upper Range Value (URV)	Set PV Upper Range Value between -32,768.0 and 65,535.0. If the PV Upper Range Value is outside the range of -32,768.0 and 65,535.0, or if PV Upper Range Value ≤ PV Lower Range Value, PV Upper Range Value will be reverted to the original value, and the execution of PID control will continue. If the PV Upper Range Value is out of the range, error code 102 will be stored in Error Status (S1+34, S1+35). However, if PV Upper Range Value is out of the range or if PV Upper Range Value ≤ PV Lower Range Value at the first scan of PIDD instruction execution, the execution of PID control will stop. The execution of PID control will start once the value within the range is set and the error is eliminated.	R/W	Yes
S1+26, S1+27	SP Low Limit	Set SP Low Limit between SP Lower Range Value (S1+44, S1+45) and SP Upper Range Value (S1+46, S1+47). If the value of SP Low Limit is lower than SP Lower Range Value (S1+44, S1+45), the instruction operates with SP Lower Range Value. If the value of SP Low Limit is higher than SP Upper Range Value (S1+46, S1+47), the instruction operates with SP Upper Range Value. If SP High Limit ≤ SP Low Limit, the instruction operates with SP Lower Range Value. Error code 106 will be stored in Error Status (S1+34, S1+35) in those cases.	R/W	Yes
S1+28, S1+29	SP High Limit	Set SP High Limit between SP Lower Range Value (S1+44, S1+45) and SP Upper Range Value (S1+46, S1+47). If the value of SP High Limit is lower than SP Lower Range Value (S1+44, S1+45), the instruction operates with SP Lower Range Value. If the value of SP High Limit is higher than SP Upper Range Value (S1+46, S1+47), the instruction operates with SP Upper Range Value. If SP High Limit ≤ SP Low Limit, the instruction operates with SP Upper Range Value. Error code 106 will be stored in Error Status (S1+34, S1+35) in those cases.	R/W	Yes

Allocation	Function	Setting Details	R/W	Modifiable during Execution
S1+30, S1+31	MV Low Limit	Set MV Low Limit between 0.0 and 100.0 (0.0 and 100.0%). If the value of MV Low Limit is less than 0.0 or greater than 100.0, the instruction operates with 0.0. If MV High Limit ≤ MV Low Limit, the instruction operates with MV Low Limit as 0.0 and MV High Limit as 100.0. Error code 103 will be stored in Error Status (S1+34, S1+35) in those cases.	R/W	Yes
S1+32, S1+33	MV High Limit	Set MV High Limit between 0.0 and 100.0 (0.0 and 100.0%). If the value of MV High Limit is less than 0.0 or greater than 100.0, the instruction operates with 100.0. If MV High Limit ≤ MV Low Limit, the instruction operates with MV Low Limit as 0.0 and MV High Limit as 100.0. Error code 103 will be stored in Error Status (S1+34, S1+35) in those cases.	R/W	Yes
S1+34, S1+35	Error Status	The PIDD instruction error status is stored as data type D (double-word). For details on error codes, see "Error Status (S1+34, S1+35)" on page 19-37.	R	—
S1+36 to S1+43	— **Reserved** —	—	—	—
S1+44, S1+45	SP Lower Range Value	Stores the same value as PV Lower Range Value (S1+22, S1+23).	R	—
S1+46, S1+47	SP Upper Range Value	Stores the same value as PV Upper Range Value (S1+24, S1+25).	R	—
S1+48, S1+49	Output Manipulated Variable while PID Control is Inhibited	Set the output manipulated variable while PID Inhibit (S3+11) is on and PID control is inhibited. Set the value as data type F (float) between 0.0 and 100.0 (0.0 and 100.0%). While PID Inhibit (S3+11) is off, the value of Output Manipulated Variable (S1+16, S1+17) for the PIDD instruction is copied to "Output Manipulated Variable while PID Control is Inhibited (S1+48, S1+49)".	R/W	Yes
S1+50, S1+51	Input Deviation (Offset)	Stores the proportion (%) of the difference (offset) between the set point (SP) and process variable (PV) for the full scale of the process variable.	R	—
S1+52 to S1+63	— **Reserved** —	—	—	—
S1+64, S1+65	Control period	Set the cycle to perform on/off control of Control Output (S3+14) from Output Manipulated Variable. Set Control Period as data type F (float) between 0.1 and 50.0 seconds. The control period operates as 0.1 seconds if set to a value less than 0.1 seconds, and it operates as 50.0 seconds if set to a value greater than or equal to 50.1 seconds. Error code 132 will be stored in Error Status (S1+34, S1+35) in those cases.	R/W	—
S1+66, S1+67	Output Manipulated Variable (MV) Analog Value	• When **Analog Output** is specified for Output Manipulated Variable: Stores the value of Output Manipulated Variable (S1+16, S1+17) converted to the full scale in the range of the minimum value and maximum value of the analog output. • When **Data Register** is specified for Output Manipulated Variable: Stores the value of Output Manipulated Variable (S1+16, S1+17) converted to the full scale in the range of the minimum value and maximum value.	R	—
S1+68 to S1+99	— **Reserved** —	—	—	—

Error Status (S1+34, S1+35)

Indicates the PIDD error status.

Error Code	Status Description	Countermeasure	Status Classification	Error Check with WindLDR
101	The analog input specified for **Analog Input** is not configured.	Check the analog I/O module settings.	PID control execution stops	Yes
102	PV Upper Range Value or PV Lower Range Value is out of the range when the PIDD instruction is started.	Check if PV Upper Range Value or PV Lower Range Value is set to an out-of-range value, and check if PV Lower Range Value ≥ PV Upper Range Value.	PID control execution stops	Yes
	PV Upper Range Value or PV Lower Range Value is set to an out-of-range value while the PIDD instruction is being executed.		PID control execution continues	
103	An out-of-range value is set to MV High Limit or MV Low Limit.	Check if MV High Limit or MV Low Limit is set to an out-of-range value, and check if MV Low Limit ≥ MV High Limit.	PID control execution continues	Yes
106	An out-of-range value is set to SP High Limit or SP Low Limit.	Check if SP High Limit or SP Low Limit is set to an out-of-range value, and check if SP Low Limit ≥ SP High Limit.	PID control execution continues	Yes
109	Process Variable (PV) is set to a value outside the range of PV Lower Range Value to PV Upper Range Value.	Check if the process variable is outside the range of PV Lower Range Value to PV Upper Range Value.	PID control execution continues	No
111	The analog I/O module settings that were set for Analog Output are incorrect.	Check the analog I/O module settings.	PID control execution stops	Yes
122	Kp (Gain) is set to a value outside the range of 0.00001 to 100.0.	Check the Kp (Gain) setting.	PID control execution continues	No
130	Manual Mode Output MV is set to a value outside the range of 0.0 to 100.0.	Check the value of Manual Mode Output MV.	PID control execution continues	No
132	Control Period is set to a value outside the range of 0.1 to 50.0.	Check the control period settings.	PID control execution continues	Yes
Others	— **Reserved** —	—	—	—

S3: Control Relay

Storage Destination	Function	Setting Details	R/W	Modifiable during Execution
S3+0	Control Action	0 (OFF): Reverse Control Action 1 (ON): Direct Control Action	R/W	Yes
S3+1	Manual Mode Select	Set to 1 (ON) to switch the PIDD instruction to Manual Mode. S3+1 automatically changes to 0 (OFF).	R/W	Yes
S3+2	Auto Mode Select	Set to 1 (ON) to switch the PIDD instruction to Auto Mode. S3+2 automatically changes to 0 (OFF).	R/W	Yes
S3+3	Cascade Control Mode Select	Set to 1 (ON) to switch the PIDD instruction to Cascade Control Mode. S3+3 automatically changes to 0 (OFF).	R/W	Yes
S3+4	Manual Mode Active	Changes to 1 (ON) while PIDD instruction is in Manual Mode. Changes to 0 (OFF) when the mode is not Manual Mode.	R	—
S3+5	Auto Mode Active	Changes to 1 (ON) while PIDD instruction is in Auto Mode. Changes to 0 (OFF) when the mode is not Auto Mode.	R	—

Storage Destination	Function	Setting Details	R/W	Modifiable during Execution
S3+6	Cascade Control Mode Active	Changes to 1 (ON) while PIDD instruction is in Cascade Control Mode. Changes to 0 (OFF) when the mode is not Cascade Control Mode.	R	—
S3+7	PV Tracking	0 (OFF): PV tracking disabled 1 (ON): PV tracking enabled	R/W	Yes
S3+8	Kp Dependent	0 (OFF): Kp dependent 1 (ON): Kp independent	R/W	Yes
S3+9	Derivative	0 (OFF): Derivative enabled 1 (ON): Derivative disabled	R/W	Yes
S3+10	Derivative Decay	0 (OFF): Derivative decay enabled 1 (ON): Derivative decay disabled	R/W	Yes
S3+11	PID Inhibit	0 (OFF): PID operation 1 (ON): PID inhibited	R/W	Yes
S3+12	PID Control Inhibit Monitor	Changes to 1 (ON) when PID control is temporarily stopped, such as when PID control parameters are changed or when PID control is inhibited.	R	—
S3+13	Derivative Action Execution Monitor	Changes to 1 (ON) while executing the derivative action.	R	—
S3+14	Control Output (Output Manipulated Variable (MV) Digital Value)	Changes between 1 (on) and 0 (off) according to the control period and output manipulated variable.	R	—
S3+15 to S3+31	— **Reserved** —	—	—	—

PID Monitor

Function Description

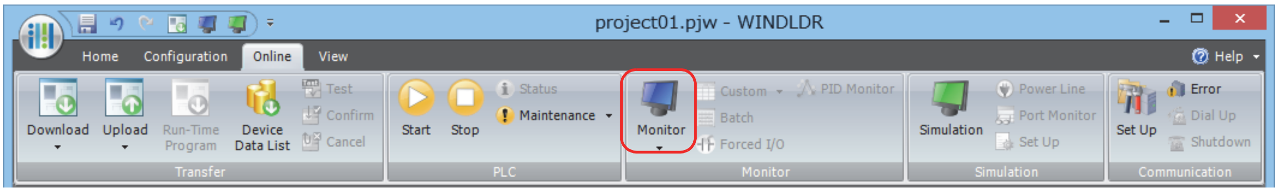
While PID control is executing, you can graphically monitor the PIDA/PIDD instruction parameters. From the **PID Monitor** dialog box, you can also directly change the values of data registers and internal relays that are being used by the PIDA/PIDD instructions, and check the operation of the PIDA/PIDD instructions while adjusting the PID control parameters.

Note: The PID monitor operates according to the PIDA/PIDD instruction settings in the ladder program displayed on the WindLDR and not according to the settings in the PLC. If the device allocations for the PIDA/PIDD instruction in the PLC and the PIDA/PIDD instruction in WindLDR are different, the PID instruction is not correctly monitored.

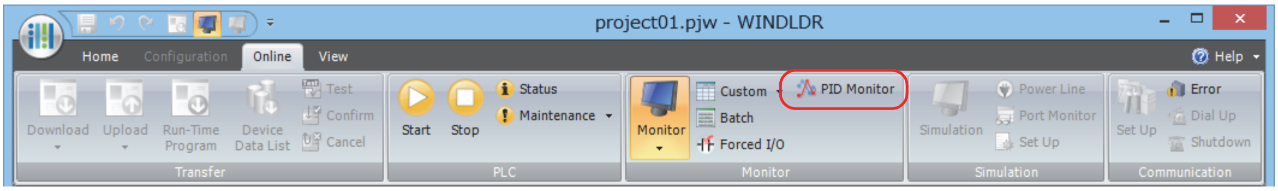
Starting the PID Monitor

Operation Procedure

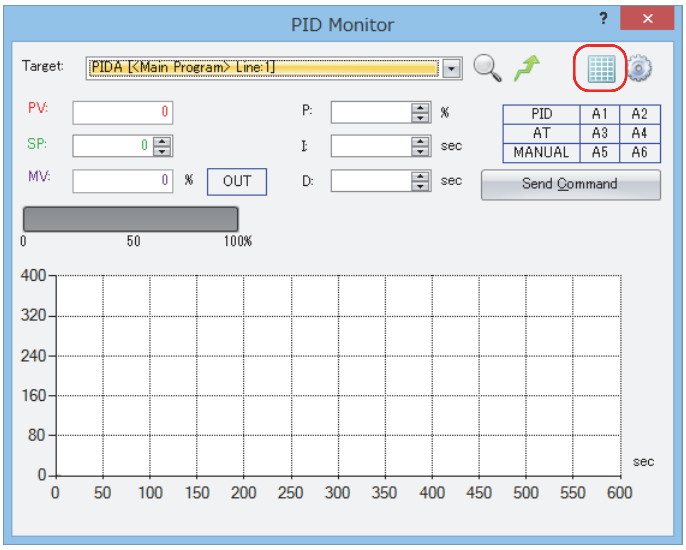
1. From the WindLDR menu bar, select **Online** > **Monitor** > **Monitor**.
   The monitor starts.



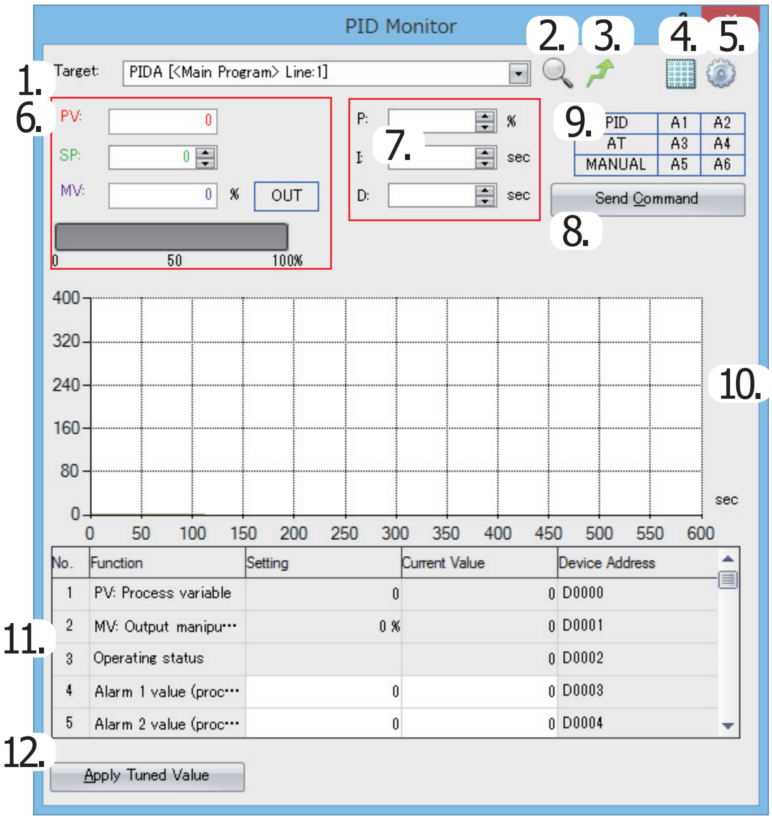
2. From the WindLDR menu bar, select **Online** > **Monitor** > **PID Monitor**.
   The **PID Monitor** dialog box is displayed.



3. On the **PID Monitor** dialog box, click ▦ button (show/hide PID monitor table).
   The PID monitor table is displayed.



PID Monitor Dialog Box



Target: PIDA [<Main Program> Line:1]

PV: 0

SP: 0

MV: 0 %

No.	Function	Setting	Current Value	Device Address
1	PV: Process variable	0	0	D0000
2	MV: Output manipu···	0 %	0	D0001
3	Operating status		0	D0002
4	Alarm 1 value (proc···	0	0	D0003
5	Alarm 2 value (proc···	0	0	D0004

1. Target

Select the PIDA/PIDD instruction to monitor. All PIDA/PIDD instructions in the ladder program opened in WindLDR are displayed. The PIDA/PIDD instructions can be identified by the program name and line number in the main program or subroutine where the instructions are used.

2. Refer to PID settings

Opens a dialog box for the PIDA/PIDD instruction being monitored so that you can confirm the settings.

3. Jump to target PID instruction

Moves the cursor of the ladder editor to the position of the PIDA/PIDD instruction being monitored.

4. Show/hide PID monitor table

You can show or hide **PID Monitor Table** (11) and **Apply Tuned Value** button (12).

5. PID monitor settings

Opens the **PID Monitor Settings** dialog box. For details, see "PID Monitor Settings Dialog Box" on page 19-44.

6. Primary parameters panel

You can check the values of the process variable (PV), set point (SP), and manipulated variable (MV). For the MV, the analog value can be checked with the horizontal bar graph, and the digital value (on/off) can be checked with the color of the OUT indicator. When the OUT indicator is white, the output is off. When it is green, the output is on.

The colors of the values for the process variable (PV), set point (SP), and manipulated variable (MV) can be changed in the **PID Monitor Settings** dialog box. For details, see "PID Monitor Settings Dialog Box" on page 19-44.

When a PIDA instruction is selected, a digital value (on/off) can be checked with the color of the OUT indicator. The OUT indicator is white when output is off, and green when output is on.

When a PIDD instruction is selected and manual mode is on, the manipulated variable (MV) text box is enabled, and the manual output manipulated variable (S1+14, S1+15) can be set. The value can be set between 0.0 and 100.0 in the unit of percentage.

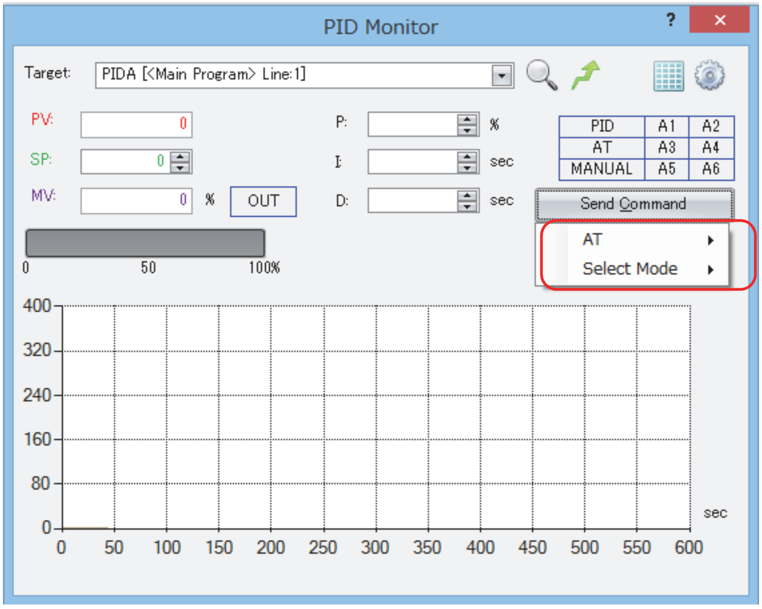
7. PID parameters panel

When a PIDA instruction is selected, you can check the values of the proportional band (P), integral time (I), and derivative time (D).

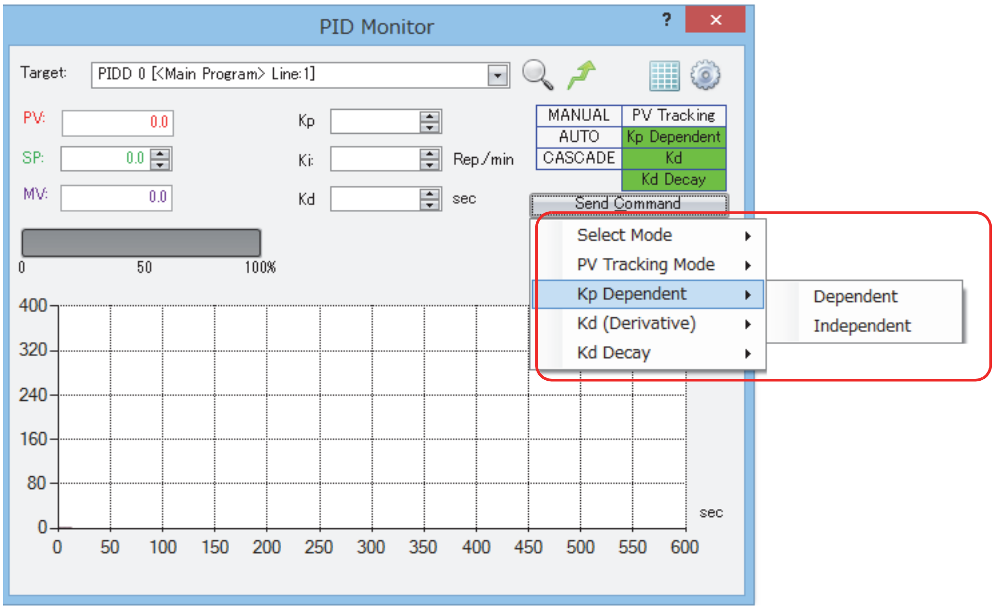
When a PIDD instruction is selected, you can check the values of the proportional gain (Kp), integral gain (Ki), and derivative gain (Kd).

8. Send Command

When a PIDA instruction is selected, you can send commands to the FC6A Series MICROSmart to execute/stop AT for PID control and to switch manual/auto mode. Click this button to display a popup menu and click the menu to send the command.



When a PIDD instruction is selected, you can send the following commands to the FC6A Series MICROSmart: Change PID control mode (manual/auto/cascade), enable/disable PV tracking, proportional gain dependent/independent, enable/disable derivative control, and enable/disable derivative decay. Click this button to display a popup menu and click the corresponding item to send the command.



9. Status indicators

You can check the primary statuses for PID control with the colored indicators.

When a PIDA instruction is selected, the indicators in the following table are displayed.

Indicator Name	Background Color	Status
PID	Gray	PID control stopped
	Green	PID control is executed
AT	Gray	AT stopped
	Green	AT is being executed
MANUAL	Gray	Auto mode
	Green	Manual mode
A1	Gray	Normal operation
	Red	Alarm 1 output is ON
A2	Gray	Normal operation
	Red	Alarm 2 output is ON
A3	Gray	Normal operation
	Red	Alarm 3 output is ON
A4	Gray	Normal operation
	Red	Alarm 4 output is ON
A5	Gray	Normal operation
	Red	Alarm 5 output is ON
A6	Gray	Normal operation
	Red	Alarm 6 output is ON

When a PIDD instruction is selected, the indicators in the following table are displayed.

Indicator Name	Background Color	Status
MANUAL	Gray	Other mode
	Green	Manual mode
AUTO	Gray	Other mode
	Green	Auto mode
CASCADE	Gray	Other mode
	Green	Cascade control mode
PV Tracking	Gray	PV tracking disabled
	Green	PV tracking enabled
Kp Dependent	Gray	Kp independent
	Green	Kp dependent
Kd	Gray	Derivative action disabled
	Green	Derivative action enabled
Kd Decay	Gray	Derivative decay disabled
	Green	Derivative decay enabled

10. Trend graph

When a PIDA instruction is selected, you can monitor the values of the process variable (PV), set point (SP), manipulated variable (MV), alarm 1 value (process high alarm), and alarm 2 value (process low alarm) with the trend graph.

When a PIDD instruction is selected, you can monitor the values of the process variable (PV), set point (SP), and output manipulated variable (MV) with the trend graph.

Shown values, colors, maximum value, and minimum value can be changed in the **PID Monitor Settings** dialog box.

For details, see "PID Monitor Settings Dialog Box" on page 19-44.

Note: When plots on the trend graph reach the right edge of the graph, the center of the graph moves to the left edge and the new log information is displayed from the center of the graph. You can check the past log information by moving the scrollbar. The log information retains a maximum of 10,000 items of information. When the log information has reached 10,000 items of information, the log information is deleted from the oldest items, and the new log information is recorded. The log information displayed on the trend graph is deleted when the dialog box is closed or when the PIDA/PIDD instruction being monitored is changed.

11. PID monitor table

You can check the function name, current value, and device address for data registers and internal relays used by the PIDA/PIDD instruction. The current values displayed in white cells can be changed. Showing/hiding items can be changed in the **PID Monitor Settings** dialog box. For details, see "PID Monitor Settings Dialog Box" on page 19-44.

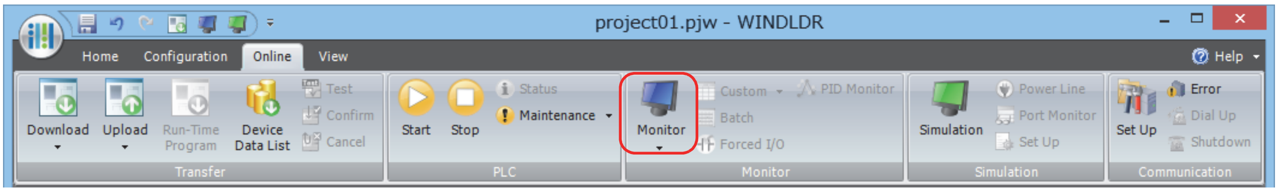
12. Apply Tuned Value

By clicking **Apply Tuned Value**, you can reflect the PID control parameters adjusted in **PID monitor table** (11) as the initial setting values for the PIDA/PIDD instruction selected in **Target** (1). To store the reflected initial setting values to FC6A Series MICROSmart ROM, you must end the monitor and download the user program to the FC6A Series MICROSmart.

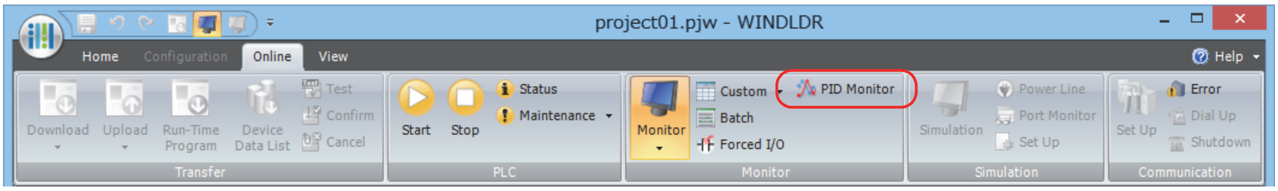
Changing PID Monitor Settings

Operation Procedure

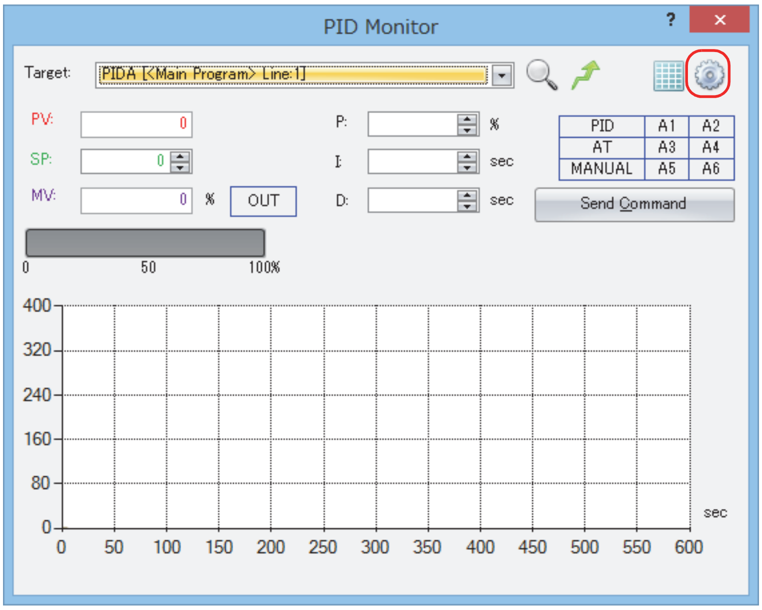
1. From the WindLDR menu bar, select **Online** > **Monitor** > **Monitor**.
   The monitor starts.



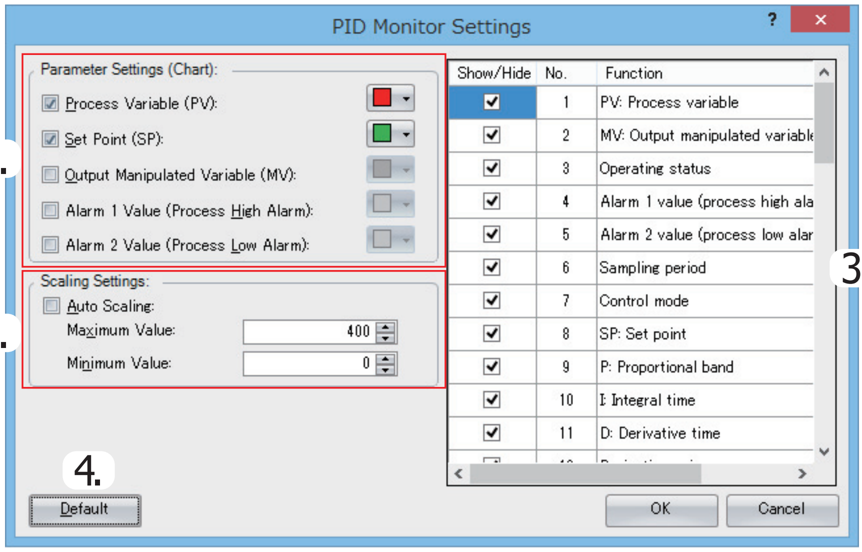
2. From the WindLDR menu bar, select **Online** > **Monitor** > **PID Monitor**.
   The **PID Monitor** dialog box is displayed.



3. In the **PID Monitor** dialog box, click ⚙ button (PID monitor settings).
   The **PID Monitor Settings** dialog box is displayed.



PID Monitor Settings Dialog Box



Parameter Settings (Chart):

- ☑ Process Variable (PV): 🟥
- ☑ Set Point (SP): 🟩
- □ Output Manipulated Variable (MV):
- □ Alarm 1 Value (Process High Alarm):
- □ Alarm 2 Value (Process Low Alarm):

Scaling Settings:

- □ Auto Scaling:
- Maximum Value: 400
- Minimum Value: 0

Show/Hide	No.	Function
☑	1	PV: Process variable
☑	2	MV: Output manipulated variabl
☑	3	Operating status
☑	4	Alarm 1 value (process high ala
☑	5	Alarm 2 value (process low alar
☑	6	Sampling period
☑	7	Control mode
☑	8	SP: Set point
☑	9	P: Proportional band
☑	10	I: Integral time
☑	11	D: Derivative time

1. Parameter Settings (Chart)

You can show or hide the parameters displayed on the trend graph in the **PID Monitor** dialog box, and also specify the colors. Select the check box for each item to show it, or clear the check box to hide it. The color of the button on the right side of each item is the display color for that item. Press ▼ button to display the list of colors and select the color to change the display color. The selected color is applied to the primary parameters panel in the **PID Monitor** dialog box.

When a PIDA instruction is selected, you can change the settings of the process variable (PV), set point (SP), output manipulated variable (MV), alarm 1 value (process high alarm), and alarm 2 value (process low alarm).

When a PIDD instruction is selected, you can change the settings of the process variable (PV), set point (SP), and output manipulated variable (MV).

2. Scaling Settings

You can set the scaling for the vertical axis of the trend graph in the **PID Monitor** dialog box. When the auto scaling check box is selected, the Maximum Value and the Minimum Value for the vertical axis are automatically adjusted for the values displayed on the trend graph. When the check box is cleared, the Maximum Value and the Minimum Value are set according to the entered values.

3. PID monitor table settings

You can show or hide the items on the PID monitor table. Select the check box for each item to show it, or clear the check box to hide it.

4. Default

All settings are set to their initial values.

Application Example

This section describes an application example using the PIDA instruction.

Note: You must change the settings according to the application's actual system configuration and operating status.

The following two system configurations are described for applications that set the set point for the control target temperature to 200°C and perform PID control.

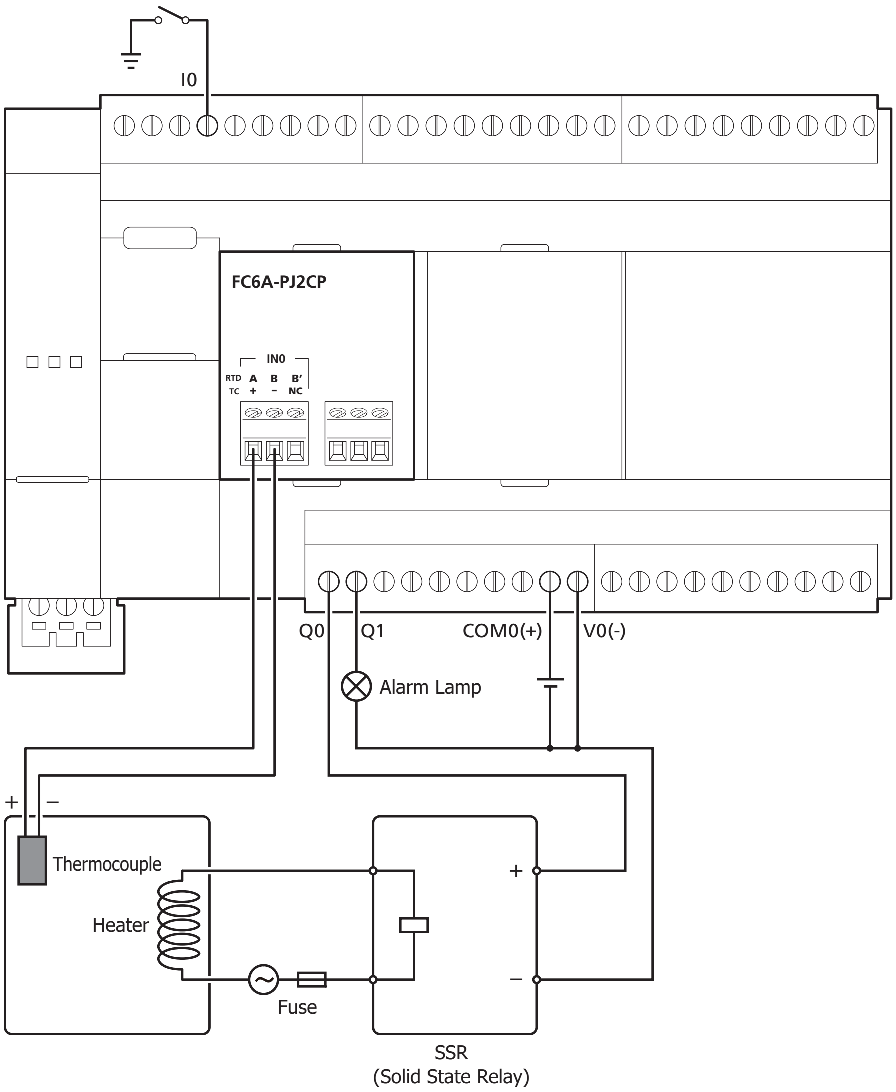
- PID control via ON/OFF output
- PID control via analog output

Operation

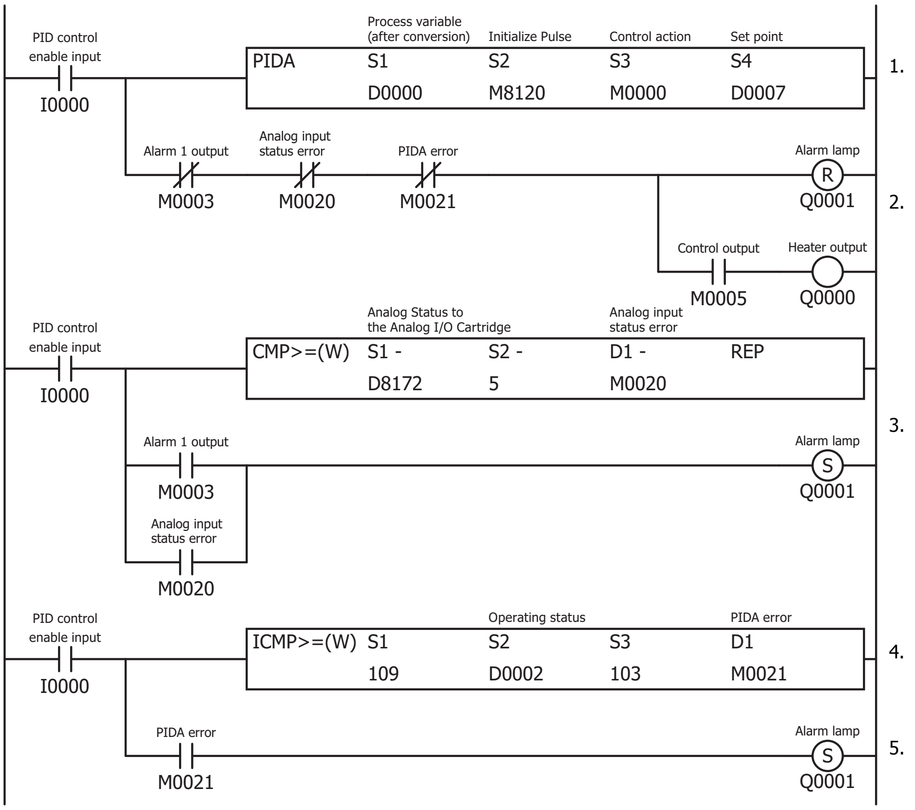
- PID control is performed based on the temperature input to the analog I/O cartridge and the manipulated variable is output.
  The set point is 200ºC.
- The control mode is PID (PID control) and the control action is reverse control action.
- Alarm 1 output (S3+3) is turned ON when the process variable reaches 250°C, and the heater is stopped or the power is adjusted.

PID control via ON/OFF output

System configuration



Ladder program



Device Address	Comment
D0000	Process variable (after conversion)
D0002	Operating status
D0007	Set point
D8172	Analog Status to the Analog I/O Cartridge
I0000	PID control enable input
Q0000	Heater output
Q0001	Alarm lamp

Device Address	Comment
M0000	Control action
M0003	Alarm 1 output
M0005	Control output
M0020	Analog input status error
M0021	PIDA error
M8120	Initialize Pulse

1. The PIDA instruction operates when I0 is ON.
2. When M0003 (Alarm 1 output (process high alarm)) is OFF and M0020 is OFF (the analog input status error for the analog I/O cartridge is 0 to 2), Q0 (heater output) turns ON/OFF according to the manipulated variable and control period calculated by PID control. The heater output turns OFF when an error occurs.
3. When the analog input status error for the analog I/O cartridge is 5, 6, or 8 to 11, M0020 turns ON.
4. When M0003 (Alarm 1 output (process high alarm)) is ON or M0020 is ON (when the analog input status error for the analog I/O cartridge is 5, 6, or 8 to 11), Q1 (alarm lamp) turns ON.
5. The operating status of the PIDA instruction is judged, and when an error occurs, M0021 turns ON and Q1 (alarm lamp) turns ON.

Note:

The analog status for the analog I/O cartridge is as follows.

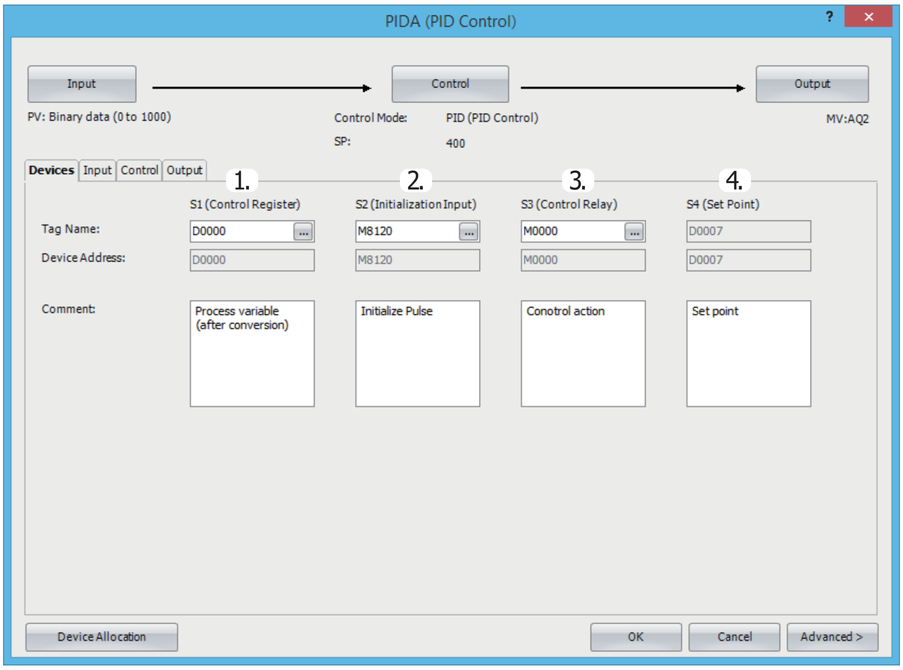
- Analog input status error
  0: Operating normally
  1: Converting data
  2: Initializing
  5, 6: Wiring error
  8 to 11: Analog I/O cartridge error

For details, see Chapter 10 "Analog I/O Cartridge" in the "FC6A Series MICROSmart User’s Manual".

PIDA (PIDA control) instruction dialog box configuration procedure

Use the default value for settings that are not described in this procedure.

1. The **Devices** tab configures the devices used with the PIDA instruction.
   - Set S1 (Control Register) to D0000 (1).
   - Set S2 (Initialization Input) to M8120 (2).
   - Set S3 (Control Relay) to M0000 (3).
   - S4 (Set Point) is automatically set to D0007 (4).



PIDA (PID Control)

PV: Binary data (0 to 1000)

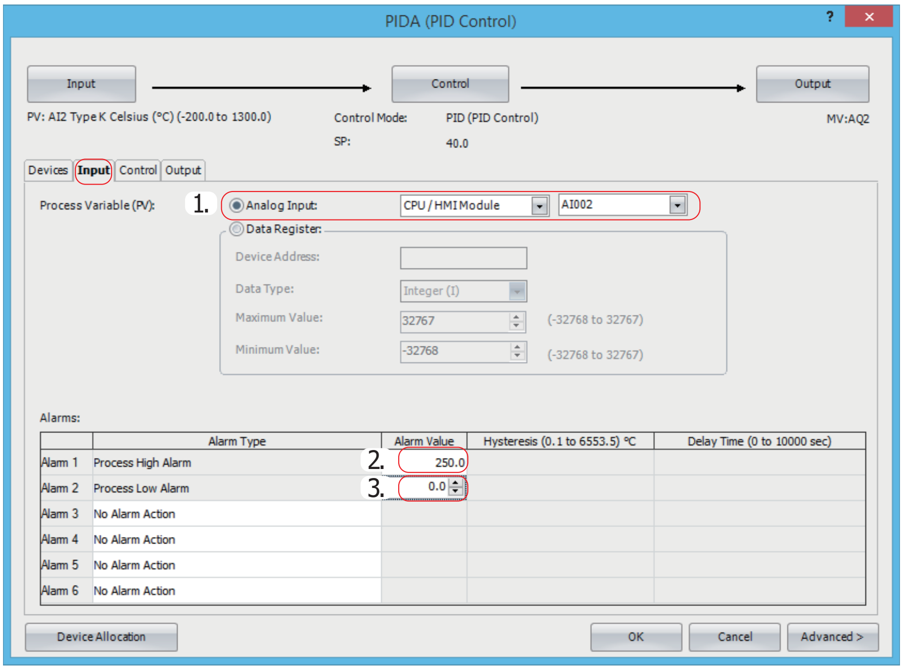
Control Mode: PID (PID Control)

SP: 400

MV:AQ2

	1. S1 (Control Register)	2. S2 (Initialization Input)	3. S3 (Control Relay)	4. S4 (Set Point)
Tag Name:	D0000	M8120	M0000	D0007
Device Address:	D0000	M8120	M0000	D0007
Comment:	Process variable (after conversion)	Initialize Pulse	Conotrol action	Set point

2. Click the **Input** tab and configure the items.
   - For the process variable (PV), select **Analog Input**, and then select **CPU/HMI Module** and **AI002** (1).
   - For Alarm 1, set the alarm value for the process high alarm to **250.0** (2).
   - For Alarm 2, set the alarm value for the process low alarm to **0.0** (3).



PIDA (PID Control)

PV: AI2 Type K Celsius (°C) (-200.0 to 1300.0)

Control Mode: PID (PID Control)

SP: 40.0

MV:AQ2

Process Variable (PV): 1. ● Analog Input: CPU / HMI Module AI002

○ Data Register:

Device Address:

Data Type: Integer (I)

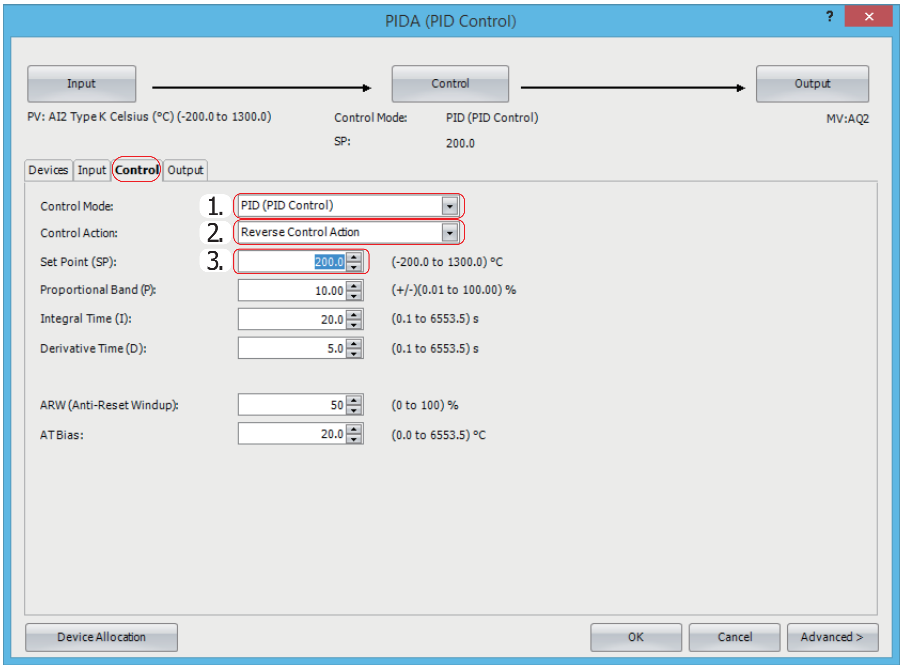
Maximum Value: 32767 (-32768 to 32767)

Minimum Value: -32768 (-32768 to 32767)

Alarms:

	Alarm Type	Alarm Value	Hysteresis (0.1 to 6553.5) °C	Delay Time (0 to 10000 sec)
Alarm 1	Process High Alarm	2. 250.0		
Alarm 2	Process Low Alarm	3. 0.0		
Alarm 3	No Alarm Action			
Alarm 4	No Alarm Action			
Alarm 5	No Alarm Action			
Alarm 6	No Alarm Action			

3. Click the **Control** tab and configure the items.
   - For Control Mode, select **PID (PID Control)** (1).
   - For Control Action, select **Reverse Control Action** (2).
   - For Set Point (SP), set **200.0** (3).



PIDA (PID Control)

Input → Control → Output

PV: AI2 Type K Celsius (°C) (-200.0 to 1300.0)

Control Mode: PID (PID Control)

SP: 200.0

MV:AQ2

Devices | Input | Control | Output

Control Mode: 1. PID (PID Control)

Control Action: 2. Reverse Control Action

Set Point (SP): 3. 200.0 (-200.0 to 1300.0) °C

Proportional Band (P): 10.00 (+/-)(0.01 to 100.00) %

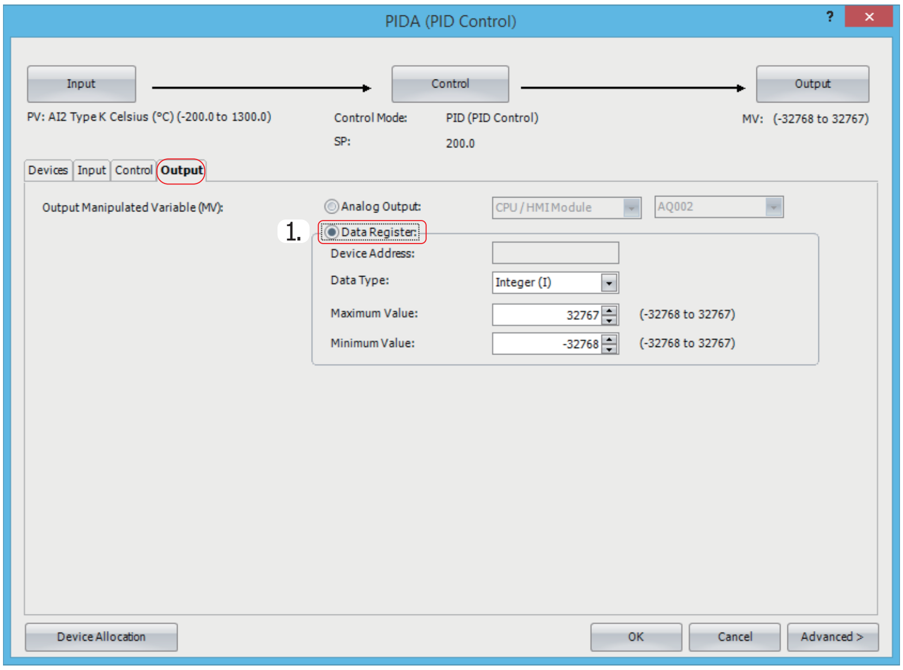
Integral Time (I): 20.0 (0.1 to 6553.5) s

Derivative Time (D): 5.0 (0.1 to 6553.5) s

ARW (Anti-Reset Windup): 50 (0 to 100) %

AT Bias: 20.0 (0.0 to 6553.5) °C

4. Click the **Output** tab and configure the items.
   - For Output Manipulated Variable (MV), select **Data Register** (1).



PIDA (PID Control)

Input → Control → Output

PV: AI2 Type K Celsius (°C) (-200.0 to 1300.0)

Control Mode: PID (PID Control)

SP: 200.0

MV: (-32768 to 32767)

Devices | Input | Control | Output

Output Manipulated Variable (MV):

○ Analog Output: CPU / HMI Module AQ002

1. ● Data Register:

Device Address:

Data Type: Integer (I)

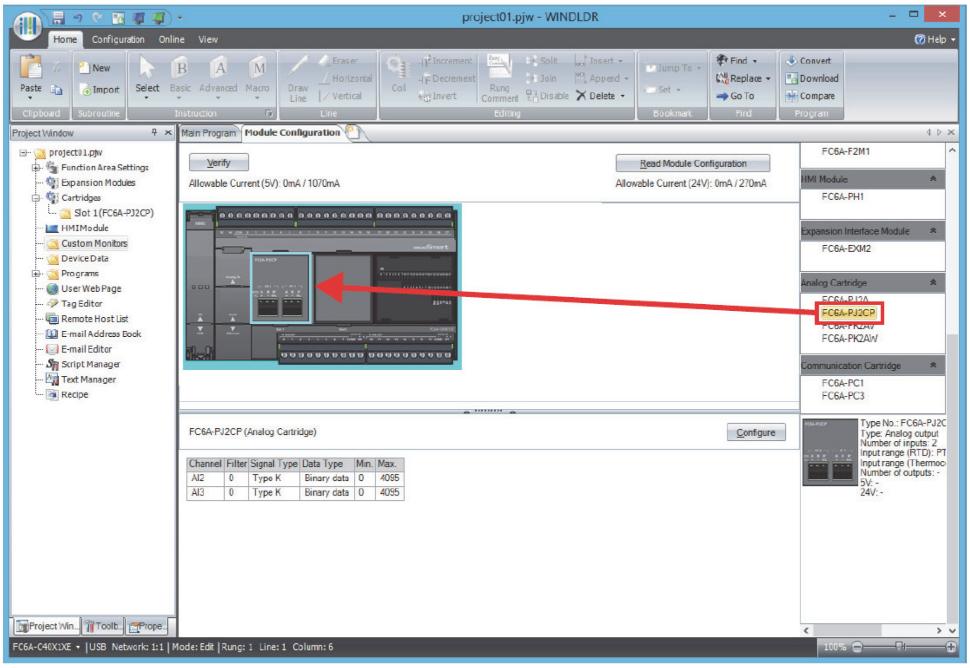
Maximum Value: 32767 (-32768 to 32767)

Minimum Value: -32768 (-32768 to 32767)

5. Click **OK** to close the **PIDA (PIDA Control)** dialog box.

6. Use the Module Configuration Editor to configure the analog input for the analog I/O cartridge. On the **Configuration** tab, in the **PLCs** group, select **Expansion Modules**.

7. Select **FC6A-PJ2CP** in the expansion modules and cartridges list and drag and drop it to the module configuration area.



Allowable Current (5V): 0mA / 1070mA

Allowable Current (24V): 0mA / 270mA

FC6A-PJ2CP (Analog Cartridge)

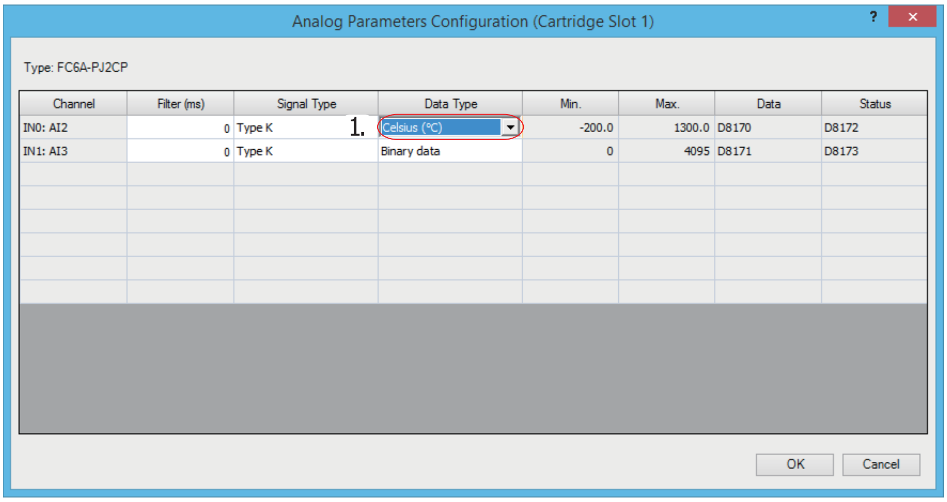
Channel	Filter	Signal Type	Data Type	Min.	Max.
AI2	0	Type K	Binary data	0	4095
AI3	0	Type K	Binary data	0	4095

Type No.: FC6A-PJ2C
Type: Analog output
Number of inputs: 2
Input range (RTD): PT
Input range (Thermoc
Number of outputs: -
5V: -
24V: -

FC6A-C40X1XE • |USB Network: 1:1 | Mode: Edit | Rung: 1 Line: 1 Column: 6

8. Click the **FC6A-PJ2CP** analog I/O cartridge that was interested in the module configuration area and click **Configure**. The **Analog Parameters Configuration (Cartridge Slot 1)** dialog box is displayed.

9. Configure the items.
   - For channel IN0 : AI2 **Data Type**, select **Celsius (°C)** (1).



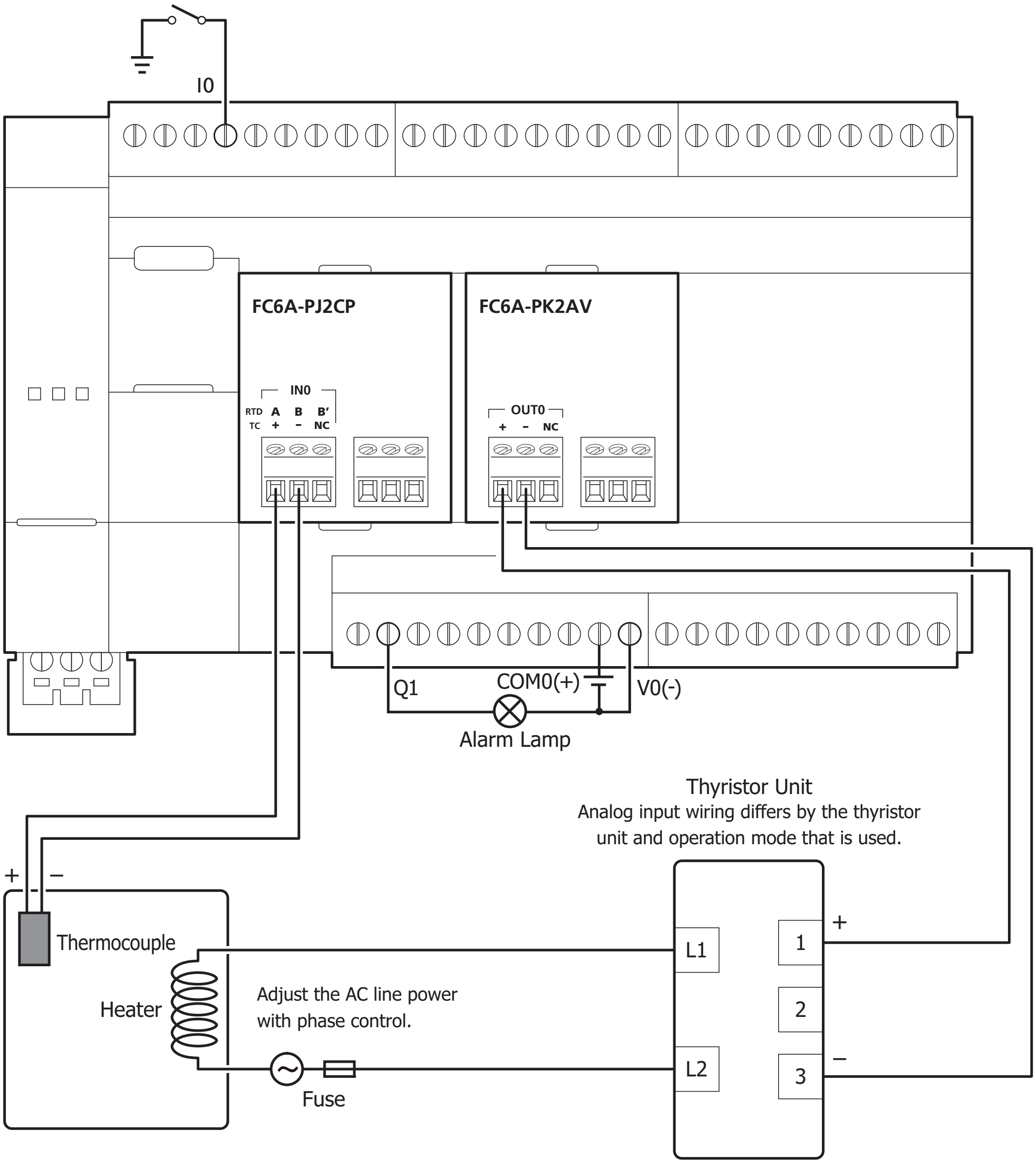
Type: FC6A-PJ2CP

Channel	Filter (ms)	Signal Type	Data Type	Min.	Max.	Data	Status
IN0: AI2	0	Type K	Celsius (°C)	-200.0	1300.0	D8170	D8172
IN1: AI3	0	Type K	Binary data	0	4095	D8171	D8173

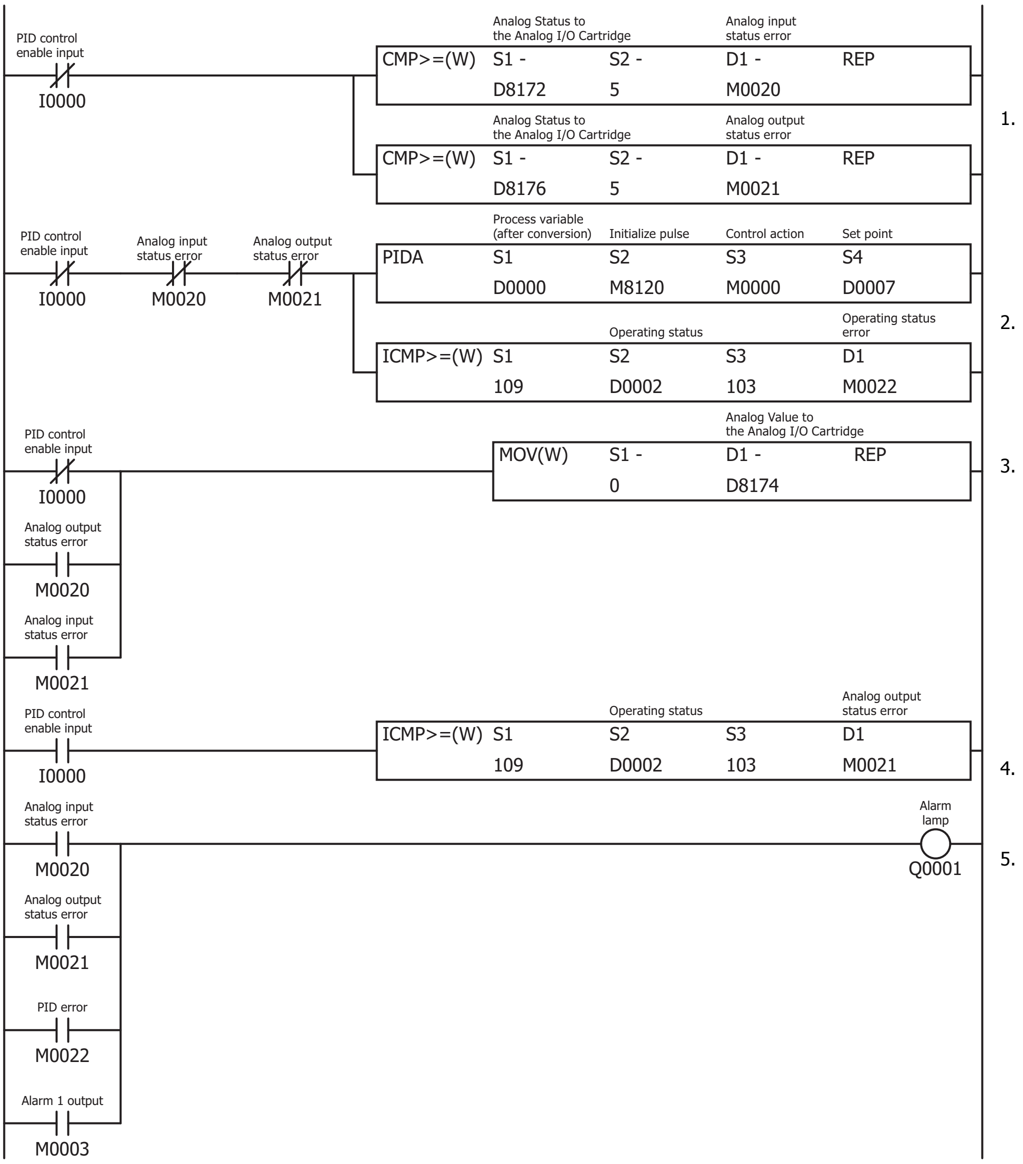
10. Click **OK** to close the **Analog Parameters Configuration (Cartridge Slot 1)** dialog box.

PID control via analog output

System configuration


I0
FC6A-PJ2CP
IN0
RTD A B B'
TC + – NC
FC6A-PK2AV
OUT0
+ – NC
Q1
COM0(+)
V0(-)
Alarm Lamp
Thyristor Unit
Analog input wiring differs by the thyristor unit and operation mode that is used.
+
–
Thermocouple
Heater
Adjust the AC line power with phase control.
Fuse
L1
L2
1
2
3
+
–

Ladder program


PID control enable input
I0000
Analog Status to the Analog I/O Cartridge
Analog input status error
CMP>=(W) S1 - D8172 S2 - 5 D1 - M0020 REP
Analog Status to the Analog I/O Cartridge
Analog output status error
CMP>=(W) S1 - D8176 S2 - 5 D1 - M0021 REP
1.
PID control enable input
I0000
Analog input status error
M0020
Analog output status error
M0021
Process variable (after conversion)
Initialize pulse
Control action
Set point
PIDA S1 D0000 S2 M8120 S3 M0000 S4 D0007
Operating status
Operating status error
ICMP>=(W) S1 109 S2 D0002 S3 103 D1 M0022
2.
PID control enable input
I0000
Analog output status error
M0020
Analog input status error
M0021
Analog Value to the Analog I/O Cartridge
MOV(W) S1 - 0 D1 - D8174 REP
3.
PID control enable input
I0000
Operating status
Analog output status error
ICMP>=(W) S1 109 S2 D0002 S3 103 D1 M0021
4.
Analog input status error
M0020
Analog output status error
M0021
PID error
M0022
Alarm 1 output
M0003
Alarm lamp
Q0001
5.

Device Address	Comment
D0000	Process variable (after conversion)
D0002	Operating status
D0007	Set point
D8178	Analog Status to the Analog I/O Cartridge
D8186	Analog Value to the Analog I/O Cartridge
D8188	Analog Status to the Analog I/O Cartridge
I0000	PID control enable input
Q0001	Alarm lamp

Device Address	Comment
M0000	Control action
M0003	Alarm 1 output
M0020	Analog input status error
M0021	Analog output status error
M0022	PID error
M8120	Initialize pulse

1. When I0000 is ON and the analog input status error for the analog I/O cartridge is 5, 6, or 8 to 11, M0020 turns ON. When the analog output status error for the analog I/O cartridge is 8 to 11, M0021 turns ON.
2. The PID instruction operates when I0000 is ON and both M0020 and M0021 are OFF. The operating status of the PID instruction is judged, and when an error occurs, M0022 turns ON.
3. When M0020 or M0021 is ON, or when I0000 is OFF (when the PID instruction is stopped), 0 is stored in D8186 and 0 V is output from AQ4.
4. The operating status of the PID instruction is judged, and when an error occurs, M0021 turns ON.
5. When either M0020, M0021, M0022, or M0003 (Alarm 1 output (process high alarm)) is ON, Q0001 (alarm lamp) turns ON.

Note:

The analog status for the analog I/O cartridge is as follows.

- Analog input status error
  0: Operating normally
  1: Converting data
  2: Initializing
  5, 6: Wiring error
  8 to 11: Analog I/O cartridge error
- Analog output status error
  0: Operating normally
  2: Initializing
  3: Parameter setting error
  5, 6: Wiring error
  8 to 11: Analog I/O cartridge error

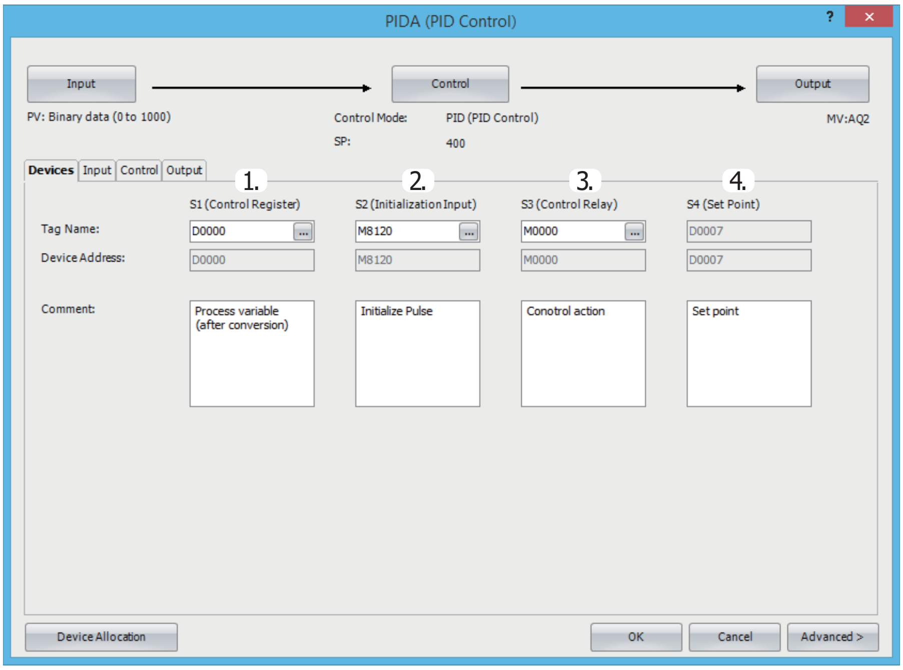
For details, see Chapter 10 "Analog I/O Cartridge" in the "FC6A Series MICROSmart User’s Manual".

PID instruction dialog box configuration procedure

Use the default value for settings that are not described in this procedure.

1. The **Devices** tab configures the devices used with the PID instruction.

- Set S1 (Control Register) to D0000 (1).
- Set S2 (Initialization Input) to M8120 (2).
- Set S3 (Control Relay) to M0000 (3).
- S4 (Set Point) is automatically set to D0007 (4).



PIDA (PID Control)

PV: Binary data (0 to 1000)

Control Mode: PID (PID Control)

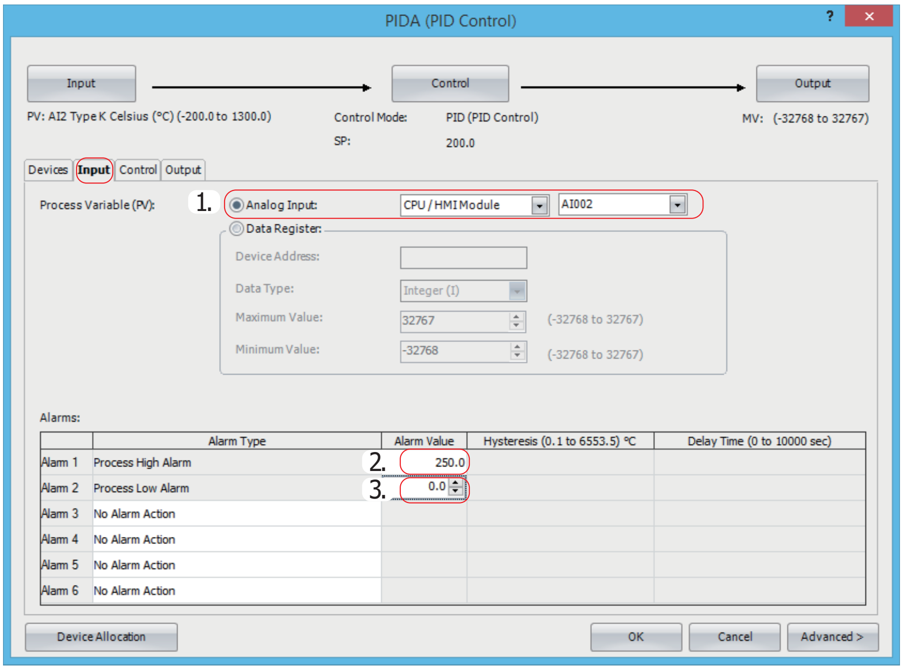
SP: 400

MV:AQ2

	1. S1 (Control Register)	2. S2 (Initialization Input)	3. S3 (Control Relay)	4. S4 (Set Point)
Tag Name:	D0000	M8120	M0000	D0007
Device Address:	D0000	M8120	M0000	D0007
Comment:	Process variable (after conversion)	Initialize Pulse	Conotrol action	Set point

2. Click the **Input** tab and configure the items.

- For Process Variable (PV), select **Analog Input**, and then select **AI002** (1).
- For Alarm 1, set the alarm value for the process high alarm to **250.0** (2).
  For Alarm 2, set the alarm value for the process low alarm to **0.0** (3).



PIDA (PID Control)

PV: AI2 Type K Celsius (°C) (-200.0 to 1300.0)

Control Mode: PID (PID Control)

SP: 200.0

MV: (-32768 to 32767)

Process Variable (PV): 1. ● Analog Input: CPU / HMI Module AI002

○ Data Register:

Device Address:

Data Type: Integer (I)

Maximum Value: 32767 (-32768 to 32767)

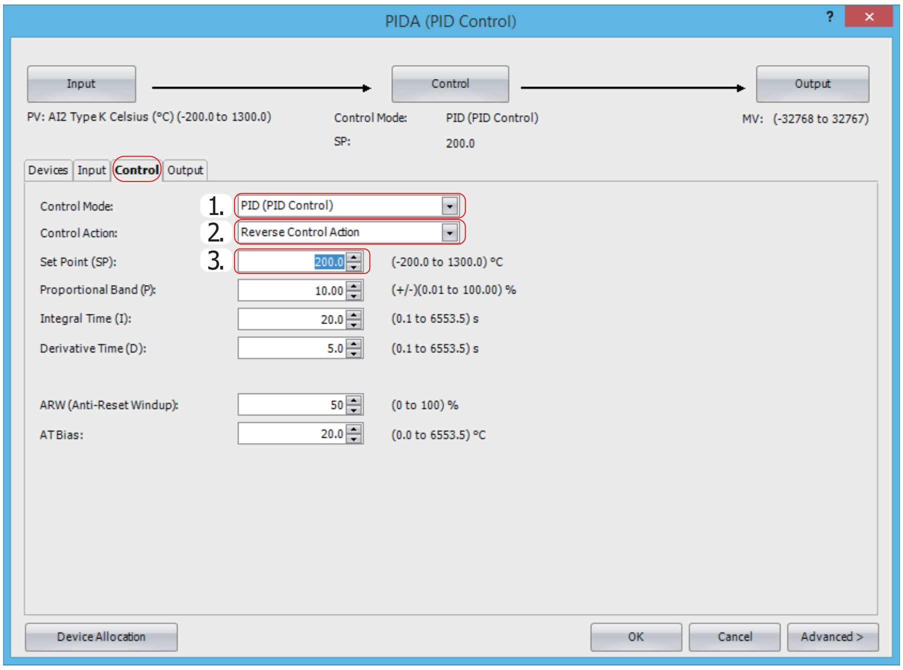
Minimum Value: -32768 (-32768 to 32767)

Alarms:

	Alarm Type	Alarm Value	Hysteresis (0.1 to 6553.5) °C	Delay Time (0 to 10000 sec)
Alarm 1	Process High Alarm	2. 250.0		
Alarm 2	Process Low Alarm	3. 0.0		
Alarm 3	No Alarm Action			
Alarm 4	No Alarm Action			
Alarm 5	No Alarm Action			
Alarm 6	No Alarm Action			

3. Click the **Control** tab and configure the items.

- For Control Mode, select **PID (PID Control)** (1).
- For Control Action, select **Reverse Control Action** (2).
- For Set Point (SP), set **200.0** (3).



PIDA (PID Control)

PV: AI2 Type K Celsius (°C) (-200.0 to 1300.0)

Control Mode: PID (PID Control)

SP: 200.0

MV: (-32768 to 32767)

Control Mode: 1. PID (PID Control)

Control Action: 2. Reverse Control Action

Set Point (SP): 3. 200.0 (-200.0 to 1300.0) °C

Proportional Band (P): 10.00 (+/-)(0.01 to 100.00) %

Integral Time (I): 20.0 (0.1 to 6553.5) s

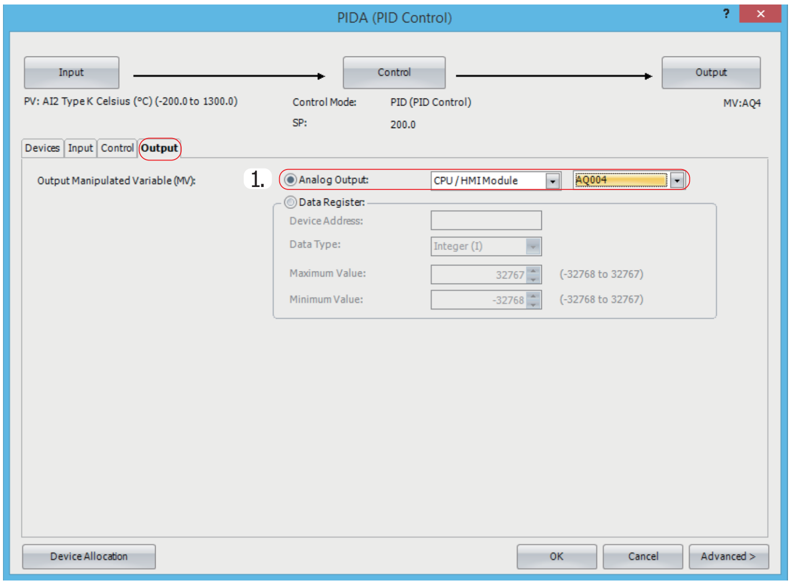
Derivative Time (D): 5.0 (0.1 to 6553.5) s

ARW (Anti-Reset Windup): 50 (0 to 100) %

AT Bias: 20.0 (0.0 to 6553.5) °C

4. Click the **Output** tab and configure the items.

- For Output Manipulated Variable (MV), select **Analog Output**, and then select **AQ004** (1).



PIDA (PID Control)

PV: AI2 Type K Celsius (°C) (-200.0 to 1300.0)

Control Mode: PID (PID Control)

SP: 200.0

MV:AQ4

Output Manipulated Variable (MV): 1. ● Analog Output: CPU / HMI Module AQ004

○ Data Register:

Device Address:

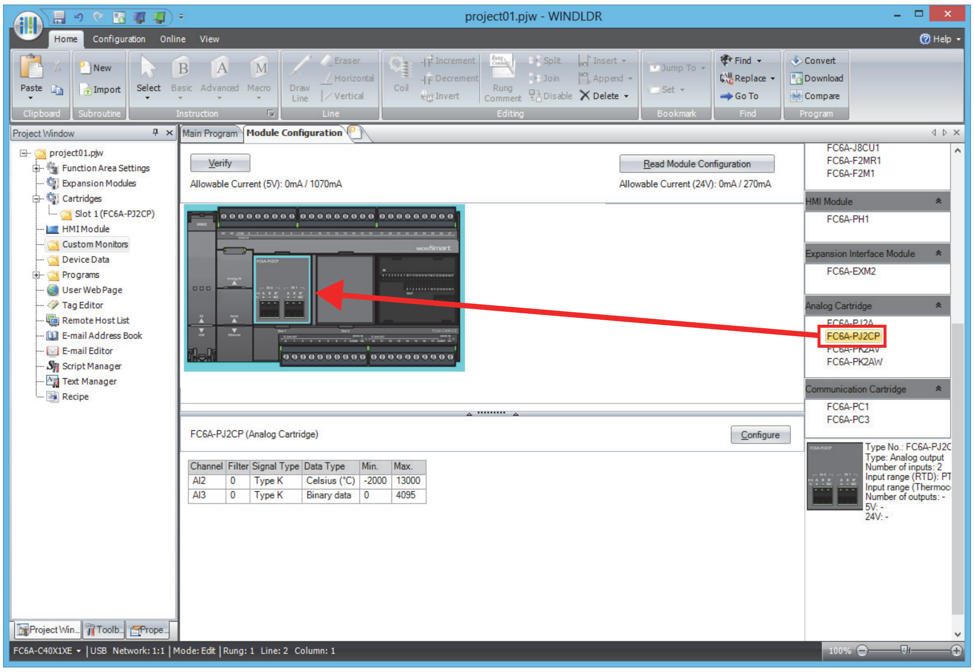
Data Type: Integer (I)

Maximum Value: 32767 (-32768 to 32767)

Minimum Value: -32768 (-32768 to 32767)

5. Use the Module Configuration Editor to configure the analog input for the analog I/O cartridge. On the **Configuration** tab, in the **PLCs** group, select **Expansion Modules**.

6. Select **FC6A-PJ2CP** in the expansion modules and cartridges list and drag and drop it to the module configuration area.



Allowable Current (5V): 0mA / 1070mA

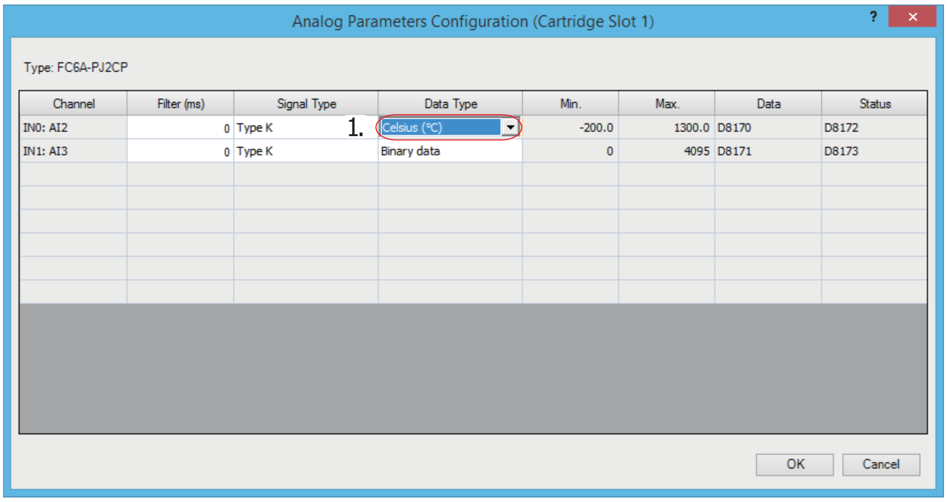
Allowable Current (24V): 0mA / 270mA

FC6A-PJ2CP (Analog Cartridge)

Channel	Filter	Signal Type	Data Type	Min.	Max.
AI2	0	Type K	Celsius (°C)	-2000	13000
AI3	0	Type K	Binary data	0	4095

7. Click the **FC6A-PJ2CP** analog I/O cartridge that was interested in the module configuration area and click **Configure**. The **Analog Parameters Configuration (Cartridge Slot 1)** dialog box is displayed.

8. Configure the items.
   - For channel IN0 : AI2 **Data Type**, select **Celsius (°C)** (1).

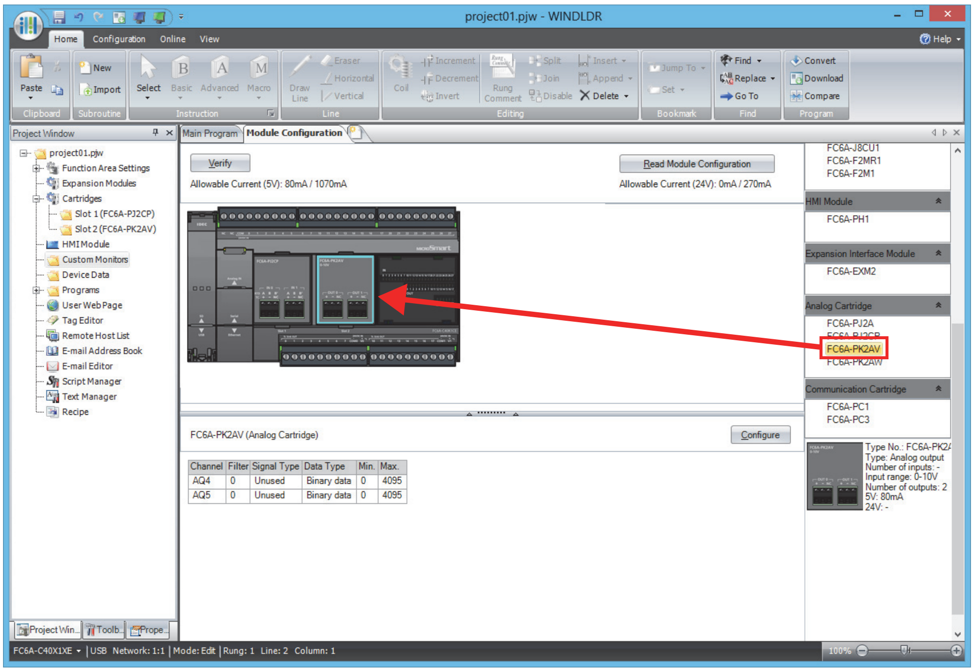


Type: FC6A-PJ2CP

Channel	Filter (ms)	Signal Type	Data Type	Min.	Max.	Data	Status
IN0: AI2	0	Type K	Celsius (°C)	-200.0	1300.0	D8170	D8172
IN1: AI3	0	Type K	Binary data	0	4095	D8171	D8173

9. Click **OK** to close the **Analog Parameters Configuration (Cartridge Slot 1)** dialog box.

10. Select **FC6A-PK2AV** in the expansion modules and cartridges list and drag and drop it to the module configuration area.



project01.pjw - WINDLDR

Verify

Allowable Current (5V): 80mA / 1070mA

Read Module Configuration

Allowable Current (24V): 0mA / 270mA

FC6A-PK2AV (Analog Cartridge)

Configure

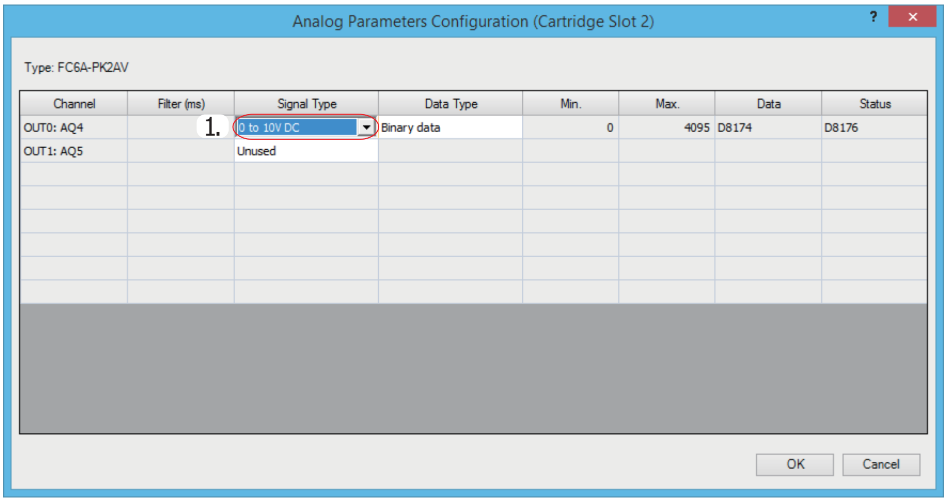
Channel	Filter	Signal Type	Data Type	Min.	Max.
AQ4	0	Unused	Binary data	0	4095
AQ5	0	Unused	Binary data	0	4095

Type No.: FC6A-PK2A
Type: Analog output
Number of inputs: -
Input range: 0-10V
Number of outputs: 2
5V: 80mA
24V: -

11. Click the **FC6A-PK2AV** analog I/O cartridge that was interested in the module configuration area and click **Configure**. The **Analog Parameters Configuration (Cartridge Slot 2)** dialog box is displayed.

12. Configure the items.

- For channel OUT0 : AQ4 **Signal Type**, select **0 to 10V DC** (1).



Analog Parameters Configuration (Cartridge Slot 2)

Type: FC6A-PK2AV

Channel	Filter (ms)	Signal Type	Data Type	Min.	Max.	Data	Status
OUT0: AQ4	1.	0 to 10V DC	Binary data	0	4095	D8174	D8176
OUT1: AQ5		Unused					

13. Click **OK** to close the **Analog Parameters Configuration (Cartridge Slot 2)** dialog box.

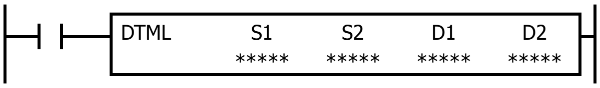
20: Dual / Teaching Timer Instructions

Introduction

This chapter describes the dual/teaching timer instructions that cyclically turn outputs on and off in the specified time interval. Four dual timers are available and the ON/OFF duration can be selected from 1 ms up to 65,535 s.

Teaching timer instruction measures the ON duration of the start input for the teaching timer instruction and stores the measured data to a designated data register, which can be used as a preset value for a timer instruction.

DTML (1-s Dual Timer)

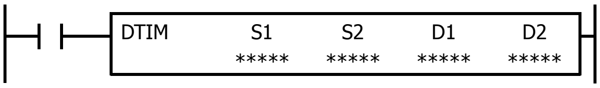


While input is on, destination device D1 repeats to turn on and off for a duration assigned by devices S1 and S2, respectively.

When the input is off, D1 turns off.

The time range is 0 through 65,535 s.

DTIM (100-ms Dual Timer)

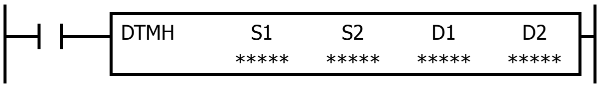


While input is on, destination device D1 repeats to turn on and off for a duration assigned by devices S1 and S2, respectively.

When the input is off, D1 turns off.

The time range is 0 through 6,553.5 s.

DTMH (10-ms Dual Timer)

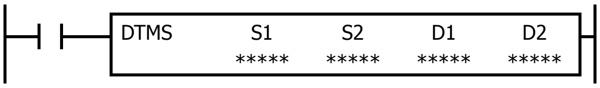


While input is on, destination device D1 repeats to turn on and off for a duration assigned by devices S1 and S2, respectively.

When the input is off, D1 turns off.

The time range is 0 through 655.35 s.

DTMS (1-ms Dual Timer)



While input is on, destination device D1 repeats to turn on and off for a duration assigned by devices S1 and S2, respectively.

When the input is off, D1 turns off.

The time range is 0 through 65.535 s.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant
S1 (Source 1)	ON duration	—	—	—	—	—	—	X	—	0-65,535
S2 (Source 2)	OFF duration	—	—	—	—	—	—	X	—	0-65,535
D1 (Destination 1)	Dual timer output	—	X	▲	X	—	—	—	—	—
D2 (Destination 2)	System work area	—	—	—	—	—	—	X	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

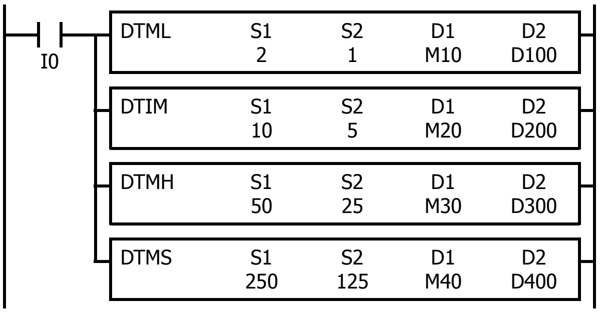
▲ Special internal relays cannot be designated as D1.

Destination device D2 (system work area) uses 2 data registers starting with the device designated as D2. The two data registers are used for a system work area. Do not use these data registers for destinations of other advanced instructions, and do not change values of these data registers using the monitor function on WindLDR. If the data in these data registers are changed, the dual timer does not operate correctly.

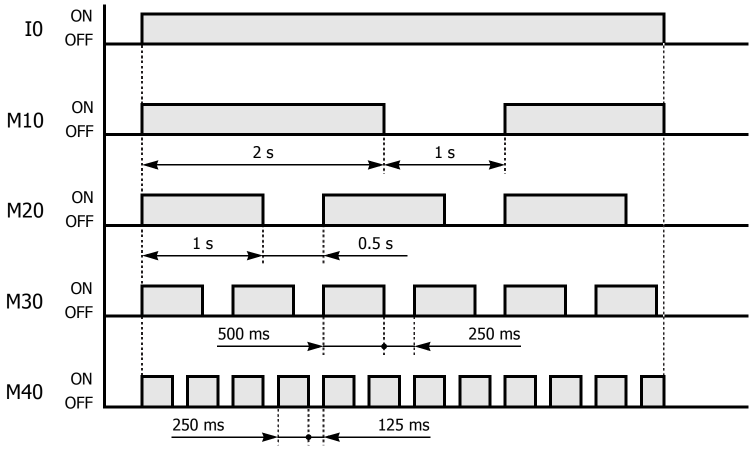
The dual timer instructions cannot be used in an interrupt program. If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled and the next instruction is executed. The data in D1 (destination 1) is unchanged. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Examples: DTML, DTIM, DTMH, DTMS


I0
DTML S1 2 S2 1 D1 M10 D2 D100
DTIM S1 10 S2 5 D1 M20 D2 D200
DTMH S1 50 S2 25 D1 M30 D2 D300
DTMS S1 250 S2 125 D1 M40 D2 D400

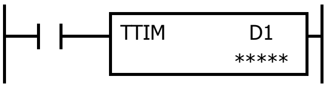
While input I0 is on, four dual timer instructions turn on and off the destination devices according to the on and off durations assigned by source devices S1 and S2.



Instruction	Increments	S1	ON Duration	S2	OFF Duration
DTML	1 s	2	1 s × 2 = 2 s	1	1 s × 1 = 1 s
DTIM	100 ms	10	100 ms × 10 = 1 s	5	100 ms × 5 = 0.5 s
DTMH	10 ms	50	10 ms × 50 = 500 ms	25	10 ms × 25 = 250 ms
DTMS	1 ms	250	1 ms × 250 = 250 ms	125	1 ms × 125 = 125 ms

For the timer accuracy of timer instructions, see "TML, TIM, TMH, and TMS (Timer)" on page 4-7.

TTIM (Teaching Timer)



While input is on, the ON duration is measured in units of 100 ms and the measured value is stored to a data register assigned by destination device D1.

When the input is off, D1 turns off.

The measured time range is 0 through 6,553.5 s.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant
D1 (Destination 1)	Measured value	—	—	—	—	—	—	X	—	—

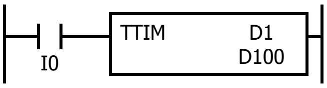
For valid device address ranges, see "Device Addresses" on page 2-1.

Destination device D1 (measured value) uses 3 data registers starting with the device designated as D1. Subsequent two data registers starting with destination device D1+1 are used for a system work area. Do not use these two data registers for destinations of other advanced instructions, and do not change values of these data registers using the monitor function on WindLDR. If the data in these data registers are changed, the teaching timer does not operate correctly.

The teaching timer instruction cannot be used in an interrupt program. If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

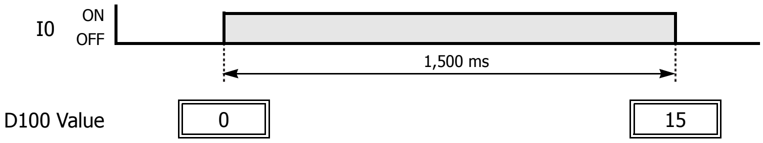
When a user program execution error occurs, the execution of the instruction is canceled and the next instruction is executed. The data in D1 (destination 1) is unchanged. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Examples: TTIM

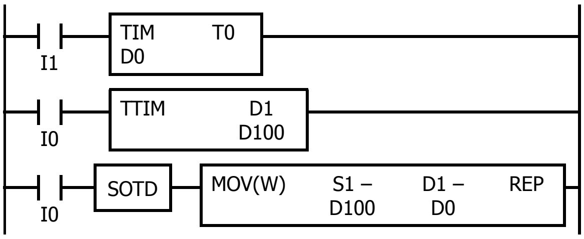


When input I0 is turned on, TTIM resets data register D100 to zero and starts to store the ON duration of input I0 to data register D100, measured in units of 100 ms.

When input I0 is turned off, TTIM stops the measurement, and data register D100 maintains the measured value of the ON duration.



The following example demonstrates a program to measure the ON duration of input I0 and to use the ON duration as a preset value for 100-ms timer instruction TIM.



When input I1 is turned on, 100-ms timer T0 starts to operate with a preset value stored in data register D0.

While input I0 is on, TTIM measures the ON duration of input I0 and stores the measured value in units of 100 ms to data register D100.

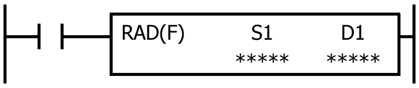
When input I0 is turned off, MOV(W) stores the D100 value to data register D0 as a preset value for timer T0.

21: TRIGONOMETRIC FUNCTION INSTRUCTIONS

Introduction

This chapter describes the trigonometric function instructions that are used to calculate sine, cosine, and tangent from an angle.

RAD (Degree to Radian)



S1·S1+1° × π/180 → D1·D1+1 rad

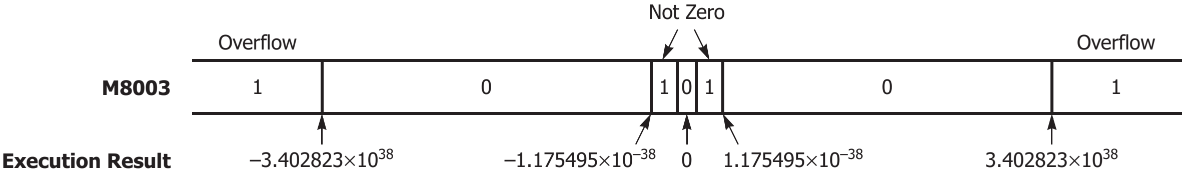
When input is on, the degree value assigned by source device S1 is converted into a radian value and stored to the destination assigned by device D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Degree value to convert into radian	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store conversion results	—	—	—	—	—	—	X	—	—	—

For the valid device address range, see "Device Addresses" on page 2-1.

When the conversion result is not within the range between -3.402823×10^{38} and $-1.175495 \times 10^{-38}$ or between 1.175495×10^{-38} and 3.402823×10^{38}, special internal relay M8003 (carry or borrow) is turned on except when the conversion result is 0. When the conversion result is between $-1.175495 \times 10^{-38}$ and 1.175495×10^{-38}, the destination device designated by D1 stores 0.



When the data designated by S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

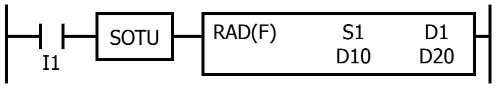
Since the RAD instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	—
I (integer)	—
D (double word)	—
L (long)	—
F (float)	X

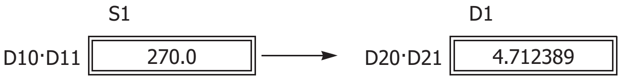
Since floating point data is used, the source and destination devices use two consecutive data registers.

Example: RAD

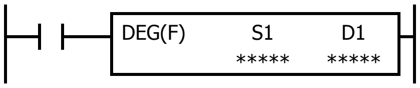


When input I1 is turned on, the degree value of data registers D10 and D11 assigned by source device S1 is converted into a radian value and stored to data registers D20 and D21 assigned by destination device D1.

270° × π/180 → 4.712389 rad



DEG (Radian to Degree)



S1·S1+1 rad × 180/π → D1·D1+1°

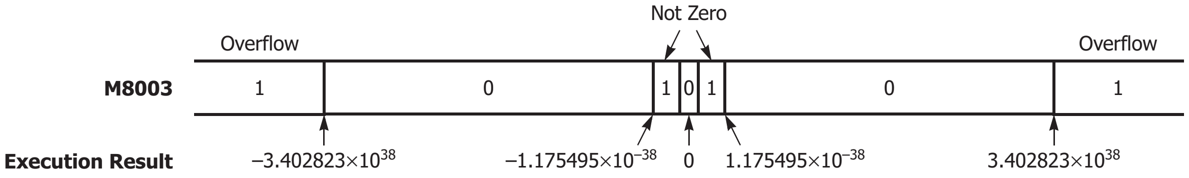
When input is on, the radian value assigned by source device S1 is converted into a degree value and stored to the destination assigned by device D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Radian value to convert into degree	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store conversion results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When the conversion result is not within the range between -3.402823×10^{38} and $-1.175495 \times 10^{-38}$ or between 1.175495×10^{-38} and 3.402823×10^{38}, special internal relay M8003 (carry or borrow) is turned on except when the conversion result is 0. When the conversion result is below -3.402823×10^{38} or over 3.402823×10^{38}, causing an overflow, the destination device designated by D1 stores a value of minus or plus infinity.



When the data designated by S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

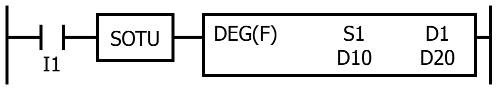
Since the DEG instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	—
I (integer)	—
D (double word)	—
L (long)	—
F (float)	X

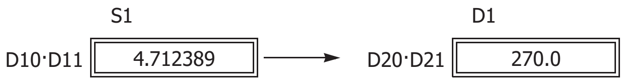
Since floating point data is used, the source and destination devices use two consecutive data registers.

Example: DEG

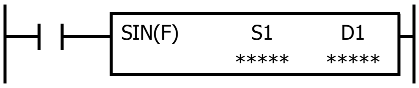


When input I1 is turned on, the radian value of data registers D10 and D11 assigned by source device S1 is converted into a degree value and stored to data registers D20 and D21 assigned by destination device D1.

4.712389 rad × 180/π → 270.0°



SIN (Sine)



sin S1·S1+1 → D1·D1+1

When input is on, the sine of the radian value assigned by source device S1 is stored to the destination assigned by device D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Radian value to convert into sine value	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store conversion results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When the data designated by S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

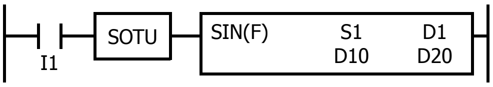
Since the SIN instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	—
I (integer)	—
D (double word)	—
L (long)	—
F (float)	X

Since the floating point data type is used, the source and destination devices use two consecutive data registers.

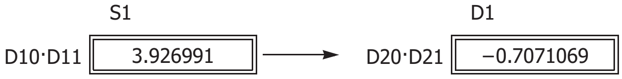
Example: SIN



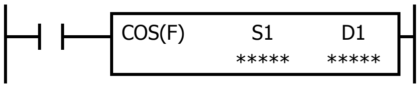
When input I1 is turned on, the sine of the radian value of data registers D10 and D11 assigned by source device S1 is stored to data registers D20 and D21 assigned by destination device D1.

3.926991 rad = 5π/4 rad

sin 5π/4 → −0.7071069



COS (Cosine)



cos S1·S1+1 → D1·D1+1

When input is on, the cosine of the radian value assigned by source device S1 is stored to the destination assigned by device D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Radian value to convert into cosine value	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store conversion results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When the data designated by S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

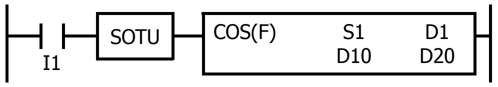
Since the COS instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	—
I (integer)	—
D (double word)	—
L (long)	—
F (float)	X

Since floating point data is used, the source and destination devices use two consecutive data registers.

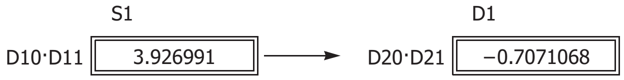
Example: COS



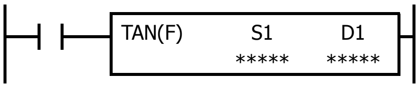
When input I1 is turned on, the cosine of the radian value of data registers D10 and D11 assigned by source device S1 is stored to data registers D20 and D21 assigned by destination device D1.

3.926991 rad = 5π/4 rad

cos 5π/4 → −0.7071068



TAN (Tangent)



tan S1·S1+1 → D1·D1+1

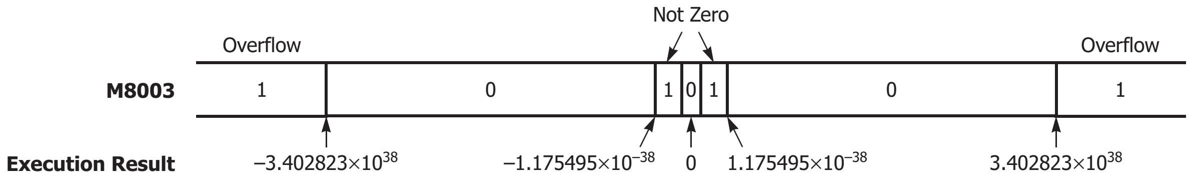
When input is on, the tangent of the radian value assigned by source device S1 is stored to the destination assigned by device D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Radian value to convert into tangent value	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store conversion results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When the conversion result is not within the range between -3.402823×10^{38} and $-1.175495 \times 10^{-38}$ or between 1.175495×10^{-38} and 3.402823×10^{38}, special internal relay M8003 (carry or borrow) is turned on except when the conversion result is 0. When the conversion result is below -3.402823×10^{38} or over 3.402823×10^{38}, causing an overflow, the destination device designated by D1 stores a value of minus or plus infinity.



When the data designated by S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

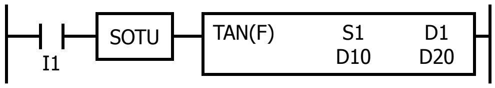
Since the TAN instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	—
I (integer)	—
D (double word)	—
L (long)	—
F (float)	X

Since floating point data is used, the source and destination devices use two consecutive data registers.

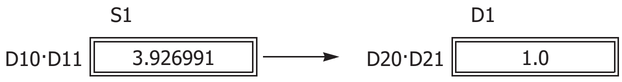
Example: TAN



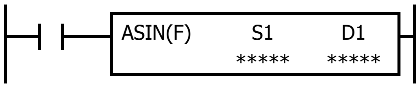
When input I1 is turned on, the tangent of the radian value of data registers D10 and D11 assigned by source device S1 is stored to data registers D20 and D21 assigned by destination device D1.

3.926991 rad = 5π/4 rad

tan 5π/4 → 1.0



ASIN (Arc Sine)



asin S1·S1+1 → D1·D1+1 rad

When input is on, the arc sine of the value assigned by source device S1 is stored in radians to the destination assigned by device D1.

The S1·S1+1 value must be within the following range:

-1.0 ≤ S1·S1+1 ≤ 1.0

If the S1·S1+1 value is not within this range, 0 is stored to D1·D1+1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Arc sine value to convert into radian	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store conversion results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When the data designated by source device S1 is not within the range between –1.0 and 1.0 or does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

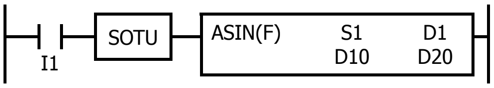
Since the ASIN instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	—
I (integer)	—
D (double word)	—
L (long)	—
F (float)	X

Since floating point data is used, the source and destination devices use two consecutive data registers.

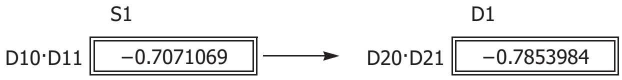
Example: ASIN



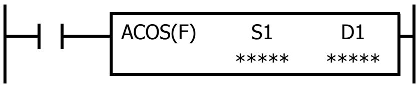
When input I1 is turned on, the arc sine of the value of data registers D10 and D11 assigned by source device S1 is stored to data registers D20 and D21 assigned by destination device D1.

asin −0.7071069 → −0.7853984 rad

−0.7853984 rad = −π/4 rad



ACOS (Arc Cosine)



acos S1·S1+1 → D1·D1+1 rad

When input is on, the arc cosine of the value assigned by source device S1 is stored in radians to the destination assigned by device D1.

The S1·S1+1 value must be within the following range:

$-1.0 \leq$ S1·S1+1 ≤ 1.0

If the S1·S1+1 value is not within this range, 0 is stored to D1·D1+1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Arc cosine value to convert into radian	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store conversion results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When the data designated by source device S1 is not within the range between –1.0 and 1.0 or does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

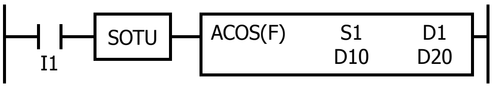
Since the ACOS instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	—
I (integer)	—
D (double word)	—
L (long)	—
F (float)	X

Since floating point data is used, the source and destination devices use two consecutive data registers.

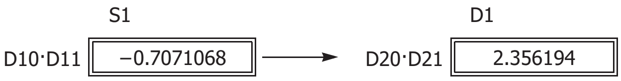
Example: ACOS



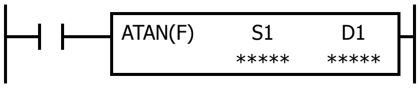
When input I1 is turned on, the arc cosine of the value of data registers D10 and D11 assigned by source device S1 is stored to data registers D20 and D21 assigned by destination device D1.

acos −0.7071068 → 2.356194 rad

2.356194 rad = 3π/4 rad



ATAN (Arc Tangent)



atan S1·S1+1 → D1·D1+1 rad

When input is on, the arc tangent of the value assigned by source device S1 is stored in radians to the destination assigned by device D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Arc tangent value to convert into radian	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store conversion results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When the data designated by source device S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

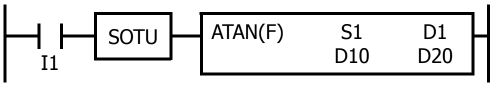
Since the ATAN instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	—
I (integer)	—
D (double word)	—
L (long)	—
F (float)	X

Since floating point data is used, the source and destination devices use two consecutive data registers.

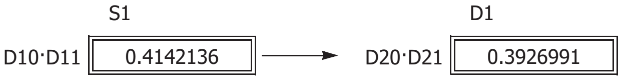
Example: ATAN



When input I1 is turned on, the arc tangent of the value of data registers D10 and D11 assigned by source device S1 is stored to data registers D20 and D21 assigned by destination device D1.

atan 0.4142136 → 0.3926991 rad

0.3926991 rad = π/8 rad

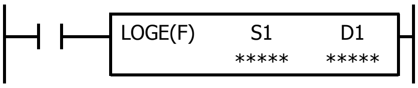


22: LOGARITHM / POWER INSTRUCTIONS

Introduction

This chapter describes logarithm function and power function instructions that are are used to calculate logarithms and powers for specified data.

LOGE (Natural Logarithm)



$\log_e$ S1·S1+1 → D1·D1+1

When input is on, the natural logarithm of the binary data assigned by source device S1 is stored to the destination assigned by device D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Binary data to convert into natural logarithm	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store conversion results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When the operation result is not within the range between -3.402823×10^{38} and $-1.175495 \times 10^{-38}$ or between 1.175495×10^{-38} and 3.402823×10^{38}, special internal relay M8003 (carry or borrow) is turned on except when the result is 0. For details, see "Carry and Borrow in Floating-Point Data Processing" on page 3-8.

When the value of S1 (source 1) is 0, a user program execution error occurs.

When the value of S1 (source 1) is less than 0, a user program execution error occurs.

When the data designated by source device S1 does not comply with the normal floating-point format, a user program execution error occurs, and the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed.

When a user program execution error occurs, special internal relay M8004 and ERR LED on the FC6A Series MICROSmart are turned on. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

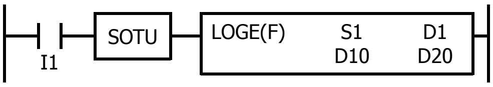
Since the LOGE instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	—
I (integer)	—
D (double word)	—
L (long)	—
F (float)	X

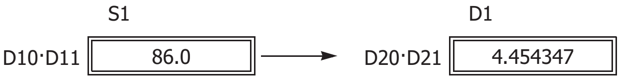
Since floating point data is used, the source and destination devices use two consecutive data registers.

Example: LOGE

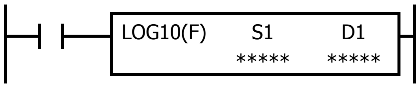


When input I1 is on, the natural logarithm of the binary data of data registers D10 and D11 assigned by source device S1 is stored to data registers D20 and D21 assigned by destination device D1.

$\log_e 86.0 \rightarrow 4.454347$



LOG10 (Common Logarithm)



$\log_{10}$ S1·S1+1 → D1·D1+1

When input is on, the common logarithm of the binary data assigned by source device S1 is stored to the destination assigned by device D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Binary data to convert into common logarithm	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store conversion results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When the operation result is not within the range between -3.402823×10^{38} and $-1.175495 \times 10^{-38}$ or between 1.175495×10^{-38} and 3.402823×10^{38}, special internal relay M8003 (carry or borrow) is turned on except when the result is 0. For details, see "Carry and Borrow in Floating-Point Data Processing" on page 3-8.

When the value of S1 (source 1) is 0, a user program execution error occurs.

When the value of S1 (source 1) is less than 0, a user program execution error occurs.

When the data designated by source device S1 does not comply with the normal floating-point format, a user program execution error occurs, and the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed.

When a user program execution error occurs, special internal relay M8004 and ERR LED on the FC6A Series MICROSmart are turned on. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

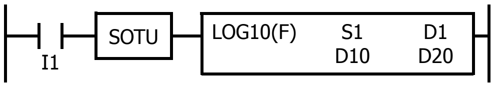
Since the LOG10 instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	—
I (integer)	—
D (double word)	—
L (long)	—
F (float)	X

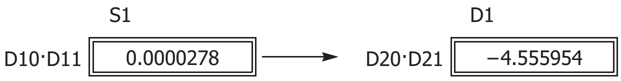
Since floating point data is used, the source and destination devices use two consecutive data registers.

Example: LOG10

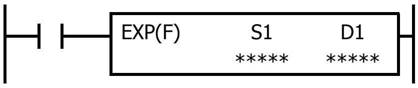


When input I1 is on, the common logarithm of the binary data of data registers D10 and D11 assigned by source device S1 is stored to data registers D20 and D21 assigned by destination device D1.

$\log_{10}$ 0.0000278 → −4.555954



EXP (Exponent)



$e^{S1 \cdot S1+1} \rightarrow$ D1·D1+1

When input is on, e is raised to the power S1·S1+1 assigned by source device S1 and is stored to the destination assigned by device D1.

e (base of natural logarithm) = 2.7182818

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Binary data of exponent	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When the operation result is not within the range between -3.402823×10^{38} and $-1.175495 \times 10^{-38}$ or between 1.175495×10^{-38} and 3.402823×10^{38}, special internal relay M8003 (carry or borrow) is turned on except when the result is 0. For details, see "Carry and Borrow in Floating-Point Data Processing" on page 3-8.

When the operation result is between $-1.175495 \times 10^{-38}$ and 1.175495×10^{-38}, the destination device designated by D1 stores 0. When the operation result is less than -3.402823×10^{38} or larger than 3.402823×10^{38}, causing an overflow, the destination device designated by D1 stores a value of minus or plus infinity.

When the data designated by source device S1 does not comply with the normal floating-point format, a user program execution error occurs, and the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed.

When a user program execution error occurs, special internal relay M8004 and ERR LED on the FC6A Series MICROSmart are turned on. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

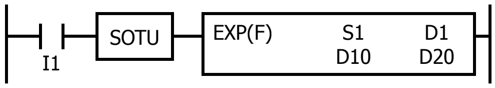
Since the EXP instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	—
I (integer)	—
D (double word)	—
L (long)	—
F (float)	X

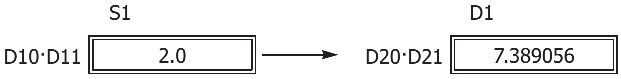
Since floating point data is used, the source and destination devices use two consecutive data registers.

Example: EXP

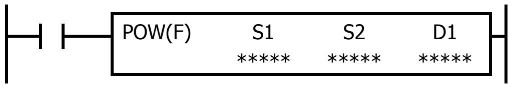


When input I1 is on, e is raised to the data of data registers D10 and D11 assigned by source device S1 and the operation result is stored to data registers D20 and D21 assigned by destination device D1.

$e^2 = 2.7182818^2 \rightarrow$ 7.389056



POW (Power)



$S1{\cdot}S1{+}1^{S2{\cdot}S2+1} \rightarrow D1{\cdot}D1{+}1$

When input is on, data assigned by source device S1 is raised to the power S2·S2+1 assigned by source device S2 and the operation result is stored to the destination assigned by device D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Binary data of base	—	—	—	—	—	—	X	—	X	—
S2 (Source 2)	Binary data of exponent	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

When the operation result is not within the range between -3.402823×10^{38} and $-1.175495 \times 10^{-38}$ or between 1.175495×10^{-38} to 3.402823×10^{38}, special internal relay M8003 (carry or borrow) is turned on, except when the result is 0. For details, see "Carry and Borrow in Floating-Point Data Processing" on page 3-8.

When the operation result is between $-1.175495 \times 10^{-38}$ and 1.175495×10^{-38}, the destination device designated by D1 stores 0. When the operation result is less than -3.402823×10^{38} or greater than 3.402823×10^{38}, causing an overflow, the destination device designated by D1 stores a value of minus or plus infinity.

When one of the following conditions occurs, a user program execution error occurs.

- The data designated by source device S1 is less than 0 and the data designated by source device S2 is not an integer.
- The data designated by source device S1 is 0 and the data designated by source device S2 is less than or equal to 0.

When the data designated by source device S1 or S2 does not comply with the normal floating-point format, a user program execution error occurs, and the execution of the instruction is canceled. The value of D1 is left unchanged and the next instruction is executed.

When a user program execution error occurs, special internal relay M8004 and ERR LED on the FC6A Series MICROSmart are turned on. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

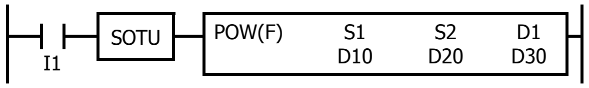
Since the POW instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	—
I (integer)	—
D (double word)	—
L (long)	—
F (float)	X

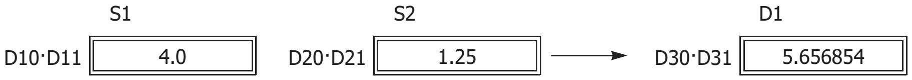
Since floating point data is used, the source and destination devices use two consecutive data registers.

Example: POW



When input I1 is on, the data of data registers D10 and D11 assigned by source device S1 is raised to the power D20·D20+1 assigned by source device S2 and the operation result is stored to data registers D30 and D31 assigned by destination device D1.

$4^{1.25} \rightarrow 5.656854$



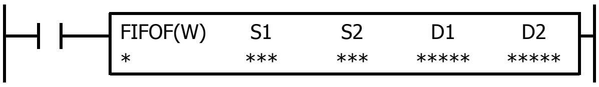
23: File Data Processing Instructions

Introduction

This chapter describes the file data processing instructions that handle the FIFO (first-in first-out) data structure. FIFOF (FIFO Format) instructions initialize the FIFO data files storing the data. FIEX (First-In Execute) instructions store new data to the FIFO data files, and FOEX (First-Out Execute) instructions retrieve the stored data from the FIFO data files. The first data to be stored to the FIFO data files by FIEX instructions will be the first data to be retrieved by FOEX instructions.

NDSRC (N Data Search) instructions search a designated value through a specified range.

FIFOF (FIFO Format)



When input is on, FIFOF instruction initializes an FIFO data file. Each data file has unique number 0 through 9. A maximum of 10 data files can be used in a user program.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
N (File Number)	File Number	—	—	—	—	—	—	—	—	0-9	—
S1 (Source 1)	Quantity of data registers per record	—	—	—	—	—	—	—	—	1-255	—
S2 (Source 2)	Quantity of records	—	—	—	—	—	—	—	—	2-255	—
D1 (Destination 1)	First data register to store FIFO data file	—	—	—	—	—	—	X	—	—	—
D2 (Destination 2)	FIFO status output	—	—	▲	—	—	—	—	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

▲ Special internal relays cannot be designated as D2.

Since the FIFOF instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	X
I (integer)	—
D (double word)	—
L (long)	—
F (float)	—

When an M (internal relay) is assigned as the D2, three internal relays starting with the device assigned by D2 are used.

When a D (data register) is assigned as the D1, S1×S2+2 data registers starting with the device assigned by D1 are used.

Destination Device D1 (FIFO Data File)

FIFO data files are initialized when corresponding FIFOF instructions are executed. FIFO data file is placed in the area starting with the device designated by D1 and occupies as many as S1×S2+2 data registers. The size of each record is equal to S1. S1-1 records of data can be stored in an FIFO data file using FIEX instructions. The stored data can be retrieved from the FIFO data file using FOEX instructions.

Device	Function	Description
D1+0	FI pointer	The FI pointer indicates the position to store new data into the FIFO data file. When an FIEX instruction is executed, the new data in data registers starting with the device designated by S1 of the FIEX instruction is stored at the position indicated by the FI pointer, and the FI pointer is incremented by 1 to indicate the position to store the next data. When the FI pointer indicates the last record of the FIFO data file, and an FIEX instruction is executed, the FI pointer will return to 0.
D1+1	FO pointer	The FO pointer indicates the position to retrieve the stored data from the FIFO data file. When an FOEX instruction is executed, the data at the position indicated by the FIFO pointer is retrieved and stored to the data registers starting with the device designated by D1 of the FOEX instruction, and the FO pointer is incremented by 1 to indicate the position to retrieve the next data. When the FO pointer indicates the last record of the FIFO data file, and an FOEX instruction is executed, the FO pointer will return to 0.
D1+2	Record 0	The first record to store the data.
...		
D1+(S1+1)		
D1+(S1+2)	Record 1	The second record to store the data.
...		
D1+(S1×2+1)		
D1+(S1×(S2–1)+2)	Record S2–1	The last record to store the data.
...		
D1+(S1×S2+1)		

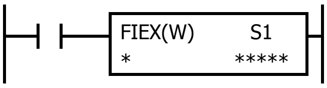
Destination Device D2 (FIFO Status Output)

When FIEX or FOEX instructions are executed, the following internal relays are turned on or off according to the execution status.

D2 Assignments	Function	Description
D2+0	Data file full output	When the value stored in the FI pointer (D1+0) is equal to the value stored in the FO pointer (D1+1) – 1, the FIFO data file is full, and no more data can be stored. If an FIEX instruction is executed when the FIFO data file is full, no operation is executed, and the data file full output (D2+0) will be turned on.
D2+1	Data file empty output	When the value stored in the FI pointer (D1+0) is equal to the value stored in the FO pointer (D1+1), the FIFO data file is empty. If an FOEX instruction is executed when the FIFO data file is empty, no operation is executed, and the data file empty output (D2+1) will be turned on.
D2+2	Pointer out of range output	The value stored in the FI or FO pointer can be 0 through S2–1. When an FIEX or FOEX instruction is executed while the FI or FO pointer value exceeds the valid range, no operation is executed, and the pointer out of range output (D2+2) will be turned on.

Note: The FIFOF instruction cannot be used in an interrupt program.
If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

FIEX (First-In Execute)



When input is on, the data stored in data registers starting with the device assigned by S1 is stored to the corresponding FIFO data file.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
N (File Number)	File number	—	—	—	—	—	—	—	—	0-9	—
S1 (Source 1)	First data register to store data to FIFO data file	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

Make sure that FIEX instructions are executed after the corresponding FIFOF instruction has initialized the FIFO data file. If FIEX instructions are executed without executing the corresponding FIFOF instruction, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled and the next instruction is executed. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

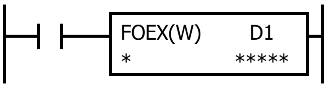
Since the FIEX instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	X
I (integer)	—
D (double word)	—
L (long)	—
F (float)	—

When a D (data register) is assigned as the source, as many data registers as the value stored in device S1 of the corresponding FIFOF instruction are used.

FOEX (First-Out Execute)



When input is on, the data is retrieved from the corresponding FIFO data file and stored to the data registers starting with the device assigned by D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
N (File Number)	File number	—	—	—	—	—	—	—	—	0-9	—
D1 (Destination 1)	First data register number to store data	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

Make sure that FOEX instructions are executed after the corresponding FIFOF instruction has initialized the FIFO data file. If FOEX instructions are executed without executing the corresponding FIFOF instruction, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled and the next instruction is executed. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Since the FOEX instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	X
I (integer)	—
D (double word)	—
L (long)	—
F (float)	—

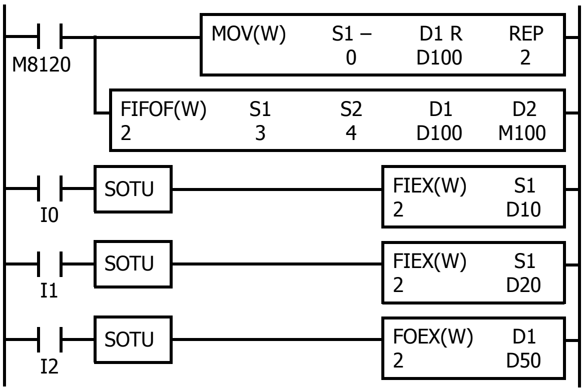
When a D (data register) is assigned as the destination, as many data registers as the value stored in device S1 of the corresponding FIFOF instruction are used.

Example: FIFOF, FIEX, and FOEX

This program demonstrates a user program of the FIFOX, FIEX, and FOEX instructions to use an FIFO data file.

File number:	2
Quantity of data registers per record:	3
Quantity of records:	4
FIFO Data file:	D100 through D113 (3×4+2 data registers)
FIFO status outputs:	M100 through M102

Ladder Diagram



M8120 is the initialize pulse special internal relay.

When the CPU starts, MOV sets 0 to FI and FO pointers, and FIFOF initializes FIFO data file 2.

When input I0 is turned on, the data in D10 through D12 are stored to the FIFO data file 2.

When input I1 is turned on, the data in D20 through D22 are stored to the FIFO data file 2.

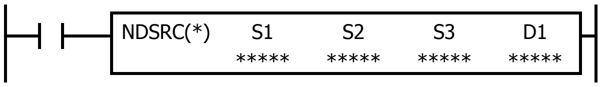
When input I2 is turned on, the first data is retrieved from the FIFO data file 2 and stored to D50 through D52.

FIFO Data File

The table below shows the data stored in FIFO data file 2 when inputs I0, I1, and I2 are turned on in this order. Only valid data managed by the FIFOF, FIEX, and FOEX instructions are shown in the table.

Function	Device Address	Input I0	Input I1	Input I2
FI Pointer	D100	1	2	2
FO Pointer	D101	0	0	1
Record 0	D102 through D104	D10, D11, D12	D10, D11, D12	—
Record 1	D105 through D107	—	D20, D21, D22	D20, D21, D22
Record 2	D108 through D110	—	—	—
Record 3	D111 through D113	—	—	—

NDSRC (N Data Search)



When input is on, a value specified by device S1 is searched for. Data registers are searched, starting with the data register assigned by device S2. Device S3 specifies the quantity of 1-word or 2-word blocks of data registers to search, depending on the data type.

The offset of the data register where a match first occurred is stored in data register assigned by device D1. The number of matches is stored in the next data register. When the search results in no match, 65,535 is stored in device D1 and 0 is stored in device D+1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Value to be sought	—	—	—	—	—	—	X	—	X	—
S2 (Source 2)	First data register number to search	—	—	—	—	—	—	X	—	—	—
S3 (Source 3)	Quantity of blocks to search	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Search result	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

Source S1 data specifies the value to be sought and the valid range depends on the data type.

The search range cannot straddle data registers and non-retentive data registers. Make sure that the sum of data register numbers designated by S1 and S2 does not result in a different data register range.

For source S3 and destination D1, 1 word is always used without regard to the data type.

Destination D1 occupies two consecutive data registers starting with the device designated by D1.

When F (float) data is selected and S1 does not comply with the normal floating-point format, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. When a user program execution error occurs, the execution of the instruction is canceled and the next instruction is executed.

When data in S2 to S2+(S3) does not comply with floating-point format, the data is skipped to continue the search and the results are stored in D1 and D1+1.

The NDSRC instruction cannot be used in an interrupt program.

If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When S3 is zero or the sum of S2 and S3 is not within the valid data register range, a user program execution error will result, turning on special internal relay M8004 and ERR LED on the FC6A Series MICROSmart. When a user program execution error occurs, the execution of the instruction is canceled and the next instruction is executed.

For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Since the NDSRC instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Valid Data Types

W (word)	X
I (integer)	X
D (double word)	X
L (long)	X
F (float)	X

When a word device such as D (data register) is assigned as the source, 1 point (word or integer data) or 2 points (double-word, long, or float data) are used.

Quantity of Source and Destination Devices

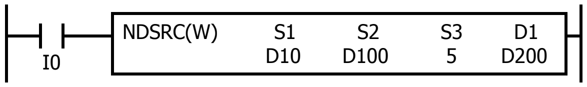
Depending on the data type, source devices S1 and S2 use a different quantity of devices. Source device S3 and destination device D1 always use 1 word without regards to the data type.

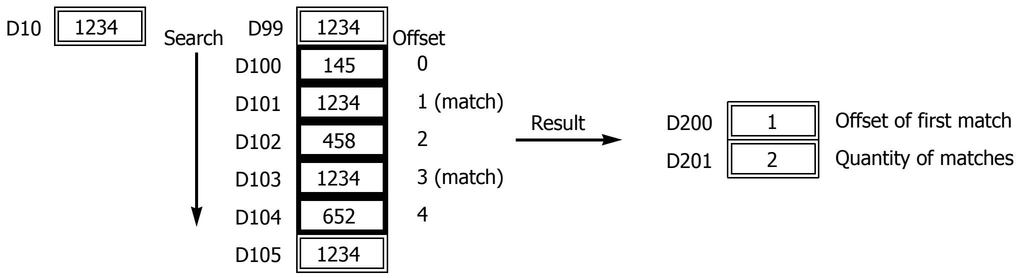
Device	W (word), I (integer)	D (double word), L (long), F (float)
S1, S2	1 word device	2 word devices
S3, D1	1 word device	1 word device

Examples: NDSRC

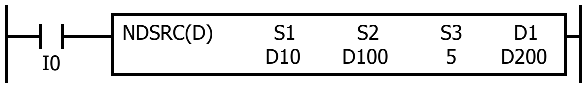
The following examples demonstrate the NDSRC instruction that searches the data of three different data types.

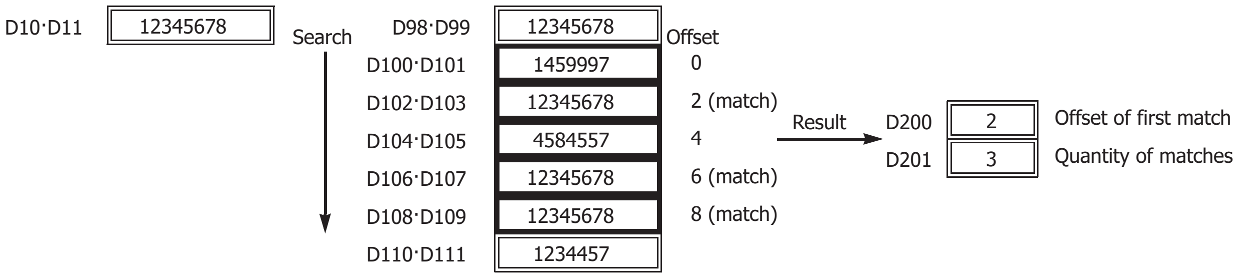
- **Data Type: Word**



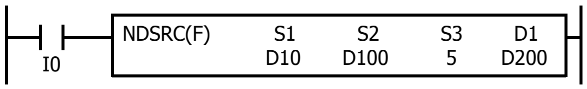


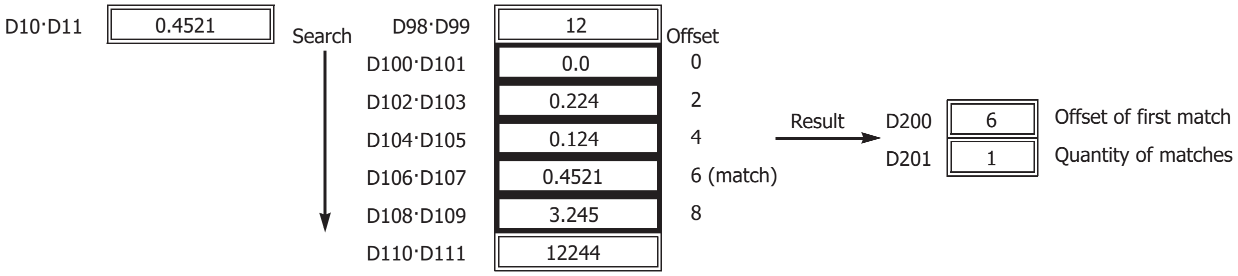
- **Data Type: Double Word**





- **Data Type: Float**





24: Clock Instructions

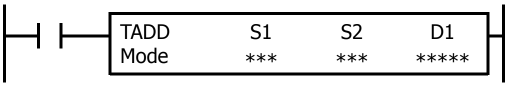
Introduction

TADD (time addition) and TSUB (time subtraction) instructions add or subtract time data in two different modes. The data can be selected from time (hour, minute, and second) or date/time (year, month, day, day of week, hour, minute, and second).

HTOS (HMS to s) and STOH (s to HMS) instructions perform conversion of time data between hours, minutes, seconds and seconds.

HOUR (hour meter) instruction measures the on duration of the input and compares the total duration to a preset value. When the preset value is reached, an output or internal relay is turned on.

TADD (Time Addition)



S1 + S2 → D1, CY

When input is on, time data assigned by source device S2 are added to date/time data assigned by source device S1, depending on the selected mode. The result is stored to destination device D1 and carry (M8003).

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
Mode	Selection of S1 data range	—	—	—	—	—	—	—	—	0, 1	—
S1 (Source 1)	Date/time data to add to	—	—	—	—	—	—	X	—	—	—
S2 (Source 2)	Time data to add	—	—	—	—	—	—	X	—	—	—
D1 (Destination 1)	Destination to store results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

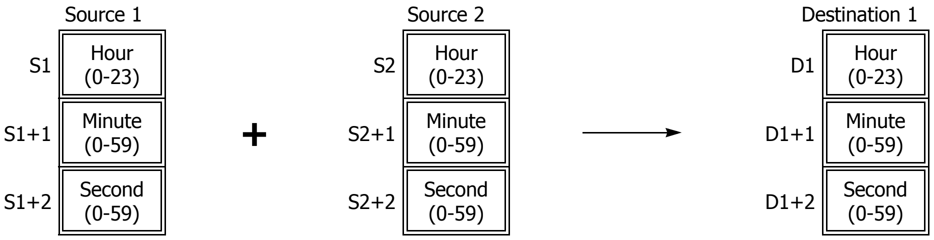
When Mode 0 is selected, source devices S1 and S2 and destination device D1 occupy 3 consecutive data registers starting with the designated device.

When Mode 1 is selected, source device S1 and destination device D1 occupy 7 consecutive data registers starting with the designated device. Source device S2 occupies 3 consecutive data registers starting with the designated device.

Since the TADD instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Mode 0

When mode 0 is selected, time data (hour, minute, and second) stored in 3 data registers starting with source device S2 are added to the time data (hour, minute, and second) stored in 3 data registers starting with source device S1. The results are stored to 3 data registers starting with destination device D1.



Hour data can be 0 through 23. Minute and second data can be 0 through 59.

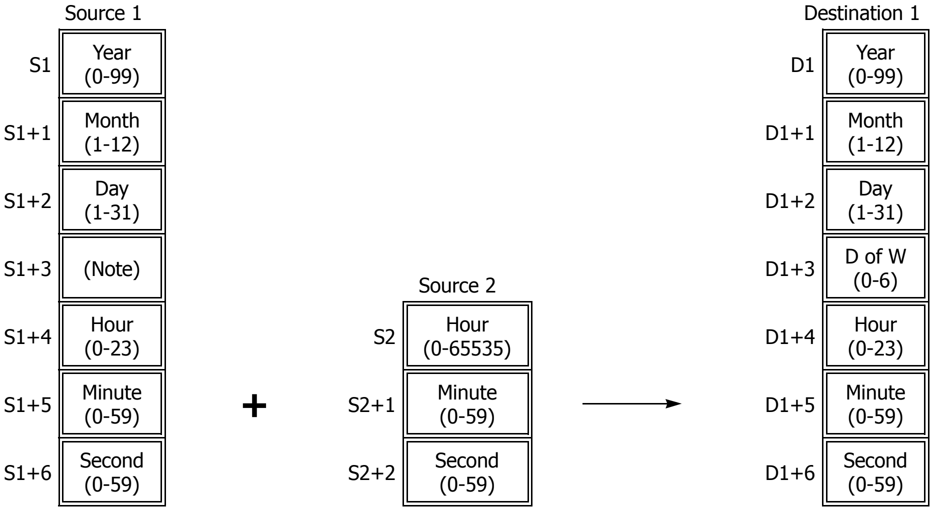
When the execution result exceeds 23:59:59, the result is subtracted by 24 hours and stored to the data register designated by destination device D1, turning on special internal relay M8003 (carry).

When any of the hour, minute, or second data exceeds the valid range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled and the next instruction is executed. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Mode 1

When mode 1 is selected, time data (hour, minute, and second) stored in 3 data registers starting with source device S2 are added to the date/time data (year, month, day, day of week, hour, minute, and second) stored in 7 data registers starting with source device S1. The results are stored to 7 data registers starting with destination device D1.



Note: Device S1+3 in source 1 is not used for execution and need not be designated.

TADD instruction supports leap years.

For source 1: Year data can be 0 through 99. Month data 1 through 12. Day data 1 through 31. Hour data 0 through 23. Minute and second data 0 through 59.

Year data 0 through 99 is processed as year 2000 through 2099.

For source 2: Hour data can be 0 through 65,535. Minute and second data can be 0 through 59.

Destination 1: The day of week is calculated automatically from the resultant year, month, and day, and stored to device D1+3.

Day of week data represent: 0 (Sunday), 1 (Monday), 2 (Tuesday), 3 (Wednesday), 4 (Thursday), 5 (Friday), and 6 (Saturday).

When source 1 contains invalid day/time data, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

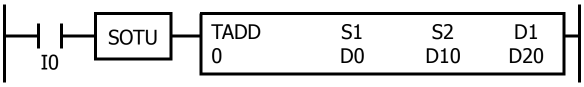
When the execution result exceeds 99 year 12 month 31 day 23:59:59, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

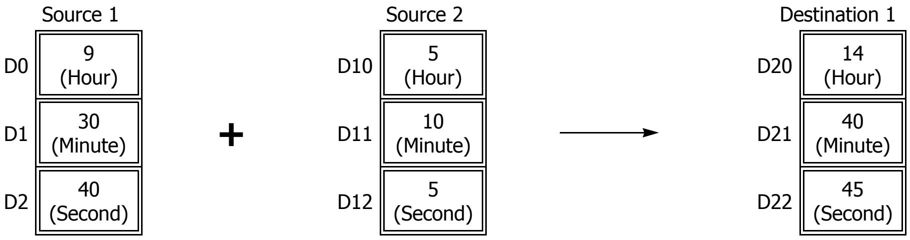
When a user program execution error occurs, the execution of the instruction is canceled and the next instruction is executed. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Examples: TADD

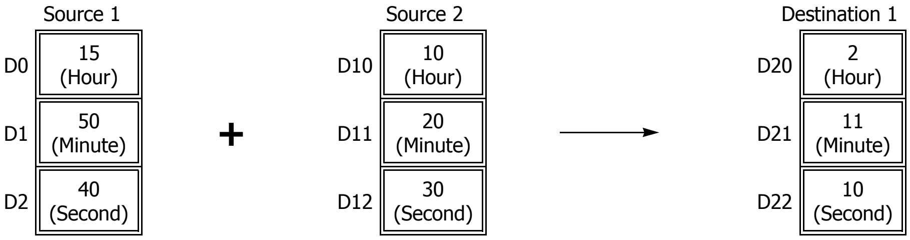
The following examples demonstrate the TADD instruction that will add time data in two different modes.

- **Mode 0**

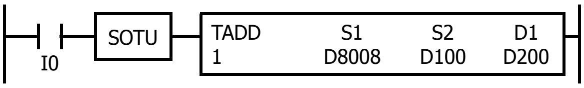


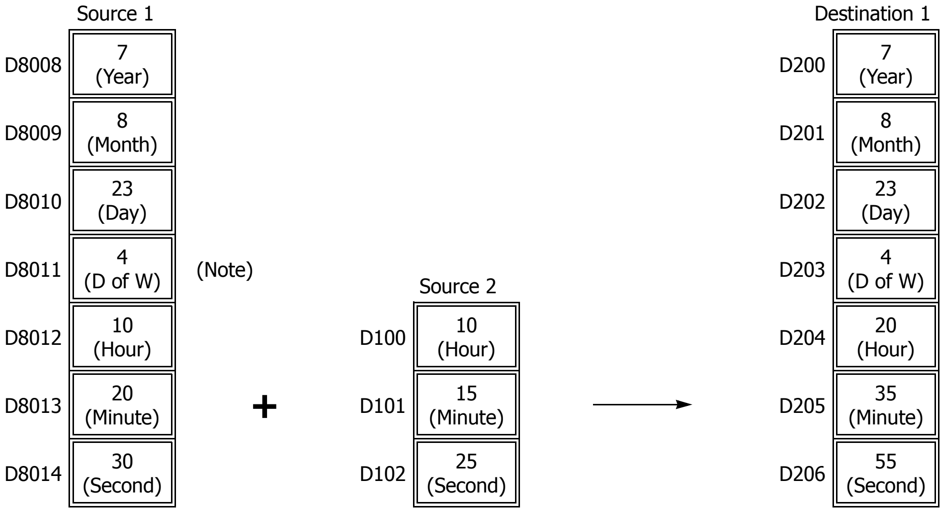


When the result exceeds 23:59:59, the resultant hour data is subtracted by 24, turning on special internal relay M8003 (carry).



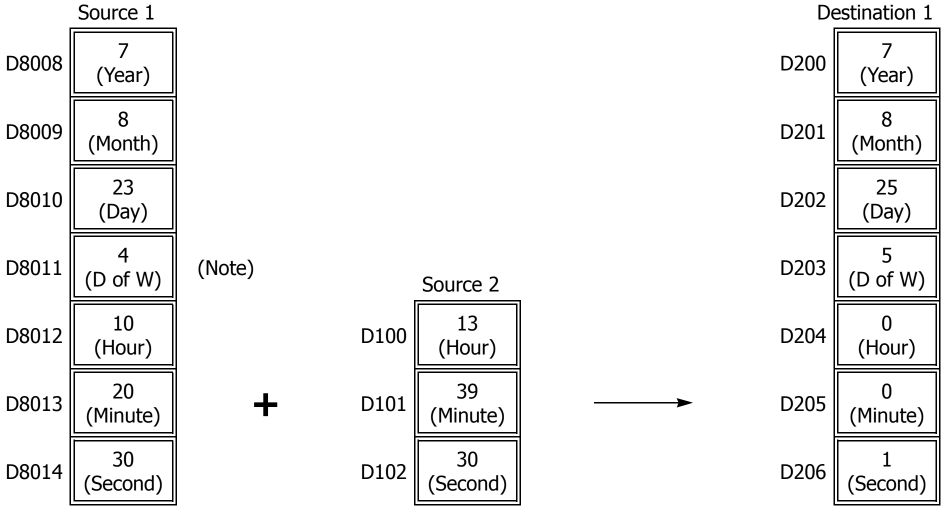
- **Mode 1**





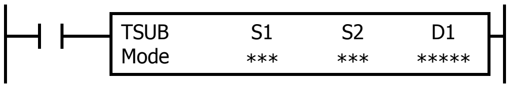
Note: D8011 in source 1 is not used for execution and does not need to be designated. The day of week is calculated automatically from the resultant year, month, and day, and stored to D203 of destination 1.

When the result exceeds 23:59:59, the resultant hour data is subtracted by a multiple of 24 and the day data is incremented.



Note: D8011 in source 1 is not used for execution and does not need to be designated. The day of week is calculated automatically from the resultant year, month, and day, and stored to D203 of destination 1.

TSUB (Time Subtraction)



S1 – S2 → D1, CY

When input is on, time data assigned by source device S2 is subtracted from date/time data assigned by source device S1, depending on the selected mode. The result is stored to destination device D1 and borrow (M8003).

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
Mode	Selection of S1 data range	—	—	—	—	—	—	—	—	0, 1	—
S1 (Source 1)	Date/time data to subtract from	—	—	—	—	—	—	X	—	—	—
S2 (Source 2)	Time data to subtract	—	—	—	—	—	—	X	—	—	—
D1 (Destination 1)	Destination to store results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

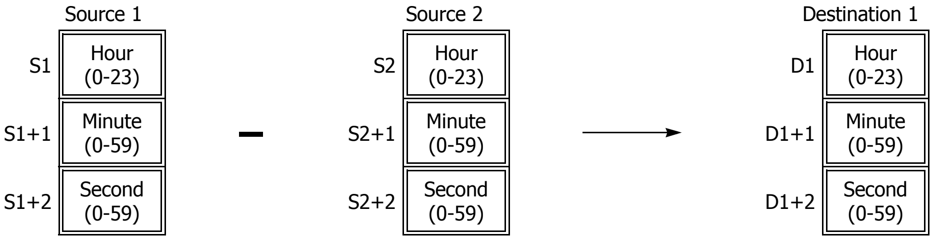
When Mode 0 is selected, source devices S1 and S2 and destination device D1 occupy 3 consecutive data registers starting with the designated device.

When Mode 1 is selected, source device S1 and destination device D1 occupy 7 consecutive data registers starting with the designated device. Source device S2 occupies 3 consecutive data registers starting with the designated device.

Since the TSUB instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Mode 0

When mode 0 is selected, time data (hour, minute, and second) stored in 3 data registers starting with source device S2 is subtracted from the time data (hour, minute, and second) stored in 3 data registers starting with source device S1. The results are stored to 3 data registers starting with destination device D1.



Hour data can be 0 through 23. Minute and second data can be 0 through 59.

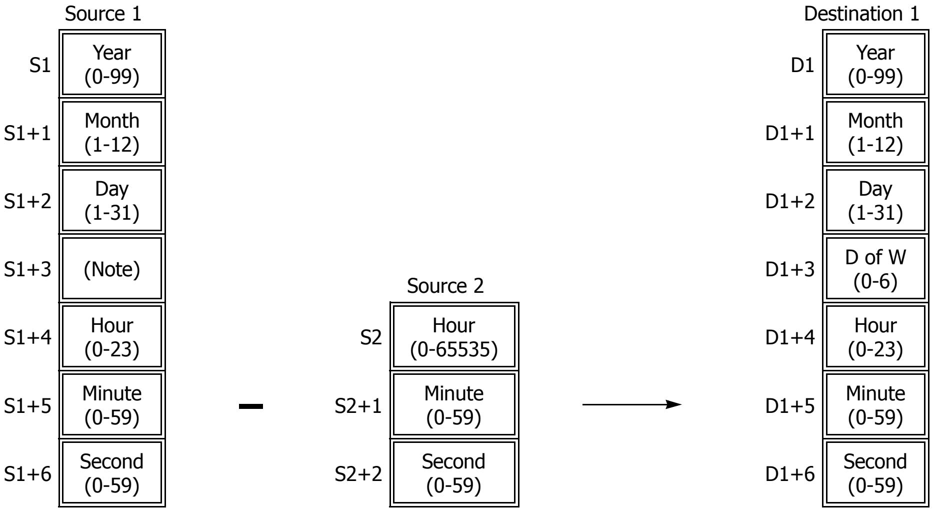
When the execution result is less than 00:00:00, 24 hour is added to the result and stored to the data register designated by destination device D1, turning on special internal relay M8003 (borrow).

When any of the hour, minute, or second data exceeds the valid range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

When a user program execution error occurs, the execution of the instruction is canceled and the next instruction is executed. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Mode 1

When mode 1 is selected, time data (hour, minute, and second) stored in 3 data registers starting with source device S2 is subtracted from the date/time data (year, month, day, day of week, hour, minute, and second) stored in 7 data registers starting with source device S1. The results are stored to 7 data registers starting with destination device D1.



Note: Device S1+3 in source 1 is not used for execution and need not be designated.

TSUB instruction supports leap years.

For source 1: Year data can be 0 through 99. Month data 1 through 12. Day data 1 through 31. Hour data 0 through 23. Minute and second data 0 through 59.

Year data 0 through 99 is processed as year 2000 through 2099.

For source 2: Hour data can be 0 through 65,535. Minute and second data can be 0 through 59.

Destination 1: The day of week is calculated automatically from the resultant year, month, and day, and stored to device D1+3.

Day of week data represent: 0 (Sunday), 1 (Monday), 2 (Tuesday), 3 (Wednesday), 4 (Thursday), 5 (Friday), and 6 (Saturday).

When source 1 contains invalid day/time data, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

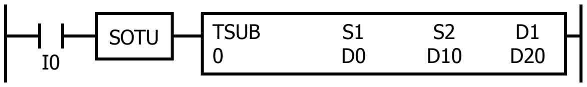
When the execution result is less than 00 year 1 month 1 day 00:00:00, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart.

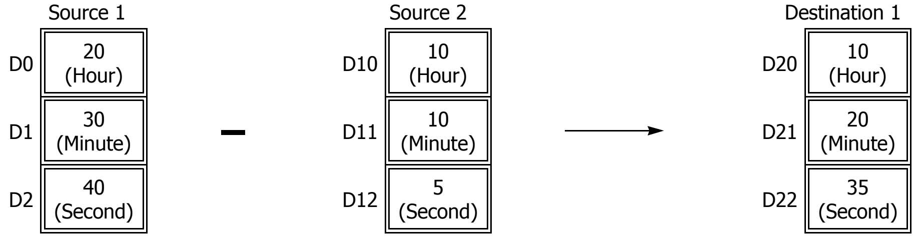
When a user program execution error occurs, the execution of the instruction is canceled and the next instruction is executed. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

Examples: TSUB

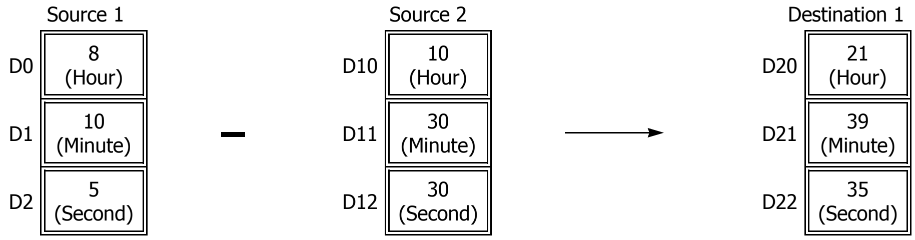
The following examples demonstrate the TSUB instruction to that will subtract time data in two different modes.

- **Mode 0**

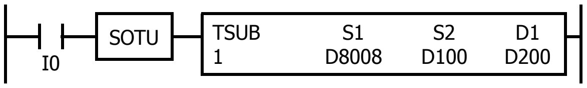


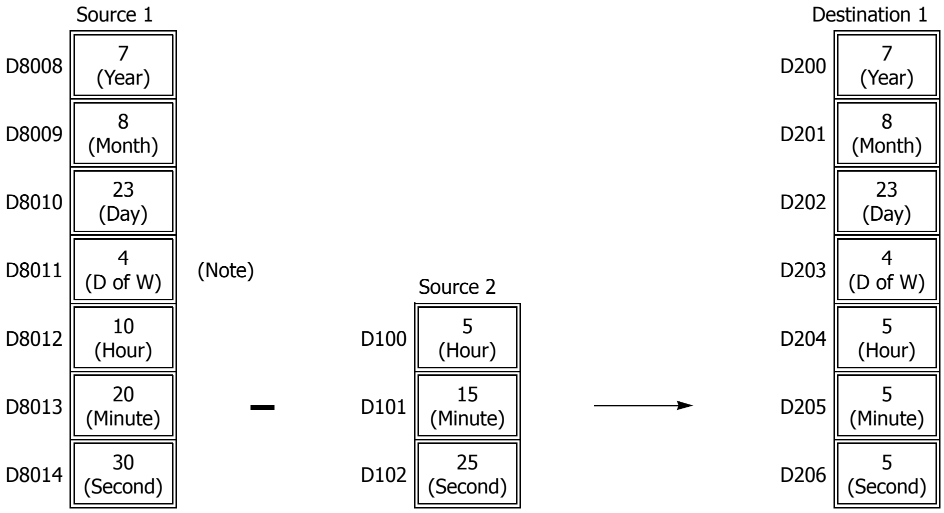


When the result is less than 00:00:00, 24 is added to the result, turning on special internal relay M8003 (borrow).



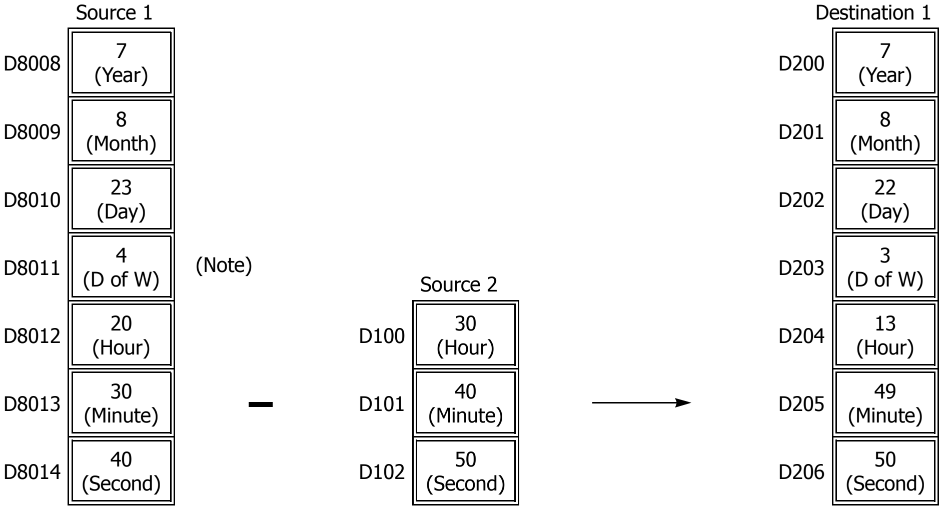
- **Mode 1**





Note: D8011 in source 1 is not used for execution and does not need to be designated. The day of week is calculated automatically from the resultant year, month, and day, and stored to D203 of destination 1.

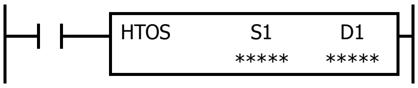
When the result is less than 00:00:00, 24 is added to the result, and the day data is decremented.



Source 1				Source 2			Destination 1	
D8008	7 (Year)						D200	7 (Year)
D8009	8 (Month)						D201	8 (Month)
D8010	23 (Day)						D202	22 (Day)
D8011	4 (D of W)	(Note)					D203	3 (D of W)
D8012	20 (Hour)			D100	30 (Hour)		D204	13 (Hour)
D8013	30 (Minute)		−	D101	40 (Minute)	→	D205	49 (Minute)
D8014	40 (Second)			D102	50 (Second)		D206	50 (Second)

Note: D8011 in source 1 is not used for execution and does not need to be designated. The day of week is calculated automatically from the resultant year, month, and day, and stored to D203 of destination 1.

HTOS (HMS to Sec)



Hours, minutes, seconds → Seconds

When input is on, time data in hours, minutes, and seconds assigned by source device S1 is converted into seconds. The result is stored to destination device D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Time data in hours, minutes, seconds	—	—	—	—	—	—	X	—	—	—
D1 (Destination 1)	Destination to store results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

Source device S1 occupies 3 consecutive data registers starting with the designated device.

Destination device D1 occupies 2 consecutive data registers to store double-word data, starting with the designated device.

Hour data can be 0 through 65,535. Minute and second data can be 0 through 59.

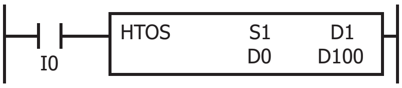
When any of the hour, minute, or second data exceeds the valid range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. When a user program execution error occurs, the execution of the instruction is canceled and the next instruction is executed.

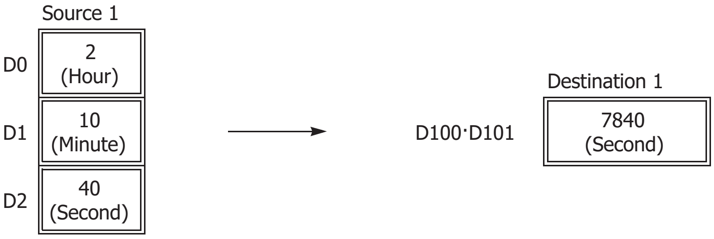
When the execution of the instruction is canceled, the data in D1 and D1+1 is left unchanged. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.

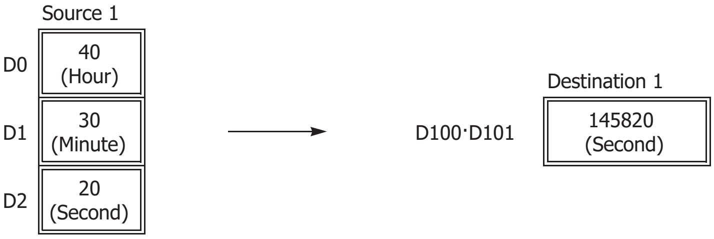
Since the HTOS instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Examples: HTOS

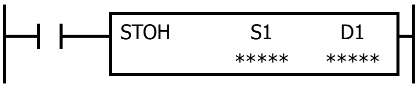
The following examples demonstrate the HTOS instruction that will convert time data in hours, minutes, and seconds into seconds and store the results to two consecutive data registers.







STOH (S to HMS)



Seconds → Hours, minutes, seconds

When input is on, time data in seconds assigned by source device S1 is converted into hours, minutes, and seconds. The result is stored to destination device D1.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Time data in seconds	—	—	—	—	—	—	X	—	X	—
D1 (Destination 1)	Destination to store results	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

Source device S1 occupies 2 consecutive data registers to store double-word data, starting with the designated device.

Destination device D1 occupies 3 consecutive data registers starting with the designated device.

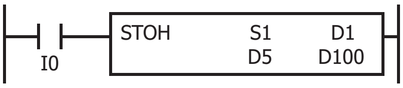
Second data for source device S1 can be 0 through 4,294,967,295.

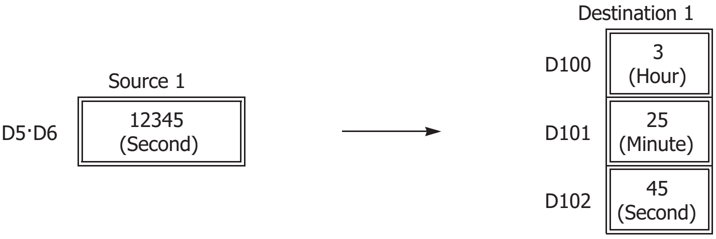
When the conversion result exceeds 65,535 hours 59 minutes 59 seconds, special internal relay M8003 (carry) is turned on. For example, when the conversion result is 65,537 hours 0 minute 0 second, destination 1 stores 1 hour 0 minute 0 second, turning on special internal relay M8003 (carry).

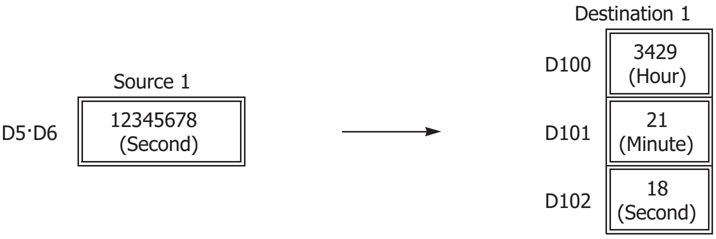
Since the STOH instruction is executed in each scan while input is on, a pulse input from a SOTU or SOTD instruction should be used.

Examples: STOH

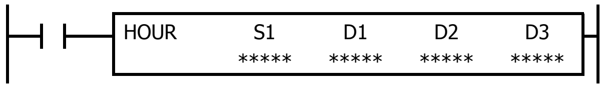
The following examples demonstrate the STOH instruction that will convert time data in seconds into hours, minutes, and seconds and store the results to three consecutive data registers.







HOUR (Hour Meter)



S1 ↔ D1 → D2

While input is on, the ON duration of the input is measured. The measured time value (hour, minute, and second) is stored to 3 consecutive data registers assigned by destination device D1 and compared with the preset value assigned by source device S1.

When the D1 value reaches the S1 value, an output or internal relay assigned by destination device D2 is turned on.

The data registers specified by D3 and D3+1 are used as the system work area.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Preset value	—	—	—	—	—	—	X	—	0-65,535	—
D1 (Destination 1)	Measured input ON duration	—	—	—	—	—	—	X	—	—	—
D2 (Destination 2)	Comparison output	—	X	▲	—	—	—	—	—	—	—
D3 (Destination 3)	System work area	—	—	—	—	—	—	X	—	—	—

For valid device address ranges, see "Device Addresses" on page 2-1.

Source device S1 can be designated by a data register or constant.

Source device S1, when designated by a data register, and destination device D1 occupy 3 consecutive data registers starting with the designated device to store hour, minute, and second data.

When source device S1 is designated by a constant, the preset value can be 0 through 65,535 in hours, then minutes and seconds are set to 0.

▲ Special internal relays cannot be designated as destination device D2.

Destination device D3 requires two data registers reserved for system work area.

Hour data can be 0 through 65,535. Minute and second data can be 0 through 59.

When the measured input ON duration value in destination device D1 reaches the preset value designated by source device S1, the comparison output designated by destination device D2 turns on. As long as the input remains on, the measured input ON duration value continues to increase. When the measured input ON duration value exceeds 65,535 hours 59 minutes 59 seconds, the value returns to 0 hours 0 minutes 0 seconds to repeat another measuring cycle, with the comparison output remaining on.

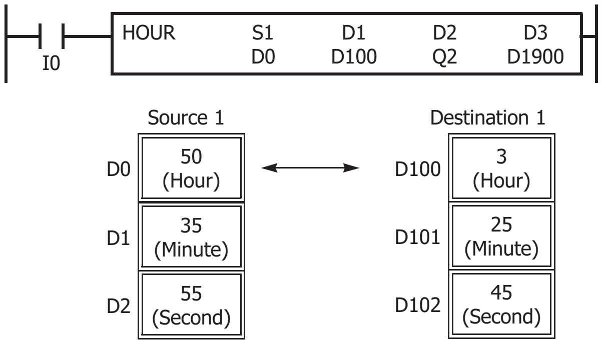
When any of the hour, minute, or second data of source device S1 exceeds the valid range, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart, but the input ON duration is measured.

When any of the hour, minute, or second data of source device S1 is changed to an invalid value after the comparison output has turned on, the comparison output is turned off. Then a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart, but the input ON duration measurement is continued. For user program execution errors, see "User Program Execution Errors" on page 3-10.

Examples: HOUR

The following examples demonstrate the HOUR instruction that will measure the input ON duration value in hours, minutes, and seconds and to compare the value in two different ways.

- **Source Device S1: Data Register**



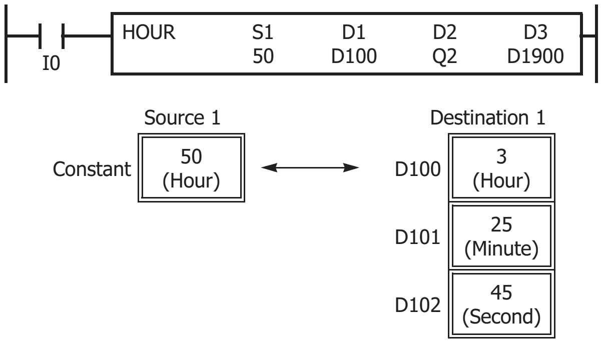
D0·D1·D2 ↔ D100·D101·D102 → Q2

While input I0 is on, the ON duration is measured. The measured time value (hour, minute, and second) is stored to data registers D100·D101·D102 assigned by destination device D1 and compared with the preset value stored in data registers D0·D1·D2 assigned by source device S1.

When the measured value reaches the preset value, output Q2 assigned by destination device D2 is turned on.

Data registers D1900 and D1901 assigned by destination device D3 are reserved for system work area.

- **Source Device S1: Constant**



50 ↔ D100·D101·D102 → Q2

While input I0 is on, the ON duration is measured. The measured time value (hour, minute, and second) is stored to data registers D100·D101·D102 assigned by destination device D1 and compared with 50 hours assigned by source device S1.

When the measured value reaches 50 hours, output Q2 assigned by destination device D2 is turned on.

Data registers D1900 and D1901 assigned by destination device D3 are reserved for system work area.

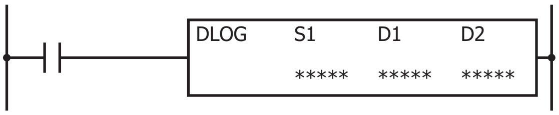
25: DATA LOG INSTRUCTIONS

Introduction

This chapter describes the data log instructions that save the log data of specified devices to the SD memory card.

The Plus CPU module can switch between two save methods: basic mode and advanced mode. In basic mode, the file name is automatically determined.
In advanced mode, you set the file name, and the file name is changed at the desired time.

DLOG (Data Log)



The DLOG instruction saves the values of the specified devices in the specified data format as a CSV file on the SD memory card.

When the input is on, the date and time and the values of the specified devices are output to the CSV file in the folder assigned by S1. When the execution of the instruction finishes, the device assigned by D1 is turned on and the execution status is stored to the device assigned by D2.

When the folder assigned by S1 does not exist on the SD memory card, that folder is created. The folder path is "FCDATA01\DATALOG\User specified folder."

The filename of created CSV files is as follows:

- **All-in-One CPU module:**

The CSV file name is "DATE.csv." The date when the DLOG instruction is turned on is used as DATE.

Example: When the date is December 30, 2015, the file name is "20151230.csv".

- **Plus CPU module:**

For the name of the CSV file, set the desired name in WindLDR. A device value and the date can be added to the set file name.

Example: If the file name is "LOG", the value is "12345" (zero suppression: 5 digits), and the date is 2017/4/1 23:59:01, then the file name is "LOG_12345_170401_235901.csv".

When a file with the same date does not exist in the folder designated by S1, the CSV file is created and the header and the log data is output.

Output example

Time	D0010	
2015/12/30 08:30:23	12345	← Log data

← Header

When a file with the same date already exists in the folder designated by S1, only the log data is appended to the CSV file.

Output example

Time	D0010	
2015/12/30 08:30:23	12345	
2015/12/30 17:30:23	1212	← Appended log data

When the execution of DLOG instruction finishes, the device designated by D1 is turned on and the status code is stored to the device designated by D2 according to the execution result. For status codes, see "3. D2 (destination 2): Execution Status" on page 25-4.

Notes:

- When the FC6A Series MICROSmart is stopped and then set to run, the header is added and output to the CSV file when the DLOG instruction is first executed after it starts running. Header output can be selected for the Plus CPU module.

Output example

Time	D0010	← Header
2015/12/30 08:30:23	12345	← Log data
Time	D0020	← Appended header
2015/12/30 17:30:23	1212	← Log data

- For details on the SD memory card specification, see Chapter 11 "SD Memory Card" in the "FC6A Series MICROSmart User’s Manual".
- For the configuration of the maximum log file size, see Chapter 5 "Functions and Settings" in the "FC6A Series MICROSmart User's Manual".

Notes:

- The number of DLOG instructions that can be programmed in a user program is 48. However, be careful so that the same folders and files are not specified by DLOG instructions. When the folders are duplicated, log data with a mixed format is output to the same CSV file.
- The DLOG instruction cannot be used in an interrupt program. If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.
- While the input to DLOG instruction is on, log data is repeatedly output to the CSV file. When you want to output the log data only one time, add a SOTU (single output up instruction) or SOTD (single output down instruction) to the input condition. For SOTU or SOTD, see "SOTU and SOTD (Single Output Up and Down)" on page 4-28.
- The process of writing data to the SD memory card for the DLOG instruction takes several scans. Once a DLOG instruction is executed, the process continues until the log data transfer is complete, regardless of any change in the instruction input. While the log data is being written to the SD memory card, instructions are not being executed, even when the inputs to DLOG instructions are turned on. To execute the DLOG instruction again, confirm that the previous data transfer process has finished, and then execute the instruction.

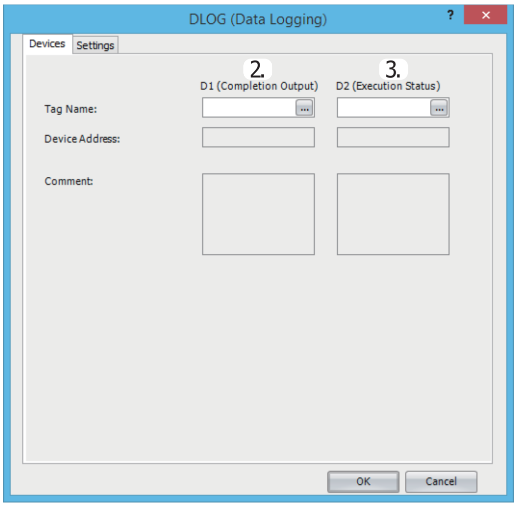
Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Folder name (Note)	—	—	—	—	—	—	—	—	—	—
D1 (Destination 1)	Completion output	—	X	▲	—	—	—	—	—	—	—
D2 (Destination 2)	Execution status	—	—	—	—	—	—	▲	—	—	—

Note: Specify the folder name by entering characters.

▲ Special internal relays cannot be designated as D1. Special data registers cannot be designated as D2.

Settings



DLOG (Data Logging)

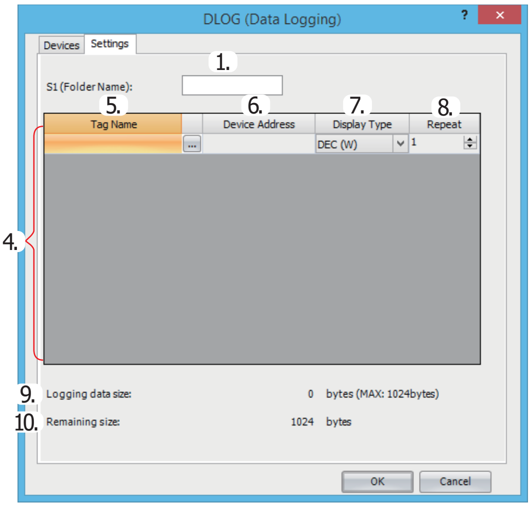
Devices | Settings

2. D1 (Completion Output)
3. D2 (Execution Status)

Tag Name:

Device Address:

Comment:



DLOG (Data Logging)

Devices | Settings

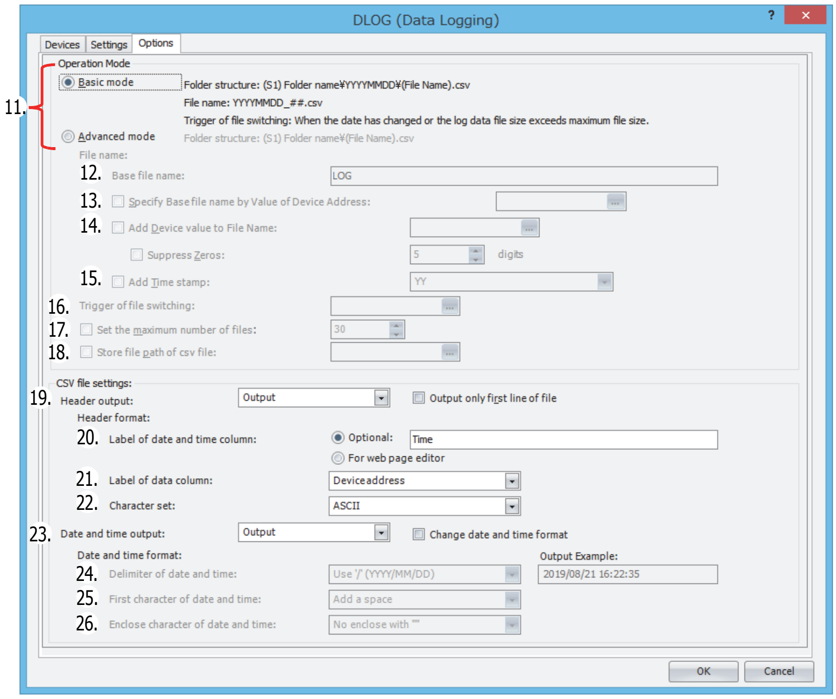
1. S1 (Folder Name):

5. Tag Name		6. Device Address	7. Display Type	8. Repeat
	...		DEC (W)	1

4.

9. Logging data size: 0 bytes (MAX: 1024bytes)

10. Remaining size: 1024 bytes



DLOG (Data Logging)

Devices | Settings | Options

Operation Mode

11.

● Basic mode Folder structure: (S1) Folder name¥YYYYMMDD¥(File Name).csv

File name: YYYYMMDD_##.csv

Trigger of file switching: When the date has changed or the log data file size exceeds maximum file size.

○ Advanced mode Folder structure: (S1) Folder name¥(File Name).csv

File name:

12. Base file name: LOG

13. □ Specify Base file name by Value of Device Address:

14. □ Add Device value to File Name:

□ Suppress Zeros: 5 digits

15. □ Add Time stamp: YY

16. Trigger of file switching:

17. □ Set the maximum number of files: 30

18. □ Store file path of csv file:

CSV file settings:

19. Header output: Output □ Output only first line of file

Header format:

20. Label of date and time column: ● Optional: Time ○ For web page editor

21. Label of data column: Deviceaddress

22. Character set: ASCII

23. Date and time output: Output □ Change date and time format

Date and time format: Output Example: 2019/08/21 16:22:35

24. Delimiter of date and time: Use '/' (YYYY/MM/DD)

25. First character of date and time: Add a space

26. Enclose character of date and time: No enclose with ""

Notes:

- The **Options** tab is displayed only with the Plus CPU module.
- Items 12. to 18. can be set only in advanced mode.
- Items 19. to 26. can be set in basic and advanced modes.

1. S1 (source 1): Folder Name

Specify the folder name on the SD memory card to store the log data with a desired text up to 8 single-byte alphanumeric characters.

Notes:

- The following single-byte characters cannot be used in the folder names:
  / \ : * ? " < > | # { } % & ~
- Consecutive periods cannot be used in folder names.
- A period cannot be used at the start or the end of the folder name.
- Single-byte spaces at the start or the end of the folder name are omitted.

2. D1 (destination 1): Completion Output

Specify the device that turns on when the transfer of log data to a SD card and the execution of the DLOG instruction are complete. This device is turned on regardless of the success or failure of transferring log data to the SD memory card.

3. D2 (destination 2): Execution Status

D2 specifies the first data register of the data registers to use in DLOG instruction.

Starting from the specified data register, 2 continuous words of data registers are used. Specify the first data register so that the range of data registers is not exceeded.

Storage Destination	Function	Setting Details	Reference
Starting number+0	Execution status	This register stores the status code.	"Execution Status" on page 25-4
Starting number+1	File size ratio*1	File size ratio = [CSV file] / [File size set for external memory device*2] × 100 (the decimal part is rounded up)	

*1 Stores the value calculated from the file size. The decimal part is rounded up.
Example: When D2 is "D0000", the file size set for external memory device is "5 MB (5120 KB)", and the CSV file is "520 KB",
File size ratio = 520 / 5120 × 100 = 10.2 → 11
"11" is stored in the register.

*2 For how to set the file size for external memory devices, see Chapter 5 "Functions and Settings" in the "FC6A Series MICROSmart User's Manual".

Execution Status

Specify the device to store the status code. One of the following status codes is stored according to the DLOG instruction execution status and result.

All-in-One CPU module

Status Code	Status	Description
0	Normal	—
1	SD memory card insertion error	The SD memory card is not inserted
2	SD memory card capacity error	The SD memory card is full
3	SD memory card writing error	Writing log data to the SD memory card fails
4	CSV file capacity error	The CSV file has exceeded the upper limit of the log data size
5	SD memory card protection error	The SD memory card is write protected
6	SD memory card access error	DLOG instruction is executed while another DLOG instruction or TRACE instruction is being executed
7	Characters conversion error	Converting log data to numeric characters fails
8	Folder creation error	Creating the folder fails
9	CSV file open error	Opening the CSV file fails
32	Executing DLOG instruction	Writing log data to SD memory card is in progress

Plus CPU module

Status Code	Status	Description
0	Normal	—
1	—	—
2	SD memory card capacity error	The SD memory card is full
3	SD memory card writing error	Writing log data to the SD memory card fails
4	CSV file capacity error	The number of CSV files in one folder for one day has exceeded 99 files (YYYYMMDD_01.csv to YYYYMMDD_99.csv)
5	SD memory card protection error	The SD memory card is write protected
6	SD memory card access error	DLOG instruction is executed while another DLOG instruction or TRACE instruction is being executed
7	Characters conversion error	Converting log data to numeric characters fails
8	Folder creation error	Creating the folder fails
9	CSV file open error	Opening the CSV file fails
10	Normal (Stored in RAM)	Because SD memory card is not inserted, log data is stored in Plus CPU module RAM. If the power to the PLC is turned off without inserting SD memory card, all logged data are cleared. When SD memory card is not inserted and the log data exceeds 1 MB, "11. RAM overflow" occurs.
11	RAM overflow	The total amount of log data exceeds 1MB while SD memory card is not inserted
12	Path storage setting error	The data register for storing the path exceeds the largest data register (D7999, D61999, D269999)
13	CSV file size error	The logging data is not saved because the CSV file exceeded the file size set by the external memory device in advanced mode
32	Executing DLOG instruction	Writing log data to SD memory card is in progress

File size ratio

The data register stores the value calculated from the file size. The decimal part is rounded up.

The file sizes set for the CSV file and external memory device are used.

For how to set the file size for the external memory device, see Chapter 5 "Functions and Settings" in the "FC6A Series MICROSmart User's Manual".

File size ratio = [CSV file] / [File size set for external memory device]

Example: When D2 is "D0000", the file size set for external memory device is "5 MB (5000 KB)", and the CSV file is "520 KB"

The decimal part of the calculated result, "10.4", is rounded up, and "11" is stored in the register.

File size ratio = 520 / 5000 = 10.4 → 11

4. Settings

The following is a list of the device and format that can be set as the data to be output to the CSV files.

Display Type	Valid Devices
DEC(W)	TC, TP, CC, CP, D
DEC(I)	D
DEC(D)	CC, CP, D
DEC(L)	D
DEC(F)	D
HEX(W)	TC, TP, CC, CP, D
HEX(D)	CC, CP, D
BIN(B)	I, Q, M, R, T, C
String(S)	—

5. Tag Name

Enter tag names or device addresses to specify the devices whose values are output to the CSV files.

6. Device Address

When the devices are specified as tag names, the corresponding device addresses are shown.

7. Display Type

Select the display type from the following table for each device for when the device values are output to the CSV file.

Display Type	Range	Maximum Characters
DEC(W)	0 to 65,535	5
DEC(I)	-32,768 to 32,767	6
DEC(D)	0 to 4,294,967,295	10
DEC(L)	-2,147,483,648 to 2,147,483,647	11
DEC(F)	-3.402823E+38 to 3.402823E+38	13
HEX(W)	0000 to FFFF	4
HEX(D)	00000000 to FFFFFFFF	8
BIN(B)	0 or 1	1
String(S)	Unicode (UTF-8) with BOM, Unicode (UTF-8) without BOM, ASCII, Japanese (Shift-JIS), Chinese (GB2312), European (ISO 8859-1 (Latin 1))	100

Notes:

- If **Display Type** is set to **String(S)**, the value stored in the device is output to the CSV file.
  For example, when storing a string in devices using the MOVC instruction, the string output to the CSV file uses the character set selected with the MOVC instruction.
- If **Display Type** is set to **String(S)** and an ASCII control character is contained in the string, "?" is output for that control character.
  Example: 0123[0x0D][0x0A]456[0x00] → 0123??456
- If **Display Type** is set to **String(S)**, the string is registered as one device regardless of the number of repeats.
- If **Display Type** is set to **String(S)** and a separating character is used in the string, enclose the string in double quotation marks ("). Double quotation marks (") are one escape sequence. As a CSV file format rule, a value enclosed in double quotation marks (") is identified as a single string.
  If the string is not enclosed in double quotation marks ("), the value up to the separating character is identified as one string.

Configuration Example 1

When the **String** is **ABC,DEF**, the **Separating Character** is **Comma (,)**, and the number of repeats is 7

- Hexadecimal

Device	Stored Value			
	Upper Byte		Lower Byte	
	ASCII	**Hexadecimal**	**ASCII**	**Hexadecimal**
D1000	A	0x41	B	0x42
D1001	C	0x43	Comma (,)	0x2C
D1002	D	0x44	E	0x45
D1003	F	0x46	NULL	0x00

- String

Time	D 1000	
2018/04/06 14:59:40	ABC	DEF
2018/04/06 14:59:50	ABC	DEF

Configuration Example 2

When the **String** is **"ABC,DEF"**, the **Separating Character** is **Comma (,)**, and the number of repeats is 9

- Hexadecima

Device	Stored Value			
	Upper Byte		Lower Byte	
	ASCII	**Hexadecimal**	**ASCII**	**Hexadecimal**
D1000	Double quotation marks (")	0x22	A	0x41
D1001	B	0x42	C	0x43
D1002	Comma (,)	0x2C	D	0x44
D1003	E	0x45	F	0x46
D1004	Double quotation marks (")	0x22	NULL	0x00

- String

Time	D 1000	
2018/04/06 14:59:40	ABC,DEF	
2018/04/06 14:59:50	ABC,DEF	

Configuration Example 3

When the **String** is **"""ABC,DEF"""**, the **Separating Character** is **Comma (,)**, and the number of repeats is 13

- Hexadecima

Device	Stored Value			
	Upper Byte		Lower Byte	
	ASCII	**Hexadecimal**	**ASCII**	**Hexadecimal**
D1000	Double quotation marks (")	0x22	Double quotation marks (")	0x22
D1001	Double quotation marks (")	0x22	A	0x41
D1002	B	0x42	C	0x43
D1003	Comma (,)	0x2C	D	0x44
D1004	E	0x45	F	0x46
D1005	Double quotation marks (")	0x22	Double quotation marks (")	0x22
D1006	Double quotation marks (")	0x22	NULL	0x00

- String

Time	D 1000	
2018/04/06 14:59:40	"ABC,DEF"	
2018/04/06 14:59:50	"ABC,DEF"	

8. Repeat

The data in as many consecutive devices as the specified repeat, starting from the specified device address, are output to the SD memory card.

For example, when the display type of D10 is DEC(W) and the repeat is set to 5, the data is output to SD memory card as follows.

Time	D0010	D0011	D0012	D0013	D0014	← Header
2015/12/30 15:40:00	12345	1	5	12	111	← Log data 1
2015/12/30 15:41:00	1212	3	7	35	222	← Log data 2
2015/12/30 15:42:00	345	4	99	79	333	← Log data 3

Note: If **Display Type** is set to **String(S)**, one repeat is two characters. A string with a maximum of 100 characters can be output with 50 repeats.

9. Logging data size

The amount of memory that the DLOG instruction uses for the current log settings is shown. The amount of memory used increases when a device to log the data is added. You can register up to a maximum of 64 devices (the total amount of memory must be less than or equal to 1,024 bytes). One byte of memory area is required for each character.

10. Remaining size

The amount of free memory (logging data size subtracted from 1,024 bytes) is shown.

11. Select Mode

Select **Operation Mode** from **Basic mode** or **Advanced mode**.

Items 12. to 18. can be set only in advanced mode.

12. Base file name

Enter the file name of the output data.

The default is **LOG**. Up to 116 single-byte characters can be entered (not including the extension).

13. Specify Base file name by Value of Device Address

Select the base file name setting.

Select this check box and a data register can be specified.

Clear this check box to use the file name in **Base file name** (12.).

Specify the data registers that will be used as the base file name. A maximum of 40 data registers can be used to set up to 80 characters (not including the extension).

The values are read in order starting from the set data register, and the data up to the NULL terminating character (0x00) or up to 80 characters is handled as base file name.

Example: When D0100 is specified for the device address, the fixed value is "IDEC".

Device	Stored Value			
	Upper Byte		Lower Byte	
	ASCII	Hexadecimal	ASCII	Hexadecimal
D1000	I	0x48	D	0x44
D1001	E	0x45	C	0x43
D1002	NULL	0x00	NULL	0x00

Notes:

- The following characters cannot be used in the file name set by **Base file name** or **Specify Base file name by Value of Device Address**.
  / \ : * ? " < > | # { } % & ~
- A file name set with **Specify Base file name by Value of Device Address** that contains a character that cannot be used will be handled as follows.
  - If a character that cannot be used was set, the file name up to that character is used.
  - If the first character is a character that cannot be used, the file name is the text set for **Base file name**.
  - Single-byte spaces at the beginning of the file name are deleted.

14. Add Device value to File Name

Select whether or not to add device values to the file name.
Select this check box and a data register can be specified.
Clear this check box and nothing can be specified.

Specify the data registers containing the values to read and add to the file name. Two words are used.

The base file name, "_" (single-byte underscore), and device value will be added to the file name in that order.

Set the device value between 0 and 999,999,999 (decimal).

If a value greater than 999,999,999 is stored in the data registers, 999,999,999 will be added to the file name.

Zero suppression that does not add "0" is also specified. Select this check box to perform zero suppression. Clear this check box and the number of digits can be specified. "0" is added to the device value so that it becomes the specified number of digits.

If the number of digits of the device values is larger than the set digits, the device value is output unchanged.

Example: When **File name** is **LOG** and the device address is **D0200**, the file name is as follows.
8 digits: LOG_00123456.CSV
6 digits: LOG_123456.CSV
4 digits: LOG_123456.CSV

Device	Stored Value	
	Decimal (Double word)	Decimal (Word)
D0200	1	123456
D0201	57920	

15. Add Timestamp

Select whether or not to add the output date and time to the file name.
Select this check box and the output date and time will be added to the file name. The format of the output date and time can be selected.
Clear this check box and the output date and time will not be added to the file name.

Select the format of the output date and time to add from the following six types.

Year, **Year+Month**, **Year+Month+Day**, **Year+Month+Day+Hours**, **Year+Month+Day+Hours+Minutes**, **Year+Month+Day+Hours+Minutes+Seconds**

The format is YYMMDD_hhmmss (YY: year, MM: month, DD: day, hh: hours, mm: minutes, and ss: seconds).

The base file name, "_" (single-byte underscore), and output date and time will be added to the file name in that order.

Example: When **File name** is **LOG** and the output date and time is 2013/9/15 23:30:50

Year:	LOG_13
Year+Month:	LOG_1309
Year+Month+Day:	LOG_130915
Year+Month+Day+Hours:	LOG_130915_23
Year+Month+Day+Hours+Minutes:	LOG_130915_2330
Year+Month+Day+Hours+Minutes+Seconds:	LOG_130915_233050

16. Trigger of file switching

Specify the device that will be used as trigger for switching files. When the specified device is turned on and the DLOG instruction is executed, the file for output data is switched. The device will be turned off automatically at the timing for switching files.

A new file will be created with the file name specified in items 12. to 15. If the file with the same name already exists, the new file will not be created. Instead, the data will be added to the relevant file.

17. Set the maximum number of files

This property limits the number of CSV files that are saved to the CSV file storage folder on the SD memory card.

Select this check box and the number of files can be limited. The limit on the number of files can be set between 1 and 100.
When the limit on the number of files is reached, the file with the oldest timestamp is deleted, and a new file is created.

If this check box is cleared, the number of files will not be limited. CSV files will continue to be created until the free space on the SD memory card is full.

Note: If the number of files saved on the SD memory card has already exceeded the limit at the start of operation, the number of files at that time becomes the limit.

18. Store file path of csv file

Select this check box to store the path of the CSV file.

Specify the first data register of the data registers used for storage. The CSV data can be displayed in Web Page Editor by setting this data register in a trend bar or other component in Web Page Editor.

The path to the CSV file is stored in the specified data registers when switching the name of the CSV file.

The CSV file path is stored in data registers using the format <D>\Folder Name\CSV File Name. For details, see Chapter 13 "Web Server" in the "FC6A Series MICROSmart Communication Manual".

Items 19. to 22. can be set in basic and advanced modes.

19. Header output

Specify whether or not to output the header to the CSV file.

To output the header only on the first row of the file when outputting the header, select the **Output only first line of file** check box.

- When not outputting a header

2018/04/06 14:59:40	30	34	28	30	60	68	56	60
2018/04/06 14:59:50	32	35	27	32	64	70	54	64
2018/04/06 15:00:00	31	35	27	32	62	70	54	64
2018/04/06 15:00:10	31	36	27	32	62	72	54	64
2018/04/06 15:00:20	32	34	28	31	64	68	56	62
2018/04/06 15:00:30	31	35	27	32	62	70	70	64

- When outputting a header

Time	D 1000	D 1001	D 1002	D 1003	D 1004	D 1005	D 1006	D 1007
2018/04/06 14:59:40	30	34	28	30	60	68	56	60
2018/04/06 14:59:50	32	35	27	32	64	70	54	64
Time	D 1000	D 1001	D 1002	D 1003	D 1004	D 1005	D 1006	D 1007
2018/04/06 15:00:20	32	34	28	31	64	68	56	62
2018/04/06 15:00:30	31	35	27	32	62	70	70	64

- When outputting a header only on the first row of the file

Time	D 1000	D 1001	D 1002	D 1003	D 1004	D 1005	D 1006	D 1007
2018/04/06 14:59:40	30	34	28	30	60	68	56	60
2018/04/06 14:59:50	32	35	27	32	64	70	54	64
2018/04/06 15:00:00	31	35	27	32	62	70	54	64
2018/04/06 15:00:10	31	36	27	32	62	72	54	64
2018/04/06 15:00:20	32	34	28	31	64	68	56	62

20. Label of date and time column

Specify the label of the execution date and time column of the CSV file header. Select from **Optional** or **For web page editor**.

The default value of **Optional** is **Time**.

Select **Optional** to output the string entered in the text box. Up to 120 bytes can be set (approximately 40 when the character set is UTF-8).

When **For web page editor** is selected, the trend bar component in Web Page Editor can be used.

Note: For details on the trend bar component, see Chapter 13 "Web Server" in the "FC6A Series MICROSmart Communication Manual".

- Optional (value: Time)

Time	D 1000	D 1001	D 1002	D 1003	D 1004	D 1005	D 1006	D 1007
2018/04/06 14:59:40	30	34	28	30	60	68	56	60
2018/04/06 14:59:50	32	35	27	32	64	70	54	64

- For web page editor

#_date_time	D 1000	D 1001	D 1002	D 1003	D 1004	D 1005	D 1006	D 1007
2018-04-06 14:59:40	30	34	28	30	60	68	56	60
2018-04-06 14:59:50	32	35	27	32	64	70	54	64

Note: If **For web page editor** is selected, the string format of the execution date is changed to YYYY-MM-DD.
To use the Web Page Editor trend bar component in Microsoft Internet Explorer, select the **Change date and time format** check box and set **Delimiter of date and time** to **Use '/' (YYYY/MM/DD)**.

21. Label of data column

Specify the label of the data column of the CSV file header. Select from **Device address**, **Tag name**, or **Comment**.

- Device address

Time	D 1000	D 1001	D 1002	D 1003	D 1004	D 1005	D 1006	D 1007
2018/04/06 14:59:40	30	34	28	30	60	68	56	60
2018/04/06 14:59:50	32	35	27	32	64	70	54	64

- Tag

Time	Temp 1	Temp 2	Temp 3	Temp 4	Hum 1	Hum 2	Hum 3	Hum 4
2018/04/06 14:59:40	30	34	28	30	60	68	56	60
2018/04/06 14:59:50	32	35	27	32	64	70	54	64

- Comment

Time	Temp_Osaka	Temp_Kyoto	Temp_Shiga	Temp_Hyogo	Hum_Osaka	Hum_Kyoto	Hum_Shiga	Hum_Hyogo
2018/04/06 14:59:40	30	34	28	30	60	68	56	60
2018/04/06 14:59:50	32	35	27	32	64	70	54	64

22. Character set

Select the character set of the CSV file from the following.

Unicode (UTF-8) with BOM, **Unicode (UTF-8) without BOM**, **ASCII**, **Japanese (Shift-JIS)**, **Chinese (GB2312)**, **European (ISO 8859-1 (Latin 1))**

When using a trend bar component in Web Page Editor, select **Unicode (UTF-8) with BOM** or **Unicode (UTF-8) without BOM**.

Notes:

- If **String(S)** is selected for **Display Type** (7.), select the same character set for this setting as the string that is stored in the devices.
- For **Unicode (UTF-8) with BOM**, the BOM values (0xEF 0xBB 0xBF) are output as the first 3 bytes of data.
- For **Unicode (UTF-8) without BOM**, the characters may be unreadable because the application does not open the file as **UTF-8** data.

23. Date and time output

Specify whether to output the execution date and time to the CSV file.

- When outputting the execution date and time

Time	D 1000
2018/04/06 14:59:40	30

- When not outputting the execution date and time

D 1000	
30	

Items 24 to 26 can be set by selecting the **Change date and time format** check box. If the **Change date and time format** check box is cleared, the execution date and time format is as follows.

- When **Optional** is set to **Label of date and time column**

Time	D 1000
2018/04/06 14:59:40	30

- When **For web page editor** is set to **Label of date and time column**

#_date_time_	D 1000
2018-04-06 14:59:40	30

24. Delimiter of date and time

Specify the delimiter for the date to use in the execution date and time format in the CSV file.
Select from **Use '/' (YYYY/MM/DD)** or **Use '-' (YYYY-MM-DD)**.

- Use '/' (YYYY/MM/DD)

Time	D 1000
2018/04/06 14:59:40	30

- Use '-' (YYYY-MM-DD)

Time	D 1000
2018-04-06 14:59:40	30

25. First character of date and time

Specify whether to use a space for the first character in the execution date and time format in the CSV file.
Select from **Add a space** or **Do not add a space**.

- Add a space

Time	D 1000
2018/04/06 14:59:40	30

- Do not add a space

Time	D 1000
2018/04/06 14:59:40	30

26. Enclose character of date and time

Specify whether to use a character to enclose the execution date and time format in the CSV file.
Select from **Enclose with ""** or **No enclose with ""**.

- Enclose with ""

Time	D 1000
"2018/04/06 14:59:40"	30

- No enclose with ""

Time	D 1000
2018/04/06 14:59:40	30

Note: When **Date and time output** is set to **Output** and the subsequently output CSV file is opened in spreadsheet software, the execution date and time is displayed according to the display format of the spreadsheet software.

CSV File Output Format and File Format Configuration

- **Output format**

The CSV file output format is as follows.

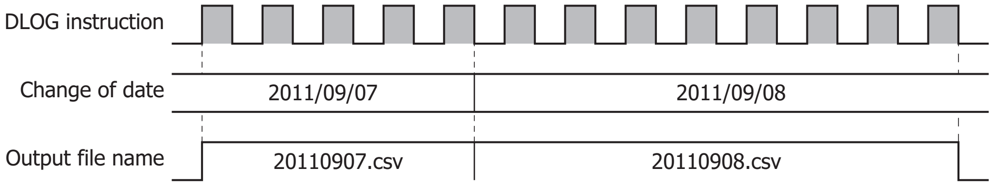
```
Time,D0010,D0020,D0030,D0050,D0060
2015/12/30 15:40:00,12345,1,5,12,111
2015/12/30 15:41:00,1212,3,7,35,222
2015/12/30 15:42:00,345,4,99,79,333
                    ·
                    ·
                    ·
```

- **Select Mode**

Basic mode

When the DLOG instruction is executed and the CSV file for the same date does not exist in the folder designated by S1, a new file is created and the header and the log data 1 are output as shown in above output format example. If the DLOG instruction is executed again on the same date, the log data 2 is appended to the CSV file. Similarly, if the DLOG instruction is executed again on the same date, log data 3 is appended to the CSV file.

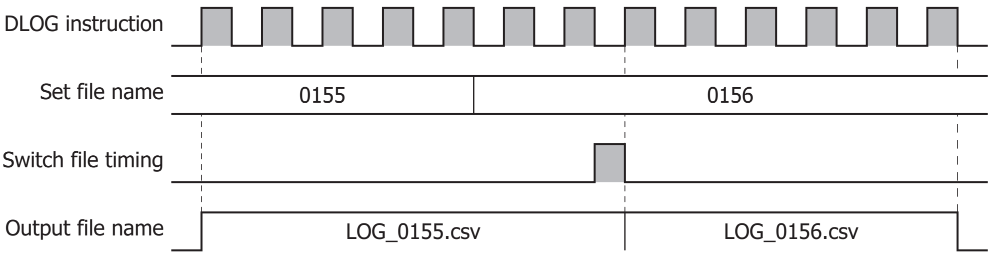
When the date changes and the DLOG instruction is executed, a new CSV file with a new file name is created and the header and the log data is output.



Advanced mode

When the DLOG instruction is executed and the CSV file for the same name does not exist in the folder designated by S1, a new file is created and the header and the log data 1 are output as shown in above output format example. If the DLOG instruction is executed again on the same name, the log data 2 is appended to the CSV file. Similarly, if the DLOG instruction is executed again on the same name, log data 3 is appended to the CSV file.

When the switch file timing trigger is turned on and the file name changes, a CSV file with a new file name is created and the header and data are output. The switch file timing trigger is automatically turned off when the DLOG instruction is executed.



- **File format**

The decimal symbol and separating character in CSV files can be specified in the function area settings. For details, see Chapter 11 "SD Memory Card" in the "FC6A Series MICROSmart User's Manual".

- **File name**

The output order of the file name is as follows.

[File name]_[Value of device address]_[YYMMDD]_[hhmmss].CSV

File name:	Text entered in **Base file name** or text in device address values specified by **Specify Base file name by Value of Device Address**
Value of device address:	Value of device address specified by **Add Device value to File Name**
YYMMDD:	Year, month, and day of date and time set in **Add Timestamp**
hhmmss:	Hours, minutes, and seconds of date and time set in **Add Timestamp**

- Configuration Example 1

Item	Setting	
Base file name	LOG	
Add Device value to File Name	Device address: D0200	D0200: 123456
Add Timestamp	Year + Month	Date and time when the data was output: September, 2013

The file name is "LOG_123456_1309.CSV".

- Configuration Example 2

<table>
<tr><th>Item</th><th colspan="2">Setting</th></tr>
<tr><td>Base file name</td><td colspan="2">LOG</td></tr>
<tr><td>Specify Base file name by Value of Device Address</td><td>Device address: D0100
The text to set is "IDEC"</td><td><table>
<tr><th rowspan="2">Device</th><th colspan="2">Stored Value</th></tr>
<tr><th>Upper Byte</th><th>Lower Byte</th></tr>
<tr><td>D0100</td><td>1</td><td>D</td></tr>
<tr><td>D0101</td><td>E</td><td>C</td></tr>
<tr><td>D0102</td><td>NULL</td><td>NULL</td></tr>
</table></td></tr>
<tr><td>Add Device value to File Name</td><td>Device address: D0200</td><td>D0200: 123456</td></tr>
<tr><td>Add Timestamp</td><td>Year + Month + Day + Minute + Second</td><td>Date and time when the data was output: September 15, 2013, 23:30:50</td></tr>
</table>

The file name is "IDEC_123456_130915_233050.CSV".

Examples: DLOG

When M0 is turned on, the decimal values of D0 through D5 (data type W (word)) and D10 (data type F (floating point)) are saved in a CSV file in the "RESULT" folder on the SD memory card every 10 seconds.

Output example

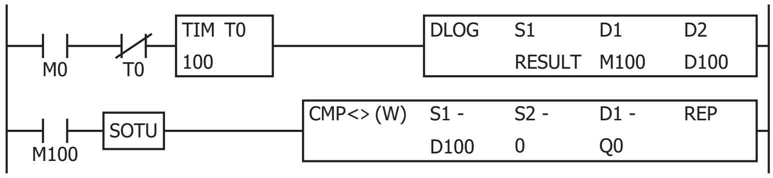
Time	D0000	D0001	D0002	D0003	D0004	D0005	D0010
2012/02/06 10:20:30	12345	0	0	56789	0	0	-3.402823E+38
2012/02/06 10:20:40	12345	0	0	56789	0	0	-3.402823E+38
2012/02/06 10:20:50	12345	0	0	56789	0	0	-3.402823E+38

The sample user program described below operates as follows.

- M100 is turned on when the transfer of log data has finished transferring to the SD memory card is complete.
- The DLOG instruction status code is stored in D100.
- The status code saved in D100 is checked and Q0 is turned on if an error occurs.

- **Configuration Procedure**

1. Create the ladder program.

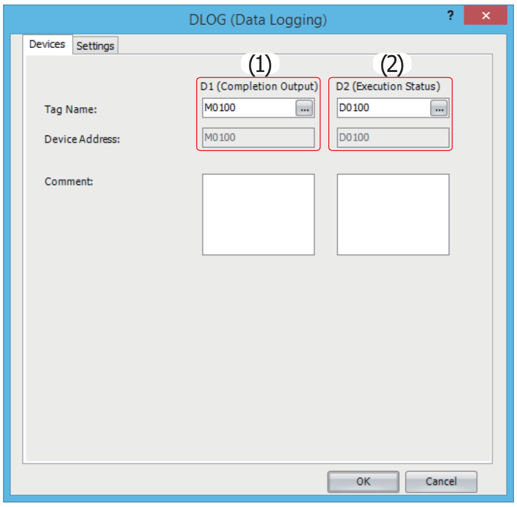


2. Configure the DLOG instruction.

Configure the Devices tab.

(1) Designate M0100 as D1 (Completion Output).

(2) Designate D0100 as D2 (Execution Status).



DLOG (Data Logging)

Devices | Settings

	(1) D1 (Completion Output)	(2) D2 (Execution Status)
Tag Name:	M0100	D0100
Device Address:	M0100	D0100
Comment:		

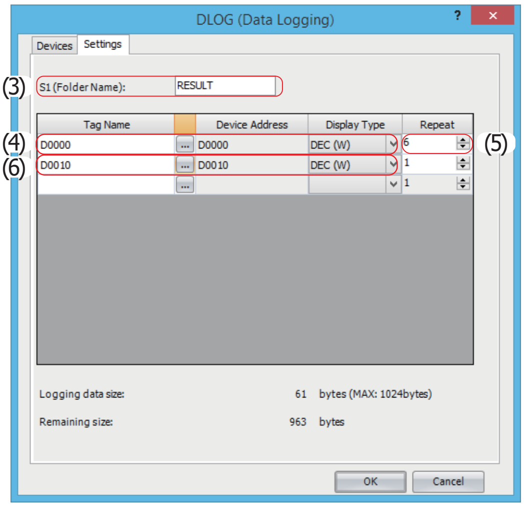
Configure the Settings tab.

(3) Enter "RESULT" in **S1 (Folder Name)**.

(4) Configure D0000 to output the decimal value of D0000 to the CSV files with the data type W (word).

(5) Set the repeat to 6 to output the values of D0000 through D0005 to the CSV files.

(6) Configure D0010 to output the decimal value of D0010 to the CSV files with the data type F (float).



S1 (Folder Name): RESULT

Tag Name	Device Address	Display Type	Repeat
D0000	D0000	DEC (W)	6
D0010	D0010	DEC (W)	1
			1

Logging data size: 61 bytes (MAX: 1024bytes)

Remaining size: 963 bytes

The configuration is now completed.

- **Operation Description**

10 seconds after M0 is turned on, the DLOG instruction is executed one time. When the DLOG instruction is executed, the data of D0 through D5 and D10 are output to the CSV file on the SD memory card as decimal values along with the current date and time.

The saved location of CSV files is FCDATA01\DATALOG\RESULT. The oldest data is saved at the top of the log data, and the latest data is saved at the bottom of the log data.

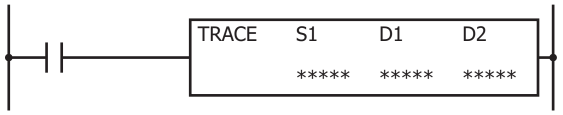
When the execution of the DLOG instruction is complete, the completion output M100 is turned on and the CMP instruction is executed once. The CMP instruction compares the status code stored in the execution status D100 with 0 and turns Q0 on or off. Q0 is turned on when an error occurs in the DLOG instruction.

While M0 is on, the log data is appended to the CSV file every 10 seconds.

Output results

Time	D0000	D0001	D0002	D0003	D0004	D0005	D0010
2012/02/06 10:20:30	12345	0	0	56789	0	0	-3.402823E+38
2012/02/06 10:20:40	12345	0	0	56789	0	0	-3.402823E+38
2012/02/06 10:20:50	12345	0	0	56789	0	0	-3.402823E+38

TRACE (Data Trace)



The TRACE instruction saves the values for the previous number of scans for the specified device in the specified data format as a CSV file on the SD memory card.

When the input is turned on, the date and time and the values of the previous scans for the specified device are output to the CSV file in the folder assigned by S1. When the execution of the instruction finishes, the device assigned by D1 is turned on and the execution status is stored to the device assigned by D2.

When the folder assigned by S1 does not exist on the SD memory card, that folder is created. The folder path is "FCDATA01\TRACE\User specified folder." For details on the folder structure, see Chapter 11 "SD Memory Card" in the "FC6A Series MICROSmart User's Manual".

The filename of created CSV files is as follows:

- **All-in-One CPU module:**

The CSV file name is "DATE.csv." The date when the DLOG instruction is turned on is used as DATE.

Example: When the date is December 30, 2015, the file name is "20151230.csv".

- **Plus CPU module:**

A new folder "YYYYMMDD" (Year, Month, and Date) is created and csv files are stored in the folder. Underscore "_" and 2-digit number "00 to 99" are appended to the csv filenames. When the file size exceeds the log data file size, a new file is created and 2-digit number is incremented and appended to the filename.

Example: When the date is April 1, 2017 and the first file in the day is created, the file is "20170401\20170401_00.csv."

When a CSV file with the same date does not exist in the folder designated by S1, the CSV file is created and the trace data is output. The oldest data is output at the top of the trace data, and the latest data is output at the bottom of the trace data.

Output example

Triggered at:	2012/02/06 08:30:23	← Header row 1
Scan	D0010	← Header row 2
Old	12345	← Data 2 scans before
	12345	← Data 1 scan before
New	12345	← Latest data

When a CSV file for the same date already exists in the folder designated by S1, the header and trace data are appended to the CSV file.

Output example

Triggered at:	2012/02/06 08:30:23	
Scan	D0010	
Old	12345	
	12345	
New	12345	
Triggered at:	2012/02/06 17:16:15	← Appended header row 1
Scan	D0010	← Appended header row 2
Old	1212	← Appended data 2 scans before
	1212	← Appended data 1 scan before
New	1212	← Appended latest data

When the execution of the TRACE instruction finishes, the device designated by D1 is turned on and the status code is stored to the device designated by D2 according to the execution result. For status codes, see "3. D2 (destination 2): Execution Status" on page 25-17.

Notes:

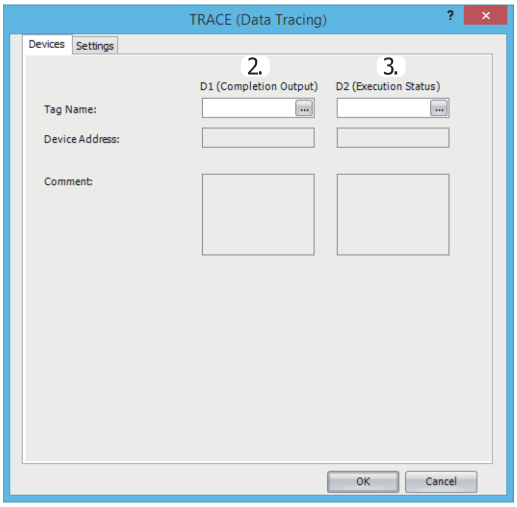
- The TRACE instruction accumulates data while FC6A Series MICROSmart is running but does not accumulate data when FC6A Series MICROSmart is stopped.
- Data is accumulated while FC6A Series MICROSmart is running even when the input to the TRACE instruction is off.
- When the input to the TRACE instruction is turned on, the accumulated data is output to the CSV file.
- Trace data is accumulated even when the MCS (master control set) instruction is on.
- For the configuration of the maximum log file size, see Chapter 5 "Functions and Settings" in the "FC6A Series MICROSmart User's Manual".
- The maximum number of TRACE instructions that can be programmed in a user program is three. Make sure that the folder name specified by the TRACE instruction is not duplicated by any of the folder names specified by the other TRACE instructions. When the folders are duplicated, trace data with a mixed format is output to the same CSV file.
- The number of scans of which the trace data can be saved to the CSV file when the TRACE instruction is executed one time depends on the number of devices specified to trace and the display type for each device. For details, see "9. Trace data size" on page 25-18.
- The TRACE instruction cannot be used in an interrupt program. If used, a user program execution error will result, turning on special internal relay M8004 and the ERR LED on the FC6A Series MICROSmart. For details about the user program execution errors, see "User Program Execution Errors" on page 3-10.
- Trace data is not accumulated when the TRACE instruction is skipped by the JMP (jump) instruction. For the JMP (jump) instruction, see "JMP (Jump) and JEND (Jump End)" on page 4-31.
- While the input to the TRACE instruction is on, the trace data is repeatedly output to the CSV file. When you want to output the trace data only one time, add a SOTU (single output up instruction) or SOTD (single output down instruction) to the input conditions. For SOTU (single output up instruction) or SOTD (single output down instruction), see "Basic Instructions" - "SOTU and SOTD (Single Output Up and Down)" on page 4-28.
- The data writing process to the SD memory card for the TRACE instruction takes several scans. Once a TRACE instruction is executed, the process continues until the trace data transfer is complete, regardless of any change in the instruction input. While the trace data is being written to the SD memory card, the instruction is not executed even when the input to the TRACE instructions are turned on. To execute the TRACE instruction again, confirm that the previous data writing process has finished, and then execute the instruction.

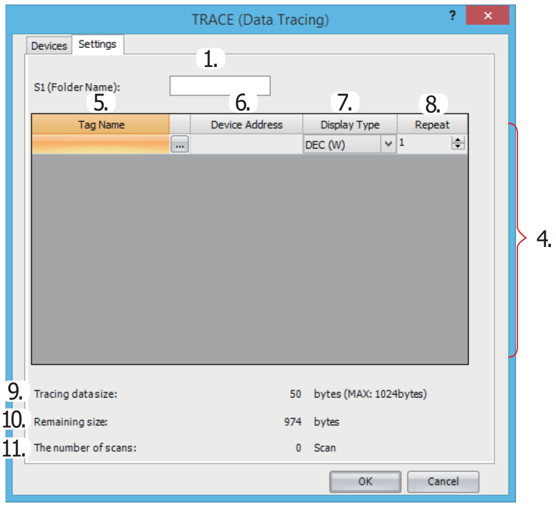
Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Folder name (Note)	—	—	—	—	—	—	—	—	—	—
D1 (Destination 1)	Completion output	—	X	▲	—	—	—	—	—	—	—
D2 (Destination 2)	Execution status	—	—	—	—	—	—	▲	—	—	—

Note: Specify the folder name by entering characters.
▲ Special internal relays cannot be designated as D1. Special data registers cannot be designated as D2.

Settings





Tracing data size: 50 bytes (MAX: 1024bytes)
Remaining size: 974 bytes
The number of scans: 0 Scan

1. S1 (source 1): Folder Name

Specify the folder name on the SD memory card that will store the trace data using up to 8 single-byte alphanumeric characters.

Notes:

- The following single-byte characters cannot be used in folder names:
  / \ : * ? " < > | # { } % & ~
- Consecutive periods cannot be used in folder names.
- A period cannot be used at the start or the end of the folder name.
- Single-byte spaces at the start or the end of the folder name are omitted.

2. D1 (destination 1): Completion Output

Specifies the device that turns on when the transfer of trace data to SD card and the execution of the TRACE instruction are complete. This device turns on regardless of the success or failure of the transfer of trace data to the SD memory card.

3. D2 (destination 2): Execution Status

Specifies the data register to store the status code. One of the following status codes is stored according to the TRACE instruction execution status and result.

All-in-One CPU module

Status Code	Status	Description
0	Normal	—
1	SD memory card insertion error	The SD memory card is not inserted
2	SD memory card capacity error	The SD memory card is full
3	SD memory card writing error	Writing trace data to the SD memory card fails
4	CSV file capacity error	The CSV file has exceeded the upper limit of the log data size
5	SD memory card protection error	The SD memory card is write protected
6	SD memory card access error	The TRACE instruction is executed while another DLOG instruction or TRACE instruction is being executed
7	Characters conversion error	Converting trace data to numeric characters fails
8	Folder creation error	Creating the folder fails
9	CSV file open error	Opening the CSV file fails
32	Executing TRACE instruction	Writing trace data to SD memory card is in progress

Plus CPU module

Status Code	Status	Description
0	Normal	—
1	—	—
2	SD memory card capacity error	The SD memory card is full
3	SD memory card writing error	Writing trace data to the SD memory card fails
4	CSV file capacity error	The number of CSV files in one folder for one day has exceeded 99 files (YYYYMMDD_01.csv to YYYYMMDD_99.csv)
5	SD memory card protection error	The SD memory card is write protected
6	SD memory card access error	The TRACE instruction is executed while another DLOG instruction or TRACE instruction is being executed
7	Characters conversion error	Converting trace data to numeric characters fails
8	Folder creation error	Creating the folder fails
9	CSV file open error	Opening the CSV file fails
10	Normal (Stored in RAM)	Because SD memory card is not inserted, trace data is stored in Plus CPU module RAM. If the power to the PLC is turned off without inserting SD memory card, all trace data are cleared. When SD memory card is not inserted and the trace data exceeds 1 MB, "11. RAM overflow" occurs.
11	RAM overflow	The total amount of trace data exceeds 1MB while SD memory card is not inserted
32	Executing TRACE instruction	Writing trace data to SD memory card is in progress

4. Settings

The following is a list of the device and display types that can be set as the data to be output to the CSV files.

Display Type	Valid Devices
Decimal (W)	TC, TP, CC, CP, D
Decimal (I)	D
Decimal (D)	CC, CP, D
Decimal (L)	D
Decimal (F)	D
Hexadecimal (W)	TC, TP, CC, CP, D
Hexadecimal (D)	CC, CP, D
Binary (B)	I, Q, M, R, T, C

5. Tag Name

Enter tag names or device addresses to specify the devices to the CSV files.

6. Device Address

When the devices are specified as tag names, the corresponding device addresses are shown.

7. Display Type

Select the display type from the following table for each device for when the device values are output to the CSV file.

Display Type	Range	Maximum Characters
Decimal (W)	0 to 65,535	5
Decimal (I)	-32,768 to 32,767	6
Decimal (D)	0 to 4,294,967,295	10
Decimal (L)	-2,147,483,648 to 2,147,483,647	11
Decimal (F)	-3.402823E+38 to 3.402823E+38	13
Hexadecimal (W)	0000 to FFFF	4
Hexadecimal (D)	00000000 to FFFFFFFF	8
Binary (B)	0 or 1	1

8. Repeat

The data in as many consecutive devices as the specified repeat, starting from the specified device address, are output to the SD memory card.

For example, when the display type of D10 is DEC(W) and the repeat is set to 8, the data is output to SD memory card as follows.

Triggered at:	2015/12/30 15:40:30								← Header row 1
Scan	D0010	D0011	D0012	D0013	D0014	D0015	D0016	D0017	← Header row 2
Old	1	9	17	25	33	41	49	57	← Data 7 scans before
	2	10	18	26	34	42	50	58	← Data 6 scans before
	3	11	19	27	35	43	51	59	← Data 5 scans before
	4	12	20	28	36	44	52	60	← Data 4 scans before
	5	13	21	29	37	45	53	61	← Data 3 scans before
	6	14	22	30	38	46	54	62	← Data 2 scans before
	7	15	23	31	39	47	55	63	← Data 1 scan before
New	8	16	24	32	40	48	56	64	← Latest data

9. Trace data size

The amount of memory that the TRACE instruction uses for the current trace settings is shown. The amount of memory used increases when a device to trace the data is added. You can register up to a maximum of 64 devices (the total amount of memory must be less than or equal to 1,024 bytes). One byte of memory area is required for each character.

10. Remaining size

The amount of free memory (the difference between the trace data size and 1,024 bytes) is shown.

11. The number of scans

How many scans of trace data can be accumulated with the current trace settings is shown. The number of scans of data that can be accumulated depends on the format of the trace data to output. If there is little data to output per scan, the data for many scans can be accumulated.

CSV File Output Format and File Format Configuration

- **Output format**

The CSV file output format is as follows.

```
Triggered at:,2015/12/30 15:40:30
Scan,D0010,D0020,D0030,D0040,D0050,D0060,D0070,D0080
Old,1,9,17,25,33,41,49,57
,2,10,18,26,34,42,50,58
,3,11,19,27,35,43,51,59
,4,12,20,28,36,44,52,60
,5,13,21,29,37,45,53,61
,6,14,22,30,38,46,54,62
,7,15,23,31,39,47,55,63
New,8,16,24,32,40,48,56,64
                    .
                    .
                    .
```

When the TRACE instruction is executed and the CSV file for the same date does not exist in the folder designated by S1, a new CSV file is created and the trace data is output as shown in above output format example. When the date changes and the TRACE instruction is executed, a new CSV file is created with a new file name.

- **File format**

The decimal symbol and separating character in CSV files can be specified in the function area settings. For details, see Chapter 11 "SD Memory Card" in the "FC6A Series MICROSmart User's Manual".

Examples: TRACE

When M0 is turned on, the accumulated data of D0 through D5 (data type W (word)) and D10 (data type F (floating point)) are saved as decimal values in a CSV file on the "RESULT" folder on the SD memory card.

Output sample

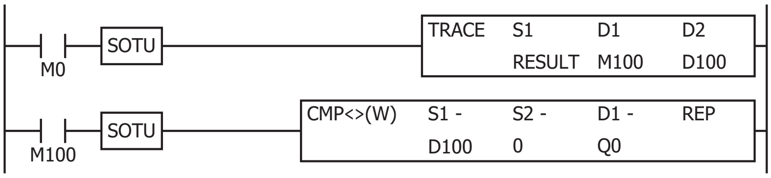
Triggered at:	2012/02/06 10:20:30						
Scan	D0000	D0001	D0002	D0003	D0004	D0005	D0010
Old	12345	2	12345	56789	1	56789	-3.402823E+38
	12345	2	12347	56789	1	56788	-3.402823E+38
	12345	2	12349	56789	1	56787	-3.402823E+38
:	:	:	:	:	:	:	:
	12345	2	12379	56789	1	56772	-3.402823E+38
	12345	2	12381	56789	1	56771	-3.402823E+38
New	12345	2	12383	56789	1	56770	-3.402823E+38

The sample user program described below operates as follows.

- M100 is turned on when the writing trace data to the SD memory card completes.
- The TRACE instruction status code is stored to D100.
- The execution status saved in D100 is checked and Q0 is turned on if an error occurs.

- **Configuration Procedure**

1. Create the ladder program.

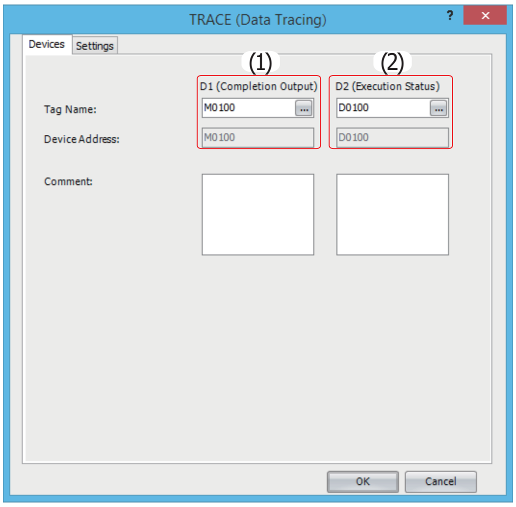


2. Configure the TRACE instruction.

Configure the Devices tab.

(1) Designate M0100 as D1 (Completion Output).

(2) Designate D0100 as D2 (Execution Status).



	D1 (Completion Output)	D2 (Execution Status)
Tag Name:	M0100	D0100
Device Address:	M0100	D0100
Comment:		

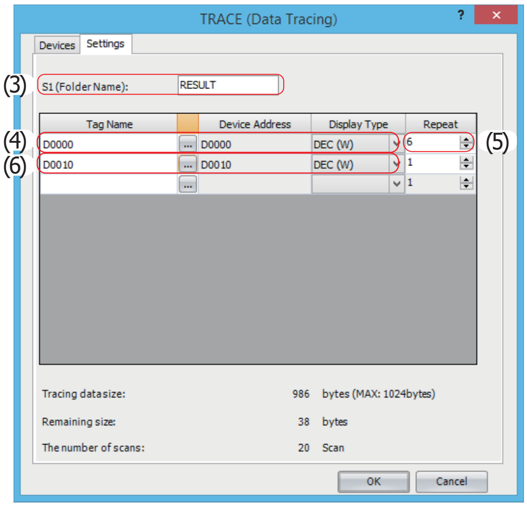
Configure the Settings tab.

(3) Enter "RESULT" in S1 (Folder Name).

(4) Configure D0000 to output the decimal value of D0000 to the CSV files with the data type W (word).

(5) Set the repeat to 6 to output the values of D0000 through D0005 to the CSV files.

(6) Configure D0010 to output the value of D0010 to the CSV files with the data type F (float).



S1 (Folder Name): RESULT

Tag Name	Device Address	Display Type	Repeat
D0000	D0000	DEC (W)	6
D0010	D0010	DEC (W)	1
			1

Tracing data size: 986 bytes (MAX: 1024bytes)

Remaining size: 38 bytes

The number of scans: 20 Scan

The configuration is now completed.

- **Operation Description**

When M0 is turned on, the TRACE instruction is executed one time. When the TRACE instruction is executed, the data of D0 through D5 and D10 in the previous 17 scans is output to a CSV file on the SD memory card as decimal values along with the date and time of the execution.

The saved location of CSV files is FCDATA01\TRACE\RESULT. The oldest data is saved at the top of the trace data, and the latest data is saved at the bottom of the trace data.

When the execution of the TRACE instruction completes, the completion output M100 is turned on and the CMP instruction is executed once. The CMP instruction compares the status code stored in the execution status D100 with 0 and turns Q0 on or off. Q0 is turned on when an error occurs in the TRACE instruction.

Output results

Triggered at:	2012/02/06 10:20:30						
Scan	D0000	D0001	D0002	D0003	D0004	D0005	D0010
Old	12345	2	12345	56789	1	56789	-3.402823E+38
	12345	2	12347	56789	1	56788	-3.402823E+38
	12345	2	12349	56789	1	56787	-3.402823E+38
	12345	2	12351	56789	1	56786	-3.402823E+38
	12345	2	12353	56789	1	56785	-3.402823E+38
	12345	2	12355	56789	1	56784	-3.402823E+38
	12345	2	12357	56789	1	56783	-3.402823E+38
	12345	2	12359	56789	1	56782	-3.402823E+38
	12345	2	12361	56789	1	56781	-3.402823E+38
	12345	2	12363	56789	1	56780	-3.402823E+38
	12345	2	12365	56789	1	56779	-3.402823E+38
	12345	2	12367	56789	1	56778	-3.402823E+38
	12345	2	12369	56789	1	56777	-3.402823E+38
	12345	2	12371	56789	1	56776	-3.402823E+38
	12345	2	12373	56789	1	56775	-3.402823E+38
	12345	2	12375	56789	1	56774	-3.402823E+38
New	12345	2	12377	56789	1	56773	-3.402823E+38

26: Script

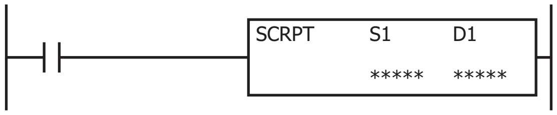
Introduction

This chapter describes the SCRPT instruction that calls and executes scripts from the ladder program.

SCRPT (Script)

Executes the specified script.

Symbol



Operation

When the input turns on, the script that corresponds to the script ID specified by S1 is executed. When the script is finished executing, the execution status and the execution time are stored in D1 and D1+1. To use the SCRPT instruction, you must create a script to execute in Script Manager dialog box beforehand.

For an overview of the script function, see "Script Function Overview" on page 26-3.

For editing scripts, see "Script Programming and Management" on page 26-5.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Script ID	—	—	—	—	—	—	X	—	1-255	—
D1 (Destination 1)	Execution results	—	—	—	—	—	—	▲	—	—	—

▲ Special data registers cannot be designated as D1.

Settings

S1 (source 1): Script ID
Specify the script ID. A constant or data register can be specified.

D1 (destination 1): Execution results
Specify the data register where the execution results are stored. Two data registers starting with the specified data register are used. The script execution status (error code at script completion) is stored in D1. The execution time from when the script execution is started until it is completed is stored in D1+1 in 100 µs increments.

Execution results

The execution results (D1, D1+1) store the execution status and the execution time.

Execution status

Numeric Value	Status	Error Cause
0	Normal termination	—
1	Arithmetic error	Division by zero, floating point format error
2	Script ID error	The specified script does not exist
3	Device access error	Invalid device specified, device boundary exceeded

If any error occurs other than the script ID error, script processing is canceled and the execution of the SCRPT instruction is terminated. If the script ID error occurs, D1 and D1+1 are updated and execution of the SCRPT instruction terminates.

Execution time

The execution time from when the execution of the specified script is started until it is completed is stored in 100 µs increments. For example, when it takes 1.45 ms to execute the specified script, 15 is stored as the execution time. When it takes 6,553.5 ms or longer to execute the specified script, 65535 is stored as the execution time. The SCRPT instruction execution time is affected by interrupt processing or other processing that occurs during the execution.

If any error occurs other than the script ID error, the script execution time from when the script execution is started until the error occurs is stored as the execution time. If the script ID error occurs, zero is stored as the execution time.

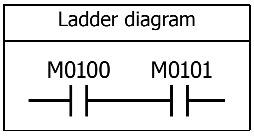
Script selection

To specify a registered script ID for S1 in WindLDR, click **Refer** button on the **SCRPT (Script)** dialog box to open Script Manager dialog box. Choose the script ID to execute and click on **Select** button on **Script Manager** dialog box. The selected ID will be entered to S1.

Script Function Overview

A script is a function for programming complicated processing with conditional branch, logical operations, arithmetic operations, and functions as text.
The programmed scripts can be executed in a ladder program.

For example, the logical AND operation is written as follows.



⇔

Script

[M0100] && [M0101]

← The script is programmed as text.

■Script Programming and Management

Scripts are programmed using the **Script Editor** dialog box in WindLDR and managed with the **Script Manager** dialog box.

Notes:

- In Script Editor, you can program scripts by selecting conditional expressions, operators, and functions from a list, and you can also check them for errors. The scripts can also be exported as text files, so they can be edited using a text editor such as Notepad. If you save an edited script as a text file, it can be imported into Script Editor.
  For details, see "Script Editor" on page 26-8.
- In Script Manager, you can manage the scripts as a group by adding scripts created in Script Editor or deleting them. For details, see "Script Manager" on page 26-7.

Script Data Type

The data type for the range of data that will be handled in the script must be configured by taking into consideration the details of the processing in the script including the maximum and minimum values of data that will be handled and whether negative numbers and real numbers are required.

Notes:

- The data type is configured in Script Editor. For how to configure it, see "Script Editor" on page 26-8.
- Scripts can also be written that designate the data type for processing enclosed in curly brackets "{ }".
- Scripts can also be written that designate data types in executable line units. See "Programming Scripts" on page 26-12.

Data Types

The following five data types can be processed in a script. For details about data types in ladder, see "Data Types for Advanced Instructions" on page 3-7.

Data Type	Symbol	Bits	Quantity of Data Registers Used	Range of Decimal Values
Word (Unsigned 16 bits)	**W**	16 bits	1	0 to 65,535
Integer (Signed 16 bits)	**I**	16 bits	1	–32,768 to 32,767
Double Word (Unsigned 32 bits)	**D**	32 bits	2	0 to 4,294,967,295
Long (Signed 32 bits)	**L**	32 bits	2	–2,147,483,648 to 2,147,483,647
Float (Floating point)	**F**	32 bits	2	-3.402823×10^{38} to 3.402823×10^{38}

Note: Some functions cannot be used if the data type is different. Check the function in the notation list.
See "Programming Scripts" on page 26-12.

Device List

This section shows the devices that can be used in Script Editor and the device notation.

This section describes available devices and its notation that can be used in the **Script Editor** dialog box.

Note: The device ranges differ depending on each FC6A Series MICROSmart. Specify the devices within the device range of the selected FC6A Series MICROSmart. For device ranges, see "Device Addresses" on page 2-1.

Device		Notation in Script Editor
Bit device	I (Inputs)	I0 to I27
	I (Expansion inputs)	I30 to I10597
	Q (Outputs)	Q0 to Q17
	Q (Expansion outputs)	Q30 to Q10597
	M (Internal relays)	M0000 to M7997, M10000 to M21247
	M (Special internal relays)	M8000 to M9997
	R (Shift registers)	R000 to R255
	T (Timer contacts)	C000 to C511
	C (Counter contacts)	T000 to T1999
	D (Data register bits)	D0000.0 ... D0000.15 to D7999.0 ... D7999.15 D10000.0 ... D10000.15 to D61999.0 ... D61999.15
	D (Special data register bits)	DD8000.0 ... D8000.15 to D8899.0 ... D8899.15
	D (Non-retentive data register bits)	D70000.0 ... D70000.15 to D269999.0 ... D269999.15
Word device	D (Data registers)	D0000 to D7999 D10000 to D61999
	D (Special data registers)	D8000 to D8899
	D (Non-retentive data registers)	D70000 to D269999
	TC (Timer current values)	TC000 to TC1999
	TP (Timer preset values)	TP000 to TP1999
	CC (Counter current values)	CC000 to CC511
	CP (Counter preset values)	CP000 to CP511

Calculations in which both bit and word devices are used are not allowed.

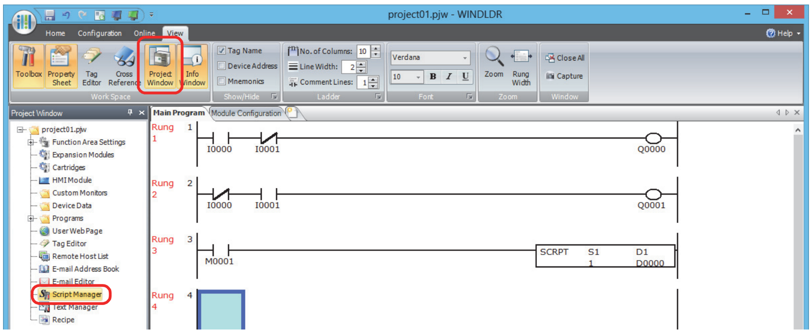
Bit devices are always processed as bits, and values of those devices are 0 (OFF) or 1 (ON).

Script Programming and Management

Script Registration Procedure

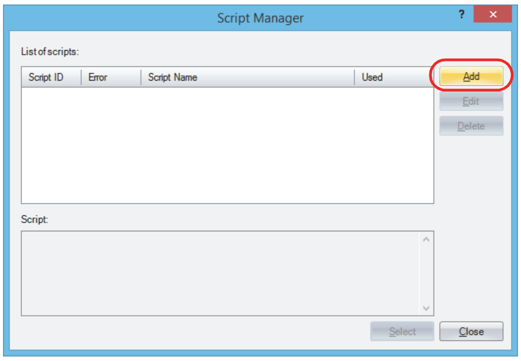
This section describes the procedure to create and register a script. The registered script can be executed by specifying the script ID in the SCRPT instruction.

1. On the **Project Window**, double-click **Script Manager**.



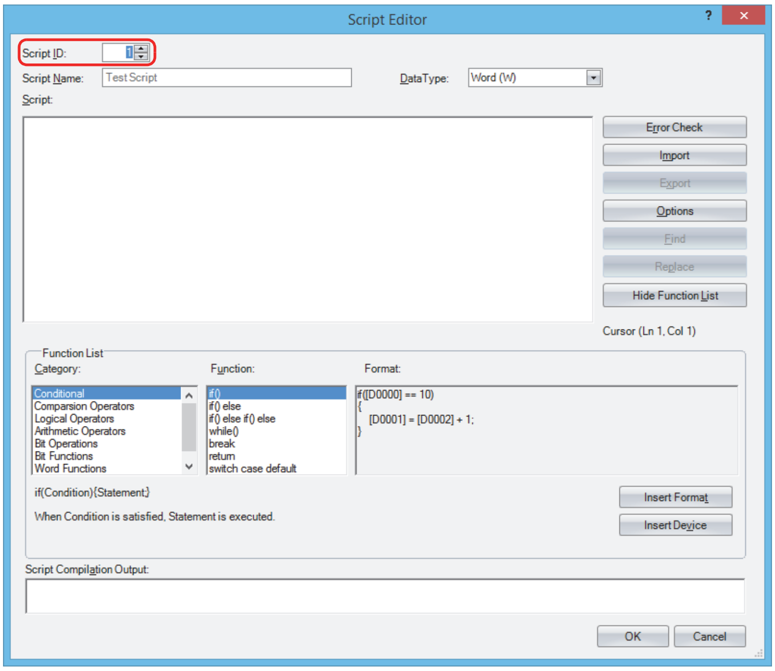
Script Manager dialog box opens.

2. Click **Add**.



Script Editor dialog box opens.

3. Specify **Script ID.**
When creating a new script, enter the script ID (1 to 255).



Script ID: 1

Script Name: Test Script

DataType: Word (W)

Cursor (Ln 1, Col 1)

```
if([D0000] == 10)
{
    [D0001] = [D0002] + 1;
}
```

if(Condition){Statement}

When Condition is satisfied, Statement is executed.

4. Enter **Script Name**.
   The script name can be entered up to 40 single-byte alphanumeric characters.
   However, the following single-byte characters cannot be used:
   / \:*?"<>|

5. Select **Data Type**.

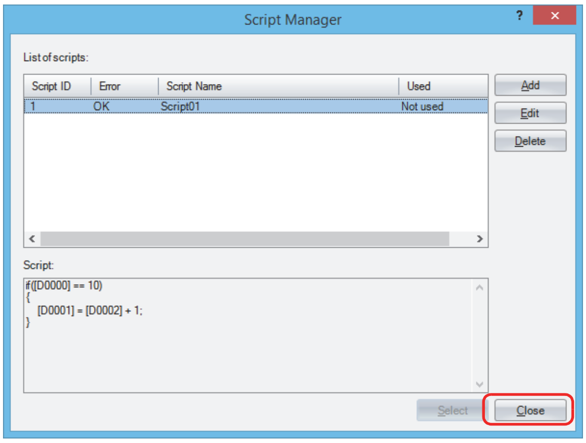
Note: The script is executed with the selected data type.

6. In **Script**, write the program.

Note: To create a script using the sample codes provided by WindLDR, under **Function list**, select **Category** and **Function**, and then click **Insert Format**. The sample code displayed in **Format** is inserted at the cursor position in **Script**.

7. After you finish creating the script, click **OK**.
   You are returned to the Script Manager dialog box and the created script is displayed in **List of scripts**.

8. Click **Close**.

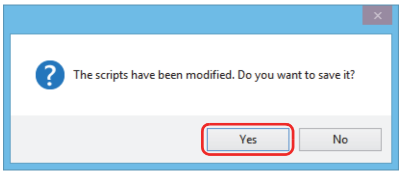


Script ID	Error	Script Name	Used
1	OK	Script01	Not used

```
if([D0000] == 10)
{
   [D0001] = [D0002] + 1;
}
```

A save confirmation message is displayed.

9. Click **OK**.

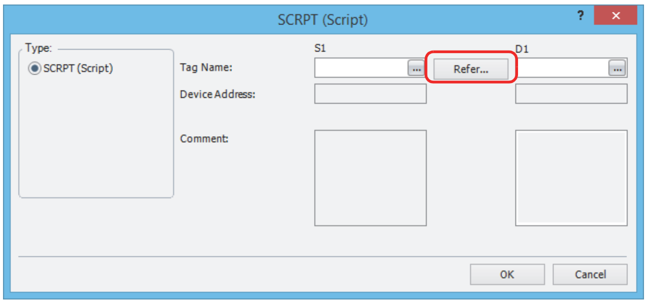


The scripts have been modified. Do you want to save it?

The script is saved and the **Script Manager** dialog box is closed.

Notes:

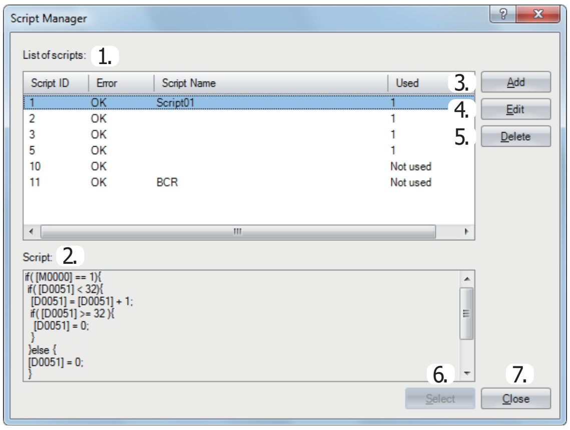
- If you click **No**, the script is not saved and the **Script Manager** dialog box is closed.
- Script Manager can also be opened from the **SCRPT (Script)** dialog box.
  Click **Refer** to open Script Manager.



Type: ● SCRPT (Script)

Script Manager

In the **Script Manager** dialog box, you can add scripts created in the **Script Editor** dialog box or delete registered scripts.



Script ID	Error	Script Name	Used
1	OK	Script01	1
2	OK		1
3	OK		1
5	OK		1
10	OK		Not used
11	OK	BCR	Not used

```
if( [M0000] == 1){
 if( [D0051] < 32){
  [D0051] = [D0051] + 1;
  if( [D0051] >= 32 ){
   [D0051] = 0;
  }
 }else {
 [D0051] = 0;
 }
```

1. List of scripts

Shows the list of registered scripts.

Script ID:	Shows the script ID of the registered scripts (1 to 255).
Error:	Shows **OK** if the registered script has no errors. Shows **NG** if there are any errors.
Script name:	Shows the name of the registered scripts.
Used:	Shows whether or not the registered scripts are used in this project.

2. Script

Shows the contents of the script selected in the **List of scripts**.

3. Add

Click this button to display the **Script Editor** dialog box in order to create and add a script.

For details, see "Script Editor" on page 26-8.

4. Edit

Click this button to open the **Script Editor** dialog box in order to edit the script selected in the **List of scripts**.
For details, see "Script Editor" on page 26-8.

5. Delete

Click this button to delete the script selected in the **List of scripts**.

6. Select

This can only be used when Script Manager has been opened from the SCRPT instruction dialog box.

Select a script in the list of scripts and click this button to enter the selected script ID in S1 on the dialog box.

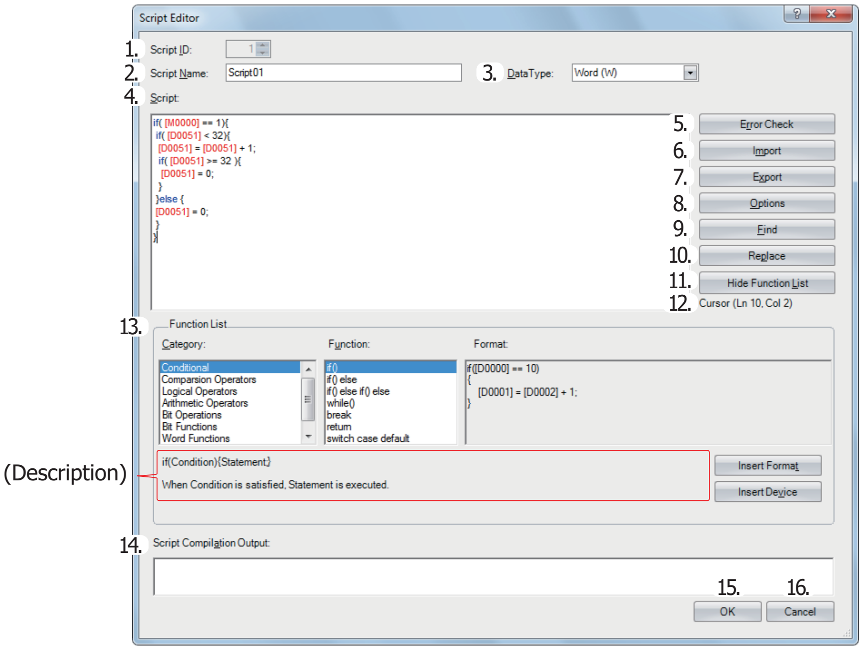
7. Close

Click this button to close Script Manager.

Note: If you click **Close** when the list of scripts was modified, a save confirmation message is displayed. Click **Yes** on the confirmation message to save the changes. Click **No** to discard the changes and close Script Manager.

Script Editor

Script Editor is where you create new scripts and edit the scripts selected in Script Manager.



Script ID: 1

Script Name: Script01

DataType: Word (W)

```
if( [M0000] == 1){
 if( [D0051] < 32){
  [D0051] = [D0051] + 1;
  if( [D0051] >= 32 ){
   [D0051] = 0;
  }
 }else {
 [D0051] = 0;
 }
}
```

Cursor (Ln 10, Col 2)

```
if([D0000] == 10)
{
   [D0001] = [D0002] + 1;
}
```

if(Condition){Statement}

When Condition is satisfied, Statement is executed.

1. Script ID

When creating a new script, enter the script ID (1 to 255). When editing an existing script, this shows the script ID that was set.

2. Script Name

Enter the script name. The script name can be entered up to 40 characters.

However, the following single-byte characters cannot be used:

/ \ : * ? " < > |

3. Data Type

Select the data type to process in the script. For details about the data types, see "Script Data Type" on page 26-3.

4. Script

Enter the script.

The restrictions for one script are a maximum of 240 characters per line and a maximum of 1,024 lines per script.

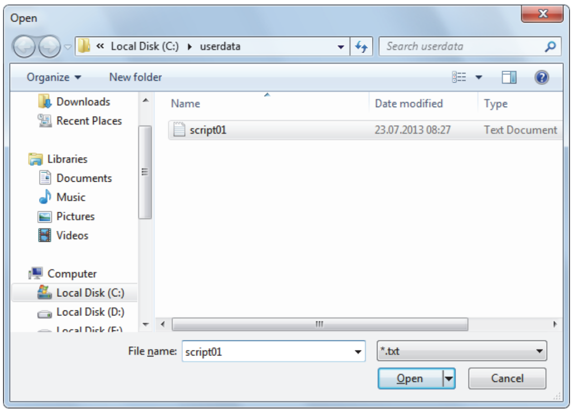
5. Error Check

Click this button to check errors in the current script.

6. Import

Click this button to display the **Open** dialog box so you can import a script program.

If you select a script that was saved (exported) as a text file (*.txt) and click **Open**, the selected script is inserted at the current cursor position in the current script.



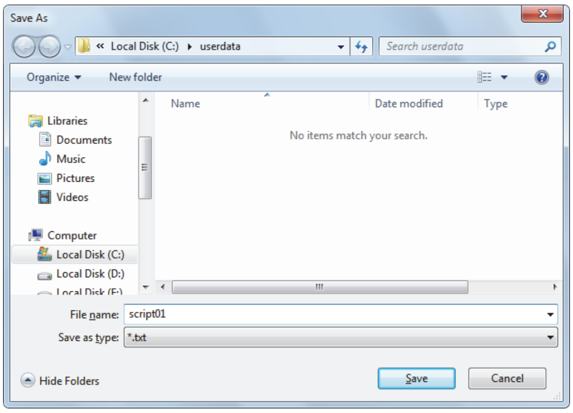
Name	Date modified	Type
script01	23.07.2013 08:27	Text Document

File name: script01

*.txt

7. Export
Click this button to display the **Save As** dialog box so you can export the script program.

If you select the location to save the script, enter a file name, and click **Save**, the current script is saved as a text file (*.txt). The saved script can be inserted with **Import**.



No items match your search.

File name: script01

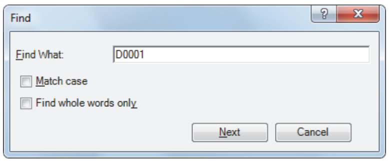
Save as type: *.txt

8. Options
Click this button to display the **Options** dialog box.

In the **Options** dialog box, you can configure the font for the text used in the **Script** text box, the colors, the tab indents, and other settings. For details, see "Script Formatting Options Dialog Box" on page 26-11.

9. Find
Click this button to display the **Find** dialog box.

You can search the entered text in the script.



Find What: D0001

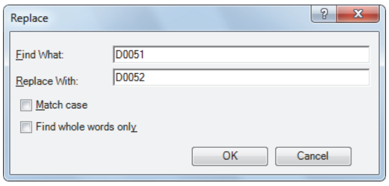
☐ Match case

☐ Find whole words only

Note: If you first select an area in the **Script** and then click **Find**, you can search only in the selected area.

10. Replace
Click this button to display the **Replace** dialog box.

Enter the text to search in the script in **Find What** and enter the text to replace it with in **Replace With**.



Find What: D0051

Replace With: D0052

☐ Match case

☐ Find whole words only

Notes:

- This is an effective tool when replacing device addresses.
- If you first select an area in the **Script** and then click **Replace**, you can search and replace text only in the selected area.

11. Show/Hide Function List
This button shows and hides **Function List** and **Output**.

Note: Click and drag the bottom right corner of Script Editor and you can change the size of the box for editing scripts. By hiding **Function List** and **Output**, you can increase the size of the script editing area (text box) and make it easier to edit the script.

12. Cursor
Shows the current cursor position in the **Script** text box as the line number and the column number.

13. Function List

Category:	Shows the function category list.
Function:	Shows the list of functions in the selected category.
Format:	Shows a programming example of the selected function.
Description:	Shows a description of the selected function.
Insert Format:	Click this button to insert the content displayed in **Format** at the cursor position.
Insert Device:	Click this button to open the **Tag Editor** dialog box. Specify a device address and click **OK** to insert the specified device address at the cursor position.

14. Script Compilation Output
Shows the details of errors when there are errors in the script in the error check.

If you double-click a comment displayed in **Script Compilation Output**, the section that corresponds to the error is highlighted in the **Script**.

Note: Depending on the error, the error may exist on a line that differs from the line displayed in the **output**, or it may show multiple errors.

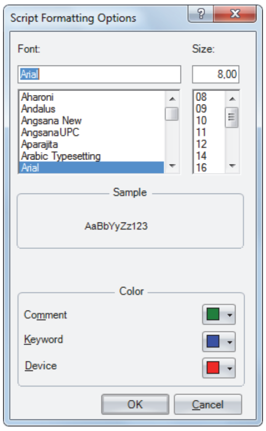
15. OK
Click this button to perform the error check on the current script, and then return to Script Manager after the script is saved.

Note: If there are any errors in the current script the save confirmation message is displayed. Scripts that contain errors can be saved.

16. Cancel
Click this button to return to Script Manager without saving the current script.

Script Formatting Options Dialog Box

This dialog box allows you to specify **Font**, **Size**, **Tab indent**, and **Color** used in the Script Editor **Script** text box.



Script Formatting Options

Font: Arial

Size: 8,00

Aharoni
Andalus
Angsana New
AngsanaUPC
Aparajita
Arabic Typesetting
Arial

08
09
10
11
12
14
16

Sample: AaBbYyZz123

Color

Comment 🟩

Keyword 🟦

Device 🟥

■Font

Enter or select the font name for text displayed in **Script**.

■Size

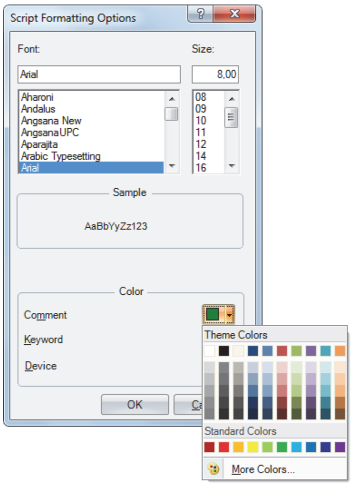
Enter or select the size (pixels) of text displayed in **Script**.

■Sample

Shows a sample of text that will be displayed in the **Script** text box with the character font and the character size specified by **Font** and **Size**.

■Colors

Shows the colors for **Comment**, **Keyword**, and **Device**.
Click the colored buttons to display the Theme Colors dialog box.



Script Formatting Options

Font: Arial

Size: 8,00

Aharoni
Andalus
Angsana New
AngsanaUPC
Aparajita
Arabic Typesetting
Arial

08
09
10
11
12
14
16

Sample: AaBbYyZz123

Color

Comment 🟩

Keyword

Device

Characters that are not comments, keywords, or devices are all shown in black.

Programming Scripts

Format List

This section describes the notation for control statements, operators, functions, data type designations, and other elements, and it also describes their operation.

Except for comments, enter everything as single-byte characters. For specific programming examples, see "Script Programming Examples" on page 26-19.

Control Statements

Conditional expressions are written here as (Cond. expr.), (Cond. expr. 1), (Cond. expr. 2).

Executable lines are written as (Exe. line), (Exe. line 1), (Exe. line 2) and so on.

■**Conditional branch**

Format		Description

if
else
else if
	if ((Cond. expr.))
	{ (Exe. line); }
		Execution line is executed if the conditional expression is satisfied.

	if ((Cond. expr.))
	{ (Exe. line 1); }
	else
	{ (Exe. line 2); }
		Execution line 1 is executed if the conditional expression is satisfied.
		Execution line 2 is executed if it is not satisfied.

	if ((Cond. expr. 1))
	{ (Exe. line 1); }
	else if ((Cond. expr. 2))
	{ (Exe. line 2); }
	else
	{ (Exe. line 3); }
		Execution line 1 is executed if the conditional expression 1 is satisfied.
		Conditional expression 2 is evaluated if conditional expression 1 is not satisfied, and execution line 2 is executed if conditional expression 2 is satisfied.
		Execution line 3 is executed if conditional expression 2 is not satisfied too.

switch
case
default
	switch ((Cond. expr.))
	{
	case constant 1:
	(Exe. line 1);
	break;
	case constant 2:
	(Exe. line 2);
	break;
	default:
	(Exe. line 3):
	break;
	}
		Execution line 1 is executed if the value of conditional expression matches constant 1.
		Execution line 2 is executed if the value of conditional expression matches constant 2.
		Execution line 3 is executed if the value of conditional expression does not match constant 1 nor constant 2.

■**Repeat**

Format		Description

while	**while** ((Cond. expr.))
	{
	(Exe. line);
	}
		Execution line is repeatedly executed while the conditional expression is satisfied.
		The execution will go into an infinite loop when the conditional expression is always satisfied, so do not use fixed values or devices that do not change as the conditional expression.

■Halt and exit

Format		Description

break	while (Cond. expr. 1)
	{
	  if (Cond. expr. 2)
	  {
	    Exe. line 1;
	    break;
	  }
	 Exe. line 2;
	}
	Exe. line 3;
		Process will be as follows while the conditional expression 1 is satisfied.
		Execution line 2 is continuously executed while the conditional expression 2 is not satisfied.
		Once the conditional expression 2 is satisfied, the execution goes out of the loop by break (not executing execution line 2), and execution line 3 is executed.

break	switch (Cond. expr.)
	{
	  case constant 1:
	    Exe. line 1;
	    break;
	  case constant 2:
	    Exe. line 2;
	    break;
	}
	Exe. line 3;
		When the conditional expression matches the constant 1, execution line 1 is executed and the switch execution is terminated by break. Evaluation for constant 2 is not executed, and the execution moves to execution line 3.

return	return;	Script will exit and the next instruction in the user program is executed.

Operator

Device, constant, and temporary device are described as a, b here, and expressions are described as Expr., Expr. 1, Expr. 2.

■Relational operator

Operator	Format	Description
==	a == b	Compares if a is equal to b.
!=	a != b	Compares if a is not equal to b.
<	a < b	Compares if a is less than b.
<=	a <= b	Compares if a is less than or equal to b.
>	a > b	Compares if a is greater than b.
>=	a >= b	Compares if a is greater than or equal to b.

■Logical operator

Operator	Format	Description
&&	(Expr. 1) && (Expr. 2)	Calculates the logical product (AND) of Expr. 1 and Expr. 2.
||	(Expr. 1) || (Expr. 2)	Calculates the logical sum (OR) of Expr. 1 and Expr. 2.
!	!(Expr.)	Inverse the logic of Expr..

■Arithmetic operator

Operator	Format	Description
+	a + b	Adds a and b.
-	a - b	Subtracts b from a.
*	a * b	Multiplies a and b.
/	a / b	Divides a by b.
%	a % b	Calculates remainder after dividing a by b.
=	a = b	Assigns b to a.

■Bit operator

Operator	Format	Description
&	a & b	Calculates the logical product (AND) of each bit of a and b. Data types W (word), I (integer), D (double word), and L (long) can be used. F (float) cannot be used.
|	a | b	Calculates the logical sum (OR) of each bit of a and b. Data types W (word), I (integer), D (double word), and L (long) can be used. F (float) cannot be used.
^	a ^ b	Calculates the exclusive logical sum (XOR) of each bit of a and b. Data types W (word), I (integer), D (double word), and L (long) can be used. F (float) cannot be used.
~	~a	Inverse the logic of each bits of a. For word devices and fixed values, 0 will be 65,535, and 65,535 will be 0. For bit device, 0 will be 1, and 1 will be 0. Data types W (word), I (integer), D (double word), and L (long) can be used. F (float) cannot be used.
<<	a << b	Shifts each bit of a to the left for b bit(s). Data types W (word), I (integer), D (double word), and L (long) can be used. F (float) cannot be used.
>>	a >> b	Shifts each bit of a to the right for b bit(s). Data types W (word), I (integer), D (double word), and L (long) can be used. F (float) cannot be used.

Function

Device, constant, and temporary device are described as a, b, c, d... here.

■Bit function

Function	Format	Description
Bit set	SET(a);	Turns bit device a to 1. This is the same result as a = 1;.
Bit reset	RST(a);	Turns bit device a to 0. This is the same result as a = 0;.
Bit reverse	REV(a);	Reverses the 1 and 0 of bit device a. This is the same result as a = ~a;.

■Word function

Arithmetic operation

Function	Format	Description
Maximum value	MAX(a,b,c)	Maximum value out of a, b, c is returned. • This can be used for all data types. • Up to 15 arguments can be defined.
Mininum value	MIN(a,b,c)	Minimum value out of a, b, c is returned. • This can be used for all data types. • Up to 15 arguments can be defined.
Exponential function	EXP(a)	Exponential function of a is returned. • This can only be used for data type F (float).
Natural logarithm (base: e)	LOGE(a)	Natural logarithm (base is e) for a is returned. • This can only be used for data type F (float). • Set a value larger than 0 for argument.
Common logarithm (Base: 10)	LOG10(a)	Common logarithm (base is 10) for a is returned. • This can only be used for data type F (float). • Set a value larger than 0 for argument.
Exponentiation	POW(a,b)	a to the b power is returned. • This can only be used for data type F (float).

Function	Format	Description
Square root	ROOT(a)	Square root of a is returned. • This can only be used for data type F (float).
Sine	SIN(a)	Sine of a (-1 to +1) is returned. Specify arbitrary formula to represent angle (units in radian) for argument a. • This can only be used for data type F (float).
Cosine	COS(a)	Cosine of a (-1 to +1) is returned. Specify arbitrary formula to represent angle (units in radian) for argument a. • This can only be used for data type F (float).
Tangent	TAN(a)	Tangent of a (-1 to +1) is returned. Specify arbitrary formula to represent angle (units in radian) for argument a. • This can only be used for data type F (float).
Arcsine	ASIN(a)	Arcsine of a (-1 to +1) in radian value (-π/2 to +π/2) is returned. Specify arbitrary formula for argument a. • This can only be used for data type F (float).
Arccosine	ACOS(a)	Arccosine of a (-1 to +1) in radian value (0 to π) is returned. Specify arbitrary formula for argument a. • This can only be used for data type F (float).
Arctangent	ATAN(a);	Arctangent of a (-1 to +1) in radian value (-π/2 to +π/2) is returned. Specify arbitrary formula for argument a. • This can only be used for data type F (float).
Conversion from Angle to Radian	RAD(a);	Value of a is converted from degree (°) to radian, and the value is returned. • This can only be used for data type F (float).
Conversion from Radian to Angle	DEG(a);	Value of a is converted from radian to degree (°), and the value is returned. • This can only be used for data type F (float).

Data type conversion

Function	Format	Description
Conversion from BCD to Binary	BCD2BIN(a)	BCD value of a is returned in binary value. • Data types W (word), I (integer), D (double word), and L (long) can be used. F (float) cannot be used.
Conversion from Binary to BCD	BIN2BCD(a)	Binary value of a is returned in BCD value. • Data types W (word), I (integer), D (double word), and L (long) can be used. F (float) cannot be used.
Conversion from float to Binary	FLOAT2BIN(a)	Float value of a is returned in binary value. Values after the decimal point is truncated. • Data types D (double word) and L (long) can be used.
Conversion from Binary to float	BIN2FLOAT(a)	Binary value of a is returned in float value. • This can be used with data types D (double word) and L (long).
Conversion from Decimal to String character	DEC2ASCII(a,b)	Decimal number value a is converted to a character string, and store in order with b as a first device. • Data types W (word), I (integer), D (double word), and L (long) can be used. F (float) cannot be used.
Conversion from String character to Decimal	ASCII2DEC(a)	Character string a is returned as decimal number value. • Data types W (word), I (integer), D (double word), and L (long) can be used. F (float) cannot be used.

Data comparison and copy

Function	Format	Description
Data comparison	`MEMCMP(a,b,c)`	a: First device of comparison target 1 b: First device of comparison target 2 c: Range of comparison (in words) Values of device a for c words and values of device b for words c are compared. 1 is returned if all the values of devices match, and 0 is returned if any of the value does not match. • Specified range is compared in word unit, and result is returned. • Up to 64 words can be compared.
Data copy	`MEMCPY(a,b,c)`	a: First device of copy source b: First device of copy target c: Range of copy (in words) Values from a for c words are copied to b for c words respectively. • Specified range is compared in word unit, and result is returned. • Up to 64 words can be copied.
Bit device (1 word length) to Bit device (1 word length)	`BITS2BITS(a,b);`	a: First device of copy target (bit device) b: First device of copy source (bit device) Copies one word worth of data from a to b. • 16 bits are processed from the starting bit device as one word.
Bit device (1 word length) to Word device	`BITS2WORD(a,b);`	a: First device of copy target (word device) b: First device of copy source (bit device) Copies one word worth of data from a to b. • 16 bits are processed from the starting bit device as one word.
Word device to Bit device (1 word length)	`WORD2BITS(a,b);`	a: First device of copy target (bit device) b: First device of copy source (word device) Copies one word worth of data from a to b. • 16 bits are processed from the starting bit device as one word.

Offset

Function	Format	Description
Indirect addressing	`OFFSET(a,b)`	a: Reference device b: Device to store the offset value (0 to 32767) Specifies the device at b words from a. Indirect read Specify OFFSET function to the right of the assignment statement. Format example: `C = OFFSET(a,b)= c` Operation: Stores the value of device at b words from a into c. Indirect write Specify OFFSET function to the left of the assignment statement. Format example: `OFFSET(a,b) = c` Operation: Stores the value of c into the device at b words from a. • Store the value appropriate for the data type as the offset value. For example, when the data type is I (integer), store the offset value of I (integer) into the device.

Character string operations

When handling strings, the NULL terminating character (0x00) is the end of the string. The NULL terminating character is not included in the length of the string.

Function	Format	Description
Character string copy	STRCUT(a , b , c , d)	a : First device of copy source b : First device that stores the source character string to copy c : Copy start position (0 to 127) d : Character count to copy (1 to 128) From the character string starting from b , character string from c bytes forward for d characters are stored into a for d characters.
Character string length	STRLEN(a)	Returns the length of the character string that starts from a .
Concatenation of character strings	STRCAT(a , b);	Concatenates the character string that starts from b to the character string that starts from a and stores a at the start.
Character string search	STRSTR(a , b);	Searches for the character string that starts from b in the character string that starts from a and returns the position in that string where it was found (number of characters from the start-1). • The maximum character string length that can be searched is 128 characters.

■Data Type Designations

Scripts written that designate the data type for processing enclosed in curly brackets "{ }".
Executable lines are written as Exe. line , Exe. line 1, Exe. line 2, Exe. line 3, and so on.

Format		Description

asword	asword
	{
	Exe. line 1 ;
	}
	Exe. line 2 ;
		Executable line 1 is executed as data type W (word).
		Executable line 2 is executed as the data type set in **Data Type** in Script Editor.

asinterger	asinterger
	{
	Exe. line 1 ;
	}
	Exe. line 2 ;
		Executable line 1 is executed as data type I (integer).
		Executable line 2 is executed as the data type set in **Data Type** in Script Editor.

asdoubleword	asdoubleword
	{
	Exe. line 1 ;
	}
	Exe. line 2 ;
		Executable line 1 is executed as data type D (double word).
		Executable line 2 is executed as the data type set in **Data Type** in Script Editor.

aslong	aslong
	{
	Exe. line 1 ;
	}
	Exe. line 2 ;
		Executable line 1 is executed as data type L (long).
		Executable line 2 is executed as the data type set in **Data Type** in Script Editor.

asfloat	asfloat
	{
	Exe. line 1 ;
	}
	Exe. line 2 ;
		Executable line 1 is executed as data type F (float).
		Executable line 2 is executed as the data type set in **Data Type** in Script Editor.

Note: The notation used for data type designations can also be written by abbreviating a portion of the keywords.

Notation	Notation Abbreviating a Portion of the Keyword
asword	asw
asinterger	asi
asdoubleword	asd
aslong	asl
asfloat	asf

There is no difference in the operation of the data type designation by using either type of notation.

Other

This section describes definitions for constant, device, temporary device, and comment.

■Constant

Constant can be defined as decimal or hexadecimal number.

Sample definition of decimal numbers

```
1234
```

Define the numeric value directly.

```
-1234
```

Define the negative number with a "-" (minus) symbol at the beginning.

```
12.34
```

Decimal number can be defined for real numbers (float).
Define a "." (period) between the whole numbers and decimal numbers.

There are 2 ways to define hexadecimal numbers.

Sample definition of hexadecimal numbers

```
0x12AB
```

Define "0" (zero) and "x" (lower case x) at the beginning of the value.

```
12ABh
```

Append "h" at the tail of the value.

■Device Address

Device Address is defined with the device symbol and address within "[" and "]".

Definition of the device

```
[Device symbol and address]
```

(Space between the device symbol and address is not required.)

Sample definition

```
[D0100]
```

■Temporary Device

Temporary device is a device that can be used only inside the script. It can store a value and can be used as a variable.
It is defined with a device symbol "@" followed by address (1 – 32).

Definition of the temporary device

```
@address
```

(Space between the device symbol "@" and address is not required.)

Sample definition

```
@2
```

Temporary device number 2

Notes:

- All the values for temporary devices are set to "0" when the execution of the script is started.
- The processing inside the curly brackets "{ }" for the data type designation cannot use temporary devices.

■Comment

A note defined in the script is called a comment. The line with "//" defined at the beginning of a line will become a comment.
"//" is defined with a single-byte. Double-byte characters can be written after "//".

Definition of comment

```
// Arbitrary note
```

Sample definition

```
// Store the initial value to calculation data [D0100] for process A
[D0100] = 1234;
          :
          :
```

← This line is not executed.

Notes:

- Describing comments to explain the contents of the script is useful especially when another person works on the scripts or when some time has passed after editing the scripts.
- Comments are ignored (not executed) when the script is executed, so they can be defined freely without interfering the execution time.

Script Programming Examples

This section describes script programming examples for control statements, arithmetic operators, and functions, as well as their operations.

1. Control Statements

Example 1.1 Conditional branch

Script

```
if ([D0100])
{
     [D0102] = 100;
}
```

Operation Description
If the value of D0100 is not 0, then 100 is stored in D0102.

Example 1.2 Conditional branch

Script

```
if ([D0100])
{
     [D0102] = [D0103] + [D0104] + [D0105];
}
```

Operation Description
If the value of D0100 is not 0, the value of D0103, D0104, and D0105 are added and the result is stored in D0102.

Example 1.3 Conditional branch

Script

```
if (0 != [D0100])
{
     if (0 != [D0102])
     {
          [D0103] = 0x1234;
     }
}
```

Operation Description
If the value of D0100 is not 0 and the value of D0102 is not 0 either, 0x1234 is stored in D0103.

If the value of D0100 is not 0 and the value of D0102 is 0, then nothing is executed.

If the value of D0100 is 0, then nothing is executed regardless of the value of D0102.

Example 1.4 Conditional branch

Script

```
if ((0 != [D0100]) || (0 != [D0102]))
{
     [D0103] = 100;
}
else
{
     [D0104] = [D0105] + 100;
}
```

Operation Description
If either the value of D0100 or the value of D0102 is not 0, then 100 is stored in D0103.

If the values of both D0100 and D0102 are 0, 100 is added to D0105 and the result is stored in D0104.

Example 1.5 Conditional branch

Script

```
if ([D0100] == 0)
{
    [D0102] = 0x1234;
}
else if ([D0100] == 1)
{
    [D0102] = 0x5678;
}
else
{
    [D0102] = 0x9999;
}
```

Operation Description
If the value of D0100 is 0, then 0x1234 is stored in D0102.

If the value of D0100 is 1, then 0x5678 is stored in D0102.

If the value of D0100 is not 0 nor 1, then 0x9999 is stored in D0102.

Example 1.6 Conditional branch

Script

```
if ([D0100])
{
    if ([D0102])
    {
        if ([D0103])
        {
            [D0104] = 100;
        }
        else
        {
            [D0104] = 200;
        }
    }
}
```

Operation Description
If the value of D0100, D0102, and D0103 are all not 0, then 100 is stored in D0104.

If the values of D0100 and the value of D0102 are not 0 and the value of D0103 is 0, then 200 is stored in D0104.

If either the value of D0100 or D0102 is 0, then nothing is executed regardless of the value of D0103.

Example 1.7 Iteration

Script

```
[D0100] = 10;
[D0102] = 10;

while (0 < [D0100])
{
    [D0102] = [D0102] + 1;
    [D0100] = [D0100] - 1;
}
```

Operation Description
If the value of D0100 is larger than 0, then 1 is repeatedly added to the value of D0102 and 1 is repeatedly subtracted from the value of D0100.

In the script example above, when the while statement repeats ten times, the value of D0100 becomes 0 and the while statement ends.

After this script is executed, the value of D0100 is 0 and the value of D0102 is 20.

Example 1.8 Iteration

Script

```
[D0100] = 0;
[D0102] = 3;
[D0103] = 5;

while ([D0100] == 0)
{
    [D0102] = [D0102] + 1;

    if ([D0103] == [D0102])
    {
        SET([M0000]);
        break;
    }
}
```

Operation Description

While the value of D0100 is 0, the while statement is repeated.
Inside the while statement, if the values of D0102 and D0103 are equal, the while statement will terminate, and after [M0000] changes to 1, execution breaks out of the while statement.
In the example script above, the values of D0102 and D0103 are equal when the while statement repeats twice, and after M0000 changes to 1, execution breaks out of the while statement loop. After execution, the value of D0100 is 0, the value of D0102 is 5, the value of D0103 is 5, and M0000 is 1.

Example 1.9 Indirect write and indirect read using iteration (while statement)

Script

```
//Transfer D0010 through D0019 to D0100 through D0109

// Initialize the offset value
[D0000] = 0;

// Loop ten times
while ([D0000] < 10)
{
    // Transfer 1 word by indirect assignment
    OFFSET([D0100] , [D0000]) = OFFSET([D0010] , [D0000]);
    // Increment indirect value
    [D0000] = [D0000] + 1;
}
```

Operation Description

This script stores the values of D0010 through D0019 in D0100 through D0109.
It operates as follows.
First, the offset value D0000 is initialized and set to 0.

First iteration (loop) : The value of D0000 is 0, so the condition "[D0000] < 10" is true and the statements inside while are executed.

- The value of D0010, 0 words from D0010, is stored in D0100, 0 words from D0100.
- 1 is added to the value of offset value D0000, so that it becomes 1.

Second iteration (loop) : The value of D0000 is 1, so the condition "[D0000] < 10" is true and the statements inside while are executed.

- The value of D0011, 1 word from D0010, is stored in D0101, 1 word from D0100.
- 1 is added to the value of offset value D0000, so that it becomes 2.

:

(Repeats in the same manner for the third to ninth iterations)

:

Tenth iteration (loop) : The value of D0000 is 9, so the condition "[D0000] < 10" is true and the statements inside while are executed.

- The value of D0019, 9 words from D0010, is stored in D0109, 9 words from D0100.
- 1 is added to the value of offset value D0000, so that it becomes 10.

The value of D0000 is 10, so the condition "[D0000] < 10" is false and execution breaks out of the while loop.
After execution, the values of D0100 through D0109 are the same with the values of D0010 through D0019.

Example 1.10 Decimal to octal conversion using a while statement

Script

```
// Convert a decimal value to octal
// - For example, convert 10 (dec) to 12 (oct), 16 (dec) to 20 (oct)
// - Convert a value to octal up to 4 digits max

@1 = 0;          // while counter
@2 = [D0100];    // gets original data
@3 = 1;          // decimal base
@4 = 0;          // calculation results

// repeat four times
while (@1 < 4)
{
    // Extract 1st octal digit from original data. Store working result in @10.
    @10 = @2 % 8;
    //Convert the extracted results to decimal and add to the results
    @4 = @4 + (@10 * @3);

    //Increase the decimal base by one digit
    @3 = @3 * 10;
    // Decrease the original data by one digit
    @2 = @2 / 8;
    // If @2 is 0, exit the white statement
    if (0 == @2)
    {
        break;
    }

    // Increment while counter by 1
    @1 = @1 + 1;

}

// Store the calculation result in D0200
[D0200] = @4;
```

Operation Description
This example converts a decimal value to octal using a while statement.
By repeating the process to divide the original decimal data by 8 and converting each digit to octal in a while statement, the conversion is implemented up to four digits.
The original decimal value is stored in D0100. After the script is executed, the converted octal value is stored in D0200.

Example 1.11 Conditional branch with switch

Script

```
switch ([D0100])
{
    case 10:
        [D0200] = 0x1234;
        break;
    case 999:
        [D0200] = 0x5678;
        SET([D0000.01]);
        break;
}
```

Operation Description
If the value of D0100 is 10, then 0x1234 is stored in D0200.
If the value of D0100 is 999, then 0x5678 is stored in D0200 and D0000.01 is set to 1.
If the value of D0100 is not 10 nor 999, then nothing is executed.

Example 1.12 Conditional branch with switch using the default statement

Script

```
switch ([D0100])
{
    case 0:
        [D0102] = 0x1234;
        break;
    case 1:
        [D0102] = 0x5678;
        break;
    default:
        [D0102] = 0x9999;
        break;
}
```

Operation Description
If the value of D0100 is 0, then 0x1234 is stored in D0102.
If the value of D0100 is 1, then 0x5678 is stored in D0102.
If the value of D0100 is not 0 nor 1, then 0x9999 is stored in D0102.

Example 1.13 Terminate the script with the return statement

Script

```
if (0x1234 == [D0100])
{
    [D0102] = 0x5678;
    return;
}
[D0103] = 0;
```

Operation Description
If the value of D0100 is not 0x1234, then 0 is stored in D0103.
If the value of D0100 is 0x1234, then 0x5678 is stored in D0102 and the script is terminated.
The return statement does not break out of a loop like the break statement, it terminates the script execution.

Example 1.14 Break out of a loop with the break statement

Script

```
[D0100] = 0;
[D0102] = 3;
[D0103] = 5;

while ([D0100] == 0)
{
    [D0102] = [D0102] + 1;

    if ([D0102] == [D0103])
    {
        SET([D0000.01]);
        break;
    }
}
```

Operation Description
While the value of D0100 is 0, the while statement is repeated until D0102 and D0103 match.
Inside the while statement, if the values of D0102 and D0103 are equal, the while statement will end and execution breaks out of the while statement.
In the example above, the values of D0102 and D0103 match when the while statement is repeated twice, and after D0000.01 changes to 1, the while statement ends. After execution, the value of D0100 is 0, the value of D0102 is 5, the value of D0103 is 5, and the value of D0000.01 is 1.

2. Relational Operators

Example 2.1 Equal to

Script

```
if ([D0100] == [D0102])
{
    [D0103] = 0x100;
}
```

Operation Description
If the value of D0100 is equal to the value of D0102, then 0x100 is stored in D0103.

Example 2.2 Not equal to

Script

```
if ([D0100] != [D0102])
{
    [D0103] = 0x100;
}
```

Operation Description
If the value of D0100 is not equal to the value of D0102, then 0x100 is stored in D0103.

Example 2.3 Less than

Script

```
if ([D0100] < [D0102])
{
    [D0103] = 0xl00;
}
```

Operation Description
If the value of D0100 is less than the value of D0102, then 0x100 is stored in D0103.

Example 2.4 Less than or equal to

Script

```
if ([D0100] <= [D0102])
{
    [D0103] = 0xl00;
}
```

Operation Description
If the value of D0100 is less than or equal to the value of D0102, then 0x100 is stored in D0103.

Example 2.5 Greater than

Script

```
if ([D0100] > [D0102])
{
    [D0103] = 0xl00;
}
```

Operation Description
If the value of D0100 is greater than the value of D0102, then 0x100 is stored in D0103.

Example 2.6 Greater than or equal to

Script

```
if ([D0100] >= [D0102])
{
    [D0103] = 0xl00;
}
```

Operation Description
If the value of D0100 is greater than or equal to the value of D0102, then 0x100 is stored in D0103.

3. Logical Operators

Example 3.1 Logical AND

Script

```
if (([D0100] == [D0200]) && ([D0300] == [D0400] + [D0500]))
{
    [D0600] = 100;
}
```

Operation Description
If the value of D0100 is equal to the value of D0200, and if the value of D0300 is equal to the value of D0400 and D0500 added together, then 100 is stored in D0600.

If either ([D0100] == [D0200]) or ([D0300] == [D0400] + [D0500]) is false, the processing in the brackets "{ }" is not executed.

Example 3.2 Logical OR

Script

```
if ((0 != [D0100]) || (0 != [D0200]))
{
    [D0300] = 100;
}
```

Operation Description
If the value of D0100 is not 0 or the value of D0102 is not 0, then 100 is stored in D0300.

If either one of them is true, the processing in the brackets "{ }" is executed.

Example 3.3 Logical inversion

Script

```
if (!([D0100] == 0x1234))
{
    [D0300] = 100;
}
```

Operation Description
If the value of D0100 is not equal to 0x1234, then 100 is stored in D0300.

Example 3.4 Logical inversion

Script

```
if (!(0 !=[D0100]))
{
    [D0300] = 100 ;
}
```

Operation Description
If the value of D0100 is 0, then 100 is stored in D0300.

This is the same as the code "if (0==[D0100]))".

4. Arithmetic Operators

Example 4.1 Addition

Script

```
[D0300] = [D0100] + [D0200];
```

Operation Description
The values of D0100 and D0200 are added together and the result is stored in D0300.

Example 4.2 Subtraction

Script

```
[D0300] = [D0100] - [D0200];
```

Operation Description
The value of D0200 is subtracted from the value of D0100 and the result is stored in D0300.

Example 4.3 Multiplication

Script

```
[D0300] = [D0100] * [D0200];
```

Operation Description
The values of D0100 and D0200 are multiplied together and the result is stored in D0300.

Example 4.4 Division

Script

```
[D0300] = [D0100] / [D0200];
```

Operation Description
The value of D0100 is divided by the value of D0200 and the result is stored in D0300.

Example 4.5 Modulo

Script

```
[D0300] = [D0100] % [D0200];
```

Operation Description
The value of D0100 is divided by the value of D0200 and the remainder is stored in D0300.

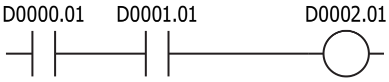
5. Bitwise Operators

Example 5.1 Logical AND

Script

```
if ([D0000.01] & [D0001.01])
{
    SET([D0002.01)]);
}
else
{
    RST([D0002.01]);
}
```

Operation Description
If the bitwise logical AND operation on the value of D0000.01 and the value of D0001.01 is 1, D0002.01 is changed to 1.
If the bitwise logical AND operation on the value of D0000.01 and the value of D0001.01 is 0, D0002.01 is changed to 0.
The operation is the same as the following ladder diagram.

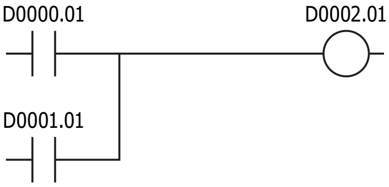


Example 5.2 Logical OR

Script

```
if ([D0000.01] | [D0001.01])
{
    SET([D0002.01]);
}
else
{
    RST([D0002.01]);
}
```

Operation Description
If the bitwise logical OR operation on the value of D0000.01 and the value of D0001.01 is 1, D0002.01 is changed to 1.
If the bitwise logical OR operation on the value of D0000.01 and the value of D0001.01 is 0, D0002.01 is changed to 0.
The operation is the same as the following ladder diagram.



Example 5.3 Logical XOR (exclusive OR)

Script

```
[D0200] = [D0100] ^ 0xFF;
```

Operation Description
The logical XOR operation on the value of D0100 and each bit in 0xFF is stored in D0200.
For example, if the value of D0100 is 15 (0x0F), then 240 (0xF0) is stored in D0200.

Example 5.4 Inversion

Script

```
[D0200] = ~[D0100];
```

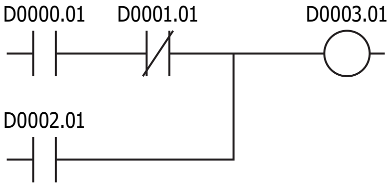
Operation Description
The bits in the value of D0100 are inverted and stored in D0200.
For example, if the value of D0100 is 0, then 65,535 is stored in D0200.

Example 5.5 Inversion

Script

```
if (([D0000.01] & ~[D0001.01]) | [D0002.01])
{
    SET([D0003.01]);
}
else
{
    RST([D0003.01]);
}
```

Operation Description
If the bitwise logical OR operation on the value of D0002.01 and the result of the bitwise logical AND operation on the value of D0000.01 and the inverted result of the value of D0001.01 is 1, then D0003.01 is changed to 1.
If the bitwise logical OR operation on the value of D0002.01 and the result of the bitwise logical AND operation on the value of D0000.01 and the inverted result of the value of D0001.01 is 0, then D0003.01 is changed to 0.
The operation is the same as the following ladder diagram.



Example 5.6 Left shift

Script

```
[D0300] = [D0100] << [D0200];
```

Operation Description
The value of D0100 is shifted to the left by the value of D0200 and the result is stored in D0300.
For example, if the value of D0100 is 1 and the value of D0200 is 3, 1 is shifted 3 bits to the left, and the result of 8 is stored in D0300.

Example 5.7 Right shift

Script

```
[D0300] = [D0100] >> [D0200];
```

Operation Description
The value of D0100 is shifted to the right by the value of D0200 and the result is stored in D0300.
For example, if the value of D0100 is 8 and the value of D0200 is 3, 8 is shifted 3 bits to the right, and the result of 1 is stored in D0300.

6. Bit Functions

Example 6.1 Set a bit

Script

```
SET([D0000.01]);
```

Operation Description
Turns D0000.01 to 1. The result is the same as `[D0000.01] = 1`.

Example 6.2 Reset a bit

Script

```
RST([D0000.01]);
```

Operation Description
Turns D0000.01 to 0. The result is the same as `[D0000.01] = 0`.

Example 6.3 Invert a bit

Script

```
REV([D0000.01]);
```

Operation Description
Inverts 1 and 0 in D0000.01. The result is the same as `[D0000.01] = ~[D0000.01]`.

7. Word Functions

Arithmetic operations

Example 7.1 Maximum value

Script

```
[D0200] = MAX([D0100], [D0110], [D0120], [D0130], [D0140]);
```

Operation Description
Out of the values stored in D0100, D0110, D0120, D0130, and D0140, the maximum value is stored in D0200.

Up to 15 arguments can be used.

Example 7.2 Mininum value

Script

```
[D0200] = MIN([D0100], [D0110], [D0120], [D0130], [D0140]);
```

Operation Description
Out of the values stored in D0100, D0110, D0120, D0130, and D0140, the minimum value is stored in D0200.

Up to 15 arguments can be used.

Example 7.3 Exponential function

Script

```
[D0010] = EXP([D0020]);
```

Operation Description
Calculates the exponential function of the value of D0020 and the result is stored in D0010.

Only the data type F (float) can be used.

Example 7.4 Natural logarithm

Script

```
[D0010] = LOGE([D0020]);
```

Operation Description
Calculates the natural logarithm of the value of D0020 and the result is stored in D0010.

Only the data type F (float) can be used.

Example 7.5 Common logarithm

Script

```
[D0010] = LOG10([D0020]);
```

Operation Description
Calculates the logarithm of the value of D0020 with 10 as the base and the result is stored in D0010.

Only the data type F (float) can be used.

Example 7.6 Power

Script

```
[D0010] = POW([D0020],[D0030]);
```

Operation Description
Calculates the power of the values.
For example, if the value of D0020 is 10 and the value of D0030 is 5, the function calculates 10 to the power of 5 and stores the result in D0010.
Only the data type F (float) can be used.

Example 7.7 Square root

Script

```
[D0010] = ROOT([D0020]);
```

Operation Description
Calculates the square root of the value of [D0020] and the result is stored in [D0010].
Only the data type F (float) can be used.

Example 7.8 Sine

Script

```
[D0010] = SIN([D0020]);
```

Operation Description
Calculates the sine of the radian value of D0020 and stores the result in D0010.
Only the data type F (float) can be used.

Example 7.9 Cosine

Script

```
[D0010] = COS([D0020]);
```

Operation Description
Calculates the cosine of the radian value of D0020 and stores the result in D0010.
Only the data type F (float) can be used.

Example 7.10 Tangent

Script

```
[D0010] = TAN([D0020]);
```

Operation Description
Calculates the tangent of the radian value of D0020 and stores the result in D0010.

Only the data type F (float) can be used.

Example 7.11 Arcsine

Script

```
[D0010] = ASIN([D0020]);
```

Operation Description
Calculates the arcsine of the value of D0020 and stores the result as radians in D0010.

Only the data type F (float) can be used.

Example 7.12 Arccosine

Script

```
[D0010] = ACOS([D0020]);
```

Operation Description
Calculates the arccosine of the value of D0020 and stores the result as radians in D0010.

Only the data type F (float) can be used.

Example 7.13 Arctangent

Script

```
[D0010] = ATAN([D0020]);
```

Operation Description
Calculates the arctangent of the value of D0020 and stores the result as radians in D0010.

Only the data type F (float) can be used.

Example 7.14 Convert angle to radians

Script

```
[D0010] = RAD([D0020]);
```

Operation Description
Converts the value of D0020 from degrees (°) to radians and stores the result in D0010.

Only the data type F (float) can be used.

Example 7.15 Convert radians to angle

Script

```
[D0010] = DEG([D0020]);
```

Operation Description
Converts the value of D0020 from radians to degrees (°) and stores the result in D0010.

Only the data type F (float) can be used.

Data type conversions

Example 7.16 Convert BCD to binary

Script

```
[D0200] = BCD2BIN([D0100]);
```

Operation Description
Converts the BCD value in D0100 to a binary value and stores it in D0200.
For example, if the BCD value 10 (16 as a binary value) is stored in D0100, 10 (binary value) is stored in D0200.

Example 7.17 Convert binary to BCD

Script

```
[D0200] = BIN2BCD([D0100]);
```

Operation Description
Converts the binary value in D0100 to a BCD value and stores it in D0200.
For example, if the binary value 16 (10 as a BCD value) is stored in D0100, 16 (BCD value) is stored in D0200.

Example 7.18 Convert float to binary

Script

```
[D0200] = FLOAT2BIN([D0100]);
```

Operation Description
Converts the float value in D0100 to a binary value and stores it in D0200.
For example, if the data type F (float) 1234.0 (0x449A4000 as a binary value) is stored in D0100, 1234 (binary value) is stored in D0200. If the data type F (float) 1234.56 (0x449A51EC as a binary value) is stored in D0100, the value after the decimal point is truncated and 1234 (binary value) is stored in D0200.

Example 7.19 Convert binary to float

Script

```
[D0200] = BIN2FLOAT([D0100]);
```

Operation Description
Converts the binary value in D0100 to a float value and stores it in D0200.
For example, if the binary value 1234 is stored in D0100, the float value 1234.0 (0x449A4000 as a binary value) is stored in D0200.

Example 7.20 Convert decimal to string

Script

```
DEC2ASCII([D0100], [D0200]);
```

Operation Description
Converts the decimal numeric value in D0200 to a string and stores it in order with D0100 as the starting address.

Notes:

- This function can be used with data types W (word), I (integer), D (double word), and L (long).
- The NULL terminating character (0x00) is added to the end of the string.

Converting 1234 (when the data type is W (word))

Device	Stored Value
D0200	1234

→

Device	Stored Value	
	Upper Byte	Lower Byte
D0100	'1' = 0x31	'2' = 0x32
D0101	'3' = 0x33	'4' = 0x34
D0102	0x00	0x00

Terminating character

Converting -12345 (when the data type is I (integer))

Device	Stored Value
D0200	-12345

⇒

Device	Stored Value	
	Upper Byte	Lower Byte
D0100	'-' = 0x2D	'1' = 0x31
D0101	'2' = 0x32	'3' = 0x33
D0102	'4' = 0x34	'5' = 0x35
D0103	0x00	0x00

Terminating character

Converting 1234567890 (when the data type is D (double word))

Device	Stored Value
D0200	1234567890
D0201	

⇒

Device	Stored Value	
	Upper Byte	Lower Byte
D0100	'1' = 0x31	'2' = 0x32
D0101	'3' = 0x33	'4' = 0x34
D0102	'5' = 0x35	'6' = 0x36
D0103	'7' = 0x37	'8' = 0x38
D0104	'9' = 0x39	'0' = 0x30
D0105	0x00	0x00

Terminating character

Converting -1234567890 (when the data type is L (long))

Device	Stored Value
D0200	-1234567890
D0201	

⇒

Device	Stored Value	
	Upper Byte	Lower Byte
D0100	'-' = 0x2D	'1' = 0x31
D0101	'2' = 0x32	'3' = 0x33
D0102	'4' = 0x34	'5' = 0x35
D0103	'6' = 0x36	'7' = 0x37
D0104	'8' = 0x38	'9' = 0x39
D0105	'0' = 0x30	0x00

Terminating character

Example 7.21 Convert string to decimal

Script

```
[D0100] = ASCII2DEC([D0200]);
```

Operation Description

Converts the stored string starting at D0200 to a decimal and stores the result in D0100.

The number of digits that can be converted is the maximum number of digits for each data type with added sign.

If the string to convert contains a NULL or characters that cannot be converted to numeric values, the string is converted up to that character.

Notes:

- This function can be used with data types W (word), I (integer), D (double word), and L (long).
- The NULL terminating character (0x00) is added to the end of the string.

Setting the string "1234" (when the data type is W (word))

Device	Stored Value	
	Upper Byte	Lower Byte
D0200	'1' = 0x31	'2' = 0x32
D0201	'3' = 0x33	'4' = 0x34
D0202	0x00	0x00

Terminating character

Device	Stored Value
D0100	1234

Setting the string "1234567" (when the data type is D (double word))

Device	Stored Value	
	Upper Byte	Lower Byte
D0200	'1' = 0x31	'2' = 0x32
D0201	'3' = 0x33	'4' = 0x34
D0202	'5' = 0x35	'6' = 0x36
D0203	'7' = 0x37	0x00

Terminating character

Device	Stored Value
D0100	12345

Setting the string "-12345" (when the data type is I (integer))

Device	Stored Value	
	Upper Byte	Lower Byte
D0200	'-' = 0x2D	'1' = 0x31
D0201	'2' = 0x32	'3' = 0x33
D0202	'4' = 0x34	'5' = 0x35
D0203	0x00	0x00

Terminating character

Device	Stored Value
D0100	-12345

String "1234567890" (when the data type is L (long))

Device	Stored Value	
	Upper Byte	Lower Byte
D0200	'1' = 0x31	'2' = 0x32
D0201	'3' = 0x33	'4' = 0x34
D0202	'5' = 0x35	'6' = 0x36
D0203	'7' = 0x37	'8' = 0x38
D0204	'9' = 0x39	'0' = 0x30
D0205	0x00	0x00

Terminating character

Device	Stored Value
D0100	1234567890

Data comparison and copying

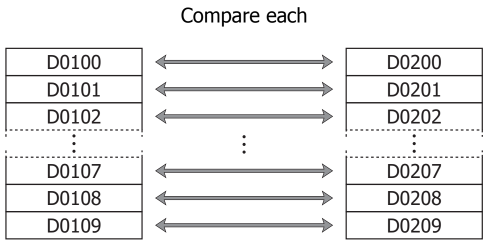
Example 7.22 Word-unit data comparison

Script

```
[D0000] = MEMCMP([D0100], [D0200], 10);
```

Operation Description
Compares the values of 10 words from D0100 (up to D0109) with the values of 10 words from D0200 (up to D0209).

If the value for each is entirely equal, 1 is stored in D0000. If even a single one is not equal, 0 is stored.



Note: Even if the data type is set to D (double word), L (long), or F (float), the comparison is performed from the first device in word units.

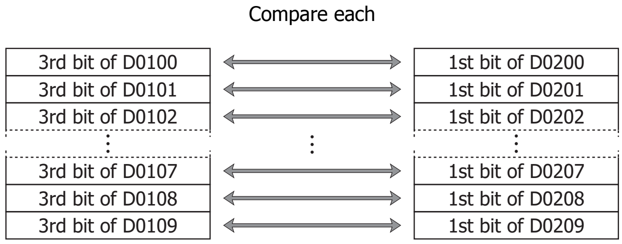
Example 7.23 Bit-unit data comparison

Script

```
[D0000] = MEMCMP([D0100.02], [D0200.00], 10);
```

Operation Description
Compares the third bit of D0100 through the third bit of D0109 with the state of the bits from the first bit of D0200 through the first bit of D0209.

If the value for each is entirely equal, 1 is stored in D0000. If even a single one is not equal, 0 is stored.



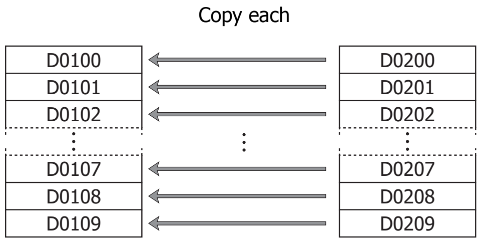
Note: Even if the data type is set to D (double word), L (long), or F (float), the comparison is performed from the first device in bit units.

Example 7.24 Word-unit data copy

Script

```
MEMCPY([D0100], [D0200], 10);
```

Operation Description
Copies the value of 10 word devices from D0200 (up to D0209) to 10 word devices from D0100 (up to D0109).



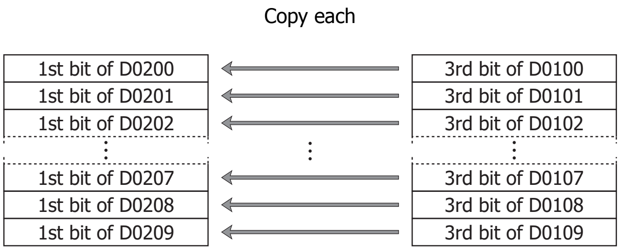
Note: Even if the data type is set to D (double word), L (long), or F (float), the data is copied from the first device in word units.

Example 7.25 Bit-unit data copy

Script

```
MEMCPY([D0200.00], [D0100.02], 10);
```

Operation Description
Copies the third bit of 10 words from D0100 (up to D0109) to the bit state for 10 bits of devices from D0200 (up to D0209).



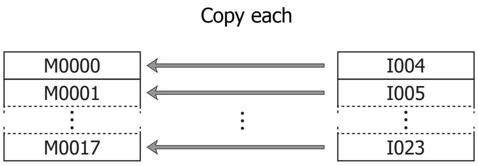
Note: Even if the data type is set to D (double word), L (long), or F (float), the bits are copied from the first device in bit units.

Example 7.26 Copy 1 word from bit devices to bit devices

Script

```
BITS2BITS([M0000], [I004]);
```

Operation Description
A value of 1 word length from I004 (I004 to I023) is copied to the bit states of a 1 word region from M0000 (M0000 to M0017).



Example 7.27 Copy 1 word from bit devices to a word device

Script

```
BITS2WORD([D0000],[I004]);
```

Operation Description
A value of 1 word length from I004 (to I023) is copied to the value of device D0000.
This is the same as BITS2BITS (I004, D0000.0);.

Example 7.28 Copy 1 word from a word device to bit devices

Script

```
WORD2BITS([M0000], [D0100]);
```

Operation Description
The value of device D0100 is copied to the bit states of a value of 1 word length from M0000 (M0000 to M0017).
This is the same as BITS2BITS (D1000.0, M0000);.

String Operations

Strings are set as continuous data registers of 1 word or more. 2 bytes of data are 1 word and they are set from upper byte to lower byte in order. Set 00h at the end of the string.

Example 7.29 Copy a string

Script

```
STRCUT([D0100], [D0200], 2, 3);
```

Operation Description
Stores in order from D0100 character count 3 (3 characters worth) from start position 2 (starting from 0, so the 3rd character) of the string "ABCDEFG" that starts from D0200.

Note: The start position can be specified in the range from 0 to 127, the character count can be specified in the range from 1 to 128.

Copy from string "ABCDEFG" at start position 2 for a character count of 3

Device		Stored Value	Start position	Character count
D0200	Upper Byte	'A' = 0x41	0	
	Lower Byte	'B' = 0x42	1	
D0201	Upper Byte	**'C' = 0x43**	← 2	3 characters
	Lower Byte	**'D' = 0x44**	3	
D0202	Upper Byte	**'E' = 0x45**	4	
	Lower Byte	'F' = 0x46	5	
D0203	Upper Byte	'G' = 0x47	:	
	Lower Byte	0x00	Terminating character	

⇨

Device		Stored Value	
D0100	Upper Byte	**'C' = 0x43**	
	Lower Byte	**'D' = 0x44**	
D0101	Upper Byte	**'E' = 0x45**	
	Lower Byte	0x00	Terminating character

Script

```
STRCUT([D0100], [D0200], 1, 4);
```

Copy from string "ABCDEFG" at start position 1 for a character count of 4

Device		Stored Value	Start position	Character count
D0200	Upper Byte	'A' = 0x41	0	
	Lower Byte	**'B' = 0x42**	← **1**	4 characters
D0201	Upper Byte	**'C' = 0x43**	← **2**	
	Lower Byte	**'D' = 0x44**	3	
D0202	Upper Byte	**'E' = 0x45**	4	
	Lower Byte	'F' = 0x46	5	
D0203	Upper Byte	'G' = 0x47	:	
	Lower Byte	0x00	Terminating character	

⇨

Device		Stored Value	
D0100	Upper Byte	**'B' = 0x42**	
	Lower Byte	**'C' = 0x43**	
D0101	Upper Byte	**'D' = 0x44**	
	Lower Byte	**'E' = 0x45**	
D0101	Upper Byte	0x00	Terminating character
	Lower Byte	0x00	

Example 7.30 Count a string

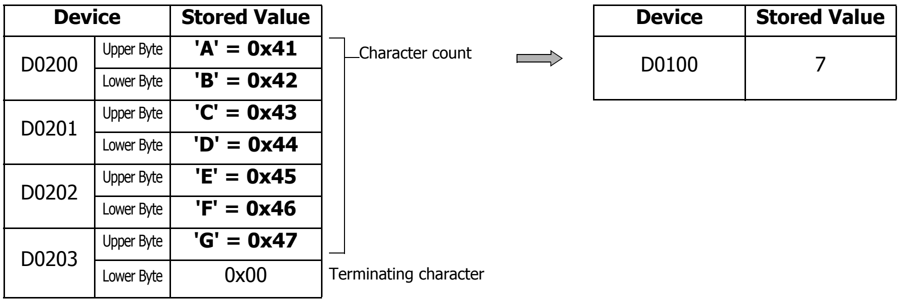
Script

```
[D0100] = STRLEN([D0200]);
```

Operation Description

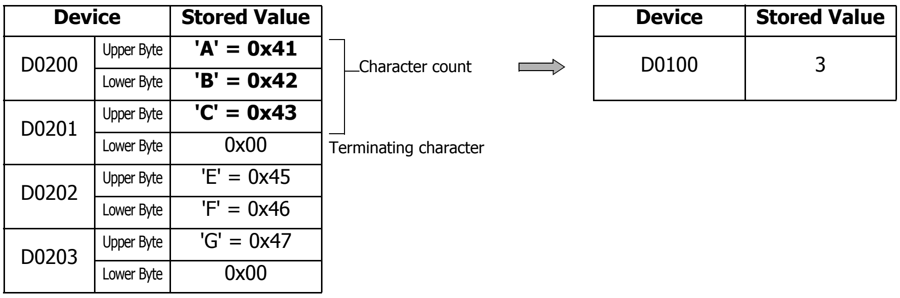
Finds the length (character count) of the string starting from D0200 and stores the result in D0100.

Note: The NULL terminating character (0x00) is the end of the string. (The terminating character is not included in the string length.)



Device		Stored Value
D0200	Upper Byte	'A' = 0x41
	Lower Byte	'B' = 0x42
D0201	Upper Byte	'C' = 0x43
	Lower Byte	'D' = 0x44
D0202	Upper Byte	'E' = 0x45
	Lower Byte	'F' = 0x46
D0203	Upper Byte	'G' = 0x47
	Lower Byte	0x00

Device	Stored Value
D0100	7



Device		Stored Value
D0200	Upper Byte	'A' = 0x41
	Lower Byte	'B' = 0x42
D0201	Upper Byte	'C' = 0x43
	Lower Byte	0x00
D0202	Upper Byte	'E' = 0x45
	Lower Byte	'F' = 0x46
D0203	Upper Byte	'G' = 0x47
	Lower Byte	0x00

Device	Stored Value
D0100	3

Example 7.31 Concatenate strings

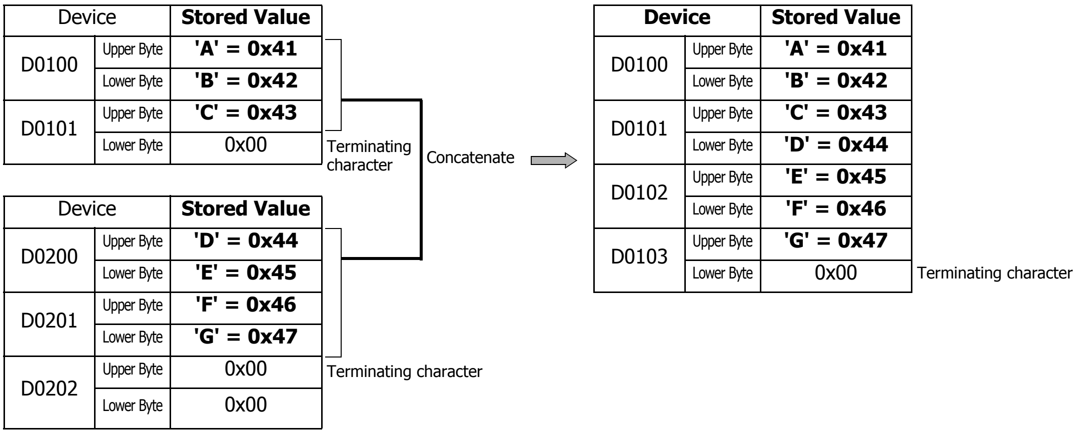
Script

```
STRCAT([D0100], [D0200]);
```

Operation Description

Appends the string starting from D0200 to the string starting from D0100.

Note: The NULL terminating character (0x00) is the end of the string. (The terminating character is not included in the string length.)



Device		Stored Value
D0100	Upper Byte	'A' = 0x41
	Lower Byte	'B' = 0x42
D0101	Upper Byte	'C' = 0x43
	Lower Byte	0x00

Device		Stored Value
D0200	Upper Byte	'D' = 0x44
	Lower Byte	'E' = 0x45
D0201	Upper Byte	'F' = 0x46
	Lower Byte	'G' = 0x47
D0202	Upper Byte	0x00
	Lower Byte	0x00

Device		Stored Value
D0100	Upper Byte	'A' = 0x41
	Lower Byte	'B' = 0x42
D0101	Upper Byte	'C' = 0x43
	Lower Byte	'D' = 0x44
D0102	Upper Byte	'E' = 0x45
	Lower Byte	'F' = 0x46
D0103	Upper Byte	'G' = 0x47
	Lower Byte	0x00

Example 7.32 Search a string

Script

```
[D0000] = STRSTR([D0100], [D0200]);
```

Operation Description
Searches for the search string "DEFG" that starts from D0200 in the string to be searched "ABCDEFGHIJKLMNO" that starts from D0100 and stores the position of the occurrence of the string in D0000. If not found, -1 is stored in D0000.

If "?" is specified as a character to search for, it is handled as any single-byte character.
When specifying "?" (0x3F) as a character, specify it as "~?" (0x7E3F) in two bytes.
When specifying "~" (0x7E) as a character, specify it as "~~" (0x7E7E) in two bytes.

Note: The maximum length of the search string is 128 characters.

When searching for "DEFG" and the string was found

Search string

Device		Stored Value	
D0200	Upper Byte	**'D' = 0x44**	
	Lower Byte	**'E' = 0x45**	
D0201	Upper Byte	**'F' = 0x46**	
	Lower Byte	**'G' = 0x47**	
D0202	Upper Byte	0x00	Terminating character
	Lower Byte	0x00	

String to be searched

Device		Stored Value	Position
D0100	Upper Byte	'A' = 0x41	0
	Lower Byte	'B' = 0x42	1
D0101	Upper Byte	'C' = 0x43	2
	Lower Byte	**'D' = 0x44**	← **3**
D0102	Upper Byte	**'E' = 0x45**	4
	Lower Byte	**'F' = 0x46**	5
D0103	Upper Byte	**'G' = 0x47**	6
	Lower Byte	'H' = 0x48	7
D0104	Upper Byte	'I' = 0x49	8
	Lower Byte	'J' = 0x4A	9
D0105	Upper Byte	'K' = 0x47	10
	Lower Byte	'L' = 0x4C	11
D0106	Upper Byte	'M' = 0x4D	12
	Lower Byte	'N' = 0x4E	13
D0107	Upper Byte	'O' = 0x4F	14
	Lower Byte	0x00	Terminating character

Search result

Device	Stored Value
D0000	3

When searching for "WXYZ" and the string was not found

Search string

Device		Stored Value	
D0200	Upper Byte	**'W' = 0x57**	
	Lower Byte	**'X' = 0x58**	
D0201	Upper Byte	**'Y' = 0x59**	
	Lower Byte	**'Z' = 0x5A**	
D0202	Upper Byte	0x00	Terminating character
	Lower Byte	0x00	

String to be searched

Device		Stored Value	Position
D0100	Upper Byte	'A' = 0x41	0
	Lower Byte	'B' = 0x42	1
D0101	Upper Byte	'C' = 0x43	2
	Lower Byte	'D' = 0x44	3
D0102	Upper Byte	'E' = 0x45	4
	Lower Byte	'F' = 0x46	5
D0103	Upper Byte	'G' = 0x47	6
	Lower Byte	'H' = 0x48	7
D0104	Upper Byte	'I' = 0x49	8
	Lower Byte	'J' = 0x4A	9
D0105	Upper Byte	'K' = 0x47	10
	Lower Byte	'L' = 0x4C	11
D0106	Upper Byte	'M' = 0x4D	12
	Lower Byte	'N' = 0x4E	13
D0107	Upper Byte	'O' = 0x4F	14
	Lower Byte	0x00	Terminating character

Search result

Device	Stored Value
D0000	-1

When searching for "?" as any single-byte character

Search string

Device		Stored Value	
D0200	Upper Byte	**'E' = 0x45**	
	Lower Byte	**'?' = 0x3F**	
D0201	Upper Byte	**'G' = 0x47**	
	Lower Byte	**'H' = 0x48**	
D0202	Upper Byte	0x00	Terminating character
	Lower Byte	0x00	

String to be searched

Device		Stored Value	Position
D0100	Upper Byte	'A' = 0x41	0
	Lower Byte	'B' = 0x42	1
D0101	Upper Byte	'C' = 0x43	2
	Lower Byte	'D' = 0x44	3
D0102	Upper Byte	**'E' = 0x45**	← **4**
	Lower Byte	**'F' = 0x46**	5
D0103	Upper Byte	**'G' = 0x47**	6
	Lower Byte	**'H' = 0x48**	7
D0104	Upper Byte	'I' = 0x49	8
	Lower Byte	'J' = 0x4A	9
D0105	Upper Byte	'K' = 0x47	10
	Lower Byte	'L' = 0x4C	11
D0106	Upper Byte	'M' = 0x4D	12
	Lower Byte	'N' = 0x4E	13
D0107	Upper Byte	'O' = 0x4F	14
	Lower Byte	0x00	Terminating character

Search result

Device	Stored Value
D0000	4

When searching for "?" as a character

Search string

Device		Stored Value	
D0200	Upper Byte	**'E' = 0x45**	
	Lower Byte	**'~' = 0x7E**	
D0201	Upper Byte	**'?' = 0x3F**	
	Lower Byte	**'G' = 0x47**	
D0202	Upper Byte	**'H' = 0x48**	
	Lower Byte	0x00	Terminating character

String to be searched

Device		Stored Value	Position
D0100	Upper Byte	'A' = 0x41	0
	Lower Byte	'B' = 0x42	1
D0101	Upper Byte	'C' = 0x43	2
	Lower Byte	'D' = 0x44	3
D0102	Upper Byte	**'E' = 0x45**	← **4**
	Lower Byte	**'?' = 0x3F**	5
D0103	Upper Byte	**'G' = 0x47**	6
	Lower Byte	**'H' = 0x48**	7
D0104	Upper Byte	'I' = 0x49	8
	Lower Byte	'J' = 0x4A	9
D0105	Upper Byte	'K' = 0x47	10
	Lower Byte	'L' = 0x4C	11
D0106	Upper Byte	'M' = 0x4D	12
	Lower Byte	'N' = 0x4E	13
D0107	Upper Byte	'O' = 0x4F	14
	Lower Byte	0x00	Terminating character

Search result

Device	Stored Value
D0000	4

When searching for "~" as a character

Search string

Device		Stored Value	
D0200	Upper Byte	**'E' = 0x45**	
	Lower Byte	**'~' = 0x7E**	
D0201	Upper Byte	**'~' = 0x7E**	
	Lower Byte	**'G' = 0x47**	
D0202	Upper Byte	**'H' = 0x48**	
	Lower Byte	0x00	Terminating character

String to be searched

Device		Stored Value	Position
D0100	Upper Byte	'A' = 0x41	0
	Lower Byte	'B' = 0x42	1
D0101	Upper Byte	'C' = 0x43	2
	Lower Byte	'D' = 0x44	3
D0102	Upper Byte	**'E' = 0x45**	← **4**
	Lower Byte	**'~' = 0x7E**	5
D0103	Upper Byte	**'G' = 0x47**	6
	Lower Byte	**'H' = 0x48**	7
D0104	Upper Byte	'I' = 0x49	8
	Lower Byte	'J' = 0x4A	9
D0105	Upper Byte	'K' = 0x47	10
	Lower Byte	'L' = 0x4C	11
D0106	Upper Byte	'M' = 0x4D	12
	Lower Byte	'N' = 0x4E	13
D0107	Upper Byte	'O' = 0x4F	14
	Lower Byte	0x00	Terminating character

Search result

Device	Stored Value
D0000	4

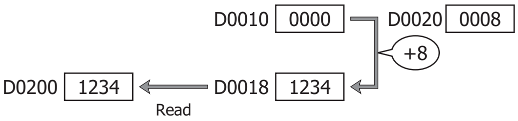
Indirect assignment

Example 7.33 Indirect read

Script

```
[D0200] = OFFSET([D0010],[D0020]);
```

Operation Description
When the value of D0020 is 8, the value of D0018, the device 8 words from D0010, is read and stored in D0200.

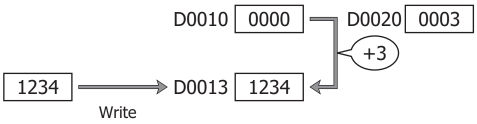


Example 7.34 Indirect write

Script

```
OFFSET([D0010],[D0020]) = 1234;
```

Operation Description
When the value of D0020 is 3, the constant 1234 is stored in D0013, the device 3 words from D0010.



8. Data Type Designations

Example 8.1 Processing that mixes F (float), D (double word), and W (word) data types using data type designations

Script

```
[D0100] = 1.5;
[D0110] = 10.5;
asd
{
     [D0200] = FLOAT2BIN([D0100]);
     [D0210] = FLOAT2BIN([D0110]);
     [D0300] = [D0200]^[D0210];
}
asword
{
     [D0400] = [D0301];
}
```

Operation Description
The value of D0100 (data type F (float)) is converted to data type D (double word) and stored in D0200 and D0201.
The value of D0110 (data type F (float)) is converted to data type D (double word) and stored in D0210 and D0211.
Logical XOR (exclusive OR) is performed on each bit of D0200 and D0210 (data type D (double word)) and the result is stored in D0300 and D0301.
The value of D0301, which is the low word of D0300 (data type D (double word)), is stored in D0400.
In the above example script, F (float) is selected as the data type in Script Editor, and **From Upper Word** is selected in **Device Settings**.
The method to store 32-bit data in devices conforms to the data storage method specified in the function area settings.
For details, see "32-bit Data Storage" on page 3-9.

Important Notes

This section describes important notes when programming scripts.

Important Notes Regarding While Definition

■Define so the execution does not go into an infinite loop.

The execution expression is repeatedly executed while the conditional expression is satisfied.
However, it will go into an infinite loop when the conditional expression is satisfied continually.

```
[D0100] = 10;

while (0 != [D0100])
{
    [D0200] = [D0200] + 1;
}
```

In the conditional expression of the while definition, it states to exit the loop when the value of D0100 turns 0. However, the value stored in D0100 is not changed after string 10 in the first line of the script, so the execution goes into an infinite loop.
When using the while definition, define it so the execution will not go into an infinite loop.
If the execution becomes an infinite loop, the scan time will exceed the watchdog timer setting and a watchdog timer error will occur.

The value of D0100 becomes 0 when the while definition is repeated 10 times in the following example, and the execution will get out of the while definition.

```
[D0100] = 10;

while (0 != [D0100])
{
    [D0200] = [D0200] + 1;
    [D0100] = [D0100] - 1;
}
```

Precautions on Data Type Designations

■Do not write scripts that use a data type designation inside another data type designation.

Data type designations are called at a maximum depth of one level.
An error will occur when adding a script with Script Editor if a data type designation is used inside another data type designation as shown in the following example.

```
[D0100] = 1.5;
[D0110] = 10.5;
asd
{
    [D0200] = FLOAT2BIN([D0100]);
    [D0210] = FLOAT2BIN([D0110]);
    [D0300] = [D0200]^[D0210];

    asword
    {
        [D0400] = [D0301];
    }
}
```

■Do not write scripts that use a temporary device inside a data type designation.

The processing inside the curly brackets "{ }" for the data type designation cannot use temporary devices.
A temporary device works only with the data type set in **Data Type** in Script Editor.
An error will occur when adding a script with Script Editor if a temporary device is used inside a data type designation as shown in the following example.

```
@1 = 1.5;
asd
{
    [D0200] = FLOAT2BIN(@1);
}
```

About the Priority of the Operator

Operators are processed in the order from the left in each line. When multiple operations are combined, the operators are processed in the following priorities.

Priority	Operator
High	()
	! ~ - (Negative number)
	* / %
	+ - (Subtraction)
	<< >>
	&
	^
	|
	< <= > >=
	== !=
	&&
	||
Low	=

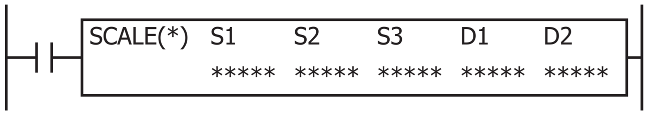
27: Flow Calculation Instructions

This chapter describes the flow calculation instructions that output flow volume and the accumulated flow volume.

SCALE (Convert Analog Input)

This instruction scales the analog input value according to the coordinates between two specified points and outputs that result.

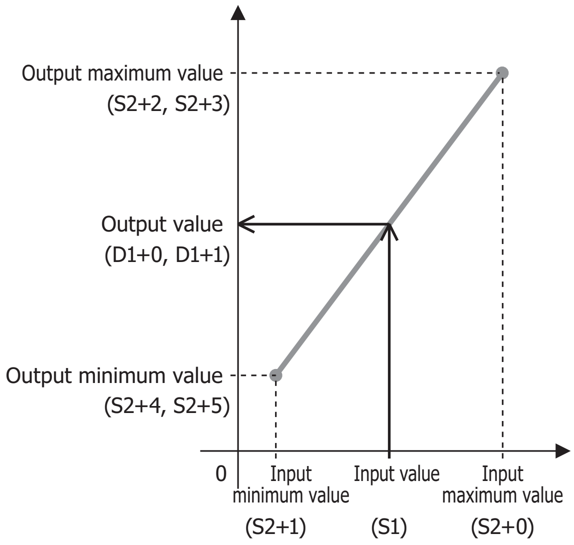
Symbol



Operation

When the input is on, the value of the data register specified by S1 is scaled according to the settings in the data register specified by S2, then the calculated result is stored in the output value (D1+0, D1+1).

As shown in the diagram below, scaling is calculated by substituting the input value into a linear function that has been derived from four points that are set by the output maximum value, output minimum value, input maximum value, input minimum value, which are configured by S2.



Notes:

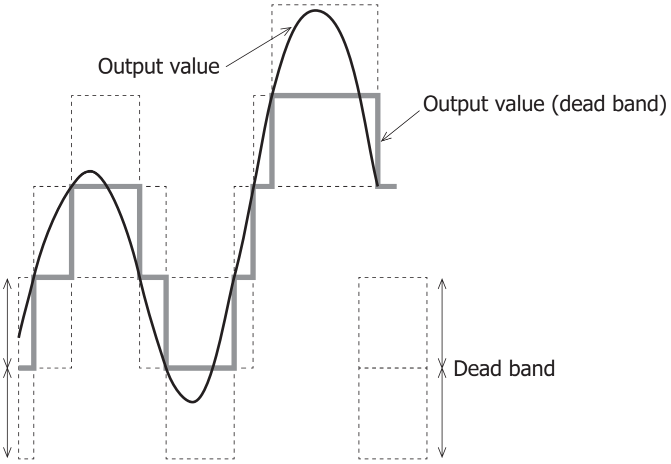
- The SCALE instruction cannot be used in an interrupt program. If used in an interrupt program, a user program execution error occurs, the execution of the instruction is canceled, and the next instruction is executed.
  For user program execution errors, see "User Program Execution Errors" on page 3-10.
- If the value of S1 or S2 is out of range, the result is a user program execution error. Error code 28 is stored in D8006, and the output value (D1+0, D1+1) and the output value (dead band) (D1+2, D1+3) are not updated.
  For user program execution errors, see "User Program Execution Errors" on page 3-10.
- In the following cases, a user program execution error will occur. Error code 46 is stored in D8006, and the output value (D1+0, D1+1) and the output value (dead band) (D1+2, D1+3) are not updated. For user program execution errors, see "User Program Execution Errors" on page 3-10.
  - When the input minimum value (S2+1) is the input maximum value (S2+0) or higher (input minimum value (S2+1) ≥ input maximum value (S2+0))
  - When the output minimum value (S2+4, S2+5) is the output maximum value (S2+2, S2+3) or higher (output minimum value (S2+4, S2+5) ≥ output maximum value (S2+2, S2+3))
  - When the output maximum value, the output minimum value, or the dead band (S2+6, S2+7) is INF (infinity) or NAN (not a number)
  - When the dead band (S2+6, S2+7) is negative

Dead Band Function

The dead band function filters minute changes in the value input to the SCALE instruction.
The SCALE instruction always retains the following two values.

Output value:	The result of scaling the input value with each scan
Output value (dead band):	The result of filtering the input value with the dead band

The fluctuation range to perform filtering is set as the dead band.
If the fluctuation range of the output value is smaller than the dead band, the output value (dead band) is not updated.



When the SCALE instruction is executed, it stores the value scaled in the first scan in output value (D1+0, D1+1) and updates the output value (dead band) (D1+2, D1+3).
The operation at this time differs according to whether or not the dead band is enabled.

When the dead band is enabled

The output value (D1+0, D1+1) and the output value (dead band) (D1+2, D1+3) are compared, and the value is stored depending on which of the absolute value of that difference (D1+4, D1+5) and the size of the dead band (S2+6, S2+7) are larger.

- If the absolute value of the difference (D1+4, D1+5) between the output value (D1+0, D1+1) and the output value (dead band) (D1+2, D1+3) is less than the dead band (S2+6, S2+7)
  The output value (D1+0, D1+1) is not stored in the output value (dead band) (D1+2, D1+3).
- If the absolute value of the difference (D1+4, D1+5) between the output value (D1+0, D1+1) and the output value (dead band) (D1+2, D1+3) is greater than or equal to the dead band (S2+6, S2+7)
  The output value (D1+0, D1+1) is stored in the output value (dead band) (D1+2, D1+3) and output value (dead band) change notification (D2+0) turns on for one scan.

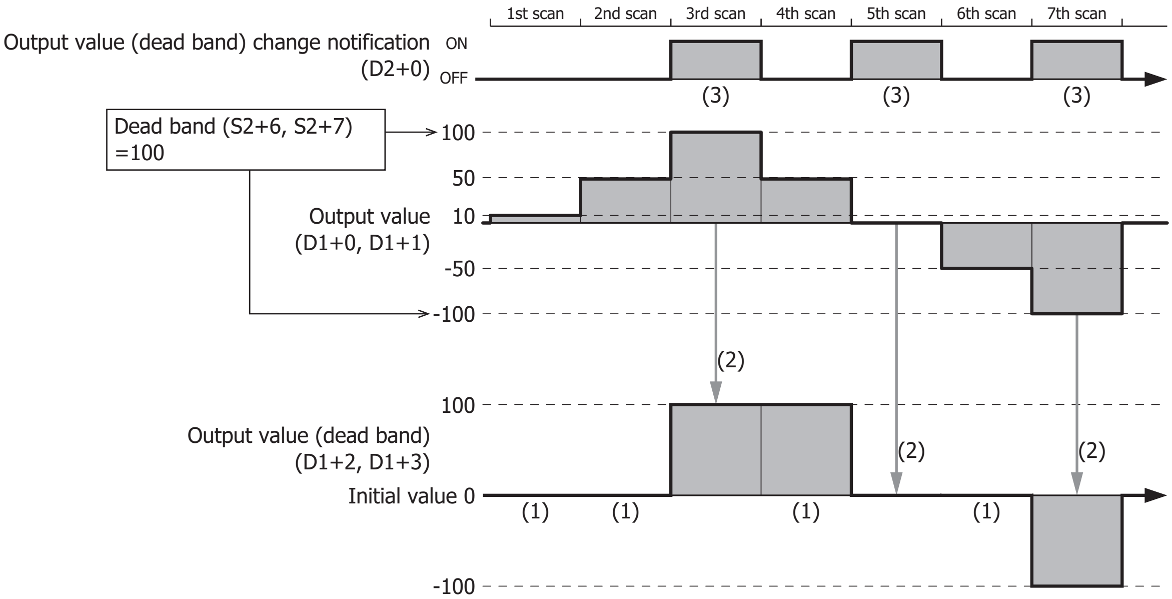
If the dead band is disabled (dead band (S2+6, S2+7)=0)

The output value (D1+0, D1+1) is stored in the output value (dead band) (D1+2, D1+3) with each scan.

Example 1: When the dead band is enabled

When the dead band (S2+6, S2+7) is 100 and initial value of the output value (dead band) (D1+2, D1+3) is 0, the operation in regard to the output value (D1+0, D1+1) is as follows.

Cycle	Output Value (D1+0, D1+1)	Output Value (dead band) (D1+2, D1+3)	Absolute Value of the Difference between the Output Value and the Output Value (dead band) (D1+4, D1+5)	Output Value (dead band) Change Notification (D2+0)
1st scan	10	0	10	OFF
2nd scan	50	0	50	OFF
3rd scan	100	100	100	ON
4th scan	50	100	50	OFF
5th scan	0	0	100	ON
6th scan	-50	0	50	OFF
7th scan	-100	-100	100	ON



(1) Since the absolute value of the difference (D1+4, D1+5) between the output value and the output value (dead band) is less than the dead band, the output value (D1+0, D1+1) is not stored in the output value (dead band) (D1+2, D1+3).

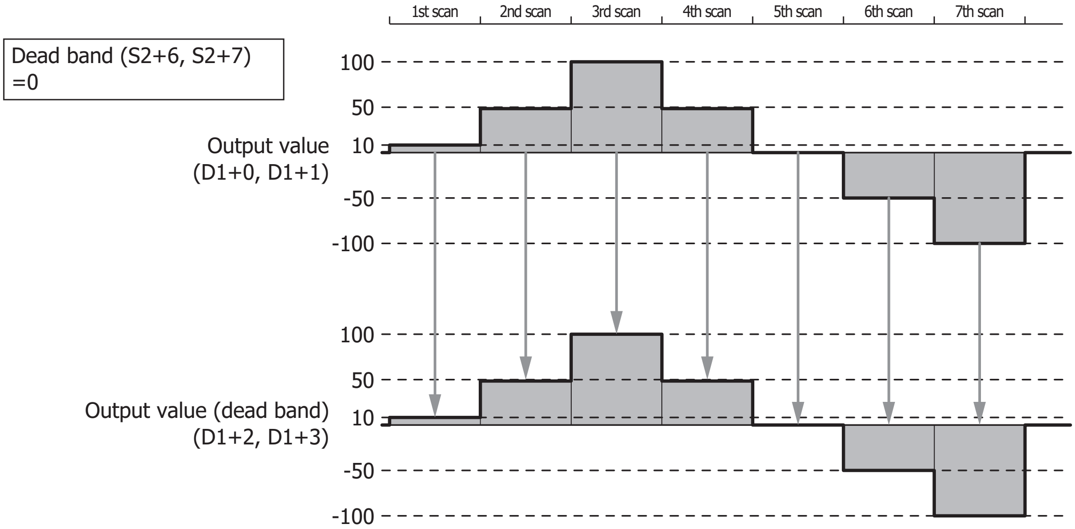
(2) Since the absolute value of the difference (D1+4, D1+5) between the output value and the output value (dead band) is greater than or equal to the dead band, the output value (D1+0, D1+1) is stored in the output value (dead band) (D1+2, D1+3).

(3) Output value (dead band) change notification (D2+0) turns on for one scan when (2) occurs.

Example 2: When the dead band is disabled

When the dead band (S2+6, S2+7) is 0 and initial value of the output value (dead band) (D1+2, D1+3) is 0, the output value (D1+0, D1+1) is stored in the output value (dead band) (D1+2, D1+3) with each scan.

Cycle	Output Value (D1+0, D1+1)	Output Value (dead band) (D1+2, D1+3)
1st scan	10	10
2nd scan	50	50
3rd scan	100	100
4th scan	50	50
5th scan	0	0
6th scan	-50	-50
7th scan	-100	-100



Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Input value	—	—	—	—	—	—	X	—	—	—
S2 (Source 2)	Control register	—	—	—	—	—	—	▲	—	—	—
S3 (Source 3)	Initialization input	X	—	X	—	—	—	—	—	—	—
D1 (Destination 1)	Output register	—	—	—	—	—	—	▲	—	—	—
D2 (Destination 2)	Output relay	—	X	▲	—	—	—	—	—	—	—

▲ Special data registers cannot be designated as S2 and D1. Special internal relays cannot be designated as D2.

Data Types

Data Type	W (word)	I (integer)	D (double word)	L (long)	F (float)
Specifiable	Yes	Yes	—	—	—

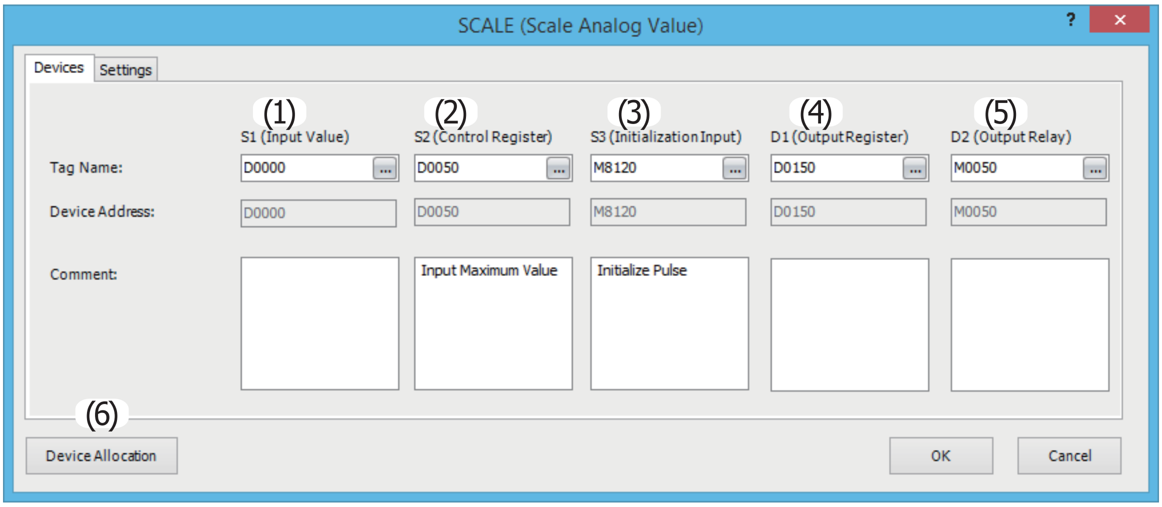
The data type setting is only reflected in the input value.

Settings

The **SCALE (Scale Analog Value)** dialog box contains the **Devices** tab and the **Settings** tab.

The **Devices** tab configures the devices used with the SCALE instruction. The **Settings** tab configures the initial values of the SCALE instruction parameters.

■ Devices tab



SCALE (Scale Analog Value)

	(1) S1 (Input Value)	(2) S2 (Control Register)	(3) S3 (Initialization Input)	(4) D1 (Output Register)	(5) D2 (Output Relay)
Tag Name:	D0000	D0050	M8120	D0150	M0050
Device Address:	D0000	D0050	M8120	D0150	M0050
Comment:		Input Maximum Value	Initialize Pulse		

(1) S1 (Source 1): Input Value

Specify the data register as the device that stores the input value to convert.

If the input value is greater than the input maximum value (S2+0) (input value > input maximum value (S2+0)), the input maximum value (S2+0) is scaled.

If the input value is less than the input minimum value (S2+1) (input minimum value (S2+1) > input value), the input minimum value (S2+1) is scaled.

(2) S2 (Source 2): Control Register

Specify the data register that stores the input maximum value, input minimum value, output maximum value, output minimum value, and the dead band.
8 continuous words are used starting from the specified data register.
For details on the control registers, see "S2: Control registers" on page 27-7.

(3) S3 (Source 3): Initialization Input

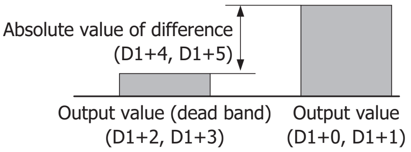
Specify the input or internal relay as the device to initialize the control registers.

When the initialization input is on, the initial values configured on the **Settings** tab in the **SCALE (Scale Analog Value)** dialog box are stored in the control registers.

When the initialization input is on, the initial values are stored in the data registers with each scan. To execute initialization only once, add the SOTU instruction (single output up) or the SOTD (single output down) instruction to the input conditions.
For how to configure the initial values, see "(2) Initial Value" on page 27-7.

(4) D1 (Destination 1): Output Register

Specify the data register that will store the output value, the output value (dead band), and the amount of output change. 6 continuous words are used starting from the specified data register.

Storage Destination	Function	Setting	Data Types
D1+0 D1+1	Output value	Stores the value after scaling with each scan.	F (float)
D1+2 D1+3	Output value (dead band)	If the dead band is disabled (dead band (S2+6, S2+7) is 0), the output value (D1+0, D1+1) is stored in the output value (dead band) with each scan. If the dead band is enabled (dead band (S2+6, S2+7) is not 0), output value (dead band) change notification (D2+0) turns on and the output value (D1+0, D1+1) is stored in the output value (dead band) only when the absolute value of the difference between the output value (D1+0, D1+1) and the output value (dead band) is greater than or equal to the dead band (S2+6, S2+7)) (difference ≥ dead band (S2+6, S2+7)). For details, see "Dead Band Function" on page 27-2.	
D1+4 D1+5	Amount of output change	Stores the absolute value of the difference between the output value (D1+0, D1+1) and the output value (dead band) (D1+2, D1+3).  Absolute value of difference (D1+4, D1+5) Output value (dead band) (D1+2, D1+3) Output value (D1+0, D1+1)	F (float)*1

*1 The data range is 0 and 1.175494E-38 to 3.402823E+38.

(5) D2 (Destination 2): Output Relay

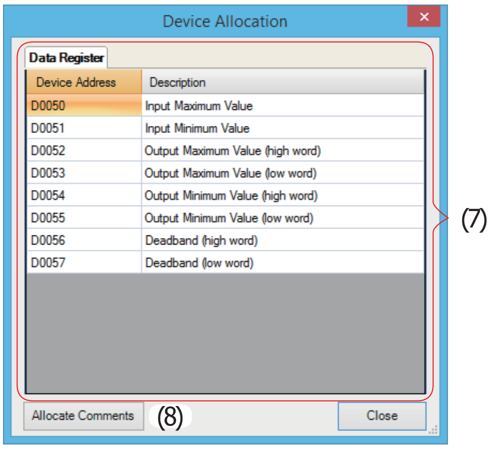
Specify the output or internal relay to store the output value (dead band) change notification and the input value alarm output. 2 continuous words are used starting from the specified device.

Storage Destination	Function	Setting
D2+0	Output value (dead band) change notification	This relay turns on for one scan if the absolute value of the difference between the output value (D1+0, D1+1) and the output value (dead band) (D1+2, D1+3) is greater than the dead band (S2+6, S2+7). At that time, the output value (D1+0, D1+1) is stored in the output value (dead band) (D1+2, D1+3).
D2+1	Input value warning output	This relay turns on when the input value (S1) is greater than the input maximum value (S2+0) or when the input value is less than the input minimum value (S2+1). At that time, if the input value (S1) is greater than the input maximum value (S2+0), it is scaled by the input maximum value (S2+0). If the input value (S1) is less than the input minimum value (S2+1), it is scaled by the input minimum value (S2+1).

(6) Device Allocation

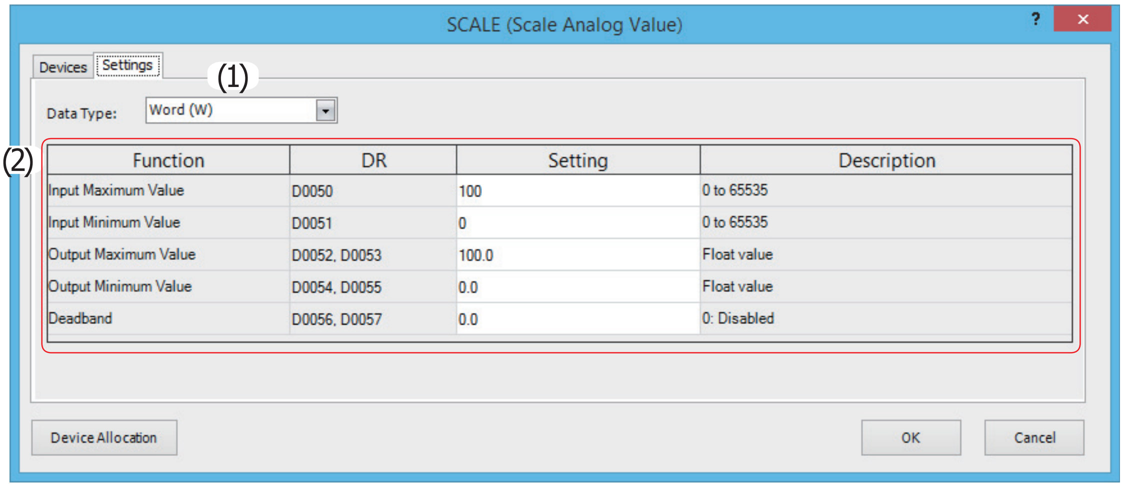
Click this button to display the **Device Allocation** dialog box. As shown next, a table of the data registers and internal relays and their corresponding SCALE instruction settings are displayed in the dialog box (7). Click **Allocate Comments** (8) and you can configure the comments for the data registers and internal relays that correspond to the content of the settings.

Device Allocation dialog box



Device Address	Description
D0050	Input Maximum Value
D0051	Input Minimum Value
D0052	Output Maximum Value (high word)
D0053	Output Maximum Value (low word)
D0054	Output Minimum Value (high word)
D0055	Output Minimum Value (low word)
D0056	Deadband (high word)
D0057	Deadband (low word)

■ Settings tab



Data Type: Word (W)

Function	DR	Setting	Description
Input Maximum Value	D0050	100	0 to 65535
Input Minimum Value	D0051	0	0 to 65535
Output Maximum Value	D0052, D0053	100.0	Float value
Output Minimum Value	D0054, D0055	0.0	Float value
Deadband	D0056, D0057	0.0	0: Disabled

(1) Data Type

Select the data type of the input value (S1) as "W (word)" or "I (integer)".

(2) Initial Value

Configure the initial values of the functions that will be stored in the control registers when the initialization input is on. For the range of initial values, see "S2: Control registers" on page 27-7.

S2: Control registers

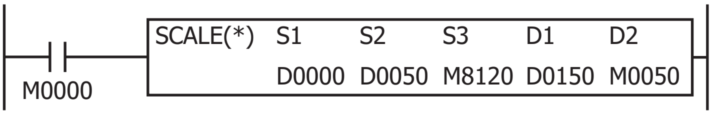
Storage Destination	Function	Setting	Data Types
S2+0	Input maximum value	Set as input maximum value > input minimum value (S2+0)[*1]. The initial value is 100.	W (word) I (integer)
S2+1	Input minimum value	Set as input minimum value < input maximum value (S2+1)[*1]. The initial value is 0.	
S2+2 S2+3	Output maximum value	Set as output maximum value > output minimum value (S2+4, S2+5)[*1]. The initial value is 100.0.	F (float)
S2+4 S2+5	Output minimum value	Set as output minimum value < output maximum value (S2+2, S2+3)[*1]. The initial value is 0.0.	
S2+6 S2+7	Dead band	The absolute value of the difference between the output value (D1+0, D1+1) and the output value (dead band (D1+2, D1+3) is sampled, and output value (dead band) change notification (D2+0) turns on only if the absolute value of the difference is greater than or equal to the dead band (difference ≥ dead band). At that time, the output value (dead band) (D1+2, D1+3) is stored in the output value (D1+0, D1+1). If the dead band is 0, the dead band function is disabled and the output value (D1+0, D1+1) is stored in the output value (dead band) (D1+2, D1+3) with each scan. The initial value is 0.	F (float)[*2]

[*1] If the conditions for the settings cannot be satisfied, a user program execution error will occur and the output value and output value (dead band) will not be updated.

[*2] The data range is 0 and 1.175494E-38 to 3.402823E+38.

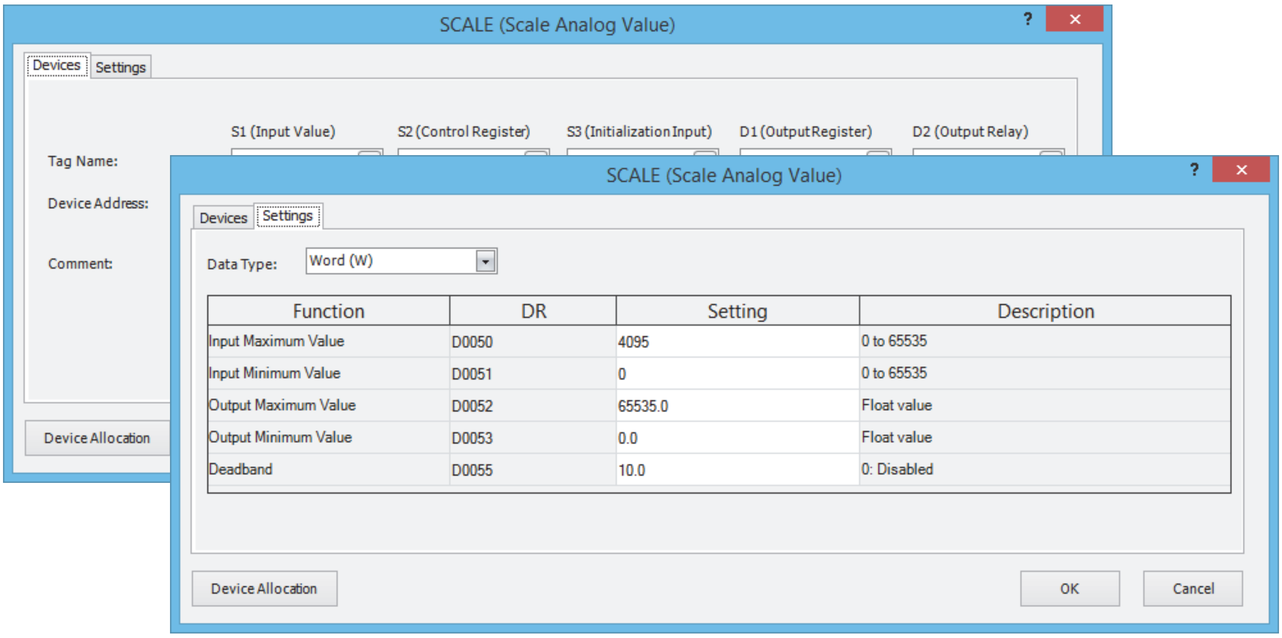
Operation Example

To convert an analog input value from 0 to 4,095 to 0 to 65,535
Create the following ladder program.



(1) Specify D0000, where the analog input value is stored, for S1.
(2) Initialize the settings with the initialize pulse, and when M0000 turns on, the calculation is performed with the SCALE instruction.

SCALE instruction settings



SCALE (Scale Analog Value)

Data Type: Word (W)

Function	DR	Setting	Description
Input Maximum Value	D0050	4095	0 to 65535
Input Minimum Value	D0051	0	0 to 65535
Output Maximum Value	D0052	65535.0	Float value
Output Minimum Value	D0053	0.0	Float value
Deadband	D0055	10.0	0: Disabled

Devices tab

Settings	Description
S1 (input value)	D0000
S2 (control register)	D0050
S3 (initialization input)	M8120
D1 (output register)	D0150
D2 (output relay)	M0050

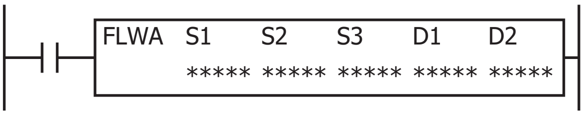
Settings tab

Settings	Description
Data type	Word (W)
Input maximum value	4,095
Input minimum value	0
Output maximum value	65,535
Output minimum value	0
Deadband	10

FLWA (Analog Flow Totalizer)

This instruction samples the instantaneous flow (volume per unit of time) and stores the accumulated flow volume (the volume of a material that has passed through for an arbitrary period) to a log.

Symbol



Operation

The flow rate is sampled at a fixed cycle and the accumulated flow volume is calculated. That result is stored in the device specified by D2 (logged data).

The FLWA instruction has a totalizer function that samples and accumulates the flow rate and a log output function that temporarily saves the accumulated volume at the desired time.

Note: When using an analog output type flow meter, the FLWA instruction can be used to monitor the flow volume by converting the analog value output from the flow meter to the flow rate with the SCALE instruction.
For details on the SCALE instruction, see "SCALE (Convert Analog Input)" on page 27-1.
When using a pulse output type flow meter, the FLWP instruction can be used to monitor the flow volume. For details on the FLWP instruction, see "FLWP (Pulse Flow Totalizer)" on page 27-18.

Notes:

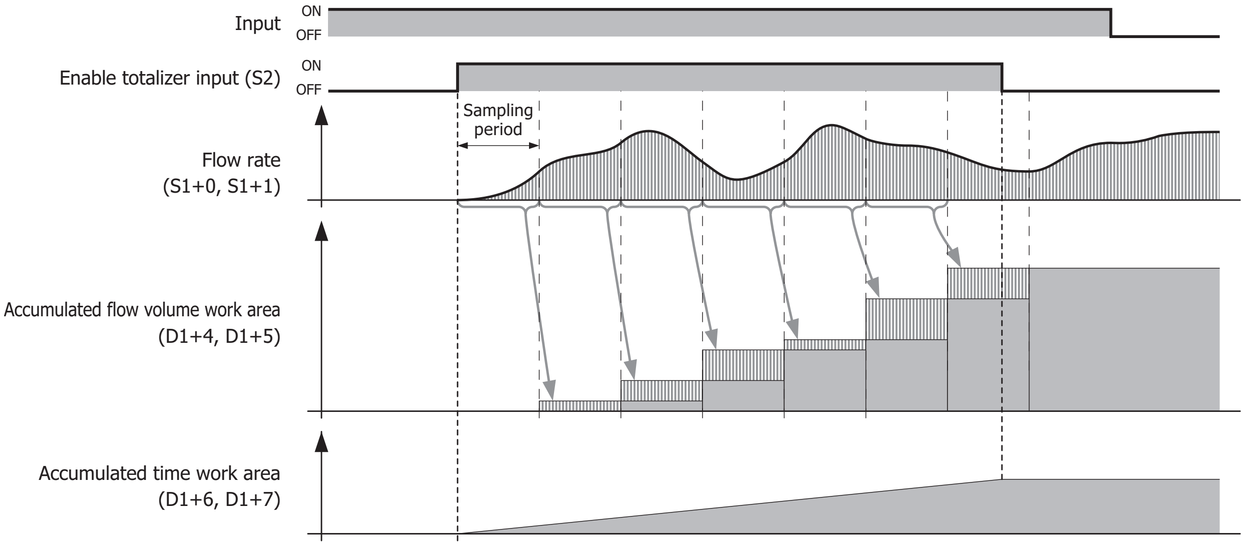
- The FLWA instruction cannot be used in an interrupt program. If used in an interrupt program, a user program execution error occurs, the execution of the instruction is canceled, and the next instruction is executed.
  For user program execution errors, see "User Program Execution Errors" on page 3-10.
- The FLWA instruction cannot be simultaneously used with the LABEL, LJMP, LCAL, LRET, JMP, JEND, MCS, and MCR instructions.

Totalizer Function Overview

When the input is on, if the enable totalizer input (S2) is turned on, the flow rate (S1+0, S1+1) starts being sampled.

When the enable totalizer input is on, totalization is performed (updating the accumulated flow volume work area and accumulated time work area).

When the enable totalizer input is off, totalization is paused. (The accumulated flow volume work area and the accumulated time work area values are retained.)

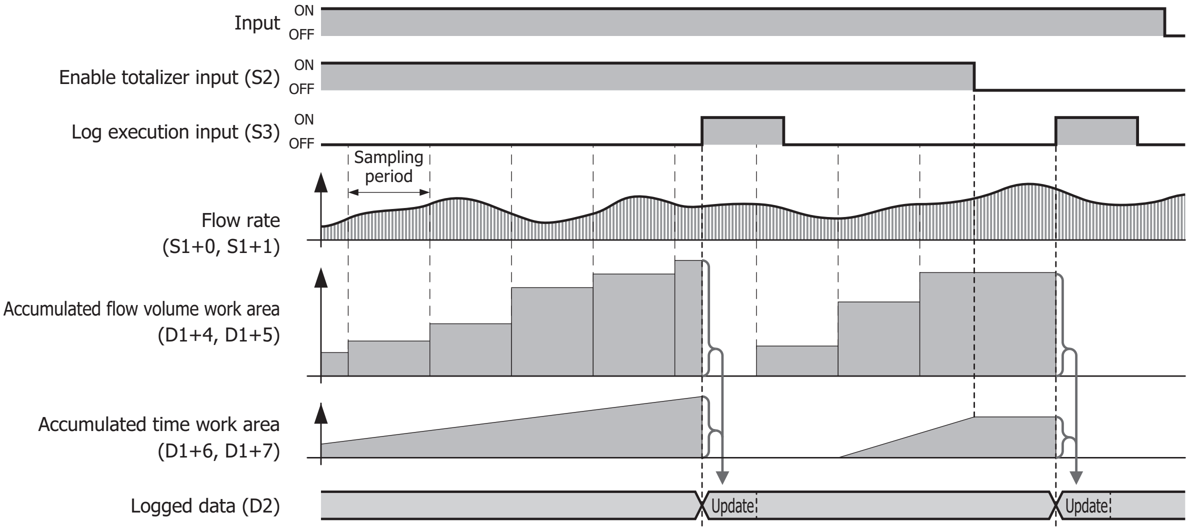

Input
ON
OFF
Enable totalizer input (S2)
ON
OFF
Sampling period
Flow rate
(S1+0, S1+1)
Accumulated flow volume work area
(D1+4, D1+5)
Accumulated time work area
(D1+6, D1+7)

Note: For details on the totalizer function such as the operation immediately after the input is turned on and the operation when the enable totalizer input (S2) is once again turned on, see "Totalizer Function" on page 27-11.

Log Output Function Overview

When the input is on, if the log execution input (S3) is turned on, the logged data (D2) is updated regardless of the enable totalizer input (S2) on/off status. The maximum amount of logged data is 35 items.

When the enable totalizer input is off, totalization is paused. (The accumulated flow volume work area and the accumulated time work area values are retained.)


Input
ON
OFF
Enable totalizer input (S2)
ON
OFF
Log execution input (S3)
ON
OFF
Sampling period
Flow rate (S1+0, S1+1)
Accumulated flow volume work area (D1+4, D1+5)
Accumulated time work area (D1+6, D1+7)
Logged data (D2)
Update
Update

Note: For details on the log output function such as the operation immediately after the input is turned on, the operation when the enable totalizer input (S2) is once again turned on, and the logged data updating when the log execution input (S3) is turned on, see "Log Output Function" on page 27-13.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Flow rate	—	—	—	—	—	—	X	—	—	—
S2 (Source 2)	Enable totalizer input	X	—	X	—	—	—	—	—	—	—
S3 (Source 3)	Log execution input	X	—	X	—	—	—	—	—	—	—
D1 (Destination 1)	Status	—	—	—	—	—	—	▲	—	—	—
D2 (Destination 2)	Logged data	—	—	—	—	—	—	▲	—	—	—

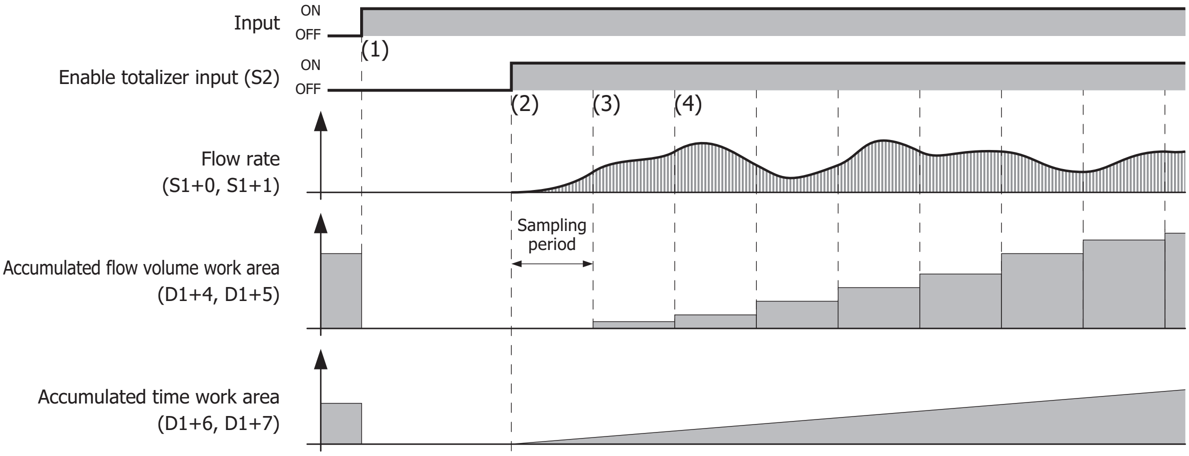
▲ Special data registers cannot be used.

Function Descriptions

■ Totalizer Function

Starting Totalization

This section describes the operation from turning on the input to starting totalization.



(1) If the input is turned on, the operation status (D1+0), the amount of logged data (D2+0), the accumulated flow volume work area, the accumulated time work area, and the logged data (D2+2 to D2+2+(6×maximum amount of logged data)) are initialized to "0".

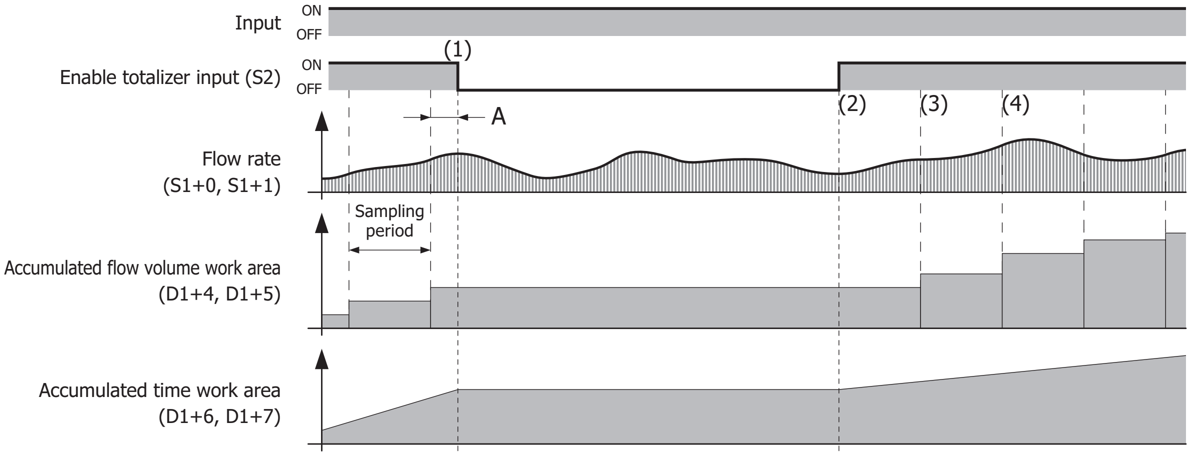
(2) If the enable totalizer input (S2) is turned on, sampling starts. The accumulated time work area (D1+6, D1+7) count starts at the same time.

(3) The amount to add calculated from the flow rate (S1+0, S1+1) when sampled and the elapsed time is added to the accumulated flow volume work area (D1+4, D1+5).

(4) Afterward the operation in (3) is repeated.

Pausing and Restarting Totalization

This section describes the operation to pause and restart totalization by turning the enable totalizer input (S2) on and off.



(1) While the totalization is being executed, if the enable totalizer input is turned off, sampling is paused and updating the accumulated flow volume work area (D1+4, D1+5) and the accumulated time work area (D1+6, D1+7) is stopped. The accumulated flow volume work area and the accumulated time work area values are retained.
During the period from the previous sampling to (1), the flow rate is not added to the accumulated flow volume work area, even if it is larger than "0".

(2) If the enable totalizer input once again changes from off to on, totalization restarts.
The processing is the same as (2) in "Starting Totalization" on page 27-11.

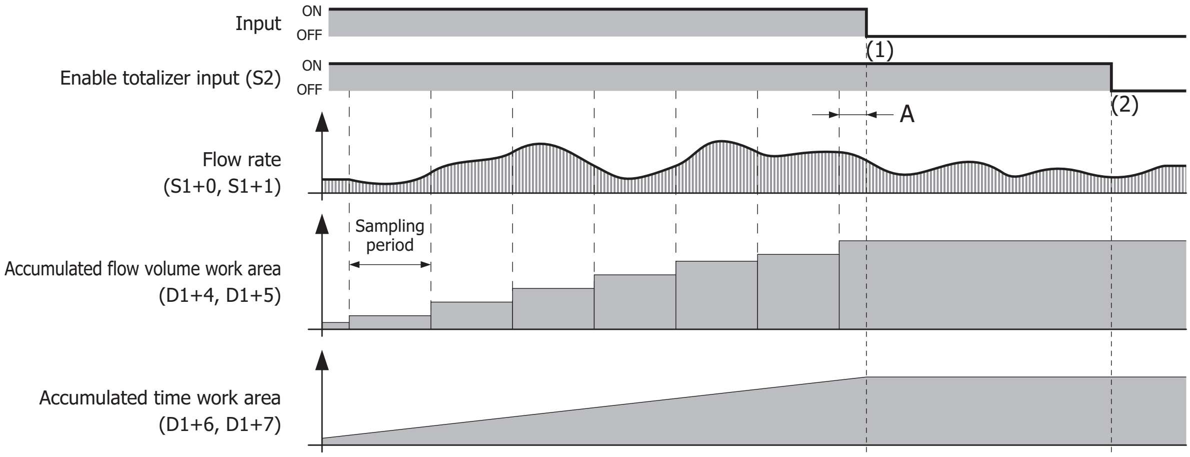
(3) The accumulated flow volume and the accumulated time for the sampling period are calculated.
The processing is the same as (3) in "Starting Totalization" on page 27-11.

(4) Afterward the operation in (3) is repeated.

Ending Totalization

To end totalization by turning off the input

This section describes the end operation for totalization when ending totalization by turning off the input.

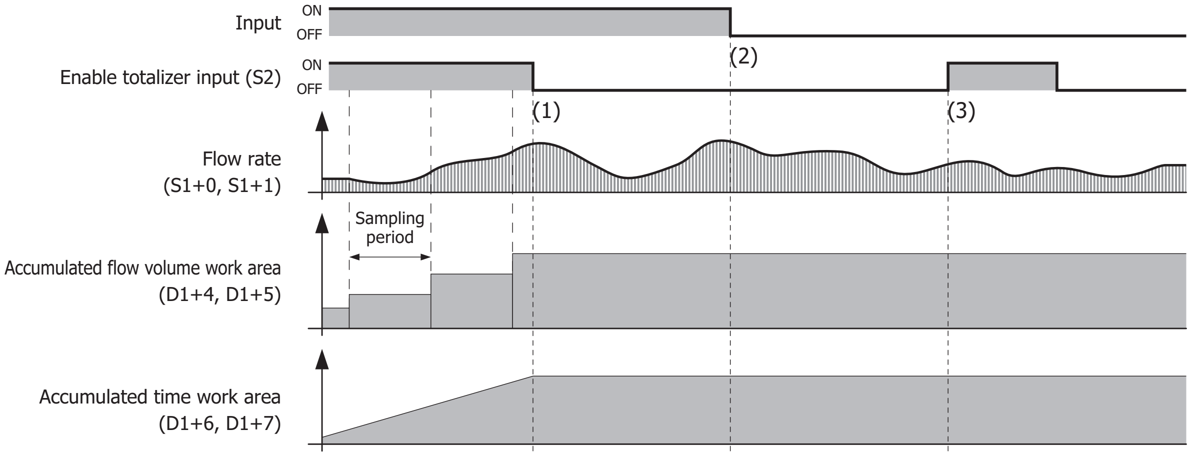

Input
ON
OFF
(1)
Enable totalizer input (S2)
ON
OFF
(2)
A
Flow rate
(S1+0, S1+1)
Sampling period
Accumulated flow volume work area
(D1+4, D1+5)
Accumulated time work area
(D1+6, D1+7)

(1) When the input is off, updating the accumulated flow volume work area and the accumulated time work area is stopped.
The accumulated flow volume work area and the accumulated time work area values are retained.
During the period from the previous sampling to (1) (=A), the flow rate is not added to the accumulated flow volume work area, even if it is larger than "0".

(2) When the input is off, the accumulated flow volume work area and the accumulated time work area values are retained, even if the enable totalizer input is turned off.

To end totalization by turning off the enable totalizer input (S2)

This section describes the end operation for totalization when ending totalization by turning off the enable totalizer input (S2).


Input
ON
OFF
(2)
Enable totalizer input (S2)
ON
OFF
(1)
(3)
Flow rate
(S1+0, S1+1)
Sampling period
Accumulated flow volume work area
(D1+4, D1+5)
Accumulated time work area
(D1+6, D1+7)

(1) When the enable totalizer input is off, totalization (updating the accumulated flow volume work area and the accumulated time work area) is stopped.
The accumulated flow volume work area and the accumulated time work area values are retained.

(2) Even if the input is turned off, no processing is performed as totalization was stopped in (1).
The accumulated flow volume work area and the accumulated time work area values are retained.

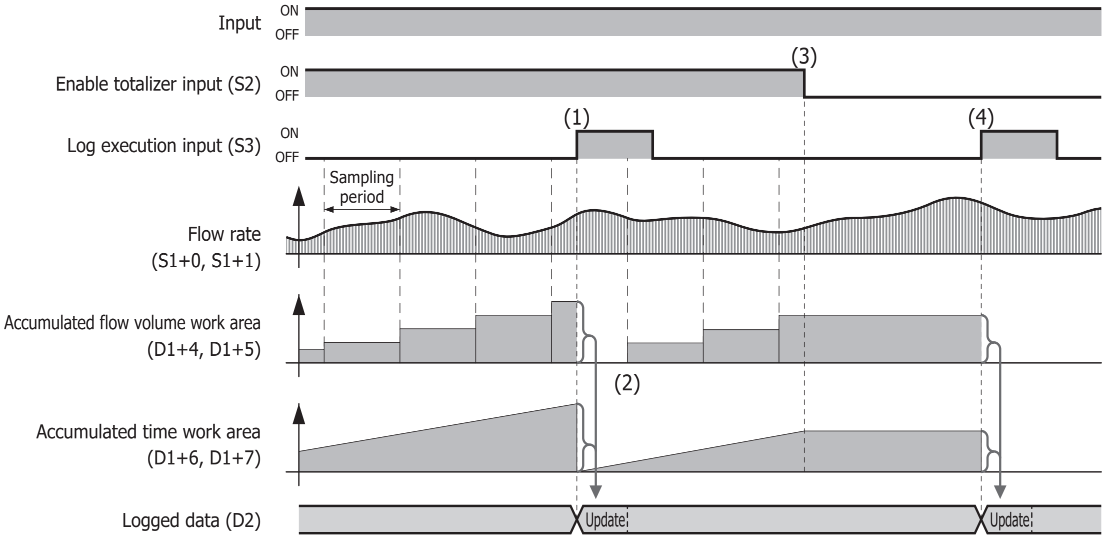
(3) When the input is off, no processing is performed even if the enable totalizer input is turned on.
The accumulated flow volume work area and the accumulated time work area values are retained.

■ Log Output Function

Log output

When the input is on

This section describes the log output operation when the input is on.



(1) When the enable totalizer input (S2) is on, if the log execution input (S3) is turned on, the log is updated and the accumulated flow volume work area and the accumulated time work area values are initialized to "0".

(2) The amount to add calculated from the flow rate (S1+0, S1+1) and the elapsed time is added to the accumulated flow volume work area.
When the enable totalizer input is on, the sampling period is retained even during the log update processing.

(3) When the enable totalizer input is off, sampling is paused and updating the accumulated flow volume work area and the accumulated time work area is stopped.
The accumulated flow volume work area and the accumulated time work area values are retained.

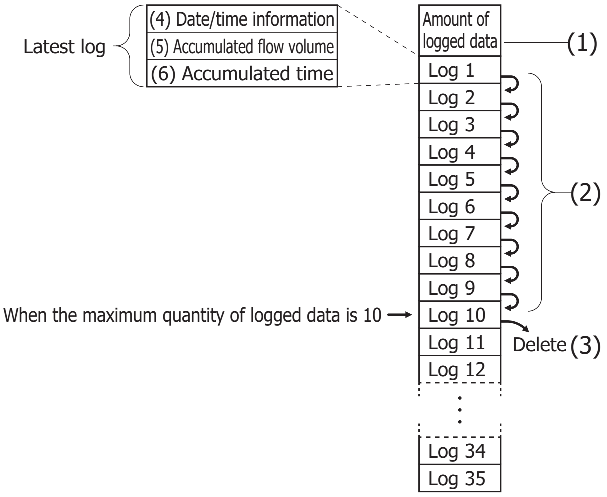
(4) Even when the enable totalizer input is off, if the input is on, the log is output in the same manner as (1).

When the input is off

When the input is off, no processing is performed even if the log execution input (S3) is turned on.

Log Updating

The logged data can be recorded from log 1 to log 35. Log 1 is the newest logged data.
Each time the log is updated, 1 is added to the amount of logged data (1) and the old logged data is pushed back by one item (2).
When the maximum amount of logged data is reached, the oldest log is deleted (3).
The date/time information (4) and accumulated flow volume (5) is stored in log 1.
The accumulated time (6) unit is converted from "ms" to "s" and stored in log 1.



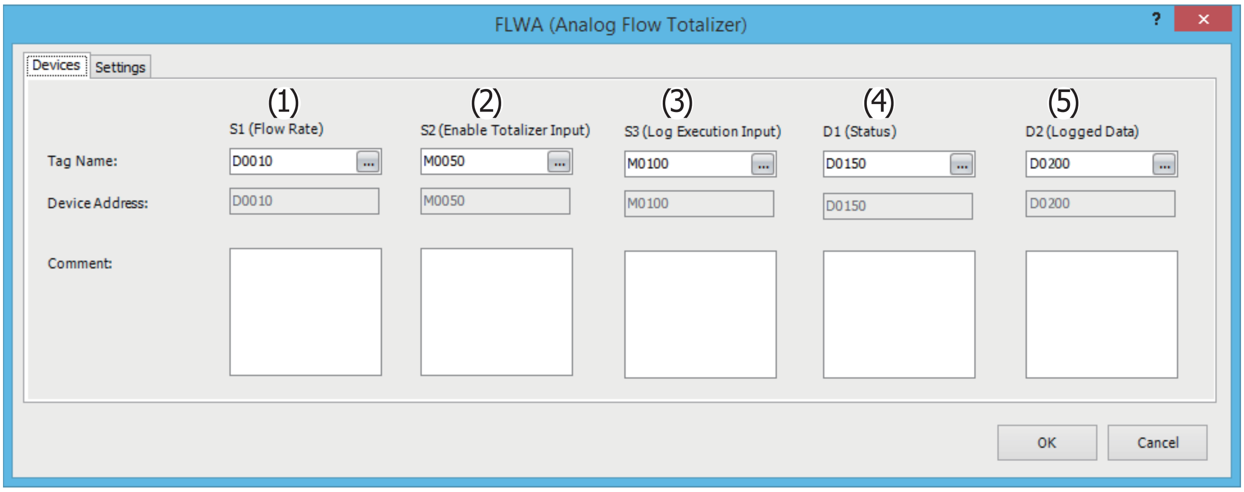
The maximum amount of logged data to use is specified on the **Settings** tab in the **FLWA (Analog Flow Totalizer)** dialog box.

Note: For details on the date/time information, the accumulated flow volume, and the accumulated time recorded in the log, see "(5) D2 (Destination 2): Logged Data" on page 27-15.

Settings

The **FLWA (Analog Flow Totalizer)** dialog box contains the **Devices** tab and the **Settings** tab.

■ Devices tab



	S1 (Flow Rate)	S2 (Enable Totalizer Input)	S3 (Log Execution Input)	D1 (Status)	D2 (Logged Data)
Tag Name:	D0010	M0050	M0100	D0150	D0200
Device Address:	D0010	M0050	M0100	D0150	D0200
Comment:					

Settings	Description
Tag name	Specifies the tag name or the device address for the device.
Device address	Shows the device address that corresponds to the tag name.
Comment	Shows the comment for the device address. This item can be edited.

(1) S1 (Source 1): Flow Rate

Specify the device that stores the flow rate. 2 continuous words (S1+0, S1+1) are used starting from the set device. The valid range for flow rate is 0 or 1.175494E-38 to 3.402823E+38 *1.

*1 The upper and lower data registers change according to the 32-bit data storage method specified. For details, see Chapter 3 "Instructions Reference" - "32-bit Data Storage" on page 3-9.

Note: If the flow rate value is outside the valid range, a user program execution error occurs, M8004 turns on, error code 28 is stored in D8006, and execution of the instruction is canceled. 10 is also stored in the status (D1) at the same time.
For user program execution errors, see "User Program Execution Errors" on page 3-10.

(2) S2 (Source 2): Enable Totalizer Input

Specify the device to enable totalization processing of the flow rate.
When the enable totalizer input is off, totalization is paused. When on, totalization is executed.

For details on the enable totalizer input, see "Function Descriptions" on page 27-11.

(3) S3 (Source 3): Log Execution Input

Specify the device to execute and stop the logging process that updates the log and initializes the accumulated flow volume work area and the accumulated time work area to "0".
When the log execution input changes from off to on, the logging process is executed.
For details on the log execution input, see "Log Output Function" on page 27-13.

(4) D1 (Destination 1): Status

Specify the device to store the status including the error during FLWA instruction execution. 10 continuous words are used starting from the set device.

Storage Destination	Function	Description
D1+0	Operation status	Stores the status including the error during FLWA instruction execution. 0 : Normal 10 : Flow rate out of range during sampling
D1+1 ⋮ D1+9	System work area	

(5) D2 (Destination 2): Logged Data

Specify the device to store the data that is logged. A maximum of 212 continuous words are used starting from the set device.

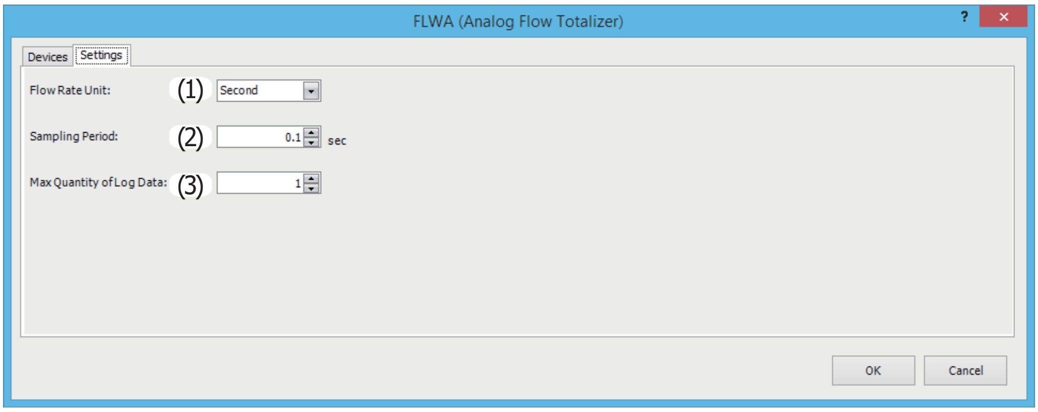
Note: The number of data registers that are used for logged data changes according to the maximum amount of logged data.
The number of data registers used for logged data is 2+6×the maximum amount of logged data.

Storage Destination	Function		Description	Data Range
D2+0	Amount of logged data		Stores the amount of logged data that has been recorded.	W (word): 0 to 35
D2+1	Reserved		—	—
D2+2 D2+3	Log 1	Log time	When the log execution input (S3) changes from off to on (during log updating), these devices store the content of special data registers D8008 to D8014 (current calendar/time values) converted to the number of seconds[*1] from January 1st, 1970.	D (double word): 0 to 4,294,967,295[*2]
D2+4 D2+5		Accumulated flow volume	When the log execution input (S3) changes from off to on (during log updating), these devices store the accumulated flow volume. For the processing when an accumulated flow volume overflow or underflow has occurred, see "Carry and Borrow in Floating-Point Data Processing" on page 3-8.	F (float): 0, 1.175494E-38 to 3.402823E+38[*2]
D2+6 D2+7		Accumulated time	When the log execution input (S3) changes from off to on (during log updating), these devices store the accumulated time. The unit is "s".	D (double word): 0 to 4,294,967,295[*2]
⋮	⋮	⋮	⋮	⋮
D2+206 D2+207	Log 35	Log time	Same as log 1.	Same as log 1.
D2+208 D2+209		Accumulated flow volume		
D2+210 D2+211		Accumulated time		

[*1] The time expressed as the number of seconds from 00:00:00 on January 1st, 1970.
(For example, if the date/time is 00:00:00 on January 1st, 2015, the value is 1,420,070,400 seconds.)

[*2] The upper and lower data registers change according to the 32-bit data storage method specified. For details, see Chapter 3 "Instructions Reference" - "32-bit Data Storage" on page 3-9.

■ **Settings tab**



Flow Rate Unit: (1) Second

Sampling Period: (2) 0.1 sec

Max Quantity of Log Data: (3) 1

(1) Flow Rate Unit

Select the time unit for flow rate that will be stored in the flow rate (S1+0, S1+1) from the following.

"Second", "Minute", "Hour", "Days"

(2) Sampling Period

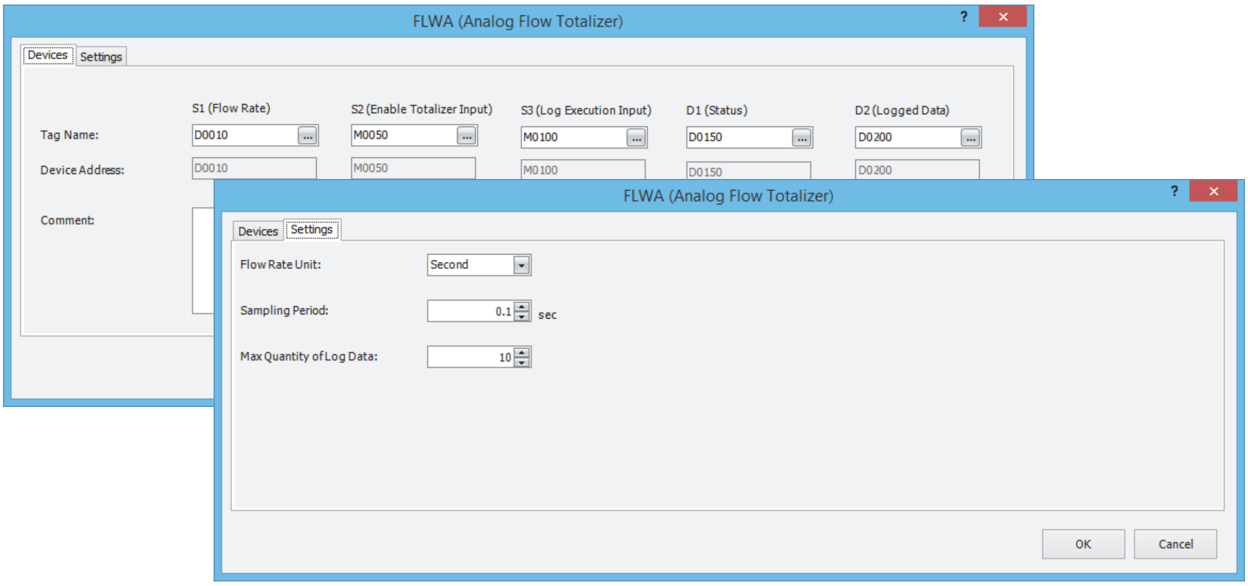
Specifies the time interval to sample the flow rate in 0.1 s increments.
The range is 0.1 to 6,553.5.

(3) Max Quantity of Log Data

Species the maximum amount of logged data that can be recorded.
The number of data registers that are used for the logged data (D2) changes according to the maximum amount of logged data.
The number of data registers used for logged data (D2) is 2+6×the maximum amount of logged data.
The range is 1 to 35.

Operation Example

Configuration Example



FLWA (Analog Flow Totalizer)

Devices | Settings

	S1 (Flow Rate)	S2 (Enable Totalizer Input)	S3 (Log Execution Input)	D1 (Status)	D2 (Logged Data)
Tag Name:	D0010	M0050	M0100	D0150	D0200
Device Address:	D0010	M0050	M0100	D0150	D0200
Comment:					

FLWA (Analog Flow Totalizer)

Devices | Settings

Flow Rate Unit: Second

Sampling Period: 0.1 sec

Max Quantity of Log Data: 10

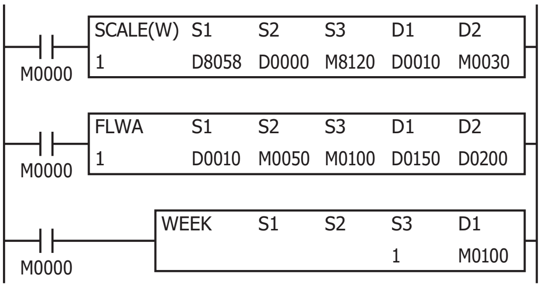
Devices tag

Item	Description	Comments
S1 (flow rate)	D0010	The value of the analog input value converted to the flow rate.
S2 (enable totalizer input)	M0050	
S3 (log execution input)	M0100	WEEK instruction output. Set to turn on for one scan every day at 00:00.
D1 (status)	D0150	
D2 (logged data)	D0200	

Settings tag

Item	Description	Comments
Flow rate unit	Second	
Sampling period	0.1 sec	
Max quantity of logged data	10	

Example program

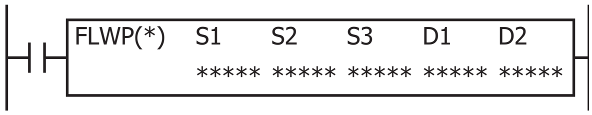


- If input M0000 turns on, the result of analog input value D8058 being converted to the flow rate is stored in D0010 with each scan.
- Sampling of the flow rate D0010 and totalization starts by turning on input M0000 and enable totalizer input M0050.
- The operation status is stored in D1 (status).
- By turning on M0100 for one scan every day at 00:00 with the WEEK instruction, S3 (log execution input) turns on and the current log information is stored in D2 (logged data).

FLWP (Pulse Flow Totalizer)

This instruction monitors a counter that measures the number of pulses and calculates the flow rate at a fixed cycle. It also stores the accumulated flow volume (the amount that flowed for an arbitrary period) to a log.

Symbol



Operation

The FLWP instruction calculates the flow rate and the accumulated flow volume by acquiring the pulse signal output from a pulse output type flow meter using a counter and monitoring the value of that counter. That result is stored in the devices specified by D1 (status) and D2 (logged data).

The FLWP instruction has a flow rate calculation function that calculates the flow rate from pulses, a totalizer function that accumulates the flow volume, and a log output function to temporarily save the accumulated volume at the desired time.

Note: When using an analog output type flow meter, the FLWA instruction can be used to monitor the flow volume. For details on the FLWA instruction, see "FLWA (Analog Flow Totalizer)" on page 27-9.

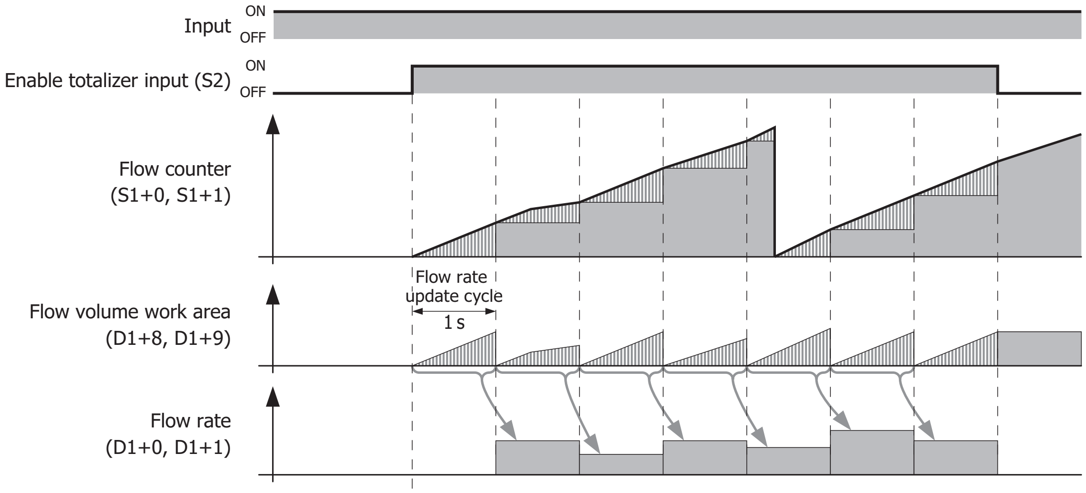
Notes:

- The maximum frequency of pulses that the FLWP instruction supports is 10 kHz.
- Output the log in a cycle within 119 hours.
- The FLWP instruction cannot be used in an interrupt program. If used in an interrupt program, a user program execution error occurs, the execution of the instruction is canceled, and the next instruction is executed.
  For user program execution errors, see "User Program Execution Errors" on page 3-10.
- The FLWP instruction must be executed with each scan as set or it cannot correctly measure the time. Therefore, it cannot be simultaneously used with the LABEL, LJMP, LCAL, LRET, JMP, JEND, MCS, and MCR instructions.

Flow Rate Calculation Function Overview

When the input is on, if the enable totalizer input (S2) is turned on, the amount that the flow counter (S1+0, S1+1) increased is added to the flow volume work area with each scan. The flow rate (D1+0, D1+1) is updated every second with the result that is calculated from the flow volume work area.

If the enable totalizer input is turned off, the flow rate is initialized to "0" and updating the flow rate is stopped.



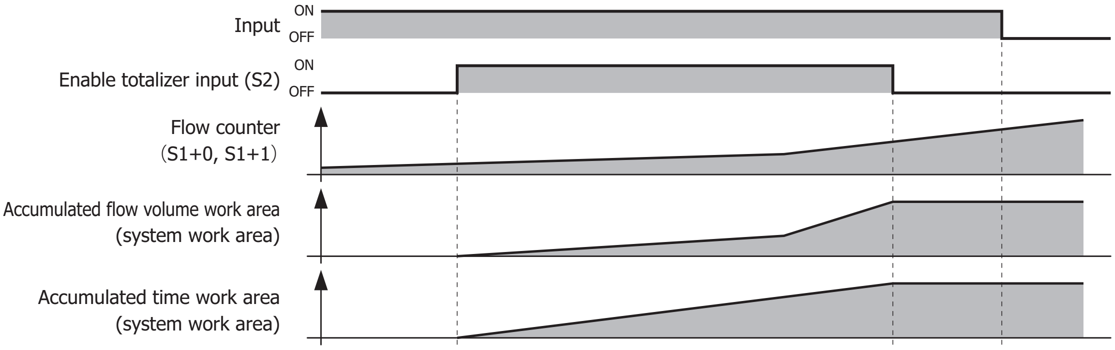
Note: For details on the flow calculation function such as the operation immediately after the input is turned on and the operation when the enable totalizer input is once again turned on, see "Flow Rate Function" on page 27-20.

Totalizer Function Overview

When the input is on, if the enable totalizer input (S2) is turned on, totalization starts.

When the enable totalizer input is on, the amount that the flow counter (S1+0, S1+1) increased is added to the accumulated flow volume work area with each scan. The accumulated time work area count also starts at the same time.

When the enable totalizer input is off, totalization is paused. (The accumulated flow volume work area and the accumulated time work area values are retained.)

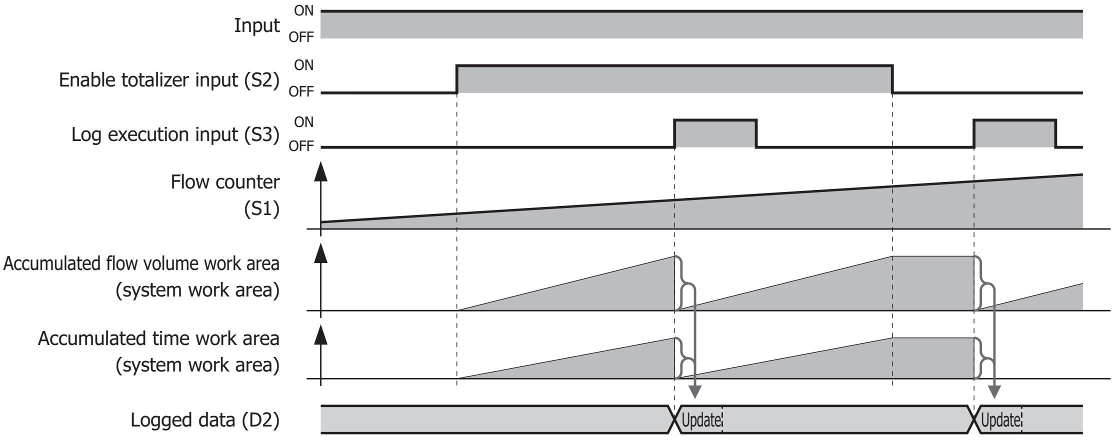

Input
ON
OFF
Enable totalizer input (S2)
ON
OFF
Flow counter
(S1+0, S1+1)
Accumulated flow volume work area
(system work area)
Accumulated time work area
(system work area)

Note: For details on the totalizer function such as the operation immediately after the input is turned on and the operation when the enable totalizer input is once again turned on, see "Totalizer Function" on page 27-22.

Log Output Function Overview

When the input is on, if the log execution input (S3) is turned on, the logged data (D2) is updated regardless of the enable totalizer input (S2) on/off status. The maximum amount of logged data is 35 items.

When the enable totalizer input is off, totalization is paused. The accumulated flow volume work area and the accumulated time work area values are retained.



Note: For details on the log output function such as the operation immediately after the input is turned on, the operation when the enable totalizer input (S2) is once again turned on, and the logged data updating when the log execution input (S3) is turned on, see "Log Output Function" on page 27-23.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	Flow counter	—	—	—	—	—	—	X	—	—	—
S2 (Source 2)	Enable totalizer input	X	—	X	—	—	—	—	—	—	—
S3 (Source 3)	Log execution input	X	—	X	—	—	—	—	—	—	—
D1 (Destination 1)	Status	—	—	—	—	—	—	▲	—	—	—
D2 (Destination 2)	Logged data	—	—	—	—	—	—	▲	—	—	—

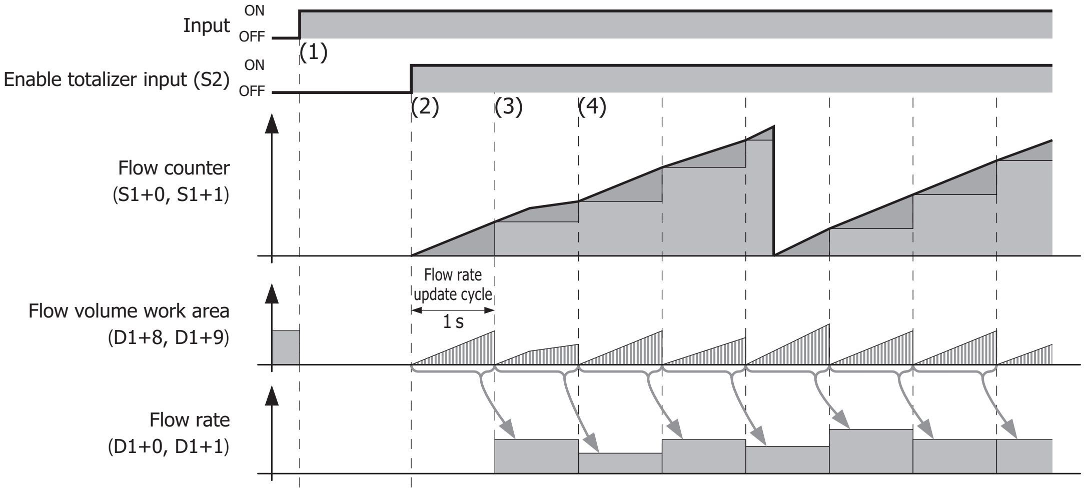
▲ Special data registers cannot be used.

Function Descriptions

■ Flow Rate Function

Starting the Flow Rate Calculation

This section describes the operation from turning on the input to starting the flow rate calculation.



(1) If the input is turned on, the flow volume work area and the flow rate (D1+0, D1+1) are initialized to "0".

(2) If the enable totalizer input (S2) is turned on, sampling of the flow counter (S1+0, S1+1) starts, and the amount that the flow counter increased is added to the flow volume work area with each scan.

(3) When the enable totalizer input is on, the flow volume is calculated from the flow volume work area each second and stored in the flow rate.
The flow rate calculation is executed at a cycle of 1 second regardless of whether the log execution input (S3) is turned on or off.

(4) (2) and (3) are then repeated.

Note: The flow rate formula is as follows.

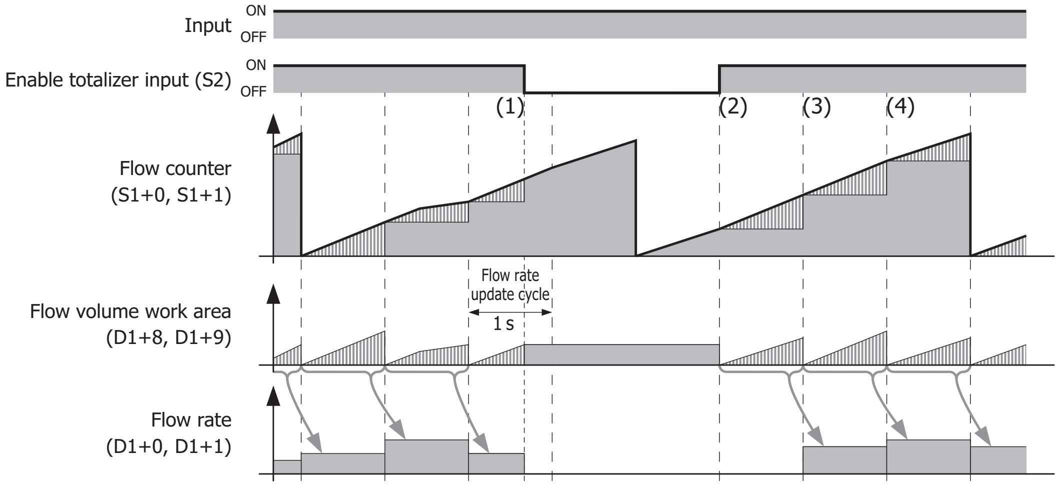
$$\textbf{Flow rate} = \frac{\textbf{Flow volume work area}}{\textbf{K factor}} \times \textbf{T}$$

T has the following values depending on the set flow rate unit.

T (s)	= 1
T (min)	= 60
T (hour)	= 60×60 = 360
T (day)	= 60×60×24 = 8,640

Pausing the Flow Rate Calculation

This section describes the operation from turning on the input to pausing the flow rate calculation.



(1) If the enable totalizer input (S2) is turned off, the flow rate is initialized to "0".

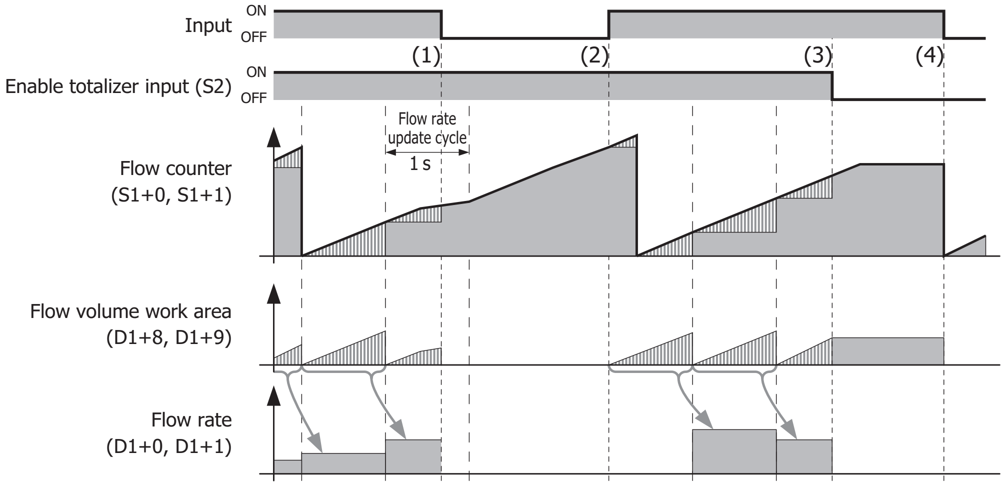
(2) If the enable totalizer input (S2) is turned on, sampling of the flow counter (S1+0, S1+1) starts, and the amount that the flow counter (S1) increased is added to the flow volume work area with each scan. The processing is the same as (2) in "Starting the Flow Rate Calculation" on page 27-20.

(3) 1 second after the enable totalizer input is turned on, the flow rate is calculated from the flow volume work area and stored in the flow rate (D1+0, D1+1).
Regardless of whether or not the log execution input (S3) is on or off, the flow rate is calculated in a 1 second cycle (the processing is the same as (3) in "Starting the Flow Rate Calculation" on page 27-20).

(4) (2) and (3) are then repeated.

Ending the Flow Rate Calculation

This section describes the operation to end the flow rate calculation.



(1) If the input is turned off when the enable totalizer input (S2) is on, the flow volume work area and flow rate are initialized to "0" and the instruction ends.

(2) If the input is turned on when the enable totalizer input is on, the flow rate calculation starts.

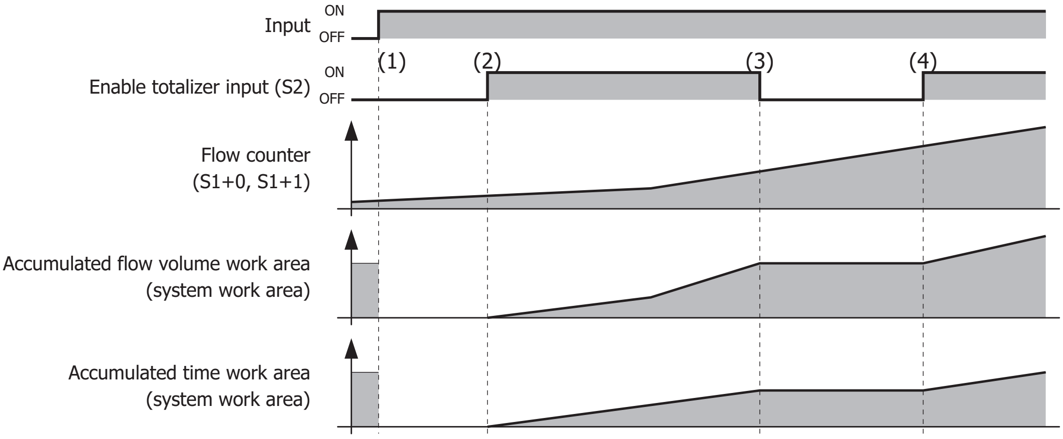
(3) If the enable totalizer input is turned off, the flow rate is initialized to "0" and the flow rate calculation is paused (the processing is the same as (1) in "Pausing the Flow Rate Calculation" on page 27-21).

(4) If the input is turned off, the flow volume work area and the flow rate (D1+0, D1+1) are initialized to "0" and the instruction ends.

■ Totalizer Function

Starting and Pausing Totalization

This section describes the operation from turning on the input to starting and pausing totalization.



(1) If the input is turned on, the operation status (D1+2), the amount of logged data (D2), the accumulated flow volume work area, the accumulated time work area, and the logged data (D2+2 to D2+2+(6×maximum amount of logged data)) are initialized to "0".

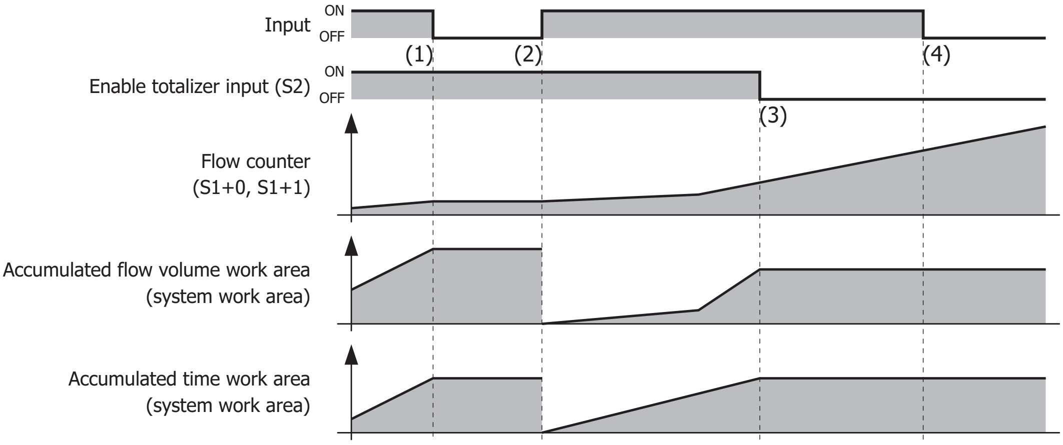
(2) If the enable totalizer input (S2) is turned on while the input is on, sampling of the flow counter starts, and the amount that the flow counter increased is added to the accumulated flow volume work area with each scan. The accumulated time work area also starts being measured at the same time.

(3) If the enable totalizer input is turned off, updating the accumulated flow volume work area and the accumulated time work area is paused.

(4) If the enable totalizer input is once again changed from off to on, the operation is the same as in (2) without initializing the accumulated flow volume work area and the accumulated time work area.

Ending Totalization

The section describes the operation to end totalization.



(1) If the input is turned off when the enable totalizer input (S2) is on, updating the accumulated flow volume work area and the accumulated time work area is stopped. The accumulated flow volume work area and the accumulated time work area values are retained.

(2) If the input is turned on, the accumulated flow volume work area and the accumulated time work area are initialized to "0". If the enable totalizer input is on at this time, updating the accumulated flow volume work area and the accumulated time work area starts.

(3) If the enable totalizer input is turned off, updating the accumulated flow volume work area and the accumulated time work area is paused.
The processing is the same as (3) in "Starting and Pausing Totalization" on page 27-22.

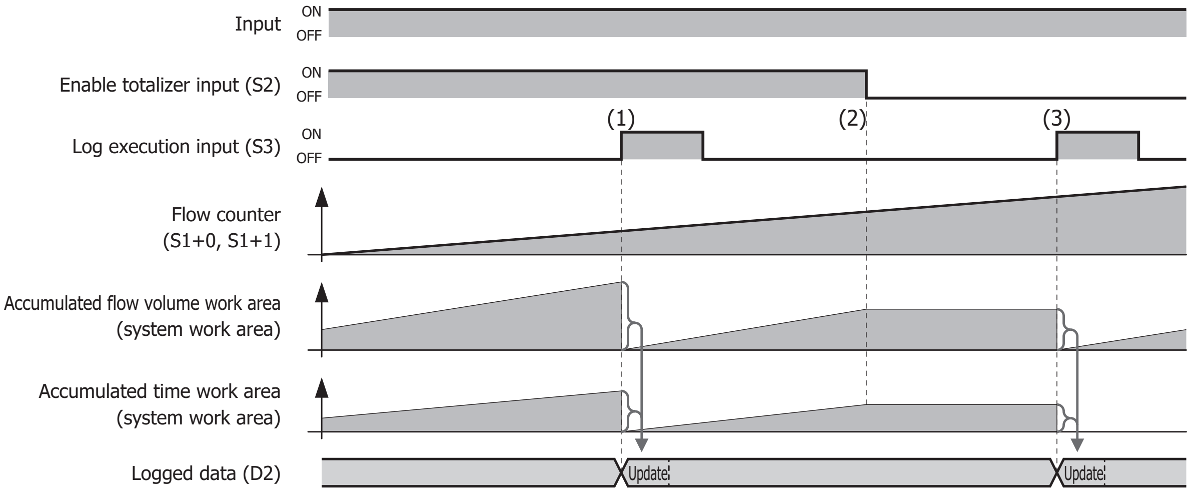
(4) If the input is turned off, the totalization processing stops. The accumulated flow volume work area and the accumulated time work area values are retained.

■ Log Output Function

Log output

When the input is on

This section describes the log output operation when the input is on.



(1) When the enable totalizer input (S2) is on, if the log execution input (S3) is turned on, the log is updated and the accumulated flow volume work area and the accumulated time work area values are initialized to "0".

(2) If the enable totalizer input is turned off, sampling is paused and updating the accumulated flow volume work area and the accumulated time work area is stopped. The accumulated flow volume work area and the accumulated time work area values are retained.

(3) Even when the enable totalizer input is off, if the input is on, the log output processing is performed in the same manner as (1).

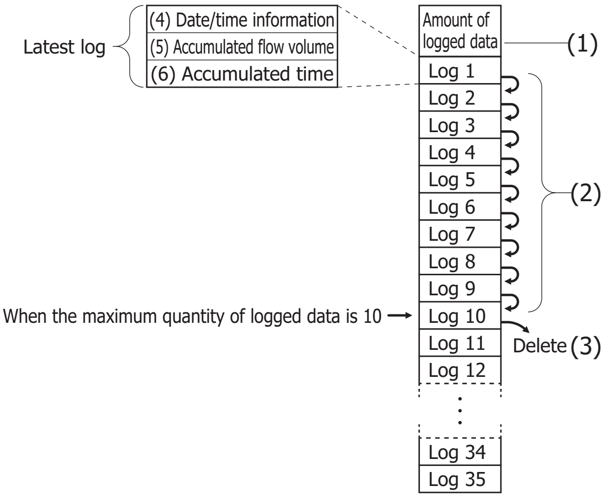
When the input is off

When the input is off, no processing is performed even if the log execution input (S3) is turned on.

Log Updating

The logged data can be recorded from log 1 to log 35. Log 1 is the newest logged data.

Each time the log is updated, 1 is added to the amount of logged data (1) and the old logged data is pushed back by one item (2).
When the maximum amount of logged data is reached, the oldest log is deleted (3).
The date/time information (4) and accumulated flow volume (5) is stored in log 1.
The accumulated time (6) unit is converted from "ms" to "s" and stored in log 1.



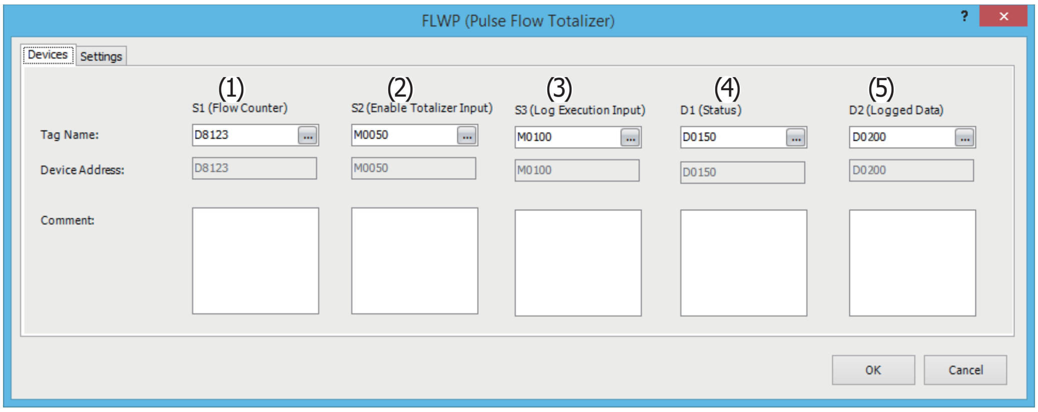
The maximum amount of logged data to use is specified on the **Settings** tab in the **FLWP (Pulse Flow Totalizer)** dialog box.

Note: For details on the date/time information, the accumulated flow volume, and the accumulated time recorded in the log, see "(5) D2 (Destination 2): Logged Data" on page 27-25.

Settings

The **FLWP (Pulse Flow Totalizer)** dialog box contains the **Devices** tab and the **Settings** tab.

■ Devices tab



Settings	Description
Tag name	Specifies the tag name or the device address for the device.
Device address	Shows the device address that corresponds to the tag name.
Comment	Shows the comment for the device address. This item can be edited.

(1) S1 (Source 1): Flow Counter

The flow counter setting differs according to the "(2) Counter Type" on page 27-26 operation type. Starting from the set device, a number of continuous devices are used according to the data type.

When Counter Type is "Free Run"

Specify the device that stores the counter value that is used to continuously add pulses. To find the number of pulse count for each scan, calculate it from the difference between the flow counter of the previous scan and latest flow counter.

When Counter Type is "Difference"

Specify the device that stores the number of pulse count for each scan.

The setting range differs according to "(2) Counter Type" on page 27-26.

If **Data Type** is "Word (W)", the range is 0 to 65,535.

If **Data Type** is "Double (D)", the range is 0 to 4,294,967,295*1.

*1 The upper and lower data registers change according to the 32-bit data storage method specified. For details, see Chapter 3 "Instructions Reference" - "32-bit Data Storage" on page 3-9.

(2) S2 (Source 2): Enable Totalizer Input

Set the device to enable flow volume totalization.

When the enable totalizer input is off, totalization is paused. When on, totalization is executed.

For details on the enable totalizer input, see "Function Descriptions" on page 27-20.

(3) S3 (Source 3): Log Execution Input

Specify the device to execute and stop the logging process that updates the log and initializes the accumulated flow volume work area and the accumulated time work area to "0".
When the log execution input changes from off to on, the logging process is executed.
While the FLWP instruction is being executed, execute the next logging process within 119 hours of the previous logging process.

For details on the log execution input, see "Log Output Function" on page 27-23.

(4) D1 (Destination 1): Status

Stores the status including the flow rate and the error during FLWP instruction execution. 16 continuous words are used starting from the set device.

Storage Destination	Function	Description	
D1+0 D1+1	Flow rate	Stores the flow rate calculated every second. The flow rate unit conforms to the K factor and flow rate unit settings.	F (float): 1.175494E-38 to 3.402823E+38*1
D1+2	Operation status	Stores the status including the error during FLWP instruction execution.	
D1+3 ⋮ D1+15	System work area		

*1 The upper and lower data registers change according to the 32-bit data storage method specified. For details, see Chapter 3 "Instructions Reference" - "32-bit Data Storage" on page 3-9.

(5) D2 (Destination 2): Logged Data

Specify the device to store the data that is logged. A maximum of 212 continuous words are used starting from the set device.

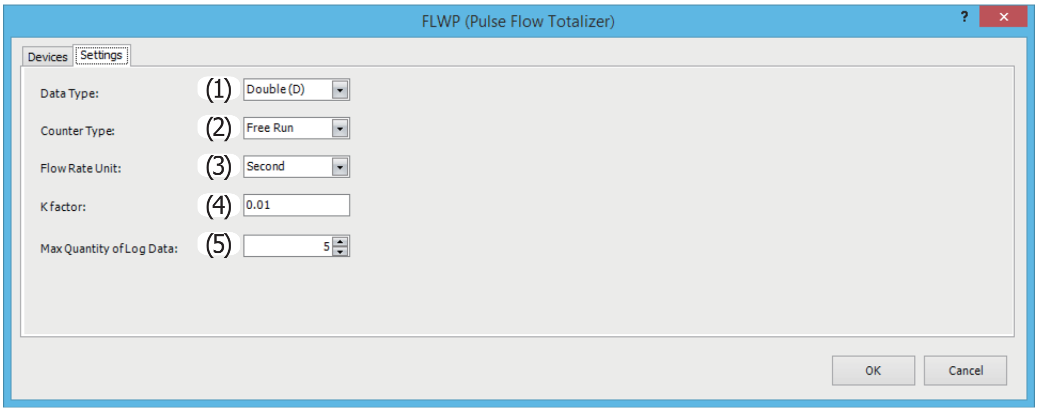
Note: The number of data registers that are used for logged data changes according to the maximum amount of logged data.
The number of data registers used for logged data is 2+6×the maximum amount of logged data.

Storage Destination	Function		Description	Range
D2+0	Amount of logged data		Stores the amount of logged data that has been recorded.	W (word): 0 to 35
D2+1	Reserved		—	—
D2+2 D2+3	Log 1	Log time	When the log execution input (S3) changes from off to on (during log updating), these devices store the content of special data registers D8008 to D8014 (current calendar/time values) converted to the number of seconds*1 from January 1st, 1970.	D (double word): 0 to 4,294,967,295*2
D2+4 D2+5		Accumulated flow volume	When the log execution input (S3) changes from off to on (during log updating), these devices store the accumulated flow volume. For the processing when an accumulated flow volume overflow or underflow has occurred, see "Carry and Borrow in Floating-Point Data Processing" on page 3-8.	F (float): 0, 1.175494E-38 to 3.402823E+38*2
D2+6 D2+7		Accumulated time	When the log execution input (S3) changes from off to on (during log updating), these devices store the accumulated time. The unit is "s".	D (double word): 0 to 4,294,967,295*2
⋮	⋮	⋮	⋮	⋮
D2+206 D2+207	Log 35	Log time	Same as log 1.	Same as log 1.
D2+208 D2+209		Accumulated flow volume		
D2+210 D2+211		Accumulated time		

*1 The time expressed as the number of seconds from 00:00:00 on January 1st, 1970.
(For example, if the date/time is 00:00:00 on January 1st, 2015, the value is 1,420,070,400 seconds.)

*2 The upper and lower data registers change according to the 32-bit data storage method specified. For details, see Chapter 3 "Instructions Reference" - "32-bit Data Storage" on page 3-9.

■ Settings tab



Data Type: (1) Double (D)

Counter Type: (2) Free Run

Flow Rate Unit: (3) Second

K factor: (4) 0.01

Max Quantity of Log Data: (5) 5

(1) Data Type

Select the flow counter data type as "Word (W)" or "Double (D)". The initial value is "Double (D)".

Word (W):	S1+0 is handled as the flow counter.
Double (D):	S1+0 and S1+1 are handled as the flow counter.

(2) Counter Type

Select the operation type for the flow counter that samples the pulses as "Free Run" or "Difference".

Free Run:	An addition-only free run operation is performed without initializing the flow counter. The free run operation resets the count to 0 and continues counting when the count exceeds the upper limit of the data type (65,535 (W), 4,294,967,295 (D)).
Difference:	The number of pulse count for each scan is stored in the flow counter. For example, use this option when you will count pulses with two different types of counters, calculate that difference in the ladder, and directly store that to the flow counter.

(3) Flow Rate Unit

Select the time unit for flow rate that will be stored in the flow rate (D1+0, D1+1) from the following.

"Second", "Minute", "Hour", "Days"

(4) K factor

Specify the constant that indicates the number of pulses per unit quantity that is output by the flow meter.
The unit differs according to the flow meter (liter, gallon, etc.), and this unit does not affect the operation of the FLWP instruction.
For example, if the flow meter outputs 10 pulses for a flow of 1 liter, the K factor is specified as 10 [pulses/liter].

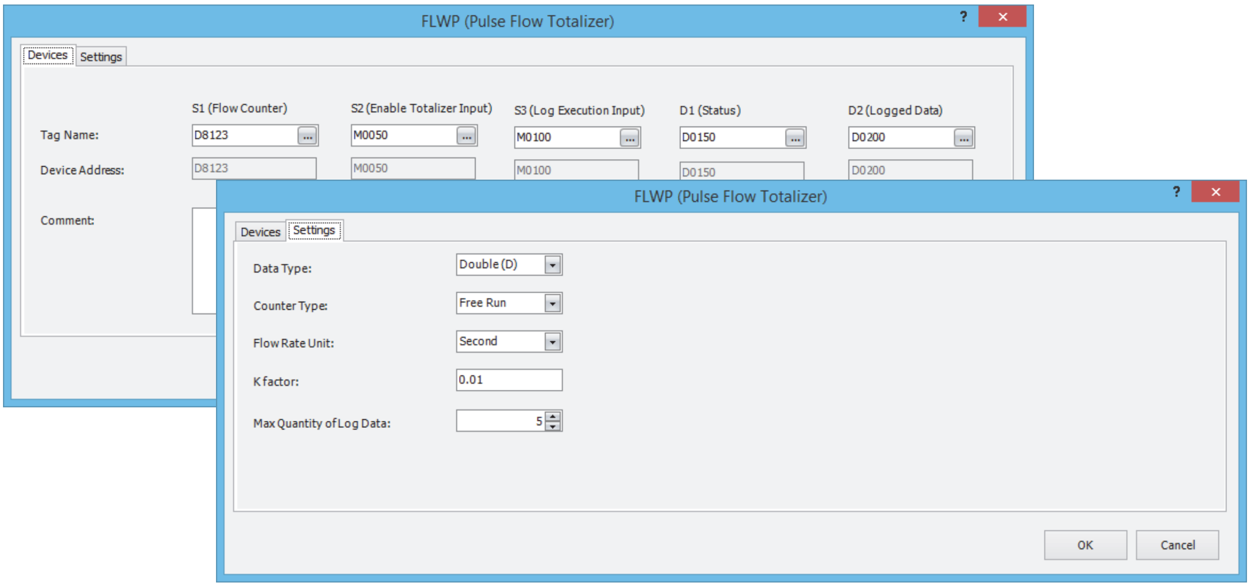
The setting range is 0 and 1.175494E-38 to 3.402823E+38.

(5) Max Quantity of Log Data

Species the maximum amount of logged data that can be recorded.
The number of data registers that are used for the logged data (D2) changes according to the maximum amount of logged data.
The number of data registers used for logged data (D2) is 2+6×the maximum amount of logged data.
The range is 1 to 35.

Operation Example

Configuration Example



FLWP (Pulse Flow Totalizer)

Devices

	S1 (Flow Counter)	S2 (Enable Totalizer Input)	S3 (Log Execution Input)	D1 (Status)	D2 (Logged Data)
Tag Name:	D8123	M0050	M0100	D0150	D0200
Device Address:	D8123	M0050	M0100	D0150	D0200
Comment:					

FLWP (Pulse Flow Totalizer)

Settings

Data Type:	Double (D)
Counter Type:	Free Run
Flow Rate Unit:	Second
K factor:	0.01
Max Quantity of Log Data:	5

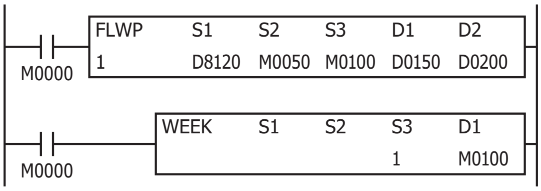
Devices tag

Item	Description	Comments
S1 (Flow Counter)	D8120	G1 high-speed counter measurement value.
S2 (Enable Totalizer Input)	M0050	
S3 (Log Execution Input)	M0100	WEEK instruction output. Set to turn on for one scan every day at 00:00.
D1 (Status)	D0150	
D2 (Logged Data)	D0200	

Settings tag

Item	Description	Comments
Data Type	Double (D)	
Counter Type	Free Run	
Flow Rate Unit	Second	
K factor	5.5344	
Max Quantity of Log Data	10	

Example program



- Totalization starts by turning on input M0000 and S2 (enable totalization input).
- The flow volume is calculated based on the accumulated value and the operation status is stored in D1 (status).
- By turning on M0100 for one scan every day at 00:00 with the WEEK instruction, S3 (logging trigger input) turns on and the current log information is stored in D2 (logged data).

28: User-defined Macro Instruction

This chapter describes the instruction that executes registered user-defined macros.

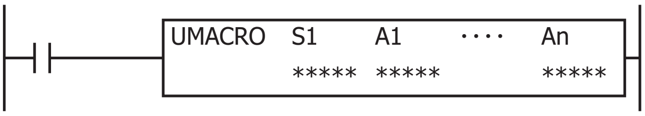
A user-defined macro is a ladder program that has been registered with an arbitrary number and name, and that can be used multiple times within the main program. For creating user-defined macros, see "User-defined macro registration procedure" on page 28-6.

For the differences between user-defined macros and subroutines, see "Differences between User-defined Macros and Subroutines" on page A-6.

UMACRO (User-defined Macro)

Executes the user-defined macro with the specified number.

Symbol



Operation

When the input is on, the user-defined macro that corresponds to the specified user-defined macro name will be executed. When the input is off, the user-defined macro that corresponds to the specified user-defined macro name will not be executed.

You can set the argument devices that will be used in the user-defined macro that is executed by the UMACRO instruction. Argument devices are devices that the user-defined macro will actually use when the UMACRO instruction is executed. For example, when I0001 is set for A1, A0001 will operate as I0001 in the user-defined macro.

Valid Devices

Device	Function	I	Q	M	R	T	C	D	P	Constant	Repeat
S1 (Source 1)	User-defined Macro Name	—	—	—	—	—	—	—	—	X[*1]	—
A1 ... An[*2]	Argument Device 1 to n (n is 1 ≤ n ≤ 100)	X	X	X	X	X	X	X	—	X	—

*1 The valid range of S1 (user-defined macro name) is 0 to 255.

*2 A1 to A100 are argument devices that will be used in the user-defined macro. The argument devices that will be used can be configured in the "Argument settings dialog box" on page 28-4, and 0 to 100 argument devices can be used.
For example, when **Number of arguments** is set to 5 in the "Argument settings dialog box" on page 28-4, argument devices A1 to A5 can be used. When **Number of arguments** is set to 0 in the "Argument settings dialog box" on page 28-4, argument devices cannot be used.

Local Devices

Local devices are the devices that can be used within the user-defined macro instruction only. By using local devices, you do not have to change the devices used in the user-defined macro when you re-use the user-defined macro in other projects.

Item	Definition
Symbol	@
Device Range	@0 to @127
Data type	Word, Integer, Double word, Long, Float

Notes:

- All local devices @0 to @127 are initializaed with zero when UMACRO instruction is executed.
- WindLDR does not support the monitoring and simulation for the local devices.

The list of instructions in which the local devices can be used is as follows:

Instruction	Support
LOD	—
LODN	—
AND	—
ANDN	—
OR	—
ORN	—
AND LOD	—
OR LOD	—
BPS	—
BRD	—
BPP	—
OUT	—
OUTN	—
SET	—
RST	—
TML	—
TIM	—
TMH	—
TMS	—
TMLO	—
TIMO	—
TMHO	—
TMSO	—
CNT	—
CDP	—
CUD	—
CNTD	—
CDPD	—
CUDD	—
CC=	X
CC≥	X
DC=	X
DC≥	X
SFR	—
SFRN	—
SOTU	—
SOTD	—

Instruction	Support
MCS	—
MCR	—
JMP	—
JEND	—
END	—
NOP	—
MOV	X
MOVN	X
IMOV	X
IMOVN	X
IBMV	X
IBMVN	X
BMOV	X
MOVC	X
NSET	X
NRS	X
XCHG	X
TCCST	X
CMP=	X
CMP<>	X
CMP<	X
CMP>	X
CMP<=	X
CMP>=	X
ICMP>=	X
LC=	X
LC<>	X
LC<	X
LC>	X
LC<=	X
LC>=	X
ADD	X
SUB	X
MUL	X
DIV	X
INC	X
DEC	X

Instruction	Support
ROOT	X
SUM	X
RNDM	X
RAD	X
DEG	X
SIN	X
COS	X
TAN	X
ASIN	X
ACOS	X
ATAN	X
LOGE	X
LOG10	X
EXP	X
POW	X
ANDW	X
ORW	X
XORW	X
SFTL	X
SFTR	X
ROTL	X
ROTR	X
BCDLS	X
WSFT	X
HTOB	X
BTOH	X
HTOA	X
ATOH	X
BTOA	X
ATOB	X
ENCO	X
DECO	X
BCNT	X
ALT	—
CVDT	X
DTDV	X
DTCB	X

Instruction	Support
SWAP	X
TXD	—
RXD	—
ETXD	—
ERXD	—
LABEL	—
LJMP	—
LCAL	—
LRET	—
DJNZ	—
DISP	—
DGRD	—
MSG	—
IOREF	—
HSCRF	—
FRQRF	—
COMRF	—
WKTBL	—
WKTIM	—
WEEK	—
YEAR	—
TADD	X
TSUB	X
HOUR	—
HTOS	X
STOH	X
DTML	—
DTIM	—
DTMH	—
DTMS	—
TTIM	—
PULS	—
PWM	—
RAMP	—
RAMPL	—
ZRN	—
ARAMP	—

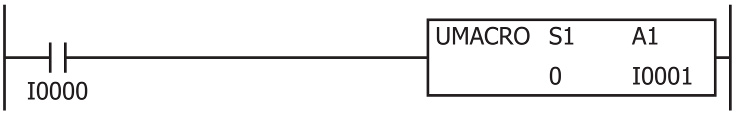
Instruction	Support
ABS	—
JOG	—
DI	—
EI	—
XYFS	—
CVXTY	—
CVYTX	—
PID	—
PIDA	—
PIDD	—
AVRG	—
RUNA	—
STPA	—
FIFOF	—
FIEX	X
FOEX	X
NDSRC	—
EMAIL	—
PING	—
SCRPT	—
DLOG	—
TRACE	—
SCALE	—
FLWP	—
FLWA	—

Operation Example

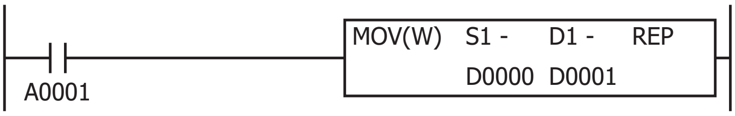
The following ladder program will execute the UMACRO instruction when main program input I0000 is turned on. When the UMACRO instruction is executed, user-defined macro name 0 will be executed.

A0001 will operate as I0001 in the user-defined macro because A1 of the UMACRO instruction has been set to I0001. Therefore, the MOV instruction will be executed when I0001 is turned on.

Main program



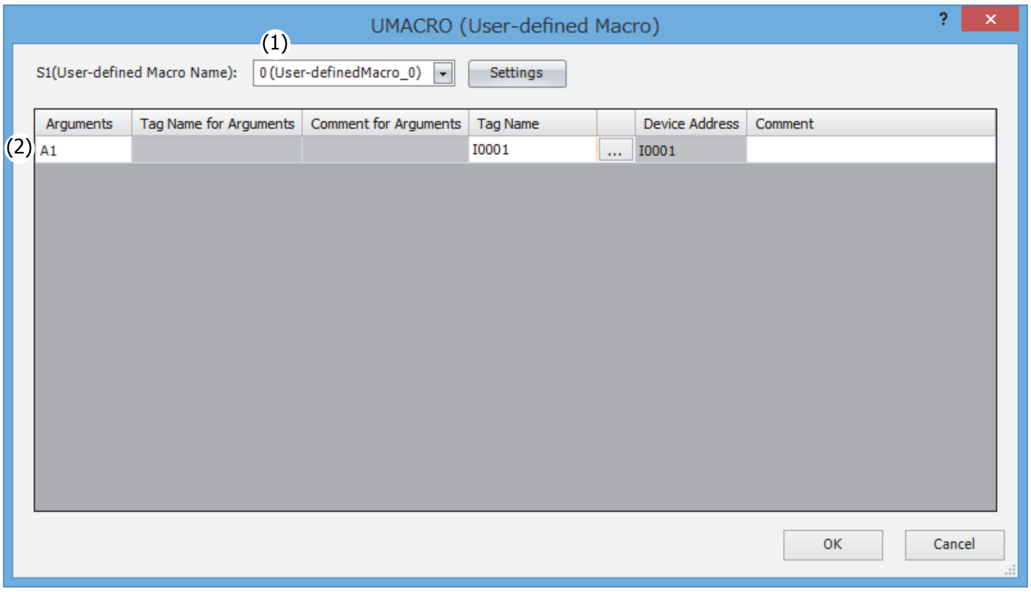
User-defined macro (user-defined macro name S1: 0)



Note: You can save time and effort in writing similar ladder programs and creating new ladder programs by turning a ladder program into a user-defined macro.

Note: The UMACRO, JMP, JEND, MCR, MCS, LCAL, LJMP, LRET, LABEL, DJNZ, and END instructions cannot be used in a user-defined macro.

Settings



UMACRO (User-defined Macro)

S1(User-defined Macro Name): 0 (User-definedMacro_0)

Arguments	Tag Name for Arguments	Comment for Arguments	Tag Name		Device Address	Comment
A1			I0001	...	I0001	

(1) S1 (source 1): User-defined Macro Name

Select the user-defined macro to execute.

Click **Settings** to display the **Argument settings** dialog box.

In the **Argument settings** dialog box, configure the argument settings for each user-defined macro number.

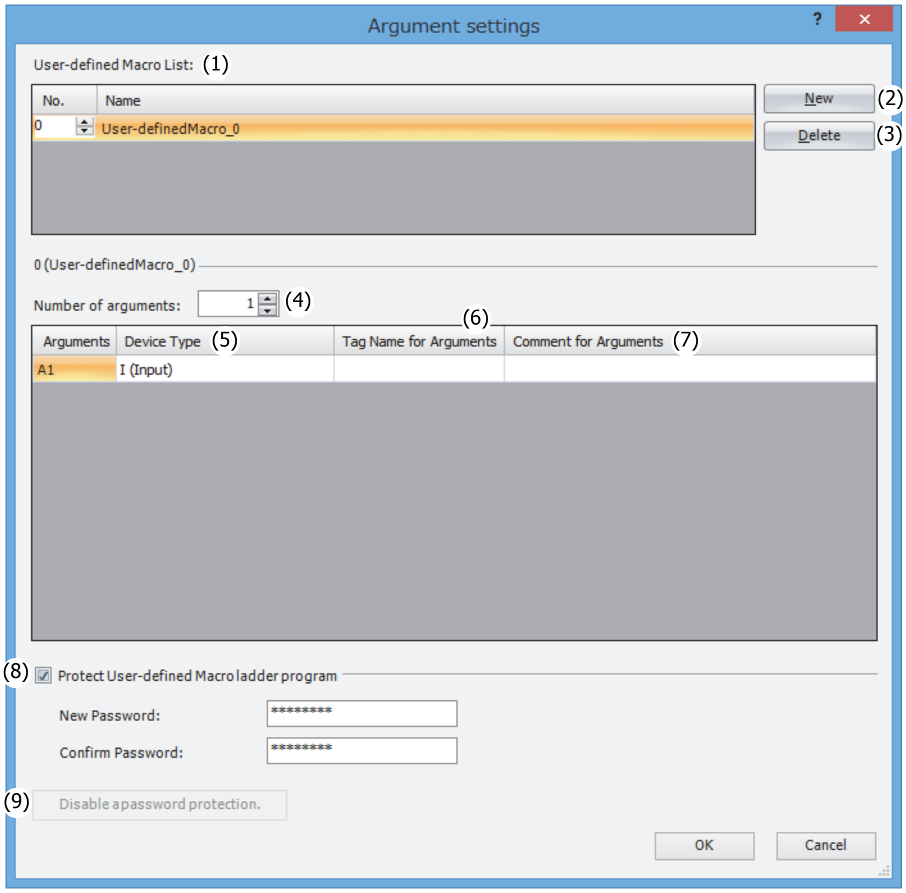
For details, see "Argument settings dialog box" on page 28-4.

(2) Argument Settings

Use this area to configure the devices that will be passed to the user-defined macro that is set with the user-defined macro name (1). The device address or tag name is set according to the device type of each argument displayed in the **Tag Name** column. The number of arguments and the device types can be changed in the **Argument settings** dialog box.

Argument settings dialog box

Select a user-defined macro or edit the user-defined macro with the selected number.



Argument settings

User-defined Macro List: (1)

No.	Name
0	User-definedMacro_0

0 (User-definedMacro_0)

Number of arguments: 1 (4)

Arguments	Device Type (5)	Tag Name for Arguments (6)	Comment for Arguments (7)
A1	I (Input)		

(8) ☑ Protect User-defined Macro ladder program

New Password: ********

Confirm Password: ********

(1) User-defined Macro List

Select a user-defined macro. Enter the name of the user-defined macro.

(2) New

Creates a new user-defined macro number.

(3) Delete

Deletes a registered user-defined macro.

(4) Number of arguments

Selects the number of arguments to use. You can specify the number between 0 and 100.

(5) Device Type

Selects the device type of each argument. When using an argument device in the user-defined macro, it will operate as the device type specified here.

(6) Tag Name for Arguments

Sets a tag name for the argument.

(7) Comment for Arguments

Sets a comment for the argument.

(8) Protect User-defined Macro ladder program

Select the **Protect User-defined Macro ladder program** check box to allow setting a password for the user-defined macro. When protection is enabled, you will be prompted to enter the password when displaying the ladder program for the user-defined macro.

For example, enable this option to allow the use of the created user-defined macro, but also to keep the ladder program for the user-defined macro private.

(9) Disable a password protection.

The argument settings cannot be changed when protection is enabled. To change the argument settings, click this button and enter the password to disable protection.

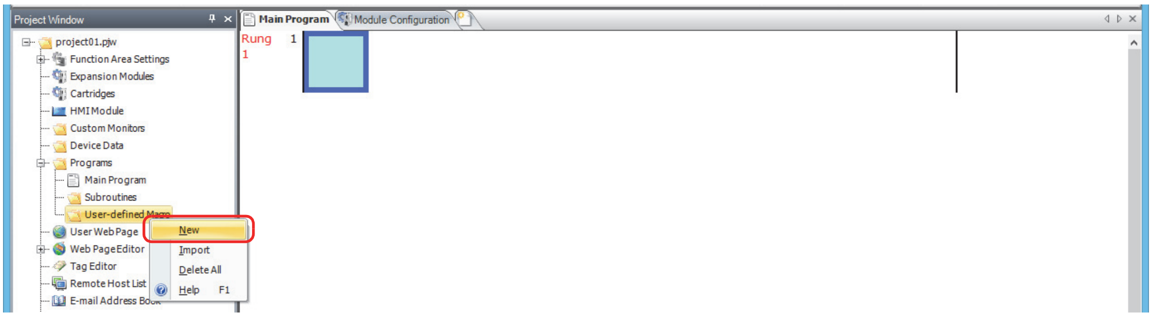
User-defined macro registration procedure

This section describes the procedure to create and register a user-defined macro.

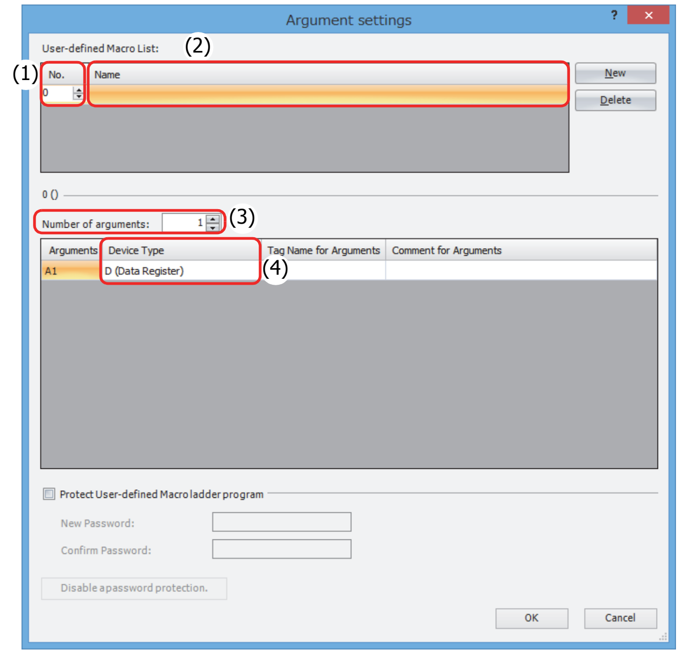
A registered user-defined macro can be set and executed with the UMACRO instruction.

1. Right-click **User-defined Macro** in the Project window and click **New**.

   The **Argument settings** dialog box will be displayed.



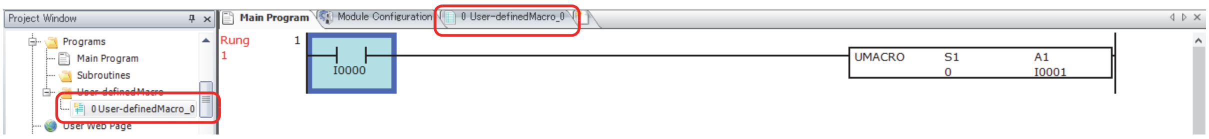
2. Configure the user-defined macro.
   - Set the user-defined macro number with **No.** (1).
   - Enter the name of the user-defined macro number in **Name** (2).
   - Select the number of arguments to use in the user-defined macro in **Number of arguments** (3).
   - Select the device type of the arguments with **Device Type** (4).



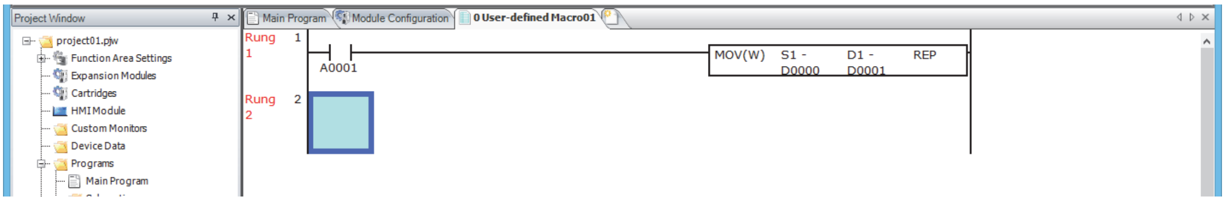
Note: You can prevent the settings in the **Argument settings** dialog box from being changed by selecting the **Protect User-defined Macro ladder program** check box and setting a password.

3. Click **OK** to close the **Argument settings** dialog box.

   The Editor will be displayed to create a ladder program in the user-defined macro.



4. Create a ladder program in the Editor for the created user-defined macro in the same manner as the main program and subroutines. The content of the argument devices (A1 to A100) configured in the **Argument settings** dialog box can be checked in the **User-defined Macro Arguments Allocation List** window and used in the ladder program as required. The created user-defined macro can be executed with the UMACRO instruction.



Note: The UMACRO, JMP, JEND, MCR, MCS, LCAL, LJMP, LRET, LABEL, DJNZ, and END instructions cannot be used in a user-defined macro.

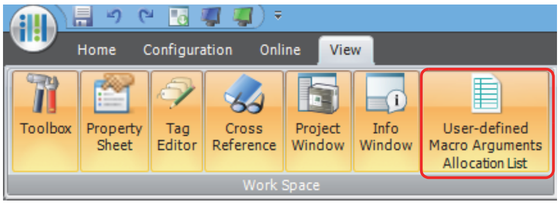
Note: The arguments configured in the **Argument settings** dialog box can be used in the user-defined macro as devices. For example, if the device type of A1 is set to data register in the **Argument settings** dialog box, A1 can be used in the same manner as a data register.

This concludes registering a user-defined macro.

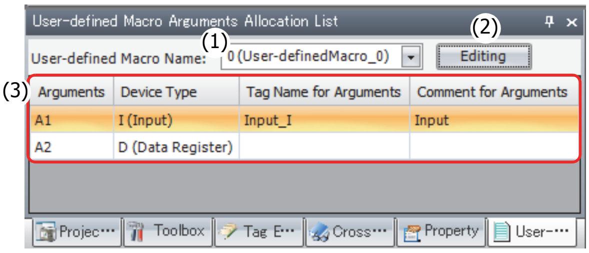
User-defined Macro Arguments Allocation List window

This window displays the content of arguments set for the user-defined macro.

1. On the **View** tab, in the **Work Space** group, click **User-defined Macro Arguments Allocation List**.
   The **User-defined Macro Arguments Allocation List** window is displayed.



2. The content that was set in the **Argument settings** dialog box can be checked here.



User-defined Macro Name: 0 (User-definedMacro_0)

Arguments	Device Type	Tag Name for Arguments	Comment for Arguments
A1	I (Input)	Input_I	Input
A2	D (Data Register)		

(1) User-defined Macro Name

Displays the user-defined macro name that was registered in the **Argument settings** dialog box.

(2) Editing

Opens the **Argument settings** dialog box. The user-defined macro settings can be edited in this dialog box.

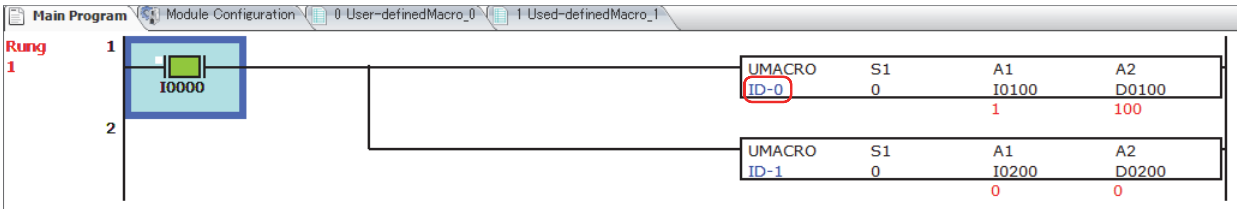
(3) Argument information

Displays the setting content of arguments set for the selected user-defined macro.

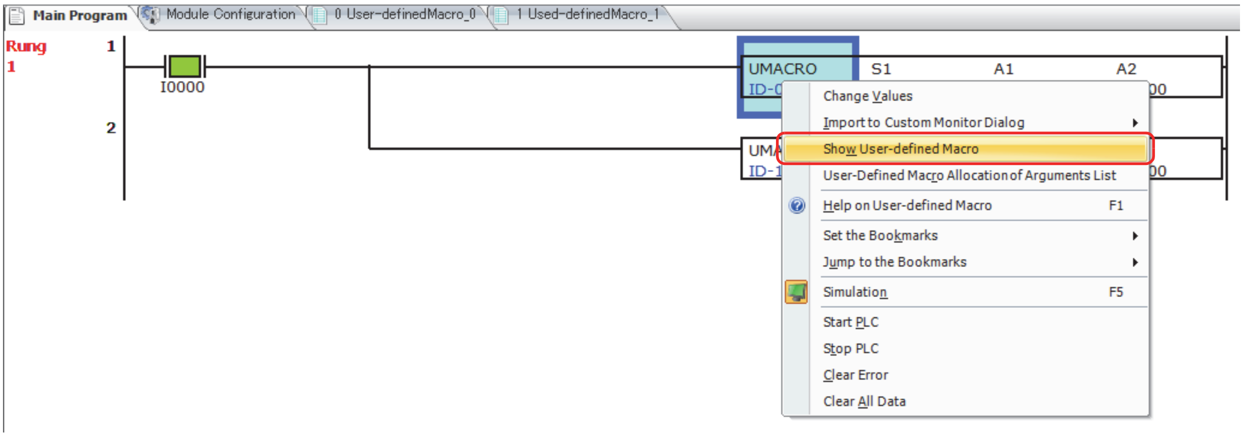
User-defined macro simulation and monitoring

This section describes the simulation and monitoring of a user-defined macro set for the UMACRO instruction. For how to use these operations, see "Simulate Operation" on page 1-9 and "Monitor Operation" on page 1-12.

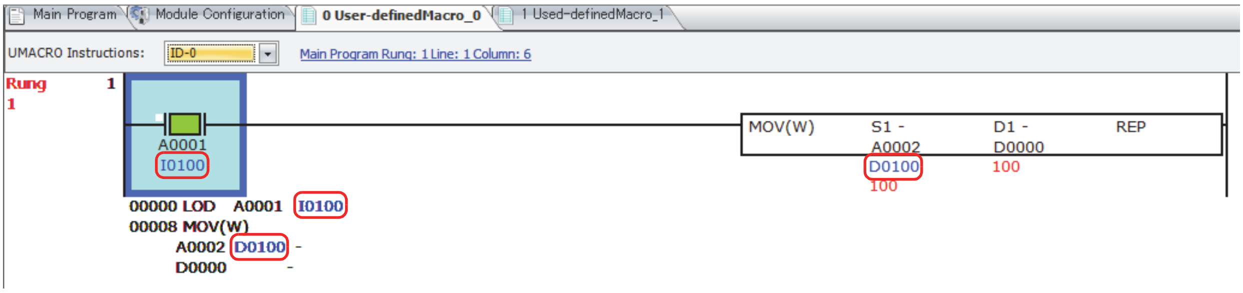
1. When the simulation and monitor are executed, the UMACRO instruction ID is displayed on the UMACRO instruction. UMACRO instruction IDs are assigned to identify the UMACRO instructions that have been set.



2. Select and right-click a UMACRO instruction, and then click **Show User-defined Macro** on the right-click menu.

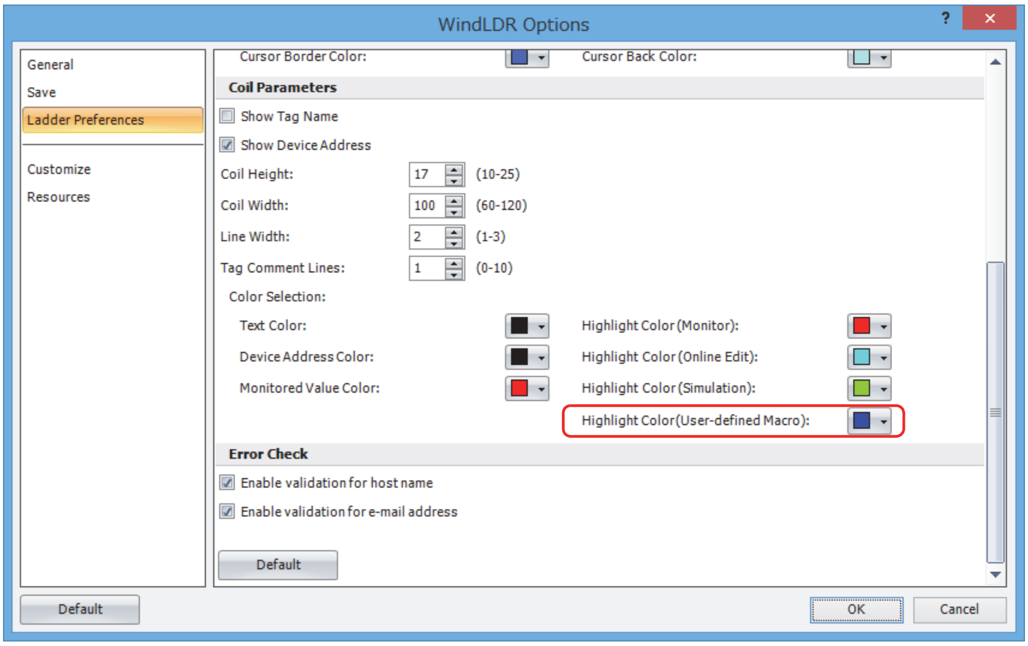


3. The user-defined macro opens that has the number set for the UMACRO instruction. The set argument devices (A1 to A100) are simulated and monitored according to the status of the device address or tag name set for the UMACRO instruction that called the user-defined macro.



```
00000 LOD    A0001  I0100
00008 MOV(W)
         A0002  D0100  -
         D0000         -
```

The colors of the ID displayed on the UMACRO instruction and the device address, tag name, and comment of the caller of the arguments set in the user-defined macro ladder program can be changed in the **WindLDR Options** dialog box.



WindLDR Options

- General
- Save
- Ladder Preferences
- Customize
- Resources

Cursor Border Color: 🟦 Cursor Back Color: ▢

Coil Parameters

☐ Show Tag Name

☑ Show Device Address

Coil Height: 17 (10-25)

Coil Width: 100 (60-120)

Line Width: 2 (1-3)

Tag Comment Lines: 1 (0-10)

Color Selection:

Text Color:	⬛	Highlight Color (Monitor):	🟥
Device Address Color:	⬛	Highlight Color (Online Edit):	🟦
Monitored Value Color:	🟥	Highlight Color (Simulation):	🟩
		Highlight Color(User-defined Macro):	🟦

Error Check

☑ Enable validation for host name

☑ Enable validation for e-mail address

APPENDIX

Execution Times for Instructions

Execution times for basic and advanced instructions of the FC6A Series MICROSmart are listed below.

Repeat is not assigned for any device.

Instruction	Device and Condition	Execution Time (μs)	
		All-in-One CPU Module/ CAN J1939 All-in-One CPU Module	Plus CPU Module
LOD	—	0.05	0.03
	Using data register	0.13	0.07
LODN	—	0.06	0.03
	Using data register	0.23	0.12
OUT	—	0.05	0.03
	Using data register	0.17	0.09
OUTN	—	0.05	0.03
	Using data register	0.17	0.09
SET, RST	—	0.05	0.03
	Using data register	0.17	0.09
AND	—	0.04	0.02
	Using data register	0.12	0.06
ANDN	—	0.05	0.03
	Using data register	0.23	0.12
OR	—	0.04	0.02
	Using data register	0.21	0.11
ORN	—	0.05	0.03
	Using data register	0.23	0.12
AND LOD, OR LOD	—	0.03	0.02
BPS	—	0.02	0.01
BRD	—	0.02	0.01
BPP	—	0.02	0.01
TML, TIM, TMH, TMS	—	3.46	1.73
TMLO, TIMO, TMHO, TMSO	—	3.44	1.72
CNT	—	3.26	1.63
CDP	—	3.48	1.74
CUD	—	3.38	1.69
CNTD, CDPD, CUDD	—	18.9	9.5
CC=, CC>=	—	1.85	0.93
DC=, DC>=	—	2.24	1.12
SFR, SFRN	N bits	7.04 + 0.04n	3.52 + 0.02n
SOTU, SOTD	—	2.04	1.02
MCS, MCR, JMP, JEND	—	—	—
END	—	—	—
NOP	—	—	—
MOV (W, I)	M→M	4.38	2.19
	D→D	0.18	0.09
MOVN (W, I)	M→M	4.46	2.23
	D→D	0.19	0.10
MOV (D, L)	M→M	14.6	7.3
	D→D	0.36	0.18
MOVN (D, L)	M→M	14.7	7.4
	D→D	0.39	0.20
MOV (F)	—	0.37	0.19
IMOV, IMOVN (W)	M+D→M+D	8.4	4.2
	D+D→D+D	8.56	4.28

Instruction	Device and Condition	Execution Time (μs)	
		All-in-One CPU Module/ CAN J1939 All-in-One CPU Module	Plus CPU Module
IMOV, IMOVN (D)	D+D→D+D	18.8	9.4
IMOV (F)	—	13.9	7.00
MOVC	—	—	—
BMOV	D→D	16.1 + 1.2n	8.1 + 0.6n
IBMV, IBMVN	M+D→M+D	13.4	6.7
	D+D→D+D	13.4	6.7
NSET (W, I)	D→D	2.28 + 2.2n	1.14 +1.1n
NSET (D, L)	D→D	2.2 + 12.6n	1.1 + 6.3n
NSET (F)	D→D	2.2 + 12.6n	1.1 + 6.3n
NRS (W, I)	D, D→D	5.84 + 0.64n	2.92 + 0.32n
NRS (D, L)	D, D→D	11.1 + 5.7n	5.6 + 2.9n
NRS (F)	D, D→D	11.0 + 5.8n	5.5 + 2.9n
XCHG (W)	D⇔D	5.32	2.66
XCHG (D)	D⇔D	26.0	13.0
TCCST (W)	D→T	4.68	2.34
TCCST (D)	D→T	15.0	7.5
CMP (=, <>, <, >, <=, >=) (W, I)	D⇔D→M	27.6	13.8
CMP (=, <>, <, >, <=, >=) (D, L)	D⇔D→M	38.0	19.0
CMP (=, <>, <, >, <=, >=) (F)	D⇔D→M	38.2	19.1
ICMP (>=)	D⇔D⇔D→M	29.0	14.5
ICMP (D, L, F)	D⇔D⇔D→M	44.6	22.3
LC (=, <>,<, >, <=, >=) (W, I)	D⇔D	4.84	2.42
LC (=, <>,<, >, <=, >=) (D, L)	D⇔D	15.2	7.6
LC (=, <>,<, >, <=, >=) (F)	D⇔D	15.3	7.7
ADD (W, I)	M+M→D	11.6	5.8
	D+D→D	11.7	5.9
ADD (D, L)	M+M→D	27.2	13.6
	D+D→D	27.2	13.6
ADD (F)	D+D→D	27.8	13.9
SUB (W, I)	M-M→D	11.6	5.8
	D-D→D	11.7	5.9
SUB (D, L)	M-M→D	27.2	13.6
	D-D→D	27.2	13.6
SUB (F)	D-D→D	27.4	13.7
MUL (W, I)	M×M→D	11.2	5.6
	D×D→D	11.3	5.7
MUL (D, L)	M×M→D	27.2	13.6
	D×D→D	27.2	13.6
MUL (F)	D×D→D	27.4	13.7
DIV (W, I)	M÷M→D	7.0	3.5
	D÷D→D	6.96	3.48
DIV (D, L)	M÷M→D	28.0	14.0
	D÷D→D	28.0	14.0
DIV (F)	D÷D→D	28.0	14.0
INC (W, I)	—	8.92	4.46
INC (D, L)	—	19.3	9.65
DEC (W, I)	—	8.92	4.46
DEC (D, L)	—	19.2	9.6
ROOT (W)	√D→D	6.24	3.12
ROOT (D)	√D→D	18.0	9.0
ROOT (F)	√D→D	19.4	9.7
SUM (W, I)	D, D→D	11.3 + 0.8n	5.7 + 0.4n
SUM (D, L)	D, D→D	16.6 + 6.0n	8.3 + 3.0n
SUM (F)	D, D→D	16.8 + 7.0n	8.4 + 3.5n

Instruction	Device and Condition	Execution Time (µs)	
		All-in-One CPU Module/ CAN J1939 All-in-One CPU Module	Plus CPU Module
RNDM	D, D→D	6.84	3.42
ANDW, ORW, XORW (W)	M・M→D	6.08	3.04
	D・D→D	6.2	3.1
ANDW, ORW, XORW (D)	D・D→D	21.6	10.8
SFTL, SFTR	N_B=100	14.10	7.05
BCDLS	D→D, S1=1	10.36	5.18
WSFT	D→D	15.6 + 1.4n	7.8 + 0.7n
ROTL, ROTR (W)	D, bits=1	9.1	4.6
HTOB (W)	D→D	4.94	2.47
BTOH (W)	D→D	4.94	2.47
HTOA (W)	D→D	7.20	3.6
ATOH (W)	D→D	5.48 + 0.88n	2.74 + 0.44n
BTOA (W)	D→D	5.88 + 0.72n	2.94 + 0.36n
ATOB (W)	D→D	5.84 + 0.76n	2.92 + 0.38n
ENCO (W)	M→D, bits=16	3.9 + 0.8n	1.95 + 0.4n
DECO	D→M	9.6	4.8
BCNT	M→D, bits=16	—	—
ALT	—	8.00	4.00
CVDT	W, I→F	10.2	5.1
	D, L→F	15.3	7.7
	F→W, I	10.3	5.2
	F→D, L	15.5	7.3
DTDV	D→D	4.78	2.39
DTCB	D→D	4.96	2.48
SWAP (W)	—	4.46	2.23
SWAP (D)	—	14.9	7.5
WKTIM	—	—	—
WKTBL	—	—	—
WEEK	—	—	—
YEAR	—	—	—
MSG	—	—	—
DISP	—	—	—
DGRD	—	—	—
LABEL	—	—	—
LJMP	—	—	—
LCAL	—	—	—
LRET	—	—	—
DJNZ	—	—	—
IOREF	—	9.48	4.74
HSCRF	—	8.32	4.16
FRQRF	—	—	—
COMRF	—	—	—
DI	—	7.44	3.72
EI	—	2.07	1.04
XYFS	—	—	—
CVXTY	—	—	—
CVYTX	—	—	—
AVRG (W, I)	S3=10	7.76	3.88
AVRG (D, L)	S3=10	13.5	6.8
AVRG (F)	S3=10	14.4	7.2
PULS	—	—	—
PWM	—	—	—
RAMP	—	—	—
RAMPL	—	—	—

Instruction	Device and Condition	Execution Time (µs)	
		All-in-One CPU Module/ CAN J1939 All-in-One CPU Module	Plus CPU Module
ZRN	—	—	—
ARAMP	—	—	—
ABS	—	—	—
JOG	—	—	—
PID	—	—	—
PIDA	—	—	—
PIDD	—	—	—
DTML, DTIM, DTMH	—	14.6	7.3
DTMS	—	13.7	6.9
TTIM	—	4.56	2.28
RAD	F→F	28.4	14.2
DEG	F→F	28.4	14.2
SIN	F→F	18.2	9.1
COS	F→F	18.4	9.2
TAN	F→F	17.6	8.8
ASIN	F→F	24.4	12.2
ACOS	F→F	24.6	12.3
ATAN	F→F	22.8	11.4
LOGE	F→F	19.00	9.5
LOG10	F→F	19.40	9.7
EXP	F→F	22.1	11.1
POW	F→F	34.0	17.0
FIFOF	—	23.2	11.6
FIEX	—	21.6	10.8
FOEX	—	21.6	10.8
NDSRC (W, I)	D, D, D→D	12.10	6.05
NDSRC (D, L)	D, D, D→D	43.0	21.0
NDSRC (F)	D, D, D→D	43.6	21.8
TADD	—	12.8	6.4
TSUB	—	12.8	6.4
HOUR	D→D, Y, D	8.0	4.0
HTOS	D→D	10.0	5.0
STOH	D→D	15.4	7.7
DLOG	—	—	—
TRACE	—	—	—
SCRPT	—	—	—
SCALE	—	—	—
FLWA	—	—	—
FLWP	—	—	—
UMACRO	—	—	—
TXD	—	—	—
RXD	—	—	—
ETXD	—	—	—
ERXD	—	—	—
PING	—	—	—
EMAIL	—	—	—

Processing in One Scan

While the FC6A Series MICROSmart is running, the FC6A Series MICROSmart performs operations repeatedly such as input refreshing, ladder program processing, and error checking.

A *scan* is the execution of all instructions from address zero to the END instruction. The time required for this execution is referred to as one *scan time*. The scan time varies with respect to program length.

The current value of the scan time is stored to special data register D8023 (scan time current value), and the maximum value of the scan time is stored to special data register D8024 (scan time maximum value). These values can be viewed on the **PLC status** dialog box of WindLDR while monitoring on a computer.

Executing Program Instructions

During the scan time, program instructions are processed sequentially starting with the first line of the ladder program, except for interrupt program execution. One scan time of a ladder program is approximately equal to the total execution time of each instruction shown on preceding pages.

Watchdog Timer

The watchdog timer monitors the time required for one program cycle (scan time) to prevent hardware malfunction. When the processing time for one scan exceeds the allowed time, a watchdog timer error occurs and the system is reset. If this is the case, place NOP instructions in the ladder diagram. The NOP instruction resets the watchdog timer. For details on the watchdog timer, see Chapter 5 "Watchdog Timer Settings" in the "FC6A Series MICROSmart User’s Manual".

Breakdown of END Processing Time

The END processing time depends on the FC6A Series MICROSmart settings and system configuration. The maximum values for the END processing time are as follows.

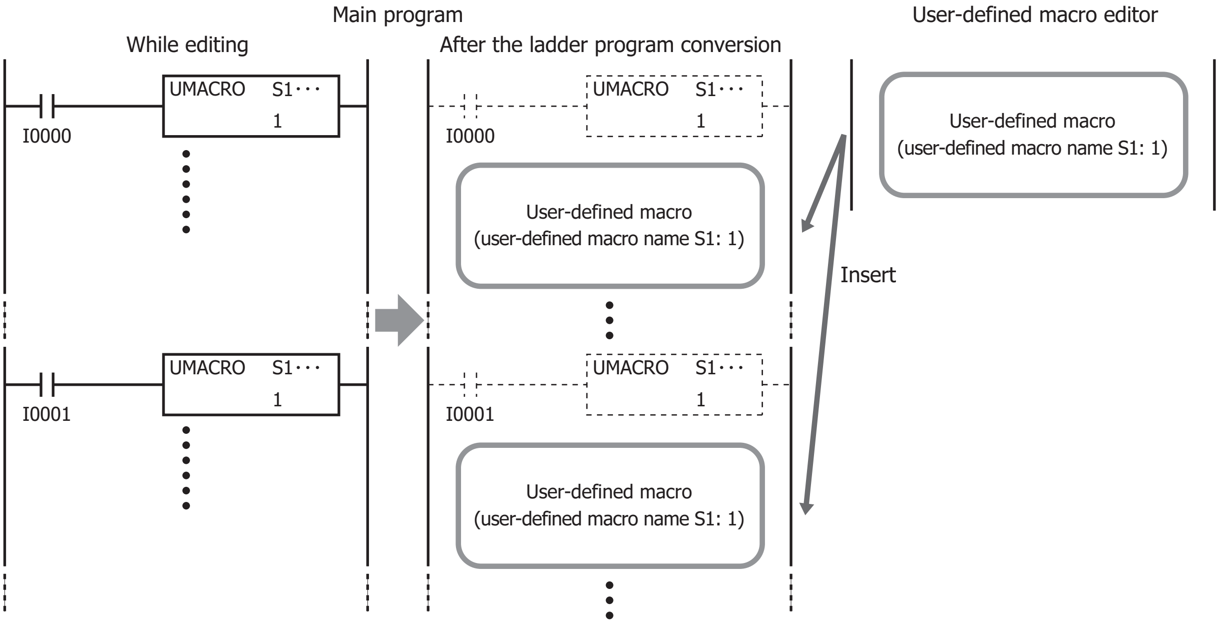
Item	Processing Time
END processing	640 µs

Differences between User-defined Macros and Subroutines

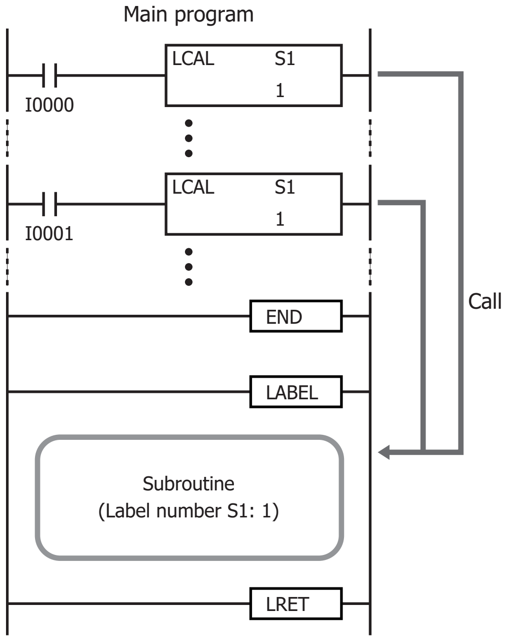
User-defined macros and subroutines are both functions that group the processing used multiple times into a single unit, and then execute that processing with a call instruction, but they have the following differences.

Item	User-defined Macro	Subroutine
Created location and call method	User-defined macros are created in the User-defined macro editor. The user-defined macro code for the corresponding user-defined macro name is inserted into the location of the UMACRO instruction.	Subroutines are created after the END instruction in the ladder program. The subroutine of the corresponding label number is called each time the LCAL instruction is executed.
User program size	When multiple UMACRO instructions are used with the same user-defined macro name, the size of the user program will increase by each use.	The size of the user program will not increase if the LCAL instruction is used multiple times with the same label number.
Argument device	Argument devices can be used.	Argument devices cannot be used.
Nested calls	Another user-defined macro cannot be called in the user-defined macro.	Another subroutine can be called in the subroutine.

■User-defined macro


Main program
While editing
After the ladder program conversion
User-defined macro editor
UMACRO S1･･･ 1
I0000
UMACRO S1･･･ 1
I0001
User-defined macro (user-defined macro name S1: 1)
Insert

■Subroutine


Main program
LCAL
S1
1
I0000
LCAL
S1
1
I0001
END
LABEL
Subroutine
(Label number S1: 1)
LRET
Call

Instruction Size (Bytes)

The byte quantities for basic and advanced instructions are listed below.

Basic Instruction	Used Data Size (Bytes)			
	All-in-One CPU Module/ CAN J1939 All-in-One CPU Module		Plus CPU Module	
	When Using Bits Specified in Data Registers	When Using Bit Devices	When Using Bits Specified in Data Registers	When Using Bit Devices
LOD	12	8	12	8
LODN	12	12	12	12
OUT	8	8	8	8
OUTN	8	8	8	8
SET, RST	8	8	8	8
AND	12	8	12	8
ANDN	12	12	12	12
OR	12	8	12	8
ORN	12	12	12	12
AND LOD	—	8	—	8
OR LOD	—	8	—	8
BPS	—	4	—	4
BRD	—	4	—	4
BPP	—	4	—	4
TML	—	12	—	12-16
TIM	—	12	—	12-16
TMH	—	12	—	12-16
TMS	—	12	—	12-16
TMLO	—	12	—	12-16
TIMO	—	12	—	12-16
TMHO	—	12	—	12-16
TMSO	—	12	—	12-16
CNT	—	12	—	12-16
CDP	—	12	—	12-16
CUD	—	12	—	12-16
CNTD	—	12	—	12-16
CDPD	—	12	—	12-16
CUDD	—	12	—	12-16
CC=	—	12	—	12-16
CC>=	—	12	—	12-16
DC=	—	12-16	—	12-24
DC>=	—	12-16	—	12-24
SFR	—	12	—	12
SFRN	—	12	—	12
SOTU	—	8	—	8
SOTD	—	8	—	8
MCS	—	4	—	4
MCR	—	4	—	4
JMP	—	12	—	12
JEND	—	4	—	4
END	—	4	—	4

Advanced Instruction	Used Data Size (Bytes)	
	All-in-One CPU Module/ CAN J1939 All-in-One CPU Module	Plus CPU Module
NOP	8	8
MOV (W, I)	16-20	16-28
MOVN (W, I)	16-20	16-28
MOV (D, L)	16-20	16-28
MOVN (D, L)	16-20	16-28
MOV (F)	16-20	16-28
IMOV, IMOVN (W)	20-28	20-44
IMOV, IMOVN (D)	20-28	20-44
IMOV (F)	20-28	20-44
MOVC	16-1044	16-1044
BMOV	16-20	16-32
IBMV, IBMVN	20-28	20-44
NSET (W, I)	16-1544	16-2060
NSET (D, L)	16-1544	16-2060
NSET (F)	16-1544	16-2060
NRS (W, I)	16-24	16-32
NRS (D, L)	16-24	16-32
NRS (F)	16-24	16-32
XCHG (W)	12-16	12-24
XCHG (D)	12-16	12-24
TCCST (W)	16-20	12-24
TCCST (D)	16-20	12-24
CMP (=, <>, <, >, <=, >=) (W, I)	20-28	16-32
CMP (=, <>, <, >, <=, >=) (D, L)	20-28	16-32
CMP (=, <>, <, >, <=, >=) (F)	20-28	16-32
ICMP (>=)	16-28	16-36
ICMP (D, L, F)	16-28	16-36
LC (=, <>, <, >, <=, >=) (W, I)	12-20	12-24
LC (=, <>, <, >, <=, >=) (D, L)	12-20	12-24
LC (=, <>, <, >, <=, >=) (F)	12-20	12-24
ADD (W, I)	16-28	16-36
ADD (D, L)	16-28	16-36
ADD (F)	16-28	16-36
SUB (W, I)	16-28	16-36
SUB (D, L)	16-28	16-36
SUB (F)	16-28	16-36
MUL (W, I)	16-28	16-36
MUL (D, L)	16-28	16-36
MUL (F)	16-28	16-36
DIV (W, I)	16-28	16-36
DIV (D, L)	16-28	16-36
DIV (F)	16-28	16-36
INC (W, I)	12	12-16
INC (D, L)	12	12-16
DEC (W, I)	12	12-16
DEC (D, L)	12	12-16
ROOT (W)	12-20	12-24
ROOT (D)	12-20	12-24
ROOT (F)	12-20	12-24
SUM (W, I)	16-24	16-36
SUM (D, L)	16-24	16-36
SUM (F)	16-24	16-36
RNDM	16-20	16-32
ANDW, ORW, XORW (W)	16-28	16-28

Advanced Instruction	Used Data Size (Bytes)	
	All-in-One CPU Module/ CAN J1939 All-in-One CPU Module	Plus CPU Module
ANDW, ORW, XORW (D)	16-28	16-28
SFTL, SFTR	16-24	16-40
BCDLS	12-16	12-24
WSFT	16-20	16-32
ROTL, ROTR (W)	12-16	12-20
HTOB (W)	12-20	12-24
BTOH (W)	12-20	12-24
HTOA (W)	16-24	16-32
ATOH (W)	16-24	16-32
BTOA (W)	16-24	16-32
ATOB (W)	16-24	16-32
ENCO (W)	16-20	16-28
DECO	16-20	16-28
BCNT	16-20	16-32
ALT	12	12
CVDT	16-20	16-28
DTDV	12-16	12-24
DTCB	12-16	12-24
SWAP (W)	16-20	16-28
SWAP (D)	16-20	16-28
WKTIM	20-24	20-40
WKTBL	12-92	12-172
WEEK	20-140	24-148
YEAR	28-220	28-228
MSG	20-	20-
DISP	16	16-24
DGRD	16-20	16-28
LABEL	12	12
LJMP	12	12-16
LCAL	12	12-16
LRET	8	8
DJNZ	12-16	12-24
IOREF	12	12
HSCRF	8	8
FRQRF	8	8
COMRF	8	8
DI	12	12
EI	12	12
XYFS	24-272	24 - 528
CVXTY	16-20	16 - 28
CVYTX	16-20	16 - 28
AVRG (W, I)	20-24	20-40
AVRG (D, L)	20-24	20-40
AVRG (F)	20-24	20-40
PULS	12-28	12-36
PWM	12-28	12-36
RAMP	12-36	12-44
RAMPL	36-48	36-48
ZRN	16, 52	16-68
ARAMP	40-316	40-332
ABS	12-16	12-20
JOG	36	36-48
PID	20-28	20-44
PIDA	112-116	112-124

Advanced Instruction	Used Data Size (Bytes)	
	All-in-One CPU Module/ CAN J1939 All-in-One CPU Module	Plus CPU Module
PIDD	272-276	272-288
DTML, DTIM, DTMH	16-24	16-36
DTMS	16-24	16-36
TTIM	12	12-16
RAD	12-20	12-24
DEG	12-20	12-24
SIN	12-20	12-24
COS	12-20	12-24
TAN	12-20	12-24
ASIN	12-20	12-24
ACOS	12-20	12-24
ATAN	12-20	12-24
LOGE	20	12-24
LOG10	20	12-24
EXP	20	12-24
POW	16-24	16-32
FIFOF	24	24-32
FIEX	12-16	12-20
FOEX	12-16	12-20
NDSRC (W, I)	16-28	16-40
NDSRC (D, L)	16-28	16-40
NDSRC (F)	16-28	16-40
TADD	16-24	16-36
TSUB	16-24	16-36
HOUR	20-24	20-40
HTOS	12-16	12-24
STOH	12-20	12-24
DLOG	28-408	28-668
TRACE	28-408	28-668
SCRPT	16-20	16-28
SCALE	36-44	36-60
FLWA	24-32	24-48
FLWP	28-36	28-52
UMACRO	24-624	12-812
TXD	20-	20-
RXD	20-	20-
ETXD	20-	20-
ERXD	20-	20-
PING	16-20	16-28
EMAIL	16-20	16-28

User Program Execution Error

This error indicates that invalid data is found during execution of a user program. When this error occurs, the ERR LED and special internal relay M8004 (user program execution error) are turned on. The detailed information of this error can be viewed from the error code stored in special data register D8006 (user program execution error code).

User Program Execution Error Code (D8006)	Error Details
1	Source/destination device exceeds range.
2	MUL result exceeds data type range.
3	DIV result exceeds data type range, or division by 0.
4	BCDLS has S1 or S1+1 exceeding 9999.
5	S1 is 10,000 or higher in the HTOB(W) instruction, or S1 is 100,000,000 or higher in the HTOB(D) instruction.
6	BTOH has any digit of S1 exceeding 9.
7	HTOA/ATOH/BTOA/ATOB has quantity of digits to convert out of range.
8	ATOH/ATOB has non-ASCII data for S1 through S1+4.
9	In the WEEK instruction, the ON time data exceeds range (hour data is 24 or higher, minute data is 60 or higher), or the OFF time data exceeds range (hour data is 25 or higher, minute data is 60 or higher, hour data is 24 but minute data is not zero). WKTIM has S1, S2, and S3 exceeding the valid range. S1: 0 through 127 S2/S3: Hour data 0 through 23, minute data 0 through 59 S2/S3 can be 10000. WKTBL instruction is not programmed or WKTIM instruction is executed before WKTBL instruction when 1 (additional days in the week table) or 2 (skip days in the week table) is set for MODE in the WKTIM instruction.
10	In the YEAR instruction, the month and day data exceeds range (month data is 13 or higher, day data is 32 or higher). In the WKTBL instruction, S1 to Sn is out of the range (month data is not between 1 and 12, day data is not between 1 and 31).
11	In the DGRD instruction, BCD and 5 digits are specified but the data exceeds 65,535.
12	CVXTY/CVYTX is executed without matching XYFS. XYFS and CVXTY/CVYTX have the same S1, but have different data types.
13	CVXTY/CVYTX has S2 exceeding the value specified in XYFS.
14	Label in LJMP, LCAL, or DJNZ is not found.
16	Executed invalid PID/PIDA instruction.
18	Attempt was made to execute an instruction that cannot be used in an interrupt program (see "Instructions Reference" on page 3-1).
19	Attempt was made to execute an instruction that is not available for the PLC.
20	Pulse output instructions have invalid values in operation parameters.
21	DECO has S1 exceeding 255.
22	BCNT has S2 exceeding 256.
23	ICMP>= has S1 < S3.
25	BCDLS has S2 exceeding 7.
26	DI or EI is executed when interrupt input or timer interrupt is not programmed in the Function Area Settings.
27	Work area is broken when using DTML, DTIM, DTMH, DTMS, or TTIM.
28	Source device data for F (float) data type instruction exceeds the valid range.
29	Result of F (float) data type instruction exceeds the data type range.
30	The shift data size or the number of bits to shift configured in the SFTL/SFTR instruction exceeds the valid range.
31	In the FIFOF instruction, the FIEX instruction or the FOEX instruction was executed before the FIFO data file was registered.
32	TADD, TSUB, HOUR, or HTOS has invalid data for source device S1.
33	RNDM has invalid data. S1 data is greater than S2 data or S1 or S2 data exceeds 32,767.
35	In the SUM instruction, the execution result exceeds the valid range for the selected data type, or S2 data is 0.

User Program Execution Error Code (D8006)	Error Details
36	DLOG/TRACE is executed but the capacity of CSV file exceeds the maximum size of a day or the number of files exceeds the maximum number of a day.
41	SD memory card is write protected.
42	The result of execution using the SCRPT instruction is a value that indicates anything other than normal completion.
46	In the SCALE instruction, the input minimum value is larger than the input maximum value, the output minimum value is larger than the output maximum value, or the dead band data exceeds the valid range.
48	Pulse output instructions using the same pulse output are executed at the same time.
49	Pulse output instructions referring to the absolute position counter are executed without initializing the absolute position counter with ABS instruction.

INDEX